Data-Driven Learning Models with Applications to Retail Operations

by

Sajad Modaresi

Business Administration
Duke University

Date: ____________________

Approved:

____
Fernando Bernstein, Supervisor

____
Nuri Bora Keskin

____
Robert Swinney

____
Peng Sun

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
2018

Abstract

Data-Driven Learning Models with Applications to Retail Operations

by

Sajad Modaresi

Business Administration
Duke University

Date: ____________________

Approved:

____
Fernando Bernstein, Supervisor

____
Nuri Bora Keskin

____
Robert Swinney

____
Peng Sun

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
2018

Abstract

Data-driven approaches to decision-making under uncertainty is at the center of many operational problems. These are problems in which there is an element of uncertainty that needs to be *estimated* (*learned*) from data in order to make *online* (*dynamic*) operational decisions. An example of such problems is estimating customer demand from transaction data to make dynamic assortment decisions. This dissertation adopts a data-driven active learning approach to study various operational problems under uncertainty with a focus on retail operations.

The first two essays in this dissertation study the classic *exploration* (i.e., parameter estimation) versus *exploitation* (i.e., optimization) trade-off from different perspectives. The first essay takes a theoretical approach and studies such trade-off in a combinatorial optimization setting. We show that resolving the exploration versus exploitation trade-off efficiently is related to solving a *Lower Bound Problem* (LBP), which simultaneously answers the questions of *what* to explore and *how* to do so. We establish a fundamental limit on the asymptotic performance of any admissible policy that is proportional to the optimal objective value of the LBP problem. We also propose near-optimal policies that are implementable in real-time. We test the performance of the proposed policies through extensive numerical experiments and show that they significantly outperform the relevant benchmark.

The second essay considers the dynamic assortment personalization problem of an online retailer facing heterogeneous customers with unknown product preferences.

We propose a prescriptive approach, called the *dynamic clustering* policy, for dynamic estimation of customer preferences and optimization of personalized assortments. We test the proposed approach with a case study based on a dataset from a large Chilean retailer. The case study suggests that the benefits of the dynamic clustering policy under the Multinomial Logit (MNL) model can be substantial and result (on average) in more than 37% additional transactions compared to a *data-intensive* policy that treats customers independently and in more than 27% additional transactions compared to a *linear-utility* policy that assumes that product mean utilities are linear functions of available customer attributes. We support the insights derived from the numerical experiments by analytically characterizing settings in which pooling transaction information is beneficial for the retailer, in a simplified version of the problem. We also show that there are diminishing marginal returns to pooling information from an increasing number of customers.

Further focusing on retail operations, the final essay studies the interplay between a retailer's return and pricing policies and customers' purchasing decisions. We characterize the retailer's optimal prices in the cases with and without product returns and determine conditions under which offering the return option to customers increases the retailer's revenue. We also explore the impact of accounting for product returns on demand estimation. The preliminary numerical results based on a real dataset suggest that our model, which accounts for product returns, increases demand estimation accuracy compared to models that do not consider product returns in their estimation.

To Grandpa.

Contents

List of Tables

List of Figures

List of Abbreviations and Symbols

Symbols

Notation for Chapter 2

The notation $x(a)$ refers to the a-th element of vector x.

A	Ground set.
$\mathcal{S}; S$	Set of all feasible solutions of the underlying combinatorial problem; A feasible solution.
N	Total number of periods.
B_n	Random cost coefficient vector in period n.
F	Common distribution of the cost coefficient vector.
l	Vector of lower bounds on the range of F.
u	Vector of upper bounds on the range of F.
S_n	Solution implemented in period n.
b_n	Realized cost coefficient vector in period n.
π	A non-anticipating policy.
$\mathcal{F}_n$	Filtration (history) in period n.
$\mathbb{E}$	Expectation.
$J^\pi(\cdot); R^\pi(\cdot)$	Expected cumulative cost associated with policy π; Regret associated with policy π.
$Comb(\nu)$	Underlying combinatorial optimization problem given the cost vector ν.

$z^*_{Comb}(\nu)$	Optimal objective value of the underlying combinatorial problem given the cost vector ν.
$\mathcal{S}^*(\nu)$	Set of optimal solutions to the underlying combinatorial problem given the cost vector ν.
μ	Mean cost coefficient vector.
$\mathbf{1}\{\cdot\}$	Indicator function of a set.
$T_n(a)$	Number of trials of ground element a prior to period n.
Δ^ν_S	Optimality gap associated with solution S given the cost vector ν.
LBP	Lower Bound Problem.
z^*_{LBP}	Optimal objective value of the LBP problem.
π^*	LBP-based policy.
$OCP(\nu)$	Optimality Cover Problem given the cost vector ν.
$z^*_{OCP}(\nu)$	Optimal objective value of the OCP problem given the cost vector ν.
$\Gamma_{OCP}(\nu)$	Set of optimal solutions to the OCP problem given the cost vector ν.
$\widehat{\mu}_n$	Vector of sample means of cost realizations of ground elements prior to period n.
n_i	Starting period of cycle i.
H	Tuning parameter of the OCP-based policy.
π_{OCP}	OCP-based policy.
$\mathrm{conv}(\cdot)$	Convex hull of a set.
$\mathrm{diag}(\nu)$	Diagonal matrix with vector ν as its diagonal.

Notation for Chapter 3

Superscript i refers to profile i and subscripts j and t refer to product j and time (customer arrival) t, respectively. Moreover, the notation $\hat{y}$ denotes an estimate for a parameter y and $\tilde{y}$ denotes a threshold on the value of a parameter y.

T	Total number of customer arrivals.
$N; \mathcal{N}$	Total number of products; Set of all products.
$I; \mathcal{I}$	Total number of profiles; Set of all profiles.
$I_k; \mathcal{I}_k$	Number of profiles in cluster k; Set of profiles in cluster k.
C	Display constraint.
r_j	Price of product j.
x^i	Unique vector of attributes associated with profile i.
p_i	Arrival probability of profile i.
P_k	Proportion of profiles in cluster k.
i_t	Profile of customer t.
$S_t; S_i^*$	Assortment offered to customer t; Optimal assortment for profile i.
$Z_{j,t}^i$	Purchasing decision of a customer with profile i arriving at time t regarding product j.
$U_{j,t}^i$	Random utility of product j for a customer with profile i arriving at time t.
$\mu_j^i; \mu^i$	Preference parameter of product j for profile i; Vector of preference parameters for profile i.
$\zeta_{j,t}^i$	Standard Gumbel random variables.
ν_j^i	Exponentiated mean utility of product j for profile i.
$\Pi_j(S, \mu^i)$	Purchase probability of product j for profile i given an assortment S.
$\mathcal{F}_t$	Filtration (history) at time t.
π	An admissible assortment selection policy.
$\mathcal{P}; \mathcal{P}'$	Set of all admissible policies; Set of all admissible consistent policies.
$\mathbb{E}_\pi$	Expectation when policy π is used.
$J^\pi(\cdot); R^\pi(\cdot)$	Expected cumulative revenue of policy π; Regret associated with policy π.

M	Mapping of profiles to clusters.
$K; \mathcal{K}$	Total number of clusters; Set of all clusters.
$F(\cdot)$	Probability distribution of purchasing decisions.
$DP(H_0, \alpha)$	Dirichlet Process with baseline prior distribution H_0 and precision parameter α.
$\mathcal{X}_t^i$	Assortment and purchase history associated with profile i at time t.
η	Number of MCMC iterations.
$c_i; c_i^*$	Cluster associated with profile i; Candidate cluster for profile i.
$n_{-i,c}$	Number of profiles, excluding profile i, that are mapped to cluster c.
$L(\cdot)$	Likelihood function.
$a(c_i^*, c_i)$	Acceptance probability of changing c_i to c_i^*.
f_i	Frequency proportion associated with mapping M_i.
$Beta(a_{j,t}, b_{j,t})$	Beta distribution with parameters $a_{j,t}$ and $b_{j,t}$.
$Q_{j,t}$	Index of product j at time t.
$\bar{\mu}_j$	Sample mean of the number of purchases for product j.
$k_j(t-1)$	Number of times that product j has been offered up to time $t-1$.
$Q_{j,t}$	Index of product j at time t.
R_{dc}	Regret of dynamic clustering policy.
$R_{d\text{-}int}$	Regret of data-intensive policy.
$R_{s\text{-}orc}$	Regret of semi-oracle policy.
R_{pool}	Regret of pooling policy.
$U_{s\text{-}orc}; U_{pool}$	Upper bound on the regret of semi-oracle policy; Upper bound on the regret of pooling policy.
$L_{d\text{-}int}$	Lower bound on the regret of data-intensive policy.
$G_1; G_2$	Gap function between the regret of data-intensive and semi-oracle policies; Gap function between the regret of data-intensive and pooling policies.

$G_{1l}; G_{2l}$	Approximate lower bound for the gap function G_1; Approximate lower bound for the gap function G_2.

Notation for Chapter 4

The notation $\tilde{y}$ denotes a threshold on the value of a parameter y.

n	Total number of products.
S	A subset (assortment) of products.
U_i^{NR}	Random utility obtained from purchasing product i in the case with no-returns.
β_i	Nominal mean utility of product i.
$p_i; p$	Price of product i; Vector of all product prices.
w_i	Nominal mean utility of product i minus its price.
μ_1	Degree of heterogeneity between products.
ϵ_i	Customer's pre-purchase uncertainty.
$\mathbb{P}_i^{NR}$	Purchase probability of product i in the case with no-returns.
U_i^{keep}	Random utility obtained from keeping product i.
U_i^{return}	Random utility obtained from returning product i.
c	Customer's hassle cost of return.
μ_2	Degree of heterogeneity between keeping and returning options.
ϵ_i^{keep}	Customer's post-purchase uncertainty for keeping product i.
ϵ_i^{return}	Customer's post-purchase uncertainty for returning product i.
$\mathbb{P}_i^{return}$	Return probability of product i.
U_i^R	Random utility obtained from purchasing product i in the case with returns.
A_i	Expected utility of purchasing product i.
$\mathbb{P}_i^R$	Purchase probability of product i in the case with returns.
k	Mean utility of the outside option in the case with returns.

$\pi_0; \pi_1$ — Seller's expected revenue in the case with no-returns; Seller's expected revenue in the case with returns.

$\pi_0^*; \pi_1^*$ — Seller's optimal expected revenue in the case with no-returns; Seller's optimal expected revenue in the case with returns.

$p_0^*; p_1^*$ — Optimal price in the case with no-returns; Optimal price in the case with returns.

Abbreviations

IID — Independent and Identically Distributed.

MIP — Mixed Integer Programming.

MNL — Multinomial Logit.

N-MNL — Nested Multinomial Logit.

MCMC — Markov Chain Monte Carlo.

MLE — Maximum Likelihood Estimation.

RMSE — Root Mean Squared Error.

Acknowledgements

I would like to express my deepest gratitude to my current and former advisors, Professors Fernando Bernstein, Denis Sauré, and Juan Pablo Vielma, for all their guidance, support, and encouragement throughout my Ph.D. studies. I would also like to thank the members of my dissertation committee, Professors Bora Keskin, Peng Sun, and Robert Swinney. Their suggestions have significantly improved this dissertation.

I am thankful to other faculty members at Duke, particularly, Professors Jeannette Song, Kevin Shang, Pranab Majumder, David Brown, and Jim Smith. I am also grateful to my dear friends, Professors Arian Aflaki, Şafak Yücel, and Asa Palley, who graduated from Duke. I am also thankful to my Ph.D. fellow friends at Fuqua, Soudipta Chakraborty, Chengyu Wu, Andrew Frazelle, Vinh Nguyen, Yunke Mai, Huseyin Gurkan, Levi DeValve, Zhenhuan Lei, Yuan-Mao Kao, Shuyu Chen, Yuan Guo, Yuexing Li, and Chen-An Lin.

Finally, I would like to thank my parents, Yousef and Maliheh, my twin brother, Sina, my sister, Zahra, and my grandparents, for all their unconditional love, wholehearted support, and great encouragement. Last but not least, I would like to thank my girlfriend, Forough, for her deep love, loving kindness, and tremendous support.

1

Introduction

Data-driven approaches to decision-making under uncertainty is at the center of many operational problems. These are problems in which there is an element of uncertainty that needs to be *estimated* (*learned*) from data in order to make dynamic operational decisions. For instance, consider shortest path problems in network applications where the arc costs (travel times) are stochastic due to network traffic (congestion). Another example in the context of retail operations is the dynamic assortment planning problem of a retailer facing customers with unknown demand. When possible, the uncertain model parameters are estimated *offline* by using historical data. However, offline estimation is not always possible as, for example, historical data (if available) might only provide partial information about uncertain model parameters. For example, a retailer offering a new assortment of products at the beginning of a new selling season has no (or limited) prior information about customer demand for such products. In such settings, the parameter estimation (e.g., demand estimation) must be conducted *online* by actively collecting data (e.g., customer transaction data), and dynamically revisited as more information becomes available. In doing so, one must balance the classic *exploration* (i.e., parameter estimation) versus *exploitation* (i.e., optimization) trade-off.

This dissertation adopts a data-driven active learning approach to study various operational problems under uncertainty with a focus on retail operations. The first two essays in this dissertation study the exploration versus exploitation trade-off from different perspectives. The first essay takes a theoretical approach and studies such trade-off in a combinatorial optimization setting. Our paper is the first to prove a lower bound on the asymptotic performance of any admissible policy. We also propose near-optimal policies that are implementable in real-time. The second essay studies the dynamic assortment personalization problem of an online retailer facing heterogeneous customers with unknown product preferences. We propose a prescriptive approach, called the *dynamic clustering* policy, for dynamic estimation of customer preferences and optimization of personalized assortments. Further focusing on retail operations, the final essay studies the interplay between a retailer's return and pricing policies and customers' purchasing decisions. We characterize the retailer's optimal prices in the cases with and without product returns and determine conditions under which offering the return option to customers increases the retailer's revenue. We also explore the impact of accounting for product returns on demand estimation.

In the first essay, *Learning in Combinatorial Optimization: What and How to Explore* (joint work with Denis Sauré and Juan Pablo Vielma), we study dynamic decision-making under uncertainty when, at each period, a decision-maker implements a solution to a combinatorial optimization problem. The objective coefficient vectors of said problem, which are unobserved prior to implementation, vary from period to period. These vectors, however, are known to be random draws from an initially unknown distribution with known range. By implementing different solutions, the decision-maker extracts information about the underlying distribution, but at the same time experiences the cost associated with said solutions. We show that resolving the implied exploration versus exploitation trade-off efficiently is related to solving a *Lower Bound Problem* (LBP), which simultaneously answers the questions of *what* to explore and *how* to do so. We establish a fundamental limit on the asymptotic performance of any admissible policy that is proportional to the optimal objective value of the LBP problem. We show that such a lower bound might be asymptotically attained by policies that adaptively reconstruct and solve LBP at an exponentially decreasing frequency. Because LBP is likely intractable in practice, we propose policies that instead reconstruct and solve a proxy for LBP. We provide strong evidence of the practical tractability of said proxy which implies that the proposed policies can be implemented in real-time. We test the performance of the proposed policies through extensive numerical experiments and show that they significantly outperform relevant benchmark in the long-term and are competitive in the short-term.

In the second essay, *A Dynamic Clustering Approach to Data-Driven Assortment Personalization* (joint work with Fernando Bernstein and Denis Sauré), we consider an online retailer facing heterogeneous customers with initially unknown product preferences. Customers are characterized by a diverse set of demographic and transactional attributes. The retailer can personalize the customers' assortment offerings based on available profile information to maximize cumulative revenue. To that end, the retailer must estimate customer preferences by observing transaction data. This, however, may require a considerable amount of data and time given the broad range of customer profiles and large number of products available. At the same time, the retailer can aggregate (pool) purchasing information among customers with similar product preferences to expedite the learning process. We propose a *dynamic clustering* policy that estimates customer preferences by adaptively adjusting customer segments (clusters of customers with similar preferences) as more transaction information becomes available. We test the proposed approach with a case study based on a dataset from a large Chilean retailer. The case study suggests that the benefits of the dynamic clustering policy under the MNL model can be substantial and result (on average) in more than 37% additional transactions compared to a *data-intensive* policy that treats customers independently and in more than 27% additional transactions compared to a *linear-utility* policy that assumes that product mean utilities are linear functions of available customer attributes. We support the insights derived from the numerical experiments by analytically characterizing settings in which pooling transaction information is beneficial for the retailer, in a simplified version of the problem. We also show that there are diminishing marginal returns to pooling information from an increasing number of customers.

In the final essay, *Uncertainty in Choice: The Role of Return Policies* (joint work with Fernando Bernstein and Gustavo Vulcano), we study the interplay between a retailer's return and pricing policies and customers' purchasing decisions. We model the customers' purchasing and returns decisions using a discrete choice model and study the seller's revenue maximization problem in the cases with and without product returns. For the single-product case, we characterize the seller's optimal price and show that the return option results in different (but not necessarily higher) optimal prices compared to the case with no returns. Moreover, we find that offering the return option is not always favorable for the retailer. In addition, we characterize conditions under which offering the return option to customers increases the retailer's revenue. Finally, we numerically study the impact of accounting for product returns on demand estimation. The preliminary numerical results based on a real dataset suggest that our model, which accounts for product returns, increases demand estimation accuracy compared to models that do not consider product returns in their estimation.

2

Learning in Combinatorial Optimization: What and How to Explore

2.1 Introduction

Motivation. Traditional solution approaches to many operational problems are based on combinatorial optimization problems and typically involve instantiating a deterministic mathematical program, whose solution is implemented repeatedly through time: nevertheless, in practice, instances are not usually known in advance. When possible, parameters characterizing said instances are estimated *off-line*, either by using historical data or from direct observation of the (idle) system. Unfortunately, off-line estimation is not always possible as, for example, historical data (if available) might only provide partial information pertaining previously implemented solutions. Consider, for instance, shortest path problems in network applications: repeated implementation of a given path might reveal cost information about arcs on such a path, but might provide no further information about costs of other arcs in the graph. Similar settings arise, for example, in other network applications (e.g., tomography and connectivity) in which feedback about cost follows from instantiating and solving combinatorial problems such as spanning and Steiner trees.

Alternatively, parameter estimation might be conducted *on-line* using feedback associated with implemented solutions, and revisited as more information about the system's primitives becomes available. In doing so, one must consider the interplay between the performance of a solution and the feedback generated from its implementation: some parameters might only be reconstructed by implementing solutions that perform poorly (relative to the optimal solution). This is an instance of the *exploration versus exploitation* trade-off which is at the center of many dynamic decision-making problems under uncertainty, and as such, it can be approached through the multi-armed bandit paradigm (Robbins, 1952). However, the combinatorial setting has salient features that distinguish it from the traditional bandit. In particular, the combinatorial structure induces correlation between the cost of different solutions, thus raising the question of *how* to collect (i.e., by implementing what solutions) and combine information for parameter estimation. Also, because of such correlation, the underlying combinatorial optimization problem might be invariant to changes in certain parameters, hence not all parameters might need to be estimated to solve said problem. Therefore, answering the question of *what* parameters to estimate is also crucial in the combinatorial setting.

Unfortunately, the features above either prevent or discourage the use of traditional bandit algorithms. First, in the combinatorial setting, traditional algorithms might not be implementable as they would typically require solving the underlying combinatorial problem at each period, for which, depending on the application, there might not be enough computational resources. Second, even with enough computational resources, such algorithms would typically call for implementing each feasible solution at least once, which in the settings of interest might take a prohibitively large amount of time (i.e., number of periods) and also result in poor performance.

Main Objectives and Assumptions. A thorough examination of the arguments behind results in the traditional bandit setting reveals that their basic principles are still applicable to the combinatorial setting. Thus, our objective can be seen as interpreting said principles and adapting them to the combinatorial setting with the goal of *developing efficient policies that are amenable to implementation*, and in the process, understanding how performance depends on the structure of the underlying combinatorial problem.

We consider a decision-maker that at each period must solve a combinatorial optimization problem with a linear objective function whose cost coefficients are random draws from a distribution that is identical in all periods and initially unknown (except for its range). We assume (without loss of generality) that the underlying combinatorial problem is that of cost minimization, and that the feasible region consists of a time-invariant nonempty collection of nonempty subsets (e.g., paths on a graph) of a discrete finite *ground set* (e.g., arcs of a graph), which is known upfront by the decision-maker. By implementing a solution, the decision-maker observes the cost realizations for the ground elements contained in said solution. Following the bulk of the bandit literature, we measure performance in terms of the cumulative *regret*, which is the additional expected cumulative cost incurred relative to that of an oracle with prior knowledge of the cost distribution.

Main Contributions. Our contributions are as follows:

1. **We establish a fundamental limit on asymptotic performance and show that it might be attained:** We prove that no policy can achieve an asymptotic (on N, which denotes the total number of periods) regret lower than $z^*_{LBP} \ln N$, where z^*_{LBP} is the optimal objective value of an instance-dependent optimization problem, which we call the *Lower Bound Problem* (LBP). This problem simultaneously answers the questions of *what* to explore and *how* to do so. More specifically, we show that in the combinatorial setting it suffices to

focus exploration on a subset of the ground set which we call the *critical set*. To the best of our knowledge, ours is the first lower bound for the stochastic combinatorial bandit setting. Then, we show that said lower bound might be asymptotically attained (up to a sub-logarithmic term) by near-optimal policies that adaptively reconstruct and solve LBP at an exponentially decreasing frequency.

2. **We develop an efficient policy amenable for real-time implementation:** The near-optimal policies alluded above reconstruct LBP adaptively over time. However, their implementation is impractical mainly because LBP depends non-trivially on the cost distribution (and thus, is hard to reconstruct), and because LBP is often an exponentially-sized problem that is unlikely to be timely solvable in practice. Nonetheless, we develop an implementable policy, which we call the OCP-based policy, by means of replacing LBP in the near-optimal policies by a proxy that distills the LBP's two main goals: the determination of what should be explored and how to do so. Said proxy, which we denote the *Optimality Cover Problem* (OCP), is a combinatorial optimization problem that is easier to reconstruct in practice as it depends solely on the vector of mean costs. While OCP is still an exponentially-sized problem, we provide strong evidence that it can be solved in practice. In particular, we show that OCP can be formulated as a Mixed-Integer Programming (MIP) problem that can be effectively tackled by state-of-the-art solvers, or via problem-specific heuristics. Finally, we show that a variant of the OCP-based policy admits an asymptotic performance guarantee that is similar to that of the near-optimal policy.

3. **We numerically show that the OCP-based policy significantly outperforms existing benchmark:** The key to the efficiency of the OCP-based

policy is that it explores as dictated by OCP (i.e., focusing exploration on the critical elements) and rarely explores every ground element, let alone every solution, of the combinatorial problem. Through extensive computational experiments we show that such a policy significantly outperforms relevant benchmark in the long-term and is competitive in the short-term settings, even when OCP is solved heuristically in a greedy way.

The optimal $\ln N$ scaling of the regret is well-known in the bandit literature (Lai and Robbins, 1985a) and can even be achieved in the combinatorial setting by traditional algorithms. The regret of such algorithms, however, is proportional to the number of solutions, which in combinatorial settings, is typically exponential. This suggests that the dependence on N might not be the major driver of performance in the combinatorial setting, especially in finite time. To this end, we aim at studying the optimal scaling of the regret with respect to the combinatorial aspects of the setting. In doing so, our performance bounds sacrifice the optimal dependence on N (by adding a sub-logarithmic term) for the sake of clarity in terms of their dependence on the underlying combinatorial aspects of the problem, thus facilitating their comparison to the fundamental performance limit. In this regard, our analysis shows that efficient exploration is achieved when exploration is focused on a critical set of elements of the ground set. Our results speak of a fundamental principle in active learning, which is somewhat obscured in the traditional bandit setting: that of only exploring what is necessary to reconstruct the optimal solution to the underlying problem, and doing so at the least possible cost.

The Remainder of the Paper. Section 2.2 reviews the related work. Section 2.3 formulates the problem and reviews ideas from the classic bandit setting. In Section 2.4 we establish a fundamental limit on performance and propose a near-optimal policy. Section 2.5 presents an efficient practical policy, amenable to implementation, whose performance is similar to that of the near-optimal policy. Section 2.6 discusses the computational aspects for solving OCP, and Section 2.7 illustrates the numerical experiments. Finally, Section 2.8 presents extensions and concluding remarks. All proofs and supporting materials are relegated to Appendix A.

2.2 Literature Review

Traditional Bandit Settings. Introduced in Thompson (1933) and Robbins (1952), the multi-armed bandit setting is a classical framework for studying dynamic decision-making under uncertainty. In its traditional formulation a gambler maximizes cumulative reward by pulling arms of a slot machine sequentially over time when limited prior information on reward distributions is available. The gambler faces the classical exploration versus exploitation trade-off: either pulling the arm thought to be the "best" (exploitation) at the risk of failing to actually identify such an arm, or trying other arms (exploration) which allows identifying the best arm but hampers reward maximization.

The seminal work of Gittins (1979) shows that, for the case of independent and discounted arm rewards, and infinite horizon, the optimal policy is of the index type. Unfortunately, index-based policies are not always optimal (see Berry and Fristedt (1985), and Whittle (1982)) or cannot be computed in closed-form. In their seminal work, Lai and Robbins (1985a) study asymptotically efficient policies for the undiscounted case. They establish a fundamental limit on achievable performance, which implies the (asymptotic) optimality of the order $\ln N$ (where N is the total number of periods) dependence in the regret (see Kulkarni and Lugosi (1997) for a finite-sample minimax version of the result). In the same setting, Auer et al. (2002) introduces the celebrated index-based UCB1 policy, which is both efficient and implementable.

Envisioning each feasible solution as an arm, the combinatorial bandit setting that we study corresponds to a bandit with correlated rewards (and many arms): only a few papers address this case (see e.g., Ryzhov and Powell (2009) and Ryzhov et al. (2012)). Alternatively, envisioning each ground element (e.g., arcs of a graph) as an arm, the combinatorial setting can be seen as a bandit with multiple simultaneous pulls: Anantharam et al. (1987) extend the fundamental bound of Lai and Robbins (1985*a*) to such a setting and propose efficient allocation rules; see also Agrawal et al. (1990). The setting we study imposes a special structure on the set of feasible simultaneous pulls, which prevents us from applying known results.

Bandit Problems with a Large Set of Arms. Bandit settings with a large number of arms have received significant attention in the last decade. In these settings, arms are typically endowed with some structure that is exploited to improve upon the performance of traditional bandit algorithms.

A first strain of (non-combinatorial) literature considers settings with a continuous set of arms, where exploring all arms is not feasible. Agrawal (1995) studies a multi-armed bandit in which arms represent points in the real line and their expected rewards are continuous functions of the arms. Mersereau et al. (2009) and Rusmevichientong and Tsitsiklis (2010) study bandits with possibly infinite number of arms when expected rewards are linear functions of an (unknown) scalar and a vector, respectively. Our paper also relates to the literature on linear bandit models (see e.g., Abernethy et al. (2008) and Dani et al. (2008)) as the model we study is a linear stochastic bandit with a finite (but combinatorial) number of arms. In a more general setting, Kleinberg et al. (2008) consider the case where arms form a metric space, and expected rewards satisfy a Lipschitz condition. See Bubeck et al. (2011) for a review of work in “continuum” bandits.

Bandit problems with some combinatorial structure have been studied in the context of assortment planning: in Rusmevichientong et al. (2010) and Sauré and Zeevi (2013), product assortments are implemented in sequence and (non-linear) rewards are driven by a choice model with initially unknown parameters. See Caro and Gallien (2007) for a similar formulation with linear rewards.

Gai et al. (2012) study combinatorial bandits when the underlying problem belongs to a restricted class, and extend the UCB1 policy to this setting. Their policy applies to the more general setting we study, and is used as a benchmark in our numerical experiments. They establish a performance guarantee that exhibits the right dependence on N, but is expressed in terms of a polynomial of the size of the ground set. We show that optimal performance dependence on the ground set is instead tied to the structure of the underlying combinatorial problem in a non-trivial manner.

Concurrent to our work, two papers examine the combinatorial setting: Chen et al. (2013) provide a tighter performance bound for the UCB1-type policy of Gai et al. (2012), which they extend to the combinatorial setting we study – their bound is still expressed as a polynomial of the size of the ground set; also, Liu et al. (2012) develop a policy for network optimization problems (their ideas can be adapted to the setting we study as well) but in a different feedback setting. Their policy collects information through implementation of solutions in a "barycentric spanner" of the solution set, which in the feedback setting of this paper could be set as a solution-cover: see further discussion in Appendix A.5. Probable performance of their policy might be arbitrarily worse than that of the OCP-based policy.

Drawing ideas from the literature of prediction with expert advice (see e.g., Cesa-Bianchi and Lugosi (2006)), Cesa-Bianchi and Lugosi (2012) study an adversarial combinatorial bandit where arms belong to a given finite set in $\mathbb{R}^d$ (see Auer et al. (2003) for a description of the adversarial bandit setting). Our focus instead is on *stochastic* (non-adversarial) settings. In this regard, our work leverages the additional structure imposed in the stochastic setting to develop efficient policies that are implementable in real-time.

2.3 Combinatorial Formulation versus Traditional Bandits

2.3.1 Problem Formulation

Model Primitives and Basic Assumptions. We consider a decision-maker that faces a combinatorial optimization problem with a linear objective function repeatedly over time. The feasible region of the combinatorial problem is time-invariant and consists of a nonempty collection $\mathcal{S}$ of nonempty subsets (e.g., paths on a graph) of a discrete finite *ground set* A (e.g., arcs in a graph). We assume that both A and $\mathcal{S}$ are known upfront by the decision-maker, and without loss of generality, that the problem is that of cost minimization.

The cost coefficient vector at each period is a vector of independent random variables. (We assume that all random variables are defined in a probability space $(\Omega, \mathcal{F}, \mathbb{P})$.) Furthermore, these random variables jointly form a sequence of i.i.d. random vectors across periods. We let $B_n(a)$ denote the random cost coefficient associated with element $a \in A$ in period $n \geqslant 1$, and define $B_n := (B_n(a) : a \in A)$ as the random cost coefficient vector in period n. (Throughout the paper, we use the notation $x(a)$ to refer to the a-th element of vector x.) Let F denote the (common) distribution of the cost coefficient vectors and $B \sim F$ with $B := (B(a) : a \in A)$ so that each B_n is an independent copy of B. We assume that F is initially unknown (by the decision-maker) except for its range: it is known that $l(a) < B(a) < u(a)$ a.s. for each $a \in A$ for given vectors $l := (l(a) : a \in A)$ and $u := (u(a) : a \in A)$ such that $l < u$ component-wise. (We also assume for simplicity that the marginal distributions of F are absolutely continuous with respect to the Lebesgue measure in $\mathbb{R}$.)

At the beginning of period n, the decision-maker selects and implements a solution $S_n \in \mathcal{S}$. Then, the random cost vector B_n is realized and the cost associated with solution S_n is incurred by the decision-maker. Finally, the decision-maker observes the realized cost coefficients *only* for those ground elements *included* in the solution implemented, i.e., the decision-maker observes $(b_n(a) : a \in S_n)$, where $b_n(a)$ denotes the realization of $B_n(a)$, $a \in A$, $n \geqslant 1$.

The decision-maker is interested in minimizing the total expected cost incurred in N periods (N is not necessarily known upfront). Let $\pi := (S_n)_{n=1}^{\infty}$ denote a non-anticipating policy, where $S_n : \Omega \to \mathcal{S}$ is an $\mathcal{F}_n$-measurable function that maps the available "history" at period n to a solution in $\mathcal{S}$, where $\mathcal{F}_n := \sigma(\{B_m(a) : a \in S_m\,,\, m < n\}) \subseteq \mathcal{F}$ for $n \geqslant 1$, with $\mathcal{F}_0 := \sigma(\varnothing)$. Finally, note that the expected cumulative cost associated with a policy π is given by

$$J^{\pi}(F, N) := \sum_{n=1}^{N} \mathbb{E}\left\{\sum_{a \in S_n} B(a)\right\}.$$

(Note that the right-hand-side above depends on the policy π through the sequence $(S_n)_{n=1}^{\infty}$).

Remark 1. In our formulation, B_n is independent of S_n. While this accommodates several applications such as shortest path, Steiner tree, and knapsack problems, it may not accommodate applications such as assortment selection problem with discrete choice models.

Full-Information Problem and Regret. Define $\mathcal{B} := \prod_{a \in A}(l(a), u(a))$. For a cost vector $\nu := (\nu(a) : a \in A) \in \mathcal{B}$, define the underlying combinatorial problem, denoted by $Comb(\nu)$, as follows:

$$z^{*}_{Comb}(\nu) := \min\left\{\sum_{a \in S} \nu(a) : S \in \mathcal{S}\right\}, \tag{2.1}$$

where $z^{*}_{Comb}(\nu)$ denotes the optimal objective value of (2.1). Let $\mathcal{S}^{*}(\nu)$ denote the set of optimal solutions to (2.1), and define $\mu(a) := \mathbb{E}\{B(a)\}$ for each $a \in A$ and $\mu := (\mu(a) : a \in A)$.

Suppose for a moment that F is known upfront: it can be seen that always implementing an optimal solution to the combinatorial problem where the cost vector equals its expected value is the best among all non-anticipating policies. That is, because of the linearity of the objective function, a clairvoyant decision-maker with prior knowledge of F would implement $S_n \in \mathcal{S}^*(\mu)$ for all $n \geqslant 1$, thus incurring an expected cumulative cost of

$$J^*(F,N) := N\, z^*_{Comb}(\mu).$$

(Note that the right-hand-side above depends on F through μ.)

In practice, the decision-maker does not know F upfront, hence no admissible policy incurs an expected cumulative cost below that incurred by the clairvoyant decision-maker. Thus, we measure the performance of a policy π in terms of its expected *regret*, which for given F and N is defined as

$$R^\pi(F,N) := J^\pi(F,N) - J^*(F,N).$$

The regret represents the additional expected cumulative cost incurred by a policy π relative to that incurred by a clairvoyant decision-maker that knows F upfront (note that regret is always non-negative).

Remark 2. Although the regret also depends on the combinatorial optimization problem through $\mathcal{S}$, we omit such dependence to simplify the notation.

2.3.2 Known Results and Incorporating Combinatorial Aspects

Traditional multi-armed bandits correspond to settings where $\mathcal{S}$ is formed by ex-ante *identical* singleton subsets of A, i.e., settings where $\mathcal{S} = \{\{a\} : a \in A\}$, and all marginal distributions of F are identical, thus the combinatorial structure is absent. In such settings, and under mild assumptions, the seminal work of Lai and Robbins (1985a) establishes an asymptotic lower bound on the regret attainable by any *consistent* policy (see Definition 25 in Appendix A.1.1). Different policies, such as the celebrated index-based UCB1 algorithm (Auer et al., 2002), have been shown to (nearly) attain such asymptotic performance limit. Combining the result by Lai and Robbins (1985a) and the performance guarantee for the UCB1 policy, we have that

$$\sum_{a \in A} (\mu(a) - \mu^*) K(a) \leqslant \liminf_{N \to \infty} \frac{R^{UCB1}(F, N)}{\ln N} \leqslant \sum_{a \in A : \mu(a) > \mu^*} \frac{8}{\mu(a) - \mu^*},$$

where $\mu^* := \min \{\mu(a) : a \in A\}$, and $K(a)$ denotes the inverse of the Kullback-Leibler divergence between F and an alternative distribution F_a under which $\mu^* = \mu(a)$. Lai and Robbins (1985a) show that consistent policies must explore (pull) each element (arm) in A at least on order $\ln N$ times. Thus, balancing the exploration versus exploitation trade-off in the traditional setting narrows down to answering how frequently to explore each element $a \in A$. (The answer to this question is given by $\ln N/N$ exploration frequency in Lai and Robbins (1985a)).

Note that the combinatorial setting can be seen as a traditional bandit with a combinatorial number of arms, where arm rewards are correlated. Thus, one might attempt to apply off-the-shelf index-based policies such as UCB1 envisioning each solution $S \in \mathcal{S}$ as an arm. However, this approach has two important disadvantages in our setting (consider that $|\mathcal{S}|$ is normally exponential in $|A|$): (i) computing an index for every solution in $\mathcal{S}$ is comparable to solving the underlying combinatorial problem by enumeration which, in most settings of interest, is impractical; and (ii) because traditional policies assume that all solutions are upfront identical, they have to periodically explore every solution in $\mathcal{S}$ with a frequency proportional to $\ln N/N$. However, because of the correlation between the solutions, this might no longer be necessary in the combinatorial setting.

To illustrate the issues above, consider two examples in which, for simplicity of

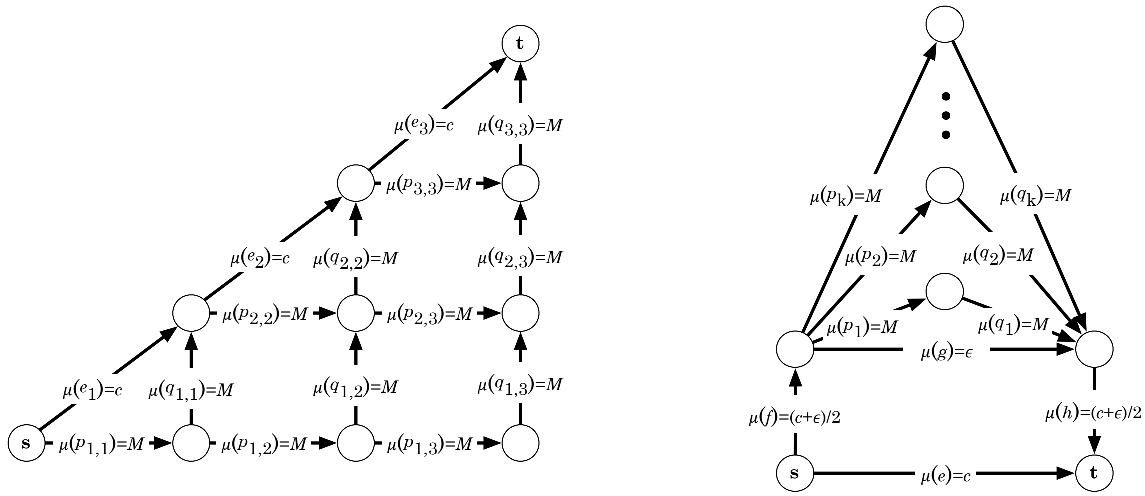

FIGURE 2.1: Graph for Example 1 (left), and Example 2 (right).

exposition, we ignore the exploration frequencies. That is, we assume that whatever elements in A are selected for exploration, they are selected persistently over time (irrespective of how), so that their mean cost estimates are accurate.

Example 1. *Consider the digraph $G = (V, A)$ for $V = \{v_{i,j} : i, j \in \{1, \ldots, k+1\}, i \leqslant j\}$ and $A = \{e_i\}_{i=1}^{k} \cup \{p_{i,j} : i \leqslant j \leqslant k\} \cup \{q_{i,j} : i \leqslant j \leqslant k\}$ where $e_i = (v_{i,i}, v_{i+1,i+1})$, $p_{i,j} = (v_{i,j}, v_{i,j+1})$, and $q_{i,j} = (v_{i,j}, v_{i+1,j})$. This digraph is depicted in the left panel of Figure 2.1 for $k = 3$. Let $\mathcal{S}$ be composed of all paths from node $s := v_{1,1}$ to node $t := v_{k+1,k+1}$.*

Consider constants $0 < \epsilon < c \ll M$ and let the distribution F be such that $\mu(e_i) = c$, $\mu(p_{i,j}) = \mu(q_{i,j}) = M$, for all $i \in \{1, \ldots, k\}$, $i \leqslant j \leqslant k$, $n \in \mathbb{N}$, and $l(a) = \epsilon$ and $u(a) = \infty$ for every arc $a \in A$. The shortest (expected) path is $S^* = \{e_1, e_2, \ldots, e_k\}$ with expected length (cost) $z^*_{Comb}(\mu) = kc$, $|A| = k(k+2)$, and $|\mathcal{S}|$ corresponds to the number of $s-t$ paths, which is equal to $\frac{1}{k+2}\binom{2(k+1)}{(k+1)} \sim \frac{4^{k+1}}{(k+1)^{3/2}\sqrt{\pi}}$ (Stanley, 1999).

A traditional bandit policy would need to explore all $\frac{1}{k+2}\binom{2(k+1)}{(k+1)}$ paths. However, the same exploration goal can be achieved while leveraging the combinatorial structure of the solution set to expedite estimation: a key observation is that one might conduct mean cost estimation for elements in the ground set, and then aggregate those to produce cost estimates for all solutions. A natural way of incorporating this observation is to explore a *minimal solution-cover* $\mathcal{E}$ of A (i.e., $\mathcal{E} \subseteq \mathcal{S}$ such that each $a \in A$ belongs to at least one $S \in \mathcal{E}$ and $\mathcal{E}$ is minimal with respect to inclusion for this property). In Example 1 we can easily construct a solution-cover $\mathcal{E}$ of size $k+1$, which is significantly smaller than $|\mathcal{S}|$.

An additional improvement follows from exploiting the ideas in the lower bound result in Lai and Robbins (1985a). To see this, note that, unlike in the traditional setting, solutions are not ex-ante identical in the combinatorial case. This opens up the possibility that information collection on some ground elements might be stopped after a finite number of periods, independent of N, without affecting asymptotic efficiency. This is illustrated in the following example.

Example 2. *Let $G = (V, A)$ be the digraph depicted in the right panel of Figure 2.1 and let $\mathcal{S}$ be composed of all paths from node s to node t. Set $l(a) = 0$ and $u(a) = \infty$ for every arc $a \in A$, and let the distribution F be such that $\mu(e) = c$, $\mu(g) = \epsilon$, $\mu(f) = \mu(h) = \frac{c+\epsilon}{2}$, $\mu(p_i) = \mu(q_i) = M$ for $n \in \mathbb{N}$ and for all $i \in \{1, \ldots, k\}$ where $0 < \epsilon \ll c \ll M$. The shortest (expected) path in this digraph is $\{e\}$.*

In Example 2, $|\mathcal{S}| = (k+2)$, and the only solution-cover of A is $\mathcal{E} = \mathcal{S}$, which does not provide an advantage over traditional approaches. However, a cover is required only if we need to explore every element in A. Indeed, feedback obtained through exploration only needs to guarantee the optimality of path $\{e\}$ with respect to all *plausible* scenarios. However, because the combinatorial problem is that of cost minimization, it suffices to check only *one* possibility: that in which every unexplored element $a \in A$ has an expected cost equal to its lowest possible value $l(a)$. In Example 2 we note that every path other than $\{e\}$ uses arcs f and h and the sum of the expected costs of f and h is strictly larger than that of e. Together with the fact that the cost of every arc has a lower bound of zero, this implies that exploring arcs f and h is sufficient to guarantee the optimality of $\{e\}$. We can explore arcs f and h by implementing any path that contains them, but the cheapest way to do so is by implementing path $\{f, g, h\}$.

Examples 1 and 2 show that in the combinatorial setting efficient policies do not need to explore every solution in $\mathcal{S}$ or even every ground element in A. In particular, Example 2 shows that the questions of *what* elements of A to explore (e.g., arcs f and h) and *how* to explore these elements (e.g., through path $\{f, g, h\}$) become crucial to construct efficient policies in the combinatorial setting. However, we still need to answer the question of *when* to explore, or more precisely, what the relative exploration frequencies must be. To achieve this, we extend the fundamental performance limit of Lai and Robbins (1985a) from the traditional multi-armed bandits to the combinatorial setting.

2.4 Bounds on Achievable Asymptotic Performance

Following the arguments in the traditional bandit setting, consistent policies must explore those subsets of suboptimal ground elements that have a chance of becoming part of any optimal solution, i.e., those subsets for which there exists an alternative cost distribution F' such that said subset belongs to each optimal solution in $\mathcal{S}^*(\mu')$, where μ' denotes the vector of mean costs under distribution F'. Because the range of F is known, for a given set $D \subseteq A$, it is only necessary to check whether D belongs to each optimal solution in $\mathcal{S}^*\left((\mu \wedge l)(D)\right)$, where

$$(\mu \wedge l)\,(D) := \left(\mu(a)\mathbf{1}\left\{a \notin D\right\} + l(a)\mathbf{1}\left\{a \in D\right\} : a \in A\right),$$

and $\mathbf{1}\{\}$ denotes the indicator function of a set. We let $\mathcal{D}$ denote the collection of all nonempty subsets of suboptimal ground elements satisfying the condition alluded above, that are minimal with respect to inclusion. We have that $D \in \mathcal{D}$ if and only if

1. $D \subseteq A$ and $D \neq \varnothing$,
2. $D \cap S^* = \varnothing$ for all $S^* \in \mathcal{S}^*(\mu)$,
3. $D \subseteq S$ for all $S \in \mathcal{S}^*\left((\mu \wedge l)(D)\right)$,
4. There is no subset $D' \subset D$ for which conditions 1-3 hold.

In other words, we take a pessimistic approach and define $\mathcal{D}$ as the collection of nonempty subsets of suboptimal ground elements that become part of any optimal solution if their mean costs are set to their lowest possible values.

As an illustration, consider the examples in the previous section. In Example 1 we have that

$$\mathcal{D} = \{\{p_{1,1}, q_{1,1}\}, \{p_{2,2}, q_{2,2}\}, \{p_{3,3}, q_{3,3}\}, \{p_{1,1}, p_{1,2}, q_{1,2}, q_{2,2}\}, \{p_{2,2}, p_{2,3}, q_{2,3}, q_{3,3}\}, \\ \{p_{1,1}, p_{1,2}, p_{1,3}, q_{1,3}, q_{2,3}, q_{3,3}\}, \{p_{1,1}, p_{1,2}, q_{1,2}, p_{2,3}, q_{2,3}, q_{3,3}\}\}$$

and in Example 2 we have that $\mathcal{D} = \{\{f\}, \{h\}\}$.

We conclude that for any $D \in \mathcal{D}$, there exists an alternative distribution F', whose mean cost vector is given by $(\mu \wedge l)(D)$, under which D is included in every optimal solution. Because said elements are suboptimal under distribution F (condition 2 above), consistent policies must distinguish F from F' to attain asymptotic optimality. The following proposition, whose proof can be found in Appendix A.1.1, shows that this can be accomplished by selecting *at least* one element in each set $D \in \mathcal{D}$ at a minimum frequency. For $n \geqslant 1$ and $a \in A$, define the random variable $T_n(a)$ as the number of times that the decision-maker has selected ground element a prior to period n, that is

$$T_n(a) := \left|\{m < n \,:\, a \in S_m\}\right|.$$

Proposition 3. *For any consistent policy π and $D \in \mathcal{D}$ we have that*

$$\lim_{N \to \infty} \mathbb{P}_F \left\{ \frac{\max \{T_{N+1}(a) : a \in D\}}{\ln N} \geqslant K_D \right\} = 1, \tag{2.2}$$

for a positive finite constant K_D.

Similar to the traditional bandit setting, K_D represents the inverse of the Kullback-Leibler divergence between F and the alternative distribution F' alluded above.

Proposition 3 characterizes *what* needs to be explored by a consistent policy by imposing a lower bound on the number of times that certain subsets of A ought to be explored. To obtain a valid performance bound, we additionally need to characterize *how* to explore these subsets in the most efficient way. In particular, in addition to selecting the set of ground elements that need to be explored, a consistent policy needs to implement solutions in $\mathcal{S}$ that include those ground elements in the most efficient manner. To assess the regret associated with implementing a solution $S \in \mathcal{S}$ given a cost vector $\nu \in \mathcal{B}$, we define

$$\Delta_S^\nu := \sum_{a \in S} \nu(a) - z^*_{Comb}(\nu).$$

The following *Lower Bound Problem* (henceforth, *LBP*) jointly determines the set of ground elements needed to be explored, a set of solutions that cover this set of ground elements, and their exploration frequencies. Furthermore, it does so in the most efficient way possible (i.e., by solving for the minimum-regret solution-cover).

Definition 4 (*LBP*)**.** *Define the lower bound problem LBP as*

$$\begin{aligned} z^*_{LBP} := \min \quad & \sum_{S \in \mathcal{S}} \Delta_S^\mu \, y(S) & \text{(2.3a)} \\ s.t. \quad & \max \{x(a) : a \in D\} \geqslant K_D, \quad D \in \mathcal{D} & \text{(2.3b)} \\ & x(a) \leqslant \sum_{S \in \mathcal{S} : a \in S} y(S), \quad a \in A & \text{(2.3c)} \\ & x(a),\, y(S) \in \mathbb{R}_+, \quad a \in A, S \in \mathcal{S}, & \text{(2.3d)} \end{aligned}$$

where z^*_{LBP} denotes the optimal objective value of the LBP problem.

Consider a solution (x, y) to the LBP problem where $x = (x(a) : a \in A)$ and $y = (y(S) : S \in \mathcal{S})$. The set $\{a \in A : x(a) > 0\}$ corresponds to the elements of the ground set that are explored to satisfy Proposition 3 and the actual values $x(a)$ represent the exploration frequencies $T_{N+1}(a)/N$. Similarly, the set $\{S \in \mathcal{S} : y(S) > 0\}$ corresponds to the *solution-cover* (which we also call the *exploration set*) of the selected ground elements, and the values $y(S)$ represent the exploration frequencies of the solutions in the cover. Indeed, constraints (2.3b) enforce exploration conditions (2.2) and constraints (2.3c) enforce the cover of the elements of A selected by (2.3b). The next result establishes a lower bound on the asymptotic regret of any consistent policy in the combinatorial setting which is proportional to $z^*_{LBP} \ln N$.

Theorem 5. *The regret of any consistent policy π is such that*

$$\liminf_{N \to \infty} \frac{R^\pi(F, N)}{\ln N} \geqslant z^*_{LBP}. \tag{2.4}$$

From Theorem 5 we see that the fundamental limit on performance is deeply connected to both the combinatorial structure of the problem, as well as the range and mean of distribution F.

Remark 3. A value of zero for z^*_{LBP} suggests that the regret may not necessarily grow as a function of N. To see how this indeed can be the case, consider the setting in Example 2 with a slight modification: set now $l(f) = l(h) = c/2 + \epsilon/4$. One can check that in this case, $\mathcal{D} = \varnothing$ as any suboptimal solution includes arcs f and h, whose cost lower bounds already ensure the optimality of solution $\{e\}$. Thus, in this case, $z^*_{LBP} = 0$ and a finite regret (independent of N) might be attainable. Indeed, this setting is such that active learning is not necessary, and information from implementing optimal solutions in $\mathcal{S}^*(\mu)$ suffices to guarantee the optimality of said solutions.

(This is not restricted to the case of shortest path problems: in Appendix A.1.2 we discuss settings in which $z^*_{LBP} = 0$ and the underlying combinatorial problem is minimum-cost spanning tree, minimum-cost perfect matching, generalized Steiner tree or knapsack.)

An Asymptotically Near-Optimal Policy. While our focus is on the finite-time performance, we note that it is possible to design policies that (nearly) attain the performance limit in Theorem 5. We here succinctly describe the intuition behind one such policy, which we call the *LBP-based* policy, and state the associated performance guarantee. We refer the reader to Appendix A.1.3 for more details on the policy and the proof of the performance guarantee.

The LBP-based policy is based on the observation that we can reconstruct the LBP problem by sporadically exploring all ground elements and that this can be done efficiently by implementing the solution to an additional optimization problem which we call the $Cover$ problem. The LBP-based policy, which we denote by $\pi^*(\gamma, \varepsilon)$ (here γ and ε are tuning parameters), solves both the $Cover$ and LBP problems (at an exponentially decreasing frequency, and using the sample mean costs) and has the following performance guarantee.

Theorem 6. *Consider $\gamma \in (0, 1)$ and $\varepsilon > 0$ arbitrary. The LBP-based policy $\pi^*(\gamma, \varepsilon)$ is such that*

$$\lim_{N\to\infty} \frac{R^{\pi^*(\gamma,\varepsilon)}(F, N)}{(\ln N)^{1+\varepsilon}} \leqslant z^*_{LBP} + \gamma\, z^*_{Cover}, \tag{2.5}$$

where z^*_{Cover} denotes the optimal objective value of the Cover problem

We observe that the accompanying constants in the lower bound and upper bound results in Theorems 5 and 6 do not match exactly. In Appendix A.1.5 we provide a detailed discussion of this gap.

2.5 An Efficient Practical Policy

A significant obstacle for the implementation of the LBP-based policy is the ability to reconstruct and solve formulation (2.3) repeatedly over time. Indeed, the right-hand-side of (2.3b) depends non-trivially on the distribution F, and while LBP is a continuous optimization problem, it has an exponential number of constraints (2.3b) that do not have a clear separation procedure. In addition, the maximum in constraint (2.3b) is known to be notoriously difficult to handle (Toriello and Vielma, 2012). For this reason, we instead concentrate on developing practical policies inspired by the exploration principles behind Theorems 5 and 6. In particular, we propose a policy that follows closely the near-optimal policy of Theorem 6, but replaces formulation (2.3) by a proxy that: (i) depends on the distribution F only through the vector of mean costs (and thus is easier to reconstruct); and (ii) can be solved effectively with modern optimization techniques. To achieve this, we distill the core combinatorial aspects of the LBP problem by concentrating on answering the questions of *what* ground elements to explore and *how* to do so (i.e., through the implementation of which solutions), while somewhat ignoring the question of when to explore (e.g., the precise exploration frequencies).

The Optimality Cover Problem. With regard to the first question above (what to explore), from Proposition 3 we know that consistent policies must try at least one element in each $D \in \mathcal{D}$ at a *specific* minimum frequency, so as to distinguish F from an alternative distribution that makes D part of any optimal solution. (Note that mean cost estimates for these elements should converge to their true values, and that ought to suffice to guarantee the optimality of the solutions in $\mathcal{S}^*(\mu)$.) Here, we consider an alternative, more direct mechanism which, in a nutshell, imposes the same exploration frequency on a set that contains at least one element from every set in $\mathcal{D}$.

Suppose that exploration is focused on a subset $C \subseteq A$ and that elements outside C would not be permanently sampled: in the long run, a consistent mean cost vector estimate $\nu \in \mathcal{B}$ will essentially be such that $\nu(a) \approx \mu(a)$ for $a \in C$, but not much can be said about $\nu(a)$ for $a \notin C$. If persistent exploration on the subset C is to guarantee the optimality of the solutions in $\mathcal{S}^*(\mu)$, independent of $(\mu(a) : a \notin C)$, then (taking a pessimistic approach) C must be such that

$$z^*_{Comb}(\mu) \leqslant z^*_{Comb}((\mu \wedge l)(A \backslash C)), \tag{2.6}$$

where we recall that $(\mu \wedge l)(A \backslash C) = (l(a)\mathbf{1}\{a \notin C\} + \mu(a)\mathbf{1}\{a \in C\} : a \in A)$. One can check that $D \cap C \neq \varnothing$ for all $D \in \mathcal{D}$ for such a subset C. This, in turn, implies that setting $x(a) = K$ for all $a \in C$, for a large enough positive constant K should lead to a feasible solution to LBP. This motivates the following definition.

Definition 7 (Critical Set)**.** *A subset $C \subseteq A$ is a* sufficient set *if and only if* (2.6) *holds. A sufficient set $C \subseteq A$ is a* critical set *if it does not contain any sufficient set $C' \subset C$.*

We may use condition (2.6) to simplify LBP by just enforcing the exploration of a critical set (what to explore). Once the critical set is identified, we can explore it efficiently (in terms of regret) by implementing a minimum-regret solution-cover (exploration set) of it (how to explore). Both the selection of the critical set and its minimum-regret solution-cover can be achieved simultaneously through the following combinatorial optimization problem.

Definition 8 (OCP)**.** *For a given cost vector $\nu \in \mathcal{B}$, we let the Optimality Cover*

Problem (henceforth, $OCP(\nu)$) be the optimization problem given by

$$z^*_{OCP}(\nu) := \min \quad \sum_{S \in \mathcal{S}} \Delta^{\nu}_S \; y(S) \tag{2.7a}$$

$$s.t. \quad x(a) \leqslant \sum_{S \in \mathcal{S}: a \in S} y(S), \quad a \in A \tag{2.7b}$$

$$\sum_{a \in S} \left(l(a)(1 - x(a)) + \nu(a)x(a) \right) \geqslant z^*_{Comb}(\nu), \quad S \in \mathcal{S} \tag{2.7c}$$

$$x(a),\, y(S) \in \{0, 1\}, \quad a \in A, S \in \mathcal{S}, \tag{2.7d}$$

where $z^*_{OCP}(\nu)$ denotes the optimal objective value of the $OCP(\nu)$ problem. Also, define $\Gamma_{OCP}(\nu)$ as the set of optimal solutions to $OCP(\nu)$.

By construction, a feasible solution (x, y) to $OCP(\mu)$ corresponds to incidence vectors of a critical set $C \subseteq A$ and a solution-cover $\mathcal{G}$ of such a set. That is, $(x, y) := \left(x^C, y^{\mathcal{G}}\right)$ where $x^C(a) = 1$ if $a \in C$ and zero otherwise, and $y^{\mathcal{G}}(S) = 1$ if $S \in \mathcal{G}$ and zero otherwise. In what follows we refer to a solution (x, y) to OCP and the induced pair of sets $(C, \mathcal{G})$ interchangeably.

Constraints (2.7c) guarantee the optimality of solutions in $\mathcal{S}^*(\nu)$ even if costs of elements outside C are set to their lowest possible values (i.e., $\nu(a) = l(a)$ for all $a \notin C$), and constraints (2.7b) guarantee that $\mathcal{G}$ covers C (i.e., $a \in S$ for some $S \in \mathcal{G}$, for all $a \in C$). Finally, (2.7a) ensures that the regret associated with implementing the solutions in $\mathcal{G}$ is minimized. Note that when solving (2.7), one can impose $y(S^*) = 1$ for all $S^* \in \mathcal{S}^*(\nu)$ without affecting the objective function, thus one can restrict attention to solutions that cover optimal elements of A.

There is a clear connection between LBP and OCP. This is formalized in the next Lemma, whose proof can be found in Appendix A.2.

Lemma 9. *An optimal solution to the linear relaxation of $OCP(\mu)$ is also optimal to formulation LBP when one replaces K_D by 1 for all $D \in \mathcal{D}$.*

Proof of Lemma 9 shows that a feasible solution to LBP can be mapped to a feasible solution to the linear relaxation of OCP (via proper augmentation), and vice versa. The above elucidates that OCP is a version of LBP that imposes equal exploration frequencies across all solutions. In this regard, the formulations are essentially equivalent up to a minor difference: optimal solutions to OCP *must* cover all optimal ground elements; this, however, can be done without affecting performance in both formulations and hence it is inconsequential. In what follows we discuss our practical policy which periodically solves the OCP problem.

OCP-based Policy. For $n \geqslant 1$, define $\widehat{\mu}_n := (\widehat{\mu}_n(a) : a \in A)$, where

$$\widehat{\mu}_n(a) := \frac{\sum_{m<n:a\in S_m} b_m(a)}{|m < n : a \in S_m|}, \quad a \in A,$$

denotes the sample mean of cost realizations for ground element a prior to period n. (Initial estimates are either collected from implementing a solution-cover or from expert knowledge.) The proposed policy, which we call the *OCP-based* policy, focuses exploration efforts at period n on the solutions in $\Gamma_{OCP}(\widehat{\mu}_n)$ at a logarithmic frequency (more precisely, by implementing solutions in $\mathcal{G}$ for some $(C, \mathcal{G}) \in \Gamma_{OCP}(\widehat{\mu}_n)$). To enforce the logarithmic exploration frequency, we use an idea known as the *doubling trick* (Cesa-Bianchi and Lugosi, 2006, Chapter 2.3). This approach also allows us to minimize the number of times that the $Comb$ and OCP problmes need to be solved. This trick divides the horizon into cycles of exponentially increasing lengths so that cycle i starts at time n_i where $n_1 = 1$ and $n_i := \max\left\{\lfloor e^{i/H} \rfloor, n_{i-1} + 1\right\}$, for all $i \geqslant 2$ and for a fixed tuning parameter $H > 0$.

At the beginning of each cycle, the OCP-based policy solves for $S^* \in \mathcal{S}^*(\widehat{\mu}_n)$, updates $\Gamma_{OCP}(\widehat{\mu}_n)$, and ensures that all elements in the critical set have been explored with sufficient frequency. If there is time remaining in the cycle, the policy implements (exploits) an optimal solution $S^* \in \mathcal{S}^*(\widehat{\mu}_n)$.

The OCP-based policy, which we denote by $\pi_{OCP}(H)$, is detailed in Algorithm 1. The main qualitative difference between the OCP-based policy and the near-optimal

Algorithm 1 OCP-based policy $\pi_{OCP}(H)$

```
Set i = 0, C = A, 𝒢 a minimal cover of A, and draw (b_1(a) : a ∈ A) randomly
from ℬ.
for n = 1 to N do
   if n = n_i then
      Set i = i + 1
      Set S* ∈ 𝒮*(μ̂_n)                                   [Update exploitation set]
      Set (C, 𝒢) ∈ Γ_OCP(μ̂_n)                             [Update OCP-exploration set]
   end if
   if T_n(a) < i for some a ∈ C then
      Set S_n = S for any S ∈ 𝒢 such that a ∈ S         [OCP-based exploration]
   else
      Set S_n = S*                                          [Exploitation]
   end if
end for
```

LBP-based policy is that: (i) it solves OCP problem instead of LBP; and (ii) it does not complement exploration on the solution to the $Cover$ problem (see Definition 29 in Appendix A.1.3 for further details).

Under mild conditions, we can prove an asymptotic performance guarantee for a variant of the OCP-based policy, denoted by π'_{OCP}, that differs from the LBP-based policy only in (i) above. The following result shows the asymptotic performance guarantee for the policy $\pi'_{OCP}(\gamma, \varepsilon, \varrho)$, where γ, ε, and ϱ are tuning parameters.

Theorem 10. *Consider $\gamma \in (0,1)$, $\varrho > 0$, and $\varepsilon > 0$ arbitrary. We then have that for ϱ sufficiently small*

$$\lim_{N\to\infty} \frac{R^{\pi'_{OCP}(\gamma,\varepsilon,\varrho)}(F,N)}{(\ln N)^{1+\varepsilon}} \leqslant z^*_{OCP}(\mu) + \gamma\, z^*_{Cover}.$$

The proof of Theorem 10 follows a similar line of arguments as that of Theorem 5 for the near-optimal LBP-based policy: we refer the reader to Appendix A.2.2 for the details of the policy π'_{OCP} and the proof of Theorem 10.

2.6 Computational Aspects for Solving OCP and Policy Implementation

In this section we address the computational aspects for the practical implementation of the OCP-based policy. We provide strong evidence that, for a large class of combinatorial problems, our policies scale reasonably well. For this, we focus our attention on the practical solvability of OCP, which our policies solve repeatedly over time. Note that $Comb(\cdot)$ and $OCP(\cdot)$ have generic combinatorial structures and hence are, a priori, theoretically hard to solve. Hence, practical tractability of said problems is essential for implementation.

Note that the OCP-based policy solves OCP at an exponentially decreasing frequency, thus ensuring its timely solvability in the long run. In the short run, a time-asynchronous version of the policy, that uses the incumbent solution to OCP until the new solution becomes available, can be implemented (see Appendix A.3.6).

As mentioned above, in general OCP might be theoretically intractable. Nonetheless, in Appendix A.3.7 we present a greedy oracle polynomial-time heuristic for OCP. The greedy heuristic requires a polynomial number of calls to an oracle for solving $Comb$. It therefore runs in polynomial time when $Comb$ is polynomially solvable. Furthermore, it provides a practical solution method for OCP when $Comb$ is not expected to be solvable in polynomial time, but is frequently tractable in practice (e.g., medium-size instances of NP-complete problems such as the traveling salesman (Applegate et al., 2011), Steiner tree (Magnanti and Wolsey, 1995; Koch and Martin, 1998; Carvajal et al., 2013), and set cover problems (Etcheberry, 1977; Hoffman and Padberg, 1993; Balas and Carrera, 1996)).

An advantage of the greedy heuristic described in Appendix A.3.7 is that it only requires an oracle for solving $Comb$ and hence does not require any knowledge of the specific structure of $Comb$. In Section 2.7 we implement a variant of the OCP-based policy that uses this greedy heuristic to solve OCP. We show that even such a myopic approach can already provide a reasonable approximation of the OCP-based policy and can significantly outperform alternative approaches. However, we would expect much better performance from heuristics or approximations that exploit the particular structure of $Comb$ for a specific class of problems. Such focus on a specific class of problems is, however, beyond the scope of this paper, thus we instead use mixed-integer programming (MIP) to exploit structure in a generic way.

Over 50 years of theoretical and computational developments in MIP (Jünger et al., 2010) have led to state-of-the-art MIP solvers with machine-independent speeds that nearly double every year (Achterberg and Wunderling, 2013; Bixby, 2012). One key to this speed is a wide range of highly effective generic primal heuristics (e.g., see Fischetti and Lodi (2011) and the "Primal Heuristic" sections of Gamrath et al. (2016); Maher et al. (2017), and Gleixner et al. (2017)). Hence, formulating OCP as a MIP opens up a wide range of exact and heuristic algorithms to solve it. However, the effectiveness of this approach is strongly contingent on constructing a formulation with favorable properties (Vielma, 2015). In what follows we focus our attention on constructing such formulations for the case in which $Comb$ is theoretically tractable, i.e., it is solvable in polynomial time. This class includes problems such as shortest path, network flow, matching, and spanning tree problems (Schrijver, 2003). For these problems we develop polynomial-sized MIP formulations of OCP, which can be effectively tackled by state-of-the-art solvers.

2.6.1 MIP Formulations of OCP for Polynomially-Solvable Problems

In this section we assume that $Comb$ is polynomially solvable. However, this does not imply that OCP is tractable or practically solvable, as it might contain an exponential (in $|A|$) number of variables and constraints. The following theorem, whose proof can be found in Appendix A.3.1, ensures that OCP remains in NP, the class of non-deterministic polynomially-solvable problems (see e.g., Cook et al. (1998)).

Theorem 11. *If Comb is in P, then OCP is in NP.*

While it is possible to establish a non-trivial jump in theoretical complexity for problems within P, we deem the study of the theoretical complexity of OCP for different problems outside the scope of the paper. Instead, here we focus on their practical solvability. For this, we first establish the existence of polynomial-sized MIP formulations when *Comb* admits a linear programming (LP) formulation. Then, we address the case when *Comb* admits a polynomial-sized extended LP formulation, and finally, the case when *Comb* does not admit such an extended formulation.

Problems with LP Formulations. We present a polynomial-sized formulation of OCP when *Comb* admits a polynomial-sized LP formulation. To describe this formulation in simple matrix notation we assume that $A := \{1, \ldots, |A|\}$. Moreover, for $v \in \mathbb{R}^r$, let $\operatorname{diag}(v)$ be the $r \times r$ diagonal matrix with v as its diagonal. Also, we remember that $l = (l(a) : a \in A)$ is the vector of lower bounds on the range of F.

Proposition 12. *Let $y^S \in \{0,1\}^{|A|}$ be the incidence vector of $S \in \mathcal{S}$, $M \in \mathbb{R}^{m \times |A|}$, and $d \in \mathbb{R}^m$ be such that $\left\{y^S\right\}_{S \in \mathcal{S}} = \left\{y \in \{0,1\}^{|A|} : My \leqslant d\right\}$ and* $\operatorname{conv}\left(\left\{y^S\right\}_{S \in \mathcal{S}}\right) =$

$\{y \in [0,1]^{|A|} : My \leqslant d\}$. *Then a MIP formulation of* $OCP(\nu)$ *is given by*

$$\min \quad \sum_{i\in A}\left(\sum_{a\in A}\nu(a)y^i(a) - z^*_{Comb}(\nu)\right) \tag{2.8a}$$

$$s.t. \qquad x(a) \leqslant \sum_{i\in A} y^i(a), \qquad a \in A \tag{2.8b}$$

$$My^i \leqslant d, \qquad i \in A \tag{2.8c}$$

$$M^T w \leqslant \mathrm{diag}(l)(\mathbf{1}-x) + \mathrm{diag}(\nu)x \tag{2.8d}$$

$$d^T w \geqslant z^*_{Comb}(\nu) \tag{2.8e}$$

$$x(a),\, y^i(a) \in \{0,1\}\,, w \in \mathbb{R}^m, \qquad a,\, i \in A, \tag{2.8f}$$

where $x = (x(a) : a \in A)$, $y^i = (y^i(a) : a \in A)$, *and* $\mathbf{1}$ *is a vector of ones.*

In the above, x represents the incidence vector of a critical set. Such a condition is imposed via LP duality, using constraints (2.8d) and (2.8e), and eliminates the necessity of introducing constraint (2.7c) for each solution in $\mathcal{S}$. Similarly, each y^i represents the incidence vector of a solution $S \in \mathcal{S}$. A formal proof of the validity of this formulation is included in Appendix A.3.3.

Formulation (2.8) has $O(|A|^2)$ variables and $O(m\,|A|)$ constraints. If m is polynomial in the size of the input of $Comb$, then we should be able to solve (2.8) directly with a state-of-the-art integer programming (IP) solver. If m is exponential, but the constraints in the LP formulation can be separated effectively, we should still be able to effectively deal with (2.8c) within a branch-and-cut algorithm. However, in such a case one would have an exponential number of w variables, which would force us to use a more intricate, and potentially less effective, branch-and-cut-and-price procedure. Nonetheless, when $Comb$ does not admit a polynomial-sized LP formulation, one can still provide formulations with a polynomial number of variables, many of them also having a polynomial number of constraints. We discuss such cases next.

Problems with Polynomial-Sized Extended Formulations. The first way to construct polynomial-sized IP formulations of OCP is to exploit the fact that many polynomially-solvable problems with LP formulations with an exponential number of constraints also have polynomial-sized *extended* LP formulations (i.e., formulations that use a polynomial number of auxiliary variables). A standard example of this class of problems is the spanning tree problem, where m in the LP formulation required by Proposition 12 is exponential in the number of nodes of the underlying graph. However, in the case of spanning trees, we can additionally use a known polynomial-sized extended formulation of the form $P := \{y \in [0,1]^{|A|} : \exists\, z \in \mathbb{R}^p, \quad Cy + Dz \leqslant d\}$ where $C \in \mathbb{R}^{m' \times |A|}$, $D \in \mathbb{R}^{m' \times p}$ and $d \in \mathbb{R}^{m'}$, with both m' and p being only cubic on the number of nodes (and hence polynomial in $|A|$) (Martin, 1991, e.g.). This formulation satisfies $\left\{y^S\right\}_{S \in \mathcal{S}} = P \cap \{0,1\}^{|A|}$ and $\operatorname{conv}\left(\left\{y^S\right\}_{S \in \mathcal{S}}\right) = P$. Then, a MIP formulation with a polynomial number of variables and constraints of OCP for the spanning tree problem is obtained by replacing (2.8c) with $Cy^i + Dz^i \leqslant d$, replacing (2.8d) with $C^T w \leqslant \operatorname{diag}(l)(\mathbf{1} - x) + \operatorname{diag}(\nu)x$ and $D^T w \leqslant 0$, and adding the polynomial number of variables z^i for $i \in \{1, \ldots, |A|\}$. Similar techniques can be used to construct polynomial-sized formulations for other problems with polynomial-sized extended LP formulations.

Problems without Polynomial-Sized Extended Formulations. It has recently been shown that there is no polynomial-sized extended LP formulations for the non-bipartite perfect matching problem (Rothvoß, 2017). Hence, we cannot use the techniques in the previous paragraph to construct polynomial-sized IP formulations of OCP for matching. Fortunately, a simple linear programming observation and a result by Ventura and Eisenbrand (2003) allow constructing a version of (2.8) with a polynomial number of variables. The observation is that a solution y^* is optimal for $\max\left\{\nu^T y : My \leqslant d\right\}$ if and only if it is optimal for

$\max\left\{\nu^T y \,:\, M_i^T y \leqslant d_i \quad \forall i \in I\left(y^*\right)\right\}$ where $I\left(y^*\right) \,:=\, \left\{i \in \{1,\ldots,m\} \,:\, M_i^T y^* = d_i\right\}$ is the set of active constraints at y^*, and M_i is the i-th row of M. The number of active constraints can still be exponential for matching. However, for each perfect matching y^*, Ventura and Eisenbrand (2003) give explicit $C \in \mathbb{R}^{m' \times |A|}$, $D \in \mathbb{R}^{m' \times p}$ and $d \in \mathbb{R}^{m'}$, such that m' and p are polynomial in $|A|$ and $\{y \in [0,1]^{|A|} \,:\, \exists\, z \in \mathbb{R}^p, \quad Cy + Dz \leqslant d\} = \{y \in \mathbb{R}^{|A|} \,:\, M_i^T y \leqslant d_i \quad \forall i \in I\left(y^*\right)\}$. Using these matrices and vectors we can then do a replacement of (2.8d) analog to that for spanning trees to obtain a version of (2.8) with a polynomial number of variables. We would still have an exponential number of constraints in (2.8c), but these can be separated in polynomial time for matching, so OCP for matching could be effectively solved by branch-and-cut.

Perfect matching is the only explicit polynomially-solvable combinatorial optimization problem that is known not to admit a polynomial-sized extended LP formulation. However, Rothvoß (2013) shows that there must exist a family of matroid problems without a polynomial-sized extended LP formulation. Fortunately, it can be shown (e.g., see Lemma 32 in Appendix A.3.2) that for matroid problems, there is a unique critical set $C \subseteq A$ that can be found in polynomial time. Once this set is obtained, we can simply replace (2.8b) by $1 \leqslant \sum_{i \in A} y^i(a)$ for all $a \in C$ and remove (2.8d)–(2.8e). We are not aware of any other polynomially-solvable combinatorial optimization problem which requires non-trivial results to formulate OCP with a polynomial number of variables.

Remark 4. Further improvements and extensions to (2.8) can be achieved. We give two such examples in Appendices A.3.4 and A.3.5. The first example shows how (2.8) for OCP can be extended to the case when $Comb$ is not in P, but admits a compact IP formulation. The second example gives a linear-sized formulation of OCP for shortest path problems.

2.7 Numerical Experiments

In this section we study the finite-time performance of the OCP-based policy from Section 2.5. In particular, we consider two policies: the OCP-based policy (as defined in Algorithm 1), and a variant that solves OCP heuristically in a greedy way (using Algorithm 5 presented in Appendix A.3.7). We refer to this latter policy as the *Greedy-Heuristic* policy. We divide the numerical experiments into two classes: long-term and short-term experiments. For each class of experiments, we first describe the benchmark policies and then discuss the settings studied.

2.7.1 Long-Term Experiments

Benchmark Policies and Implementation Details

Benchmark Policies. Our benchmark policies are versions of UCB1 (Auer et al., 2002), adapted to improve performance in the combinatorial setting. The UCB1 policy implements solution S_n in period n, where

$$S_n \in \underset{S \in \mathcal{S}}{\operatorname{argmin}} \left\{ \frac{\sum_{m<n:S_m=S} \sum_{a \in S} b_m(a)}{|m<n : S_m = S|} - \sqrt{\frac{2\ln(n-1)}{|m<n : S_m = S|}} \right\}.$$

Note that the estimate cost for solution S is based solely on past implementations of that solution. We improve performance of UCB1 by: (i) conducting parameter estimation at the ground element level to reduce variance of estimation; (ii) using $\min\{T_n(a) : a \in S\}$ instead of $|m<n : S_m = S|$ to adjust confidence interval length to better reflect the amount of information used in estimating parameters; (iii) adjusting said length so that confidence bounds remain within the bounds implied by the range of F; and (iv) reducing the solution set so that it only includes solutions that are minimal with respect to inclusion – this could improve performance by preventing UCB1 to implement solutions that are clearly suboptimal. The resulting

policy, which we denote UCB1+, implements solution S_n in period n, where

$$S_n \in \underset{S \in \mathcal{S}}{\operatorname{argmin}} \left\{ \max \left\{ \sum_{a \in S} \widehat{\mu}_n(a) - \sqrt{\frac{2 \ln(n-1)}{\min \{T_n(a) : a \in S\}}}, \sum_{a \in S} l(a) \right\} \right\}.$$

In a similar setting, Gai et al. (2012) propose an alternative adaptation of UCB1: a modified version of such a policy in period n implements

$$S_n \in \underset{S \in \mathcal{S}}{\operatorname{argmin}} \left\{ \sum_{a \in S} \max \left\{ \widehat{\mu}_n(a) - \sqrt{\frac{(\mathcal{K}+1) \ln(n-1)}{T_n(a)}}, l(a) \right\} \right\},$$

for a tuning parameter $\mathcal{K} > 0$. We denote this policy as Extended UCB1+.

Remark 5. Note that computing S_n in Extended UCB1+ can be accomplish by solving an instance of $Comb(\cdot)$. Implementing UCB1+ in contrast, requires solving for S_n via explicit enumeration.

Implementation Details. We report results when the marginal distributions of F are exponential (we normalize the mean cost vector so that the maximum solution cost is at most one): we tested many cost distributions and observed consistent performance. For the OCP-based and Greedy-Heuristic policies, we report the results for $H = 5$: preliminary tests using $H \in \{5, 10, 20\}$ always resulted in logarithmic regrets. When choosing a solution from the exploration set to implement, in case of a tie, our proposed policies select the solution that contains the most number of critical elements. In case of a second tie, they select a solution with the smallest average cost. We implemented UCB1+ and Extended UCB1+ with and without truncating indices at the implied lower bounds. Here, we present the point-wise minimum regret among both versions of each policy. We set $\mathcal{K} = 1$ in Extended UCB1+, as this selection outperformed the recommendation in Gai et al. (2012), and also is the natural choice for extending the UCB1 policy. Finally, all policies start with an initialization phase in which each solution in a common minimum-size cover of A is implemented.

All figures in this section report average performance for $N = 2000$ over 100 replications, and dotted lines represent 95% confidence intervals. All policies were implemented in MATLAB R2011b. Shortest path problems were solved using Dijkstra's algorithm except when implementing UCB1+ (note that because of the index computation, the optimization problem must be solved by enumeration). For Steiner tree and knapsack problems, we solved standard IP formulations using GUROBI 5.0 Optimizer. The OCP-based policy solves formulation (2.7) of OCP using GUROBI 5.0 Optimizer. All experiments ran on a machine with an Intel(R) Xeon(R) 2.80GHz CPU and 16GB of memory. The average running time for a single replication was around 30 seconds for the UCB1+, Extended UCB1+ and Greedy-Heuristic policies, and around 1.5 minutes for the OCP-based policy. (Note, however, that while the running time of the OCP-based policy grows (roughly) logarithmically with the horizon, those of UCB1+ and Extended UCB1+ grow linearly.)

Settings and Results

We present settings complementary to those in Examples 1 and 2 in the sense that critical sets are large, thus the OCP-based and Greedy-Heuristic policies do not have an immediate advantage. (See Appendix A.4 for numerical experiments on Examples 1 and 2.) The settings are comprised of the shortest path, Steiner tree and knapsack problems. We observed consistent performance of our policies across these settings: here we only show a representative from each class.

Shortest Path Problem. We consider a shortest path problem on a randomly generated layered graph – see panel (a) of Figure 2 in Ryzhov and Powell (2011) for an example of a layered graph. The graph consists of a source node, a destination node, and 5 layers in between, each containing 4 nodes. In each layer, every node (but those in the last layer) is connected to 3 randomly chosen nodes in the next layer. The source node is connected to every node in the first layer and every node in the last layer is connected to the destination node. Mean arc costs are selected randomly from the set $\{0.1, 0.2, \ldots, 1\}$ and then normalized. The representative graph is such that $|A| = 56$, $|\mathcal{S}| = 324$, and while the minimum-size cover of A is of size 13, the solution-cover to $OCP(\mu)$ is of size 16 with an implied critical set of size 40.

Knapsack Problem. Here the set A represents items that might go into the knapsack to maximize total utility. The solution set $\mathcal{S}$ consists of the subsets of items whose total weights do not exceed the knapsack weight limit. Weight and utility of items, as well as the weight limit, are selected randomly. The representative setting is such that $|A| = 20$, $|\mathcal{S}| = 24680$, the minimum-size cover is of size 4, and the solution-cover to $OCP(\mu)$ is of size 8 with an implied critical set of size 17.

Minimum Steiner Tree Problem. We consider a generalized version of the Steiner tree problem (Williamson and Shmoys, 2011), where for a given undirected graph with non-negative edge costs and a set of pairs of vertices, the objective is to find a minimum-cost subset of edges (tree) such that every given pair is connected in the set of selected edges. The graphs as well as the pairs of vertices are generated randomly, as well as the mean cost values. The representative setting is such that $|A| = 18$, $|\mathcal{S}| = 10651$, the minimum-size cover is of size 2. We consider two settings: one where the lower bound vector is $l = 0$ (the solution-cover to $OCP(\mu)$ is of size 7 and the critical set is of size 17) and one where lower bounds are positive numbers that are selected randomly (the solution-cover to $OCP(\mu)$ is of size 5 and the critical set is of size 12).

Results. The left and right panel in Figure 2.2 depict the average performance of different policies for the shortest path and knapsack settings, respectively. We see that in both settings, the OCP-based and Greedy-Heuristic policies significantly outperform the benchmark. The left panel in Figure 2.3 depicts the average performance

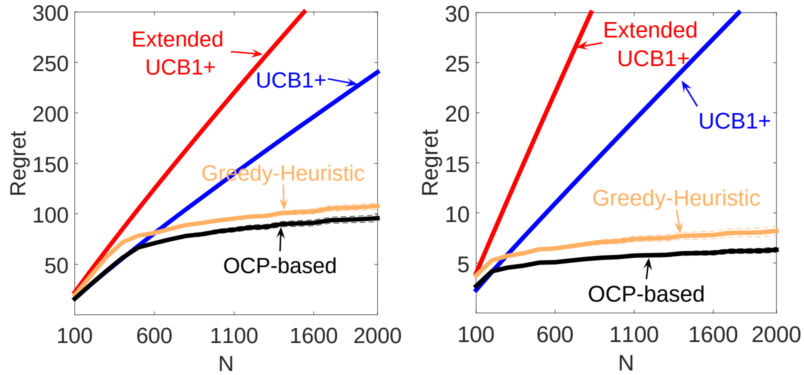

FIGURE 2.2: Average performance of different policies on the representative setting for the shortest path (left) and knapsack (right) problems.

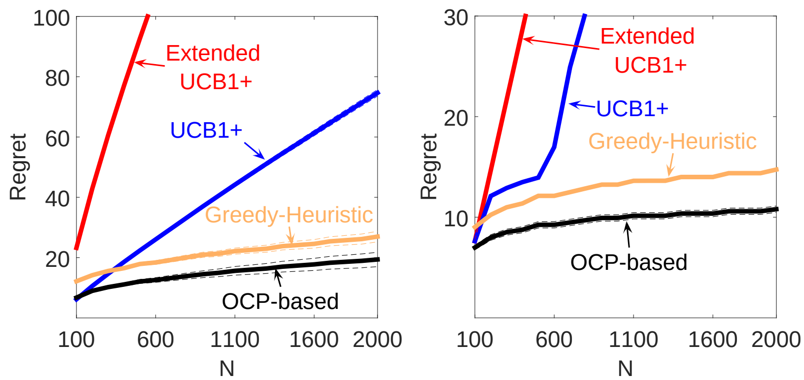

FIGURE 2.3: Average performance of different policies on the representative setting for the Steiner tree problem with zero (left) and positive (right) lower bounds.

of different policies for the Steiner tree setting when all cost lower bounds are set to zero. In this case, all arcs (but those trivially suboptimal) are critical, however, the OCP-based and Greedy-Heuristic policies still outperform the benchmark. The right panel in Figure 2.3 depicts average performance in the setting where lower bounds are positive numbers. Note that the OCP-based policy significantly outperforms the benchmark as it successfully limits exploration to a critical set. Also note that the non-concave behavior of the regret curve of UCB1+ arises only in the transient as a by-product of truncation, and it disappears at around $n = 1200$.

Sample Path Regret Comparison. The results in this section show that the average performance of our policies is significantly better than that for the benchmark. It turns out that our policies outperform the benchmark not only in terms of average, but also in terms of worst-case regret: we compared the sample path final regrets (i.e., at time period $N = 2000$) of OCP-based policy with those of UCB1+ and Extended UCB1+ policies: out of 700 sample paths in the numerical experiments in Section 2.7.1 (and including those in the Appendix A.4), the OCP-based policy outperforms the UCB1+ and Extended UCB1+ policies in all 700 (i.e., 100% of sample paths) and 697 (i.e., 99.6% of sample paths), respectively.

2.7.2 Experiment with Size of the Ground Set

In Section 2.1 we argued that, in the combinatorial setting, it is the constant accompanying the $\ln N$ term in a performance guarantee that is worth characterizing. However, prior work (see Section 2.2), lacking a fundamental performance limit, instead writes such an accompanying constant as a function of the size of the ground set (i.e., $|A|$). However, following Theorem 5, we know that such a constant is not a trivial function of $|A|$. Thus, the question of how said constant scales *in practice* with the size of the underlying combinatorial problem is of much relevance. For this reason, we next explore how performance of various policies varies with the size of the ground set.

We experiment with the shortest path problem on a layered graph (see Section 2.7.1 for a description) with $\mathfrak{L}$ layers, 2 nodes in each layer, complete connections between layers, and a direct arc from the source s to sink t. We experiment with $\mathfrak{L} = 2, 4, 6, 8, 10$ which results in $|A| = 9, 17, 25, 33, 41$ and $|\mathcal{S}| = 5, 17, 65, 257, 1025$, respectively.

We add a direct $s - t$ arc (path) to the original description of the layered graph so as to isolate the effect of size of the ground set on the performance of different policies. To this end, we let the expected cost of the $s - t$ arc (path) be 0.1, while all other arcs have an expected cost of $0.2/(\mathfrak{L}+1)$ where $\mathfrak{L}$ is the number of layers. Therefore, the $s-t$ path is the expected shortest path while all other paths (each of which has $\mathfrak{L}+1$ arcs) have an expected cost of 0.2, regardless of the size of the ground set. Thus, increasing the size of the ground set does not affect the cost (regret) of different paths in different instances. We run the experiments for $N = 20000$ and 40 replications.

For the OCP-based policy, we solve the OCP problem using the linear-sized formulation (A.22) presented in the Appendix A.3.5. We observe a behavior similar to the graph on the left panel of Figure 2.5 for all choices of $\mathfrak{L}$. That is, the cumulative regret of all three policies grow similar to a function $\mathfrak{K}\ln(n)$ for some policy-dependent constant $\mathfrak{K}$. We consider two estimates for such a constant: (i) K_{Final}, which we find by dividing the average final regret, which we denote by $\hat{R}(20000)$, by $\ln(20000)$, that is, $K_{Final} := \hat{R}(20000)/\ln(20000)$; (ii) K_{LS}, which is found by fitting the function $K_{LS}\ln(n)$ to the sample of average regrets for $n = 100, 200, \ldots, 20000$ (by minimizing the sum of squared errors). We present the value of both constants for the three policies and varying $|A|$ in Figure 2.4. We also present the average performance and computation time of different policies for the instance with $\mathfrak{L} = 10$ ($|A| = 41$ and $|S| = 1025$) as a representative setting in Figure 2.5 as we observed similar behavior in other instances. As can be seen in the left panel of Figure 2.5 (and also from Figure 2.4), the OCP-based policy significantly outperforms both benchmark policies regardless of the size of the ground set. Moreover, the constants K_{LS} and K_{Final} are significantly smaller for the OCP-based policy than those for the benchmark policies. In addition, such constants grow with a much smaller rate for the OCP-based policy than the benchmarks. Moreover, as illustrated by the right panel of Figure 2.5, the computation time of the OCP-based policy grows logarithmically with N. Furthermore, there is a significant variation if we consider computation times. This is shown in Table 2.1, which presents the average running time for a com-

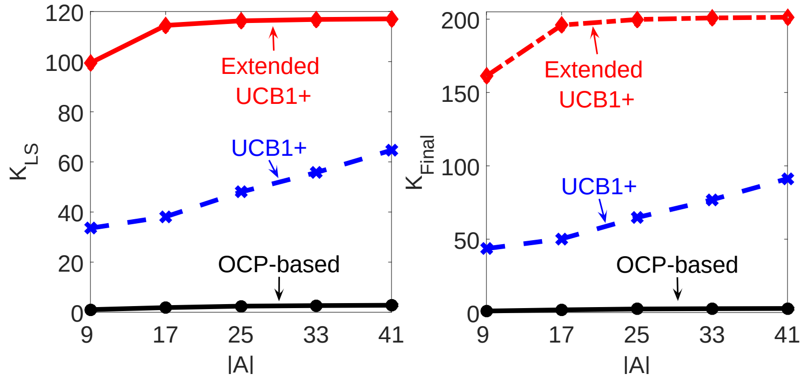

FIGURE 2.4: Constants K_{LS} (left) and K_{Final} (right) when increasing the size of the ground set.

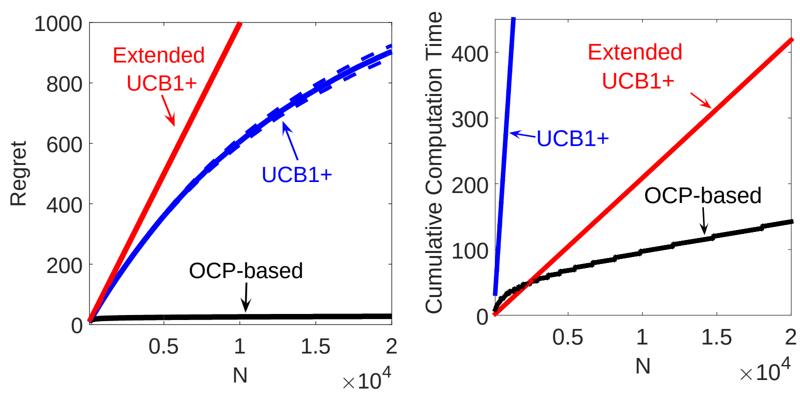

FIGURE 2.5: Average performance (left) and computation time (right) as a function of N for the instance with $\mathfrak{L} = 10$, $|A| = 41$, and $|S| = 1025$.

plete replication for each policy. This time includes all calculations required by the policy (e.g., for the OCP-based policy, it includes the solution time of all instances of OCP and $Comb$ as dictated by Algorithm 1). We can see that the OCP-based

Table 2.1: Average total computation time (in seconds) for each replication of $N = 20000$.

	$\|A\|$				
	9	17	25	33	41
OCP	75.54	79.43	81.18	92.60	142.38
UCB1+	65.47	127.38	376.56	1483.71	6686.70
Extended UCB1+	103.59	190.64	267.22	342.93	418.83

policy runs faster than both benchmark policies for (almost) all instances (we note that although for much larger instances, one expects the Extended UCB1+ to run faster than the OCP-based policy, the Extended UCB1+ performs very poorly, in terms of regret, regardless of the size of the instance). Moreover, UCB1+, which is the more "competitive" benchmark policy in terms of performance, is significantly slower than the OCP-based policy. These observations further pronounce the practical advantage of the OCP-based policy both in terms of performance (i.e., regret) and computation time.

2.7.3 Short-Term Experiments

Benchmark Policies and Implementation Details

Benchmark Policies. Our benchmark policies are adaptations of the Knowledge-Gradient (KG) policy in Ryzhov et al. (2012) and the Gittins index approximation in Lai (1987) to our setting. Both policies require prior knowledge of the time horizon N, and because of this, several runs of the benchmark policies are necessary to construct their cumulative regret curves.

The KG policy requires a prior distribution for the cost and hyper-parameters. In our implementation, we use the Exponential-Gamma conjugate prior for each ground element. That is, the algorithm assumes that for each $a \in A$, $B(a)$ follows an exponential distribution with rate $\mu(a)$, but this rate itself is random, and initially distributed according to a Gamma distribution with parameters $\alpha_{a,0}$ and $\beta_{a,0}$. At period n, the posterior distribution of $\mu(a)$ is a Gamma with parameters

$$\alpha_{a,n} = \alpha_{a,0} + T_n(a), \quad \beta_{a,n} = \beta_{a,0} + \sum_{m<n:a\in S_m} b_m(a), \quad a \in A.$$

Thus at period n, the KG algorithm implements solution S_n^{KG}, where

$$S_n^{KG} \in \operatorname*{argmin}_{S\in\mathcal{S}} \left\{ \sum_{a\in S} \frac{\beta_{a,n}}{\alpha_{a,n}-1} - (N-n)\, \mathbb{E}_S^n \left\{ \min_{S'\in\mathcal{S}} \left\{ \sum_{a\in S'} \frac{\beta_{a,n}}{\alpha_{a,n}-1} \right\} - \min_{S'\in\mathcal{S}} \left\{ \sum_{a\in S'} \frac{\beta_{a,n+1}}{\alpha_{a,n+1}-1} \right\} \right\} \right\},$$

where the expectation is taken with respect to B_n. The expectation above corresponds to the knowledge gradient term $v_S^{KG,n}$ in the notation of Ryzhov et al. (2012). Unlike in that paper, there is no closed-form expression for $v_S^{KG,n}$ in our setting. Our plain vanilla implementation of the KG algorithm computes such a term via Monte Carlo simulation, and performs the outer minimization via enumeration. The complexity of the implementation limited the size of the settings we tested.

The second benchmark is an approximation based on the Gittins index rule which in the finite-horizon undiscounted settings takes the form of an *average productivity* index (see Niño-Mora (2011)), and although it is not optimal in general, it is still applied heuristically. Our implementation assigns an index to each ground element, and computes the index of a solution as the sum of the indexes of the ground elements included in that solution. The policy requires a parametric representation of the uncertainty. To mimic a setting where the functional form of the cost distributions is unknown, we consider the approximation in Lai (1987) based on normally distributed costs and use Normal/Normal-Gamma conjugate priors (this is motivated by a central limit argument): in our approximation, the index of a ground element $a \in A$ at period n is given by

$$g_{n,N}^{a}(\mu_{a,n}, \lambda_{a,n}, \alpha_{a,n}, \beta_{a,n}) = \left(\mu_{a,n} - \sqrt{\frac{\beta_{a,n}}{(\alpha_{a,n}-1)\lambda_{a,n}}}\, h\left(\frac{\lambda_{a,n}-\lambda_{a,0}}{N-n+1+\lambda_{a,n}-\lambda_{a,0}}\right)\right)^{+},$$

where $\mu_{a,n}$ and $\lambda_{a,n}$ are the mean and variance of the normal posterior, respectively, $\alpha_{a,n}$ and $\beta_{a,n}$ are the hyper-parameters of the Gamma posterior, respectively, and $h(\cdot)$ approximates the boundary of an underlying optimal stopping problem. The policy implements solution S_n^{Gitt}, where

$$S_n^{Gitt} \in \underset{S \in \mathcal{S}}{\operatorname{argmin}} \left\{ \sum_{a \in S} g_{n,N}^{a}(\mu_{a,n}, \lambda_{a,n}, \alpha_{a,n}, \beta_{a,n}) \right\}.$$

Implementation Details. The implementation details are as in the long-term experiments. The average running time for a single replication ranged from around one second for the OCP-based policy to around 2 seconds for Gittins to less than 10 minutes for KG. We exclude the results for the UCB1+ and Extended UCB1+ policies, because they were consistently outperformed by the OCP-based policy.

Settings and Results

We consider randomly generated (structure and costs) settings of shortest path, Steiner tree and knapsack problems. We observed consistent performance of the policies across settings, and show only a representative setting for each class of problems. There, the total number of periods is selected so as to visualize the value at which the OCP-based policy begins outperforming the benchmark. In all settings, the benchmark policies initially provide a better performance compared to the OCP-based policy, but the latter policy eventually surpasses the benchmarks for moderate values of N. The same holds true for the case of the Greedy-Heuristic policy.

Shortest Path Problem. The left panel at Figure 2.6 depicts the average performances for a shortest path problem in a layered graph with 5 layers, each with 4 nodes, and 2 connections between each inner layer. The representative setting is such that $|A| = 40$, $|\mathcal{S}| = 64$, the minimum-size cover is of size 9, and the solution-cover to $OCP(\mu)$ is of size 10 with an implied critical set of size 23.

Minimum Steiner Tree Problem. The central panel at Figure 2.6 depicts the average performances on a representative setting for the Steiner tree problem. The representative setting is such that $|A| = 9$, $|\mathcal{S}| = 50$, the minimum-size cover is of size 2, and the solution-cover to $OCP(\mu)$ is of size 4 with an implied critical set of size 8.

Knapsack Problem. The right panel at Figure 2.6 depicts the average performances on a representative setting for the knapsack problem. (Here we report on the average behavior over 500 replications so that the confidence intervals do not cross.) The representative setting is such that $|A| = 11$, $|\mathcal{S}| = 50$, the minimum-size cover is of size 7, and the solution-cover to $OCP(\mu)$ is of size 2 with an implied critical set of size 5.

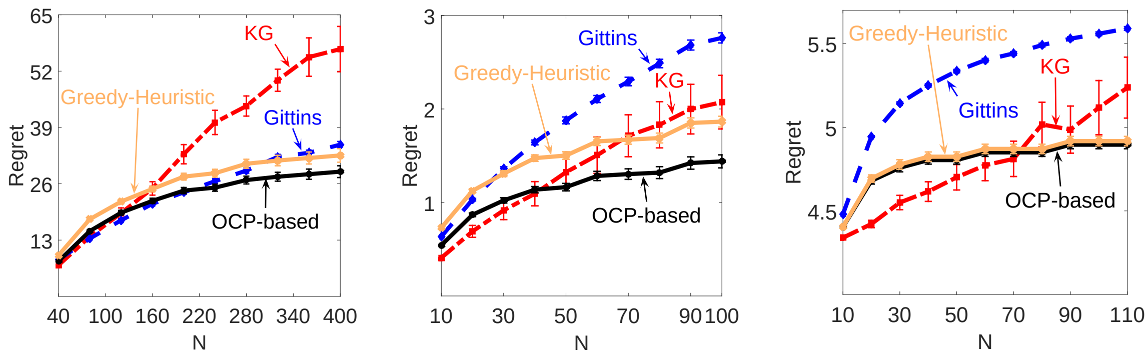

FIGURE 2.6: Average performance of different policies on the representative setting for the shortest path (left), Steiner tree (center) and knapsack (right) problems – the vertical lines show the 95% confidence intervals.

2.8 Conclusion

In this paper we study a class of sequential decision-making problems where the underlying single-period decision problem is a combinatorial optimization problem, and there is initial uncertainty about its objective coefficients. By framing the problem as a *combinatorial* multi-armed bandit, we adapt key ideas behind results in the classical bandit setting to develop efficient practical policies. We show that in addition to answering the question of *when* (with what frequency) to explore, which is key in the traditional bandit setting, in the combinatorial setting the key questions to answer are *what* and *how* to explore. We answer such questions by solving an optimization problem which we call the Lower Bound Problem (LBP). We establish a fundamental limit on the asymptotic performance of any admissible policy that is proportional to the optimal objective value of the LBP problem. We show that such a lower bound might be asymptotically attained by near-optimal policies that adaptively reconstruct and solve LBP at an exponentially decreasing frequency. Because LBP is likely intractable in practice, we propose a simpler and more practical policy, namely the OCP-based policy, that instead reconstructs and solves a proxy for LBP, which we call the Optimaltiy Cover Problem (OCP). This proxy explicitly solves for the cheapest optimality guarantee for the optimal solution to the underlying combinatorial problem. We prove a performance guarantee for a variant of the OCP-based policy, which is proportional to the optimal objective value of the OCP and can be compared to that of LBP. We also provide strong evidence of the practical tractability of OCP which in turn implies that the proposed OCP-based policies are scalable and implementable in real-time. Moreover, we test performance of the proposed policies through extensive numerical experiments and show that they significantly outperform relevant benchmark in the long-term and are competitive in the short-term.

The flexibility of the OCP-based policies allows them to be easily extended or combined with other techniques that consider similar what-and-how-to-explore questions. For instance, the OCP-based policy can be easily combined with the "barycentric spanner" of Awerbuch and Kleinberg (2004) to extend our results from element-level observations to set- or solution-level observations. Indeed, it can be shown that in such feedback settings, efficient exploration amounts to focus exploration on the solution to a variant of OCP. Moreover, the performance guarantee in Theorem 10 would remain valid with the constants associated with this alternative formulation. See Appendix A.5 for further details.

From our results, we observe a performance gap between the fundamental limit on (asymptotic) performance (Theorem 5) and the upper bound on the regret associated with near-optimal policies (Theorem 6). Although we provide a detailed discussion of this gap in Appendix A.1.5, future research can further investigate the possibility of closing this gap. Moreover, relaxing some of the assumptions in this work (e.g., considering unbounded cost distributions with heavy tails) could introduce additional interesting challenges. Furthermore, studying combinatorial bandit settings with non-linear objective functions is another direction for future research.

3

A Dynamic Clustering Approach to Data-Driven Assortment Personalization

3.1 Introduction

Motivation and Objective. According to a recent study by eMarketer (eMarketer 2017), worldwide business-to-consumer e-commerce sales will reach more than $4 trillion in 2020. With the rapid growth of online sales, retailers are finding unprecedented opportunities to both enhance the customer experience and increase revenue. A customer-centric practice that is becoming increasingly pervasive in online retailing is assortment personalization. This is the practice of offering a product mix that is tailored to each customer's taste based on previously collected data (Arora et al. 2008). The benefits of personalization are twofold: on one hand, it results in higher revenue for the retailer due to the increase in sales that results from providing customers with a set of products that more accurately matches their preferences (Arora et al. 2008); on the other hand, it attracts customer attention and fosters customer loyalty and satisfaction (Ansari and Mela 2003).

There are numerous examples of personalization in online retailing. For example, Amazon.com uses collaborative filtering to personalize recommendations to its users (Linden et al. 2003; Arora et al. 2008). A recent New York Times article reviews a growing number of start-ups that are investing in highly personalized online shopping technology (Wood 2014). Among these start-ups is Stitch Fix, an online personalized clothing service that periodically sends its customers boxes containing five pieces of clothing based on each customer's taste (e.g., size, favorite brand and color, budget). Trunk Club is another personalized clothing service.

As noted in the New York Times article, in personalized shopping "the magic comes from data." Online retailers collect an abundance of customer data (e.g., demographic, transactional, etc.). However, given the broad range of customer profiles, collecting a sufficient amount of transaction data on each customer profile may not be possible. This, in turn, limits the retailer's ability to accurately estimate preferences and offer personalized assortments. The goal of this paper is to explore the efficient use of data for assortment personalization in online retailing.

Model. We consider an online retailer that sells multiple products over a finite selling season. Customers arrive sequentially and the retailer offers each customer an assortment of products. The retailer may face display or capacity constraints that limit the number of products in the offered assortment. The customer then decides whether or not to make a purchase. The retailer's objective is to maximize expected cumulative revenue over the selling season.

Customers are divided into different profiles according to their observable attributes, such as demographic profiles (e.g., gender, age, location) and past transaction information (e.g., purchase history, payment method). This information is exogenous and available upon arrival via the customer's login information or internet cookies. We assume that, from the perspective of the retailer, customers with a common profile are homogeneous with respect to their product preferences (in practice, the definition of profiles reflects the degree of customer information available to the retailer and the level of granularity that allows the retailer to operationalize other marketing decisions).

A central assumption in this work is that the retailer has limited prior information on customers' preferences. Personalizing assortments thus requires the estimation of such preferences which, in turn, requires observing the customers' purchasing decisions (that are themselves affected by the retailer's assortment decisions). This gives rise to an *exploration* (learning preferences) versus *exploitation* (earning revenue) trade-off. As such, we formulate the assortment selection problem as a multi-armed bandit problem (Thompson 1933; Robbins 1985) with multiple simultaneous plays: each product represents an arm and including a product in the offered assortment is equivalent to pulling that arm – see Caro and Gallien (2007).

Off-the-shelf bandit approaches to the assortment problem call for solving an independent instance for each customer profile. However, practical instances of the problem might have scores of different profiles. This implies that arriving to reasonable preference estimates for a given profile might take an unreasonably large amount of time. More importantly, such an approach ignores the possibility that customers with different profiles may share similar product preferences. In this paper, we show that the retailer can exploit these similarities by pooling information across customers with similar purchasing behavior. To that end, we consider the existence of an underlying mapping of profiles to clusters, where a *cluster* is defined as a set of customer profiles with the same product preferences. This mapping is initially *unknown* to the retailer.

We propose a *dynamic clustering* policy under which the retailer estimates both the underlying mapping of profiles to clusters and the preferences of each cluster. Assortment decisions are based on the estimated mapping by adapting decision rules from traditional bandit algorithms. We use a Bayesian framework, called the Dirichlet Process Mixture model, to represent uncertainty about the mapping of profiles to clusters. This model arises as a natural selection given the discrete nature of the mapping and allows us to draw inference without having to predetermine the number of clusters upfront. The dynamic clustering policy estimates the mapping of profiles to clusters as well as the preference parameters for each cluster by using a Markov Chain Monte Carlo (MCMC) sampling scheme.

Main Contributions. The contributions of this paper are as follows:

We propose a prescriptive approach, called the dynamic clustering policy, for assortment personalization in an online setting. The proposed policy combines existing tools from the Bayesian data analysis literature (for estimating customer preferences through dynamic clustering) and from the machine learning / operations management literature (to prescribe personalized assortment decisions). This approach is motivated by the retailers' interest in identifying clusters of customers with similar preferences and in offering personalized assortments to their customers. Unlike most existing work that focuses on *offline* settings (i.e., using historical data), we propose an *online* tool for assortment personalization that can be implemented in real-time. Moreover, the proposed dynamic clustering policy is fairly general and flexible as the number of clusters (segments) is endogenous and does not need to be predetermined.

We illustrate the practical value of the dynamic clustering policy in a realistic setting by using a dataset from a large Chilean retailer. We use the case study to quantify the efficiency and demonstrate the implementability of the dynamic clustering policy. The dataset consists of roughly 95,000 customer-tied click records in the retailer's website for 19 products in the footwear category. We contrast the performance of the proposed policy to that of a *data-intensive* policy that ignores any potential similarity in preferences across profiles and thus estimates product preferences for each profile separately. We find that, under the MNL model, the proposed policy generates more than 37% additional transactions (on average) compared to the data-intensive policy in the case study. This finding quantifies the potential benefit of leveraging similarity in customer preferences by adaptively pooling information across profiles with similar purchasing behavior to expedite the learning process. The performance of the dynamic clustering policy is remarkable, considering that the underlying customer population in the case study is rather heterogeneous. We also compare the performance of the proposed policy to that of a *linear-utility* policy that assumes a more structured model of customer preferences. In particular, the linear-utility policy assumes that demand is driven by an MNL choice model in which mean utilities are linear functions of customer attributes. The findings from the case study show that the proposed policy generates more than 27% additional transactions (on average) compared to the linear-utility policy under the MNL demand model. While this can be partially explained by the fact that preferences in the dataset do not exhibit a linear structure, the advantage of the dynamic clustering policy persists even when using synthetic data (generated based on the dataset) under which mean utilities are linear functions of customer attributes by construction. This finding reinforces the benefits of pooling information through the proposed dynamic clustering approach.

To support the insights derived from the numerical experiments, we analyze a simplified version of the dynamic assortment selection problem in which a single product is offered to each arriving customer. First, we compare the performance of the data-intensive policy to that of a *semi-oracle* policy that knows upfront the mapping of customer profiles to clusters and thus conducts preference estimation and assortment optimization independently for each cluster (as opposed to each profile). This policy exploits the structure of preferences across profiles. Aligned with intuition, the semi-oracle policy outperforms the data-intensive policy, indicating that pooling information is beneficial for the retailer. We also show that there are diminishing marginal returns to pooling information from an increasing number of customer profiles. Next, we contrast the performance of the data-intensive policy with that of a *pooling* policy that aggregates information across all profiles (regardless of whether their preferences are similar or not). This scenario favors the data-intensive policy, as pooling information across all customers may lead to erroneous estimates and thus to suboptimal assortment offerings. Despite its shortcomings, we characterize conditions under which the pooling policy outperforms the data-intensive policy. This result highlights the benefit of pooling information in the short-term, when there is insufficient data to accurately estimate preferences for each customer profile.

Organization of the Essay. Section 3.2 provides a review of the relevant literature. Section 3.3 describes the model. Section 3.4 presents the dynamic clustering policy, while Section 3.5 illustrates its effectiveness through a case study. Section 3.6 discusses theoretical results characterizing settings in which pooling information is beneficial for the retailer. Section 3.7 provides concluding remarks. All proofs are relegated to Appendix B.1. Appendix B.2 discusses the extension of the results of Section 3.6 for Thompson Sampling.

3.2 Literature Review

This paper proposes a prescriptive approach that integrates dynamic clustering (segmentation) and demand learning with dynamic assortment personalization. The paper contributes to three streams of research: dynamic assortment planning with demand learning, personalization, and segmentation methods. Papers in the related literature generally focus on customer segmentation or assortment optimization as independent decisions. This paper considers both decisions jointly in a dynamic setting. In what follows, we review the relevant literature in more detail.

Most of the work in the related literature considers static segmentation and static assortment decisions. These papers generally focus on assortment planning for a homogeneous population of customers. We refer to Kök et al. (2015) for a comprehensive review of the assortment planning literature and industry practices. See also Chen et al. (2015), who consider personalized assortment decisions in an offline setting given customers' contextual information using Logit models. We compare the performance of our policy to a linear-utility MNL model similar to that in Chen et al. (2015).

Most recent work considers static segmentation with dynamic assortment decisions. There are two bodies of work in this stream. The first line of research mostly focuses on dynamic assortment planning for a homogeneous population of customers with unknown demand. These include Caro and Gallien (2007), Ulu et al. (2012), Rusmevichientong et al. (2010), Sauré and Zeevi (2013), and Agrawal et al. (2016). Most of these papers take a multi-armed bandit approach for dynamic assortment optimization. The second line of research considers dynamic assortment personalization for a heterogeneous population of customers. However, in these papers customer segments are known a priori and remain static over time, whereas in our paper, customer segments are dynamically updated over time as more information becomes available. This stream of work includes Bernstein et al. (2015) and Golrezaei et al. (2014) – both papers study dynamic assortment planning problems with limited inventory. Gallego et al. (2015) study a problem of resource allocation with applications to personalized assortment optimization. Jasin and Kumar (2012) and Ciocan and Farias (2014) study network revenue management problems. Closer to our work is the paper by Kallus and Udell (2016), which studies a high-dimensional dynamic assortment personalization problem with a fixed, but large, number of customer types. The paper uses a nuclear-norm regularized maximum likelihood approach (i.e., a frequentist approach) for estimation, while we use dynamic Bayesian estimation which updates the number of customer segments based on the observed transaction data.

In contrast to the existing literature, our paper considers simultaneous dynamic customer segmentation and dynamic assortment personalization decisions. Wedel and Kamakura (2012) provide a comprehensive review of market segmentation methodologies such as clustering, mixture models, and profiling segments. Our proposed Bayesian representation of uncertainty on the mapping of profiles to clusters is based on the Dirichlet Process Mixture model – see Heller and Ghahramani (2005) for a Bayesian hierarchical clustering algorithm that can be used as a deterministic alternative (approximation) to MCMC inference in Dirichlet Process Mixture models. Our adaptation of the Dirichlet Process Mixture model is based on Neal (2000). While, to the best of our knowledge, ours is the first paper in the operations management literature to employ this model, researchers in other fields have used the Dirichlet Process Mixture model to capture heterogeneity in the customer population. For example, Ansari and Mela (2003) use this model to customize the design and content of emails to increase web traffic (click-through). Burda et al. (2008) use this model to estimate the parameters of a Logit-Probit choice model. In these papers, the estimation is conducted offline using historical data.

From a methodological standpoint, the dynamic assortment selection problem we study can be thought of as a multi-armed bandit problem with multiple plays per period. In their seminal work, Lai and Robbins (1985*a*) prove a fundamental limit on the achievable performance of any so-called consistent policy in a classic bandit setting (we use this lower bound for the analysis in Section 3.6). Anantharam et al. (1987) extend the fundamental limit of Lai and Robbins (1985*a*) to a multi-armed bandit problem with multiple plays. The dynamic assortment selection problem can alternatively be formulated as a combinatorial multi-armed bandit problem envisioning each feasible assortment as an arm. This is the approach used in Rusmevichientong et al. (2010) and Sauré and Zeevi (2013). We refer the reader to Modaresi et al. (2014) and the references therein for a combinatorial multi-armed bandit formulation and a review of the relevant literature.

3.3 Model and Preliminaries

In this section, we formalize the retailer's assortment personalization problem. In particular, we introduce the notion of *clusters*, which captures the presence of heterogeneity in the customer population, and discuss the connection of the model to the multi-armed bandit problem.

Problem Definition. Consider an online retailer endowed with N products and let T denote the total number of customers that arrive during the selling season. Let $\mathcal{N} := \{1, \ldots, N\}$ denote the set of all products. The retailer has a limited display or capacity constraint of C products, i.e., the retailer can show a selection of at most C products to each arriving customer. Without loss of generality, we assume that $C \leqslant N$. Such display constraints have been motivated and used in different settings in previous studies (see, e.g., Besbes and Sauré (2016), Fisher and Vaidyanathan (2014), Rusmevichientong et al. (2010), and Caro and Gallien (2007)). In an online retail setting, this constraint may be related to limitations on the time customers spend searching for a product, or the number of products displayed in the webpage. In the context of companies such as Ropazi (mentioned in Section 3.1), customers are shown a subset of the entire product set. For $j \in \mathcal{N}$, we let r_j denote product j's unit price, which we assume to be fixed throughout the selling season.

Customers arrive sequentially throughout the selling season. We use t to index customers according to their arrival times, so that time $t = 1$ corresponds to the first customer arrival and time $t = T$ to the last one. The retailer classifies customers according to their profile information. Each profile is encoded as a unique vector of attributes (e.g., gender, age, location, past transactions, payment method). For example, in the case study presented in Section 3.5, we use customers' gender, age, and location to define their profile information. Thus, in that case, a customer profile is described by the attribute vector $x = (x_{gender}, x_{age}, x_{location})$. The profile of an arriving customer is observed by the retailer via the customer's login information or internet cookies. Let $\mathcal{I} := \{1, \ldots, I\}$ denote the set of customer profiles, where each profile i is associated with a unique vector of attributes x^i, $i \in \mathcal{I}$. A customer with profile i arrives with probability p_i, where $0 < p_i < 1$ for $i \in \mathcal{I}$ and $\sum_{i \in \mathcal{I}} p_i = 1$. Let $i_t \in \mathcal{I}$ denote the profile of customer t. Upon arrival, a customer is offered an assortment of at most C products. Let $\mathcal{S}$ denote the set of feasible assortments, i.e., $\mathcal{S} := \{S \subseteq \mathcal{N} : |S| \leqslant C\}$, where $|S|$ denotes the cardinality of set S, and let $S_t \in \mathcal{S}$ denote the assortment offered to customer t.

Demand Model. The retailer's revenue is contingent on the customers' purchasing decisions. Let $Z^i_{j,t}$ denote the purchasing decision of a customer with profile i arriving at time t regarding product $j \in S_t$. More specifically, $Z^i_{j,t} = 1$ if customer t is from profile i and purchases product $j \in S_t$ and $Z^i_{j,t} = 0$, otherwise. We consider two cases in terms of the underlying demand model:

MNL Demand. In this setting, a customer with profile i arriving at time t assigns a (random) utility $U^i_{j,t}$ to product j, with

$$U^i_{j,t} := \mu^i_j + \zeta^i_{j,t}, \quad j \in \mathcal{N} \cup \{0\},$$

where μ^i_j is the mean utility of product j for profile i (which is unknown to the retailer), $\zeta^i_{j,t}$ are independent (across i, j, and t) and identically distributed random variables drawn from a standard Gumbel distribution, and product 0 denotes the no-purchase alternative. We assume, without loss of generality, that $\mu^i_0 = 0$ for all $i \in \mathcal{I}$. We do not assume any particular relation between the mean utilities and customer attributes. In Section 3.5.5, we compare this model to another approach that assumes that mean utilities are linear functions of customer attributes. When offered an assortment $S_t \in \mathcal{S}$, customer t selects the product with the highest utility (including, possibly, the no-purchase alternative).

For a given assortment $S \in \mathcal{S}$ and a given vector of mean utilities $\mu^i := (\mu^i_1, \ldots, \mu^i_N)$, let $\Pi_j(S, \mu^i)$ denote the probability that a customer with profile i purchases product $j \in S \cup \{0\}$. We have that

$$\Pi_j(S, \mu^i) := \frac{\nu^i_j}{1 + \sum_{j' \in S} \nu^i_{j'}}, \; j \in S \cup \{0\}\,, \quad \Pi_j(S, \mu^i) := 0, \text{ otherwise}, \tag{3.1}$$

where $\nu^i_j := \exp(\mu^i_j)$ are the exponentiated mean utilities for $j \in \mathcal{N} \cup \{0\}$. Thus, for a customer with profile i arriving at time t and offered assortment S_t, $Z^i_{j,t} = 1$ with probability $\Pi_j(S_t, \mu^i)$. Moreover, we let $Z^i_{0,t} = 1$ if a customer with profile i arriving at time t opts not to purchase any product, and $Z^i_{0,t} = 0$ otherwise.

Independent Demand. In this setting, we assume that the purchasing decision $Z^i_{j,t}$ of a customer with profile i arriving at time t for product j is a Bernoulli random variable independent of the customer's purchasing decision for the other products[1]. In particular, we assume that a customer with profile i purchases product $j \in S_t$ with probability μ^i_j, independent of the assortment in which it is offered. Thus, in this setting, the vector $\mu^i = (\mu^i_1, \ldots, \mu^i_N)$ represents the purchase probabilities (i.e., mean of the Bernoulli distributions) for a customer with profile i. These purchase probabilities are unknown to the retailer. For a given assortment $S \in \mathcal{S}$ and a given vector of purchase probabilities μ^i, we let $\Pi_j(S, \mu^i)$ denote the probability that a customer with profile i purchases product $j \in S$, i.e.,

$$\Pi_j(S, \mu^i) := \mu^i_j, \quad j \in S. \tag{3.2}$$

A setting in which this demand model may be appropriate is one in which customers

[1] We focus on the Bernoulli distribution for clarity of exposition. The framework introduced in this paper applies to other distributions as well.

make purchasing decisions across various products that are not necessarily substitutes, as may be in the cases of Stitch Fix and Ropazi mentioned in Section 3.1.

Because of the dual role of the parameters $(\mu^i, i \in \mathcal{I})$ in these two demand models (as mean utilities under MNL demand and as purchase probabilities under independent demand), throughout the paper, we use the terms *purchase probabilities* and *preferences* interchangeably.

Assortment Selection. Let $W_t := (i_t, Z_t)$ denote the profile of customer t together with the vector of purchasing decisions, where $Z_t := (Z^{i_t}_{1,t}, \ldots, Z^{i_t}_{N,t})$. Let $\mathcal{F}_t := \sigma\left((S_\tau, W_\tau), 1 \leqslant \tau \leqslant t\right), t = 1 \ldots, T$, denote the filtration (history) associated with the assortment and purchasing decisions up to (and including) time t, with $\mathcal{F}_0 = \varnothing$. An admissible assortment selection policy π is a mapping from the available history to assortment decisions such that $S_t \in \mathcal{S}$ is non-anticipating (i.e., S_t is $\mathcal{F}_{t-1}$-measurable) for all t. Let $\mathcal{P}$ denote the set of admissible policies. The retailer's objective is to choose an assortment selection policy $\pi \in \mathcal{P}$ to maximize expected cumulative revenue over the selling season:

$$J^\pi(T, I) := \mathbb{E}_\pi\left(\sum_{t=1}^{T}\sum_{i=1}^{I}\sum_{j \in S_t} r_j Z^i_{j,t}\right),$$

where $\mathbb{E}_\pi$ denotes the expectation when policy $\pi \in \mathcal{P}$ is used.

Market Heterogeneity (Clusters). Although profiles differ in their observable attributes, customers with different profiles may have similar preferences for products. We define a *cluster* (or *segment*) as a set of customer profiles that have identically distributed preferences.[2] (We use the terms cluster and segment interchangeably in the paper.) This implies the existence of an underlying mapping of profiles to

[2] In order to formalize the definition of a cluster, we define it as a set of customer profiles with identically distributed preferences. However, the notion of "similar taste" is embedded in the dynamic clustering algorithm (introduced in Section 3.4) that produces the clusters.

clusters $M : \mathcal{I} \to \mathcal{K}$, where $\mathcal{K} := \{1, \ldots, K\}$ is the set of cluster labels and $K \leqslant I$ is the number of clusters. The mapping M assigns a cluster label $M(i) \in \mathcal{K}$ to each profile $i \in \mathcal{I}$, so that any two profiles with the same cluster label share the same set of preference parameters. That is, $\mu^i = \mu^{i'}$ if i and i' are such that $M(i) = M(i')$. The underlying mapping M of profiles to clusters is unknown to the retailer. In this regard, the case study presented in Section 3.5 as well as the analytical results in Section 3.6 show that the retailer benefits from estimating the mapping of profiles to clusters as it helps expedite the estimation of preferences. This, in turn, translates into higher revenue for the retailer.

Connection to the Multi-Armed Bandit Problem. The retailer does not know the customers' preferences, so assortment personalization requires estimating such preferences by observing the customers' purchasing decisions. The history of purchasing decisions is, in turn, affected by past assortment decisions. This leads to an *exploration* (learning preferences) versus *exploitation* (earning revenue) trade-off. The multi-armed bandit problem is the standard framework for addressing this trade-off. The assortment selection problem can be formulated as a multi-armed bandit by means of the following analogy: each product $j \in \mathcal{N}$ corresponds to an arm, and offering a product (i.e., including that product in the offered assortment) is equivalent to pulling that arm (see, e.g., Caro and Gallien (2007)). Thus, one can think of the problem as a finite horizon multi-armed bandit with multiple plays per period, where at each point in time, at most C out of N arms are pulled. Following the bandit literature, we restate the retailer's objective of maximizing expected cumulative revenue in terms of the *regret*. To that end, we first define $S_i^* \in \mathcal{S}$ as the optimal assortment that the retailer would offer to customers with profile i if μ^i was known. That is,

$$S_i^* \in \underset{S \in \mathcal{S}}{\operatorname{argmax}} \sum_{j \in S} r_j \, \Pi_j(S, \mu^i).$$

We define the regret associated with any policy π as[3]

$$R^{\pi}(T, I) := \sum_{i \in \mathcal{I}} p_i \left(\sum_{j \in S_i^*} r_j \, \Pi_j(S_i^*, \mu^i) \right) T - J^{\pi}(T, I). \tag{3.3}$$

The regret measures the retailer's expected cumulative revenue loss relative to a clairvoyant retailer that knows the purchase probabilities (and thus the underlying mapping of profiles to clusters). That is, the regret represents the retailer's expected cumulative revenue loss due to the lack of prior knowledge of purchase probabilities which results in suboptimal assortment offerings. Maximizing expected cumulative revenue is equivalent to minimizing the regret over the selling season.

An assortment selection policy in this setting is comprised of two elements: an *estimation* tool for estimating the customers' preferences and an *optimization* tool for deciding what assortment to offer to each arriving customer. As stated earlier, we focus on bandit algorithms as the optimization tool (we discuss this in more detail in Section 3.4.4). When comparing the performance of different policies, we assume that they follow the same bandit algorithm.

Model Discussion. This paper focuses on the efficient use of information to make personalized assortment offerings. In particular, we investigate the retailer's potential revenue benefit from aggregating transaction information across customers with similar product preferences. To this end, we make a few assumptions to facilitate the study. We assume perfect inventory replenishment for the retailer, and that the retailer incurs no operational costs (e.g., switching costs) for offering different assortments to different customers. Such assumptions are common in the dynamic assortment planning literature and allow us to isolate the role of dynamic personalized assortment planning in maximizing retailer's revenue. We also assume that

[3] Note that $J^{\pi}(T, I)$ and $R^{\pi}(T, I)$ are functions of μ^i and p_i for all $i \in \mathcal{I}$ as well, but we drop such dependence to simplify the notation.

the products' prices are constant throughout the selling season. This assumption is also common in the assortment planning literature and facilitates analysis (see, e.g., Sauré and Zeevi (2013)). Finally, we assume that customers' purchasing decisions are independent over time and across customers (i.e., we ignore word-of-mouth and other related effects).

3.4 Dynamic Assortment Personalization

In this section, we introduce a prescriptive approach for dynamic assortment personalization which we call the *dynamic clustering* policy. This policy adaptively estimates both the customers' preferences and the mapping of profiles to clusters in a Bayesian manner. In what follows, we first present the Bayesian model of preferences in Section 3.4.1, followed by the dynamic clustering policy in Section 3.4.2. Section 3.4.3 discusses the estimation procedure based on the observed purchase history, while Section 3.4.4 reviews the bandit policies we use. We illustrate the performance of the dynamic clustering policy in a case study in Section 3.5.

3.4.1 Bayesian Model of Preferences

In this section, we present a Bayesian framework to model customers' preferences. In section 3.4.3, we present a Markov Chain Monte Carlo (MCMC) sampling technique to estimate the model discussed in this section.

Recall that $Z^i_{j,t}$ denotes the random variable that captures the purchasing decision of a customer with profile i arriving at time t regarding product $j \in S_t$. We define $Z^i_t := (Z^i_{1,t}, \ldots, Z^i_{N,t})$ and let $F(\cdot|\mu^i)$ denote the distribution of Z^i_t as a function of the vector of parameters μ^i. This distribution is independent of t as preferences are time-homogeneous. For the case of MNL demand, $F(Z^i_t = e^j|\mu^i) = \Pi_j(S, \mu^i)$, where $\Pi_j(S, \mu^i)$ is as defined in (3.1) and e^j denotes the j-th unit vector (although $F(\cdot|\mu^i)$ also depends on the assortment S, we drop such dependence to simplify notation).

For the case of independent demand, $F(Z^i_{j,t} = 1|\mu^i) = \Pi_j(S, \mu^i)$, where $\Pi_j(S, \mu^i)$ is as defined in (3.2) and $Z^i_{j,t}$ are independent across j. The retailer knows the family of distributions F, but does not know the vector of parameters μ^i that characterizes this distribution for customers with profile i.

We adopt a hierarchical Bayesian model to represent the retailer's uncertainty with respect to the underlying mapping M from profiles to clusters and to the vector of parameters μ^i governing the preferences of customer profiles. More specifically, we model the distribution from which the Z^i_t's are drawn as a mixture of distributions of the form $F(\cdot|\mu^i)$. We denote the mixing distribution over μ^i by H and let the prior distribution of H be a Dirichlet Process (Ferguson 1973; Antoniak 1974). A Dirichlet Process prior is a natural selection as its realizations are (discrete) probability distributions. The Dirichlet Process is specified by a distribution H_0, which serves as a baseline prior for H, and a precision parameter α (which is a positive real number) that modulates the deviations of H from H_0 – the larger the precision parameter α, the more concentrated the Dirichlet Process prior is around the baseline location H_0. We denote the Dirichlet Process by $\mathrm{DP}(H_0, \alpha)$. We therefore model the uncertainty over customers' preferences as follows:

$$
\begin{aligned}
Z^i_t|\mu^i &\sim F(\cdot|\mu^i) && \text{(3.4a)}\\
\mu^i|H &\sim H && \text{(3.4b)}\\
H &\sim \mathrm{DP}(H_0, \alpha). && \text{(3.4c)}
\end{aligned}
$$

Being an infinite mixture model, the Dirichlet Process Mixture provides a flexible framework for capturing heterogeneity in the customer population without the need to predetermine the number of clusters. In fact, the number of clusters is endogenously determined based on the observed transaction data. Further details on the Dirichlet Process can be found in Ferguson (1973), Ferguson (1983), and Neal (2000).

3.4.2 Dynamic Clustering Policy

We introduce the dynamic clustering policy by presenting a general description of the sequence of events that takes place for each customer arrival t. Let Φ denote the set of time indices (periods) in which the dynamic clustering policy updates the mapping of profiles to clusters. For instance, in the case study in Section 3.5, we take $\Phi = \{100, 200, 300, \ldots\}$; that is, we update the mapping after every 100 customer arrivals (to expedite the computation time of the algorithm).

Step 1 (Arrival). Observe the profile of the arriving customer t (i.e., i_t).

Step 2 (Assortment Selection). Follow a bandit algorithm to determine the assortment $S_t \in \mathcal{S}$ to offer customer t based on the current preference and mapping estimates. (For the first customer arrival, start with an arbitrary mapping from profiles to clusters and select preferences randomly.) See Section 3.4.4 for details.

Step 3 (Transaction). Observe the purchasing decision of customer t and update the assortment and purchase history.

Step 4 (Mapping and Preference Estimation). If $t \in \Phi$, then perform the estimation procedure described in Section 3.4.3 to estimate preferences and mapping of profiles to clusters using all available information; otherwise, only update the preference estimates using the prevailing mapping estimate.

Figure 3.1 illustrates the steps in the dynamic clustering policy. This policy adapts existing tools from the Bayesian data analysis and machine learning/operations management literature. In Step 2 (i.e., optimization step), we use a bandit algorithm to determine the assortment to offer each arriving customer based on the current mapping and preference estimates. We discuss the implementation details of the

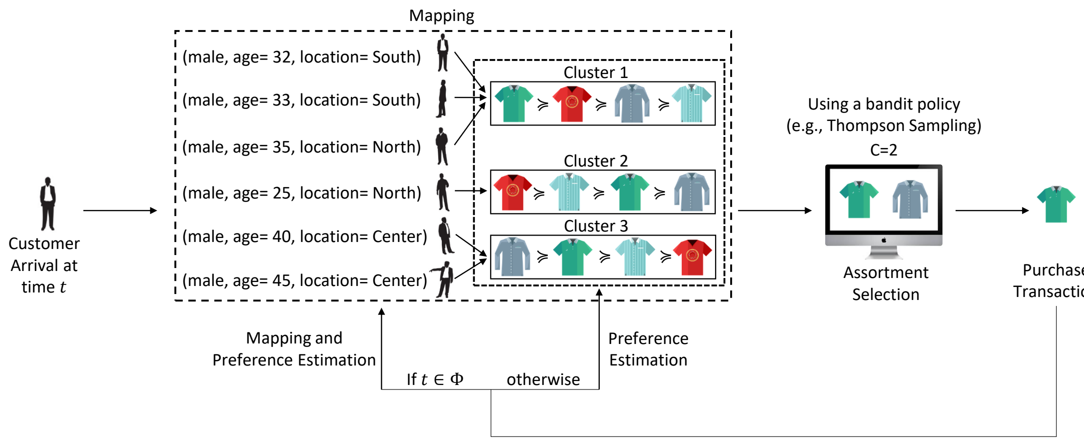

FIGURE 3.1: Illustration of the dynamic clustering policy.

bandit algorithm in Section 3.4.4. In Step 4 (i.e., estimation step), we implement the estimation procedure that will be introduced in Section 3.4.3 to update the mapping of profiles to clusters and the corresponding preference estimates associated with each cluster.

3.4.3 Mapping and Preference Estimation

We estimate the mapping and preferences in Step 4 of the dynamic clustering policy by approximating the posterior distribution of the parameters in model (3.4). To that end, we use an MCMC sampling scheme comprised of a Metropolis-Hastings step to update the mapping of profiles to clusters followed by a Gibbs sampling step to update the posterior distribution of preference parameters for each cluster. This implementation is an adaptation of the sampling scheme in Neal (2000), which is tailored for the case of a Dirichlet Process Mixture model. The output of the MCMC sampling scheme is a sequence of mappings of profiles to clusters and preference parameter vectors. The sampling procedure is tailored so that the samples approximate a set of independent draws from the posterior distribution of the model parameters. (See Gelman et al. (2014) for further details on MCMC methods.)

Next, we provide details on the sampling scheme within the MCMC procedure. Consider an arriving customer t, with $t \in \Phi$. After observing the profile of the customer, offering an assortment, and recording the purchasing decision, the dynamic clustering algorithm approximates the posterior distribution of the model parameters (mapping and preferences). Let $\mathcal{X}_t^i$ denote the assortment and purchase history associated with profile i up to (and including) time $t \leqslant T$. That is, $\mathcal{X}_t^i := \{(S_u, Z_u^{i_u}) : i_u = i, 1 \leqslant u \leqslant t\}$, and set $\mathcal{X}_0^i = \varnothing$. Define $\mathcal{X}_t := (\mathcal{X}_t^1, \cdots, \mathcal{X}_t^I)$. The following sampling procedure generates a sequence of mappings and parameter vectors (one for each cluster in each mapping). Let the tuning parameter η denote the number of samples to be collected.

MCMC Sampling. Start with an arbitrary mapping M_1 from profiles to clusters and sample the preference parameters from H_0 for each cluster. For $s = 1, \ldots, \eta$, repeat the sampling process as follows:

- **Step 1 (Cluster Update).** Let $c_i = M_s(i)$ denote the cluster associated with profile i under the mapping M_s and let c denote a generic cluster. For each profile $i \in \mathcal{I}$, update the cluster label c_i associated with that customer profile as follows. Let $n_{-i,c}$ be the number of profiles, excluding profile i, that are mapped to an existing cluster c under the mapping M_s. Draw a candidate cluster label c_i^* according to the following probability distribution:[4]

$$\begin{aligned} \mathbb{P}\left(\text{assign } c_i \text{ to an existing cluster } c\right) &= \frac{n_{-i,c}}{I - 1 + \alpha} \\ \mathbb{P}\left(\text{assign } c_i \text{ to a new cluster}\right) &= \frac{\alpha}{I - 1 + \alpha}. \end{aligned}$$

  If $c_i^* \in \{c_1, \ldots, c_I\}$, then use the corresponding parameter vector $\mu^{c_i^*}$. If $c_i^* \notin \{c_1, \ldots, c_I\}$, i.e., if the candidate cluster does not correspond to any of the existing clusters under M_s, then sample $\mu^{c_i^*}$ from H_0.

[4] This distribution is derived in Neal (2000), where α is the precision parameter of the Dirichlet Process Mixture model.

Set the new value of c_i to c_i^* with probability

$$a(c_i^*, c_i) := \min\left\{1, \frac{L(\mathcal{X}_t^i, \mu^{c_i^*})}{L(\mathcal{X}_t^i, \mu^{c_i})}\right\}$$

and do not change c_i with probability $1-a(c_i^*, c_i)$, where $L(\mathcal{X}_t^i, \mu^{c_i})$ denotes the likelihood function given the purchase history $\mathcal{X}_t^i$ and the vector of parameters μ^{c_i}. Let M_{s+1} be the updated mapping given by the new assignment of profiles to clusters (i.e., updated c_i's).

- **Step 2 (Preference Update).** Update the vector of preference parameters for each cluster: for each $c \in \{c_1, \ldots, c_I\}$, compute the posterior distribution of μ^c (given the history $\mathcal{X}_t$) and draw a new realization for μ^c from its posterior distribution.

To approximate the posterior distribution of the mapping, we discard the first η_b samples ("burn-in" period), and select every other η_d-th draw (e.g., every 10th draw) from the remaining samples (both η_b and η_d are tuning parameters). Let $m' = (\eta - \eta_b)/\eta_d$ denote the number of MCMC draws used for estimation. Denote the corresponding (distinct) mappings by $M_1, M_2, \ldots, M_m$, with $m \leqslant m'$ as the mappings corresponding to several sample points may be identical. Let $0 < f_1, f_2, \ldots, f_m \leqslant 1$ be the associated frequency proportions (i.e., the relative number of occurrences of each mapping in the set of selected samples), with $\sum_{n=1}^{m} f_n = 1$. We approximate the posterior distribution of the mapping as a discrete probability distribution that takes the value M_n with probability f_n, $n \leqslant m$. Note that the number of possible mappings from profiles to clusters is combinatorial in I, the number of profiles. In this regard, the approximation we propose alleviates the complexity of calculating the posterior distribution of the mapping. We discuss further implementation details in Section 3.5.

The preference update in Step 2 above depends on the underlying demand model. Under MNL demand, it is necessary to introduce a separate Metropolis-Hastings step to update the posterior distribution of μ^c, as there is no conjugate prior for the MNL model. This, however, comes at the expense of additional computational effort. To alleviate this computational burden, we approximate the parameters of the MNL model by using frequentist point estimates. (Because of this approximation, we do not need to specify the prior distribution H_0.) Specifically, suppose that $t-1$ customers have arrived so far and that the first step of the MCMC has resulted in a mapping of profiles to clusters denoted by M_s. In Step 2 of the MCMC, we estimate the exponentiated mean utility ν_j^i by $\hat{\nu}_{j,t}^{M_s(i)}$, where

$$\hat{\nu}_{j,t}^{M_s(i)} := \frac{\sum_{l=1}^{t-1} Z_{j,l}^{i_l} \mathbf{1}\left\{j \in S_l, M_s(i_l) = M_s(i)\right\}}{\sum_{l=1}^{t-1} Z_{0,l}^{i_l} \mathbf{1}\left\{j \in S_l, M_s(i_l) = M_s(i)\right\}}, \ j \in \mathcal{N}. \tag{3.6}$$

We then estimate $\mu_j^{M_s(i)}$ for cluster $M_s(i)$ by $\hat{\mu}_{j,t}^{M_s(i)} := \ln(\hat{\nu}_{j,t}^{M_s(i)})$. Note that this parameter estimation is conducted at the product level by exploiting the independence of irrelevant alternatives (IIA) property of the MNL model. Moreover, such estimates are obtained for each cluster by pooling transaction data across customers within the same cluster. The numerical results reported in Section 3.5 suggest that this approximation results in a reasonable performance and computation time. Under independent demand, we take H_0 in the Dirichlet Process Mixture to be the product of independent Beta distributions, as the Beta distribution is the conjugate prior of the Bernoulli distribution. Thus, the posterior distribution of μ^c in Step 2 can be computed in closed-form using Bayes' rule.

3.4.4 Assortment Optimization

We next describe the bandit policies used for the assortment selection rule.

MNL demand. For the MNL model, we adapt Algorithm 3 of Sauré and Zeevi (2013) to our setting. This algorithm determines whether to explore or exploit for each arriving customer t, as follows. If all products have been explored at least a number of times (which is of order $\ln(t)$), then the algorithm exploits the current optimal assortment. Otherwise, it offers an assortment containing under-tested products (exploration). We refer to Sauré and Zeevi (2013) for further details. Sauré and Zeevi (2013) assume a homogeneous population of customers. However, in the Bayesian setup of the dynamic clustering policy, estimates are derived from the approximation to the posterior distribution of the mapping and customer preferences. Thus, we adapt the algorithm in Sauré and Zeevi (2013) for the dynamic clustering policy. To that end, suppose that $t-1$ customers have arrived so far, and the Mapping and Preference Estimation procedure (discussed in Section 3.4.3) has resulted in the distinct mappings $M_1, M_2, \ldots, M_m$ with frequency proportions $f_1, f_2, \ldots, f_m$, respectively. Let customer t have profile i. We estimate the exponentiated mean utilities ν_j^i by $\hat{\nu}_{j,t}^i$, where

$$\hat{\nu}_{j,t}^i := \sum_{l=1}^{m} f_l \ \hat{\nu}_{j,t}^{M_l(i)},$$

and $\hat{\nu}_{j,t}^{M_l(i)}$ is as defined in (3.6). Moreover, we let

$$T_j^i(t) := \sum_{l=1}^{m} f_l \ T_j^{M_l(i)}(t),$$

where $T_j^{M_l(i)}(t)$ is the number of times that product j has been offered to a customer from cluster $M_l(i)$ up to (and excluding) time t. In other words, $T_j^i(t)$ is the *average* number of times (over different mappings) that product j has been offered to a customer from the cluster associated with profile i up to (and excluding) time t. The quantities $\hat{\nu}_{j,t}^i$ and $T_j^i(t)$ are used to select the assortment to offer customer t in Algorithm 3 of Sauré and Zeevi (2013).

Independent demand. For the independent demand model, we adapt the Thompson

Sampling policy (Thompson 1933) to our setting. We first present details of this policy for a classic setting (i.e., a homogeneous population of customers, a single product offering to each customer, and equal product prices) and then discuss how we adapt this policy to our setting.

- In a classic bandit setting, let $Beta(1,1)$, i.e., a uniform distribution, be the conjugate prior of the purchase probability of customers for each product. Let $Beta(a_{j,t}, b_{j,t})$ denote the posterior Beta distribution with parameters $a_{j,t}$ and $b_{j,t}$ for product $j \in \mathcal{N}$, where $a_{j,0} = b_{j,0} = 1$ for all products $j \in \mathcal{N}$, $a_{j,t} = a_{j,t-1}+1$ and $b_{j,t} = b_{j,t-1}$ if customer t purchases product $j \in S_t$, and $a_{j,t} = a_{j,t-1}$ and $b_{j,t} = b_{j,t-1} + 1$ if customer t does not purchase product $j \in S_t$. Moreover, $a_{j,t} = a_{j,t-1}$ and $b_{j,t} = b_{j,t-1}$ for all other products $j \in \mathcal{N} \backslash S_t$ (i.e., the products that are not offered to customer t). Sample $Q_{j,t}$ randomly from the posterior distribution $Beta(a_{j,t-1}, b_{j,t-1})$, and offer product $S_t \in \underset{j \in \mathcal{N}}{\operatorname{argmax}} \{Q_{j,t}\}$ at time t.

- In our setting, suppose that the Mapping and Preference Estimation procedure has resulted in the distinct mappings $M_1, M_2, \ldots, M_m$ with frequency proportions $f_1, f_2, \ldots, f_m$, respectively. Let $Q_{j,t}^{M_l(i_t)}$ be the index of product j corresponding to cluster $M_l(i_t)$. Note that the functions $a_{j,t}$ and $b_{j,t}$ are defined (and updated) separately for each cluster. Set product j's index as $Q_{j,t} = \sum_{l=1}^{m} f_l Q_{j,t}^{M_l(i_t)}$. Offer an assortment S_t that contains C products with the highest $Q_{j,t}$ indices.[5]

Remark 6. An alternative adaptation of Thompson Sampling would first randomly draw a mapping from $M_1, M_2, \ldots, M_m$ according to the frequency proportions $f_1, f_2, \ldots, f_m$, and then draw from the corresponding posterior Beta distributions.

[5] We can similarly adapt any index-based bandit policy, e.g., UCB1 of Auer et al. (2002). We obtained similar numerical results for UCB1 and therefore report only those based on Thompson Sampling in Section 3.5.3.

We favor our proposed approach because different mappings may only differ in terms of the composition of a few clusters. By averaging over different mappings, we take into account the similarities across different mappings. Our approach performs well, as evidenced in the case study.

3.5 Case Study

In this section, we discuss the results of several numerical experiments conducted on a dataset from a large Chilean retailer. We first provide a brief overview of the dataset in Section 3.5.1. We then discuss implementation details in Section 3.5.2. In Section 3.5.3, we compare the performance of the dynamic clustering policy to those of the data-intensive and linear-utility policies. We then study the impact of a finer representation of customer attributes (leading to a larger number of profiles) on the performance of policies in Section 3.5.4. We finally provide a more detailed comparison between the dynamic clustering policy and the linear-utility model in Section 3.5.5. The case study demonstrates the practical value of the dynamic clustering policy in a realistic setting. We find that the dynamic clustering policy outperforms the data-intensive and linear-utility policies as it benefits from pooling information and learning about customer preferences relatively faster. The case study also demonstrates the efficiency and scalability of the dynamic clustering policy in terms of computation time.

3.5.1 Dataset

The dataset that we use for the case study is from a chain of department stores owned by a Chilean multinational company headquartered in Santiago, Chile. The company sells clothing, footware, furniture, housewares, and beauty products, both through its network of department stores and through its online channel. In 2014, the company reported US$ 4.4 billion in gross profit. The dataset was collected as part of a larger field study by the retailer. In this study, the assortments offered to customers were chosen randomly without testing any assortment personalization strategy. The dataset consists of 94,622 customer-tied click records for a set of 19 products in the footware category (see Figure 3.2 below).[6] The dataset used in this study was collected through an experiment in which the 19 products were randomly assigned to 8 different assortments of 4 products each (i.e., $N = 19$ and $C = 4$). The experiment was conducted during a 32-day period through the retailer's online channel. Each arriving customer was shown one of these 8 assortments, chosen at random. Figure 3.2 illustrates an example of an assortment shown to customers. If the customer clicked on one of the products, that click was recorded in the dataset. Otherwise, a no-click was recorded. Therefore, each customer visit resulted in at most one click record. The dataset recorded the assortment history as well as the purchase/no-purchase decision (i.e., click/no-click decision in the context of the experiment) of each customer. The company uses information about the customers' location in Chile (according to a partition of the country into 7 different regions, determined by the retailer's marketing department: "Far North", "North," "Center," "South," "Far South," "Santiago West," and "Santiago East"), age group (the retailer uses three age groups, namely, $[0, 29]$, $[30, 39]$, and $[40, 99]$), and gender. This leads to a total of 42 unique vector of customer attributes. (The dataset in fact contains more granular information on the age of customers, but we mostly use the age groups as determined by the retailer.)

3.5.2 Implementation Details

Estimation of Underlying Demand Model. We begin by using the dataset to estimate the parameters of the model. First, we estimate the distribution of customer

[6] Each product is actually a "banner" that directs the customer to a page containing footware of that particular style/manufacturer – e.g., the second banner in Figure 3.2 leads to a page containing shoes with a 1970's style.

Figure 3.2: Example of an assortment shown on the retailer's website.

arrivals (i.e., p_i) based on the number of transactions associated with each customer profile. Next, we estimate the underlying demand model from data. Because the set of available products belongs to the same category (shoes in this study), we use the MNL demand model as it accounts for product substitution. We also report on experiments based on the independent demand to explore the robustness of the results with respect to the underlying demand model.

For the MNL demand model, we estimate the exponentiated mean utility of each product for each customer profile separately. That is, we estimate 19 parameters for each of the 42 customer profiles using the transaction data only from that customer profile in the dataset. We do not assume any particular relation between the mean utilities and customer attributes when estimating the underlying demand model from data. Formally, for a profile i and product j, we estimate the exponentiated mean utility ν_j^i by

$$\hat{\nu}_j^i := \frac{\sum_u Z_{j,u}^i \mathbf{1}\{j \in S_u, i_u = i\}}{\sum_u Z_{0,u}^i \mathbf{1}\{j \in S_u, i_u = i\}}, \ j \in \mathcal{N}, \ i \in \mathcal{I}.$$

Note that, as in (3.6), the parameter estimation is conducted at the product level by exploiting the IIA property of the MNL model. For the case of independent demand, we estimate the purchase probability of each product for each profile as the sample mean of the number of purchases from data.

Data-Intensive Policy. We compare the performance of the dynamic clustering policy to that of the *data-intensive* policy, in which assortment decisions are made by treating each customer profile independently (as if each customer profile had a different distribution of preferences for products – even if this is not the case). Thus, under the data-intensive policy, the retailer assumes a deterministic mapping of customer profiles to clusters where each profile is mapped to a distinct cluster, i.e., $M(i) = i$ for $i \in I$. The data-intensive policy emphasizes the accuracy of preference estimation. That is, under this policy the retailer eventually learns the customers' preferences accurately, but at the expense of requiring a considerable amount of transaction data on each customer profile. Therefore, the data-intensive policy is prone to suffer from a slow learning speed.[7] We find that the dynamic clustering policy outperforms the data-intensive policy as a result of faster learning achieved by pooling information across customers with similar preferences.

Linear-Utility Policy. We also compare the performance of the dynamic clustering policy to that of a policy that assumes a linear structure on the underlying demand model in terms of the dependence of utilities on customer attributes. We describe the MNL model in terms of the specific dataset available from the retailer. In particular, for this model, $x = (x_M, x_F, x_{A_1}, x_{A_2}, x_{A_3}, x_{L_1}, x_{L_2}, \ldots, x_{L_7})$ denotes the vector of attributes of a customer, where each variable is binary – x_M and x_F identify the gender of the customer (male and female, respectively), $x_{A_i},\ i \in \{1, 2, 3\}$ identifies the age group, and $x_{L_j},\ j \in \{1, \ldots, 7\}$ identifies the location of the customer. The mean utility μ_j^x of a product j for a customer with profile x is assumed to take the form

$$\mu_j^x := \beta_j^\top x = \beta_j^0 + \beta_j^M x_M + \beta_j^{A_2} x_{A_2} + \beta_j^{A_3} x_{A_3} + \beta_j^{L_2} x_{L_2} + \beta_j^{L_3} x_{L_3} + \cdots + \beta_j^{L_7} x_{L_7}, \quad (3.7)$$

[7] We measure the speed of learning by evaluating the root mean squared error (between the actual and estimated parameters) over time.

where β_j denotes the vector of coefficients, and β_j^0 captures the nominal utility of product j together with the effect of attributes "female", first age group (i.e., $[0, 29]$), and Location 1. The vector of coefficient β_j is unknown to the retailer and must be estimated from the customers' transaction data. Similar to the data-intensive policy, the linear-utility policy treats each customer profile independently for optimization (i.e., following the same bandit algorithm). However, because this policy assumes that the underlying mean utilities of products are linear functions of customer attributes, the estimation of the β_j's is based on maximum likelihood estimation (MLE). Because different profiles might share some attributes, the MLE leverages information from similar profiles to estimate the preference parameters.

MCMC and Operating Machine. In order to estimate the mapping and customers' preferences under the dynamic clustering policy, we use $\eta = 300$ and a burn-in period of $\eta_b = 100$ iterations in the MCMC sampling scheme, after which every η_d =10th MCMC draw is used to estimate the mapping of profiles to clusters (this alleviates the auto-correlation between the MCMC draws). We also set the precision parameter of the Dirichlet Process to $\alpha = 1$. To expedite the computation time of the dynamic clustering policy, we set $\Phi = \{100, 200, 300, \ldots\}$; that is, we update the mapping of profiles to clusters every 100 customer arrivals. In between these periods, we use the prevailing mapping to update the preference parameters. (For consistency, we also update the estimates of the attribute-specific parameters in the linear-utility policy every 100 customer arrivals and use the prevailing estimates in between these periods.) All experiments were run on a machine with an Intel(R) i7-6700 3.40GHz CPU and 16GB of memory. In what follows we discuss the results of numerical experiments.

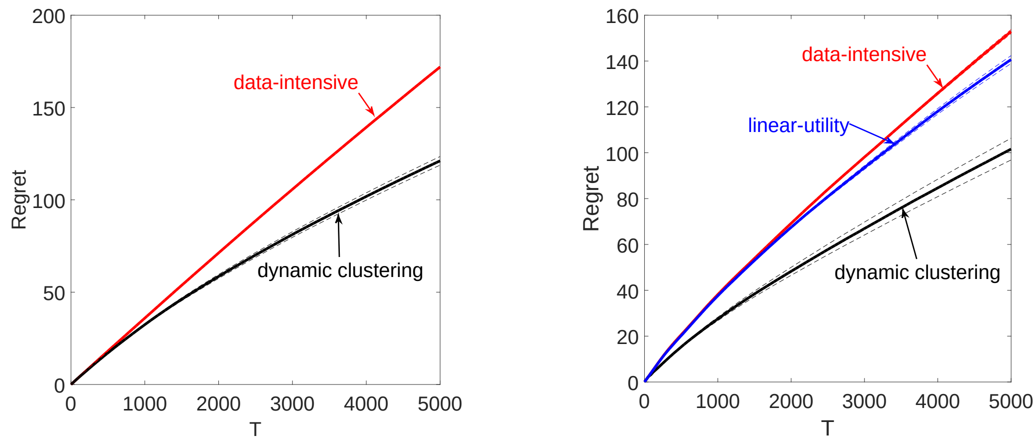

FIGURE 3.3: Average performance in a market with $I = 42$ profiles and independent demand (left), and MNL demand (right).

3.5.3 Performance Comparison

In this section, we compare the performance (in terms of regret) of the dynamic clustering policy to that of the data-intensive and linear-utility policies using the dataset from the Chilean retailer. We consider the cases of both MNL and independent demands as the underlying demand models. We also compare the speed of learning between the dynamic clustering and data-intensive policies.

The underlying demand model and distribution of customer arrivals are estimated from the dataset. We run the experiments in markets with $T = 5000$ customers. There are nineteen products ($N = 19$) and the display constraint is of size four ($C = 4$). Moreover, there are $I = 42$ distinct customer profiles. We set prices $r_j = 1$ for all $j \in \mathcal{N}$. The reported performances are averaged over 100 replications and the dashed lines around a regret function represent the 95% confidence interval.

Figure 3.3 illustrates the average performance for the case of independent demand (left panel) and MNL demand (right panel). The dynamic clustering policy significantly outperforms the data-intensive and linear-utility policies as a result of pooling information. At the same time, the linear-utility policy outperforms the data-intensive policy as the latter suffers from a relatively slower learning speed.

Figure 3.4 (left panel) illustrates the evolution of the root mean squared error (RMSE) of estimated MNL parameters (i.e., exponentiated mean utilities) for the dynamic clustering and data-intensive policies in the case of the MNL demand model. This error is averaged over all products and profiles. As noted from the graph, the RMSE associated with the dynamic clustering policy decreases significantly faster than that of the data-intensive policy, implying a faster learning speed. The right panel of Figure 3.4 shows the evolution of the average number of clusters that emerge from the dynamic clustering policy over time (i.e., over customer arrivals). As can be noted from the graph, early on in the selling season, when only a limited number of transactions have been observed, the average number of clusters is small. That is, the policy pools transaction information across a large number of customer profiles. As more transaction data is collected, the dynamic clustering policy refines the composition of customer segments (clusters) and better personalizes the assortment offerings using a larger number of clusters.

In sum, the dynamic clustering policy outperforms the data-intensive and linear-utility policies in terms of regret (i.e., revenue collection) in the case study based on the Chilean retailer's dataset. We find that, under the MNL model, the dynamic clustering policy results (on average) in 37.7% and 27.3% additional transactions compared to the data-intensive and linear-utility policies, respectively. Moreover, the dynamic clustering policy results in more than 65% additional transactions compared to a randomized assortment policy (which was used by the retailer while collecting the data). Furthermore, the dynamic clustering policy has a significantly faster learning speed compared to the data-intensive policy. The proposed policy pools information across most profiles early on in the selling season, but personalizes the assortment offerings as more transaction data becomes available.

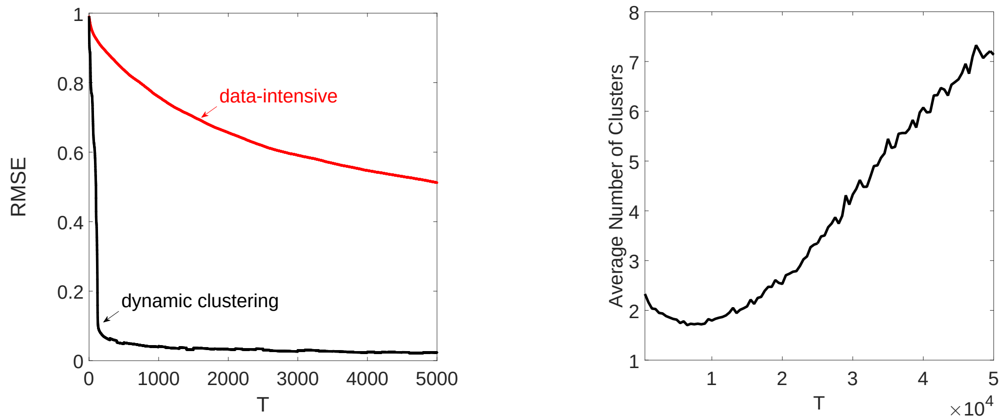

FIGURE 3.4: Speed of learning for different policies (left) and the evolution of the average number of clusters under the dynamic clustering policy (right) for MNL demand and $I = 42$.

3.5.4 *Customer Attributes*

In this section, we study the impact of a finer definition of customer attributes on policy performance. Moreover, we compare the computation times of different policies and illustrate the efficiency and scalability of the dynamic clustering policy. In addition, we discuss an approach to further expedite the computation time of the dynamic clustering policy. We finally discuss how one can incorporate management knowledge about customer similarity into the dynamic clustering policy.

As discussed before, the retailer uses 42 different customer profiles, based on their understanding of the Chilean retail market. For example, the attribute corresponding to the customer's location is based on the retailer's knowledge about the customer base in Chile (e.g., urban versus rural locations, etc.). The retailer also groups customers according to their ages into three age groups $[0, 29]$, $[30, 39]$, and $[40, 99]$, presumably based on an understanding of different purchasing patterns of customers in each of these three groups. In order to study the impact of a finer definition of customer attributes, we have extended the study beyond the 42 original customer profiles used by the retailer. The raw dataset available to us contains

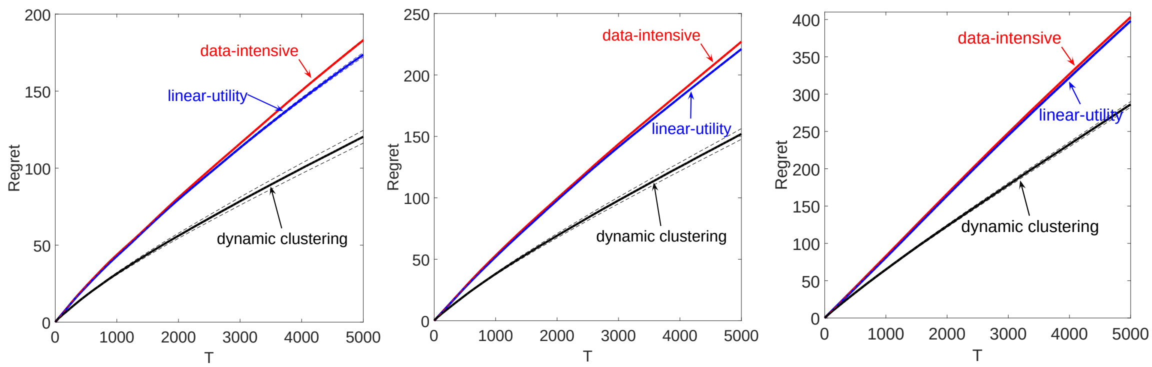

FIGURE 3.5: Average performance in a market with MNL demand and $I = 75$ (left), $I = 140$ (middle), and $I = 450$ (right).

more granular information about the customers' age (but not about the customers' location). As a result, we consider three additional studies that progressively refine the age attribute definition. These three studies consist of 75, 140, and 450 profiles, respectively. The first experiment with 75 distinct profiles uses age groups $[0, 30], [31, 40], [41, 50], \ldots, [91, 99]$. The second one has 140 customer profiles using age groups $[0, 20], [21, 25], [26, 30], \ldots, [96, 99]$. The third set of experiments has 450 customer profiles using exact ages $18, 19, 20, \ldots, 99$. In each experiment, we only kept the profiles for which there was at least one no-purchase transaction in the dataset.

Figure 3.5 illustrates the average performance of the dynamic clustering, data-intensive, and linear-utility policies for the case of MNL demand. In all settings, the dynamic clustering policy outperforms the other policies in terms of regret (and thus revenue). That is, the better performance of the dynamic clustering policy relative to the other policies is robust with respect to the definition of customer attributes.

A finer set of customer attributes leads to an increased number of profiles, potentially slowing down the mapping estimation process and thus affecting computation times. Table 3.1 reports the computation time of all three policies for the cases of 42, 75, 140, and 450 customer profiles (all under the MNL demand model). The dynamic clustering policy requires the estimation of the mapping (by running the MCMC). In Table 3.1, we separate the running time of a customer arrival for which the mapping is updated (noted as MCMC) and the running time of those arrivals for which there is no mapping update (in which case the policy uses the prevailing mapping of profiles to clusters). Similarly, the linear-utility model requires the estimation of attribute-specific parameters through the maximum likelihood estimation (MLE). Table 3.1 reports the running time of a customer arrival for which the attribute-specific parameters are updated (noted as MLE) and the running time of those arrivals in which there is no parameter update (and for which the linear-utility policy operates under the prevailing estimates). The computation time of the MCMC and MLE increases with the granularity of customer attributes (and therefore the number of customer profiles).

The computation time of the dynamic clustering policy is reasonable and scales well with the number of profiles. All experiments were run on a personal computer – a retailer with more sophisticated computational resources would experience even faster results. Note that, as expected, the data-intensive policy has the fastest computation time (at the expense of a lower revenue performance), as this policy treats each profile independently. The dynamic clustering policy can also handle a large number of products without significantly affecting computation times. In the estimation stage, the number of products impacts the calculation of the likelihood functions and the updates of preference parameters, which are computed in closed-form as discussed in Section 3.4.3. For the assortment optimization stage, there are efficient algorithms for the MNL model (for example, the algorithm in Rusmevichientong et al. (2010) scales polynomially in the number of products). The optimization in the case of independent demand is trivial.

We next discuss an approach to further improve the computation time of the dynamic clustering policy, without significantly impacting its performance. This

Table 3.1: Average computation time (in seconds) of different policies for MNL demand and $T = 5000$.

	Dynamic Clustering		Linear-Utility		Data-Intensive
	No mapping update	MCMC	No parameter update	MLE	Each customer arrival
$I = 42$	0.00031	1.0818	0.00017	2.7827	0.00015
$I = 75$	0.00039	1.7121	0.00018	3.9762	0.00015
$I = 140$	0.00054	2.9850	0.00018	5.9750	0.00016
$I = 450$	0.00117	8.1729	0.00018	9.1619	0.00017

approach involves reducing the number of customer profiles for clustering purposes. In particular, this version of the dynamic clustering policy considers customer profiles within each specific location in isolation and therefore the policy runs the MCMC for different locations in parallel. We refer to this version of the dynamic clustering policy as "Location-DC." Table 3.2 below reports the computation times of this version of the dynamic clustering policy, together with that of the original policy (which we refer to as "Original-DC"), for the setting with $I = 450$ profiles and MNL demand. As can be noted from the table, the "Location-DC" version of the policy brings significant savings in terms of computation time. While this version is slightly outperformed by the original dynamic clustering policy in terms of regret, it still performs significantly better than the data-intensive and linear-utility policies.

Table 3.2: Average computation time (in seconds) of the original and a location-based version of the dynamic clustering policy for MNL demand with $I = 450$ and $T = 5000$.

	No mapping update	MCMC
Original-DC	0.00117	8.1729
Location-DC	0.00048	1.5575

We finally discuss how one can incorporate management knowledge about customer similarity into the dynamic clustering policy. Suppose that existing management insight indicates that "neighboring" profiles (e.g., two profiles with the same gender and location and very close in age) are likely to have "similar" preferences for products. The dynamic clustering algorithm can accommodate management knowledge about customers by restricting attention to mappings that only group "neighboring" profiles. (This can be implemented in the MCMC sampling procedure: while updating the cluster label c_i^* of a customer with profile given by a vector x^i, the candidate cluster label is drawn only from clusters that currently include profiles that are "similar" to x^i – the notion of similarity can be formally defined based on attributes.) An alternative approach consists of merging upfront profiles that are known to have similar preferences for products. For example, it may be that, based on an understanding of the customer base, management is confident that customers from two particular ZIP codes or locations have similar tastes for products. Unlike the previous approach, these customer profiles with similar preferences may not have similar attributes. One can use this additional information to speed up the learning process by grouping such profiles together before running the dynamic clustering algorithm.

In sum, we find that the better performance of the dynamic clustering policy relative to the other policies is robust with respect to the definition of customer attributes. At the same time, a finer definition of customer attributes (i.e., a larger number of customer profiles) increases the computation time. However, we show that the computation time of the dynamic clustering policy is still reasonable and scales well with the number of profiles. One can further expedite the computation time of the policy by reducing the number of customer profiles for clustering purposes.

3.5.5 Comparison to Linear-Utility Model

In this section, we compare the dynamic clustering policy to the linear-utility approach in more detail. We first discuss the advantages of each approach and then introduce a set of experiments to compare the performance of the two policies.

In addition to better performance of the dynamic clustering policy over the linear-utility model, the dynamic clustering policy has several other advantages. Retailers are generally interested in identifying customer segments (i.e., clusters of customers with similar preferences). These segments are part of the output of the dynamic clustering policy and can be interpreted based on customers' attributes – see Figure

3.6 below for a representative example based on the dataset from the Chilean retailer. Figure 3.7 also illustrates the optimal assortments for different clusters in an example in which the profiles only differ by their geographical location in Chile. Moreover, the mean utilities may not be linear in customer attributes. The dynamic clustering policy makes no assumption about the structure of the mean utilities with respect to customer attributes. In particular, the transaction dataset from the Chilean retailer exhibits a non-linear dependence between mean utilities and attributes. In such cases, the linearity assumption could result in inaccurate estimates which, in turn, could hurt the retailer's revenue. Also, the dynamic clustering policy takes a Bayesian approach and is designed to expedite the learning process, especially in the short-term when the amount of transaction data is limited. Overall, the dynamic clustering policy leads to 27.3% more transactions (on average) than the linear-utility policy in the experiments based on the dataset under the MNL demand model. The linear-utility approach, however, uses maximum-likelihood estimation and thus is better suited for (offline) settings with large amounts of transaction data. As such, the linear-utility approach can identify whether an attribute is statistically significant (relevant) through the estimates of the attribute-specific parameters. One can make similar observations based on the output of the dynamic clustering policy. For example, as noted in Figure 3.6, customers in Santiago West tend to have different preferences for products than customers from Santiago East.

Because the linear-utility approach may be better suited for an offline setting, we designed an additional set of experiments which mimic an offline setting. More specifically, because the estimation approach (and underlying model assumption) is different under the dynamic clustering and linear-utility policies, we further compare the performance of these policies in a setting in which the performance is mainly affected by the quality of estimation which, in turn, impacts the assortment decisions made by different policies. To that end, we consider separation-based versions

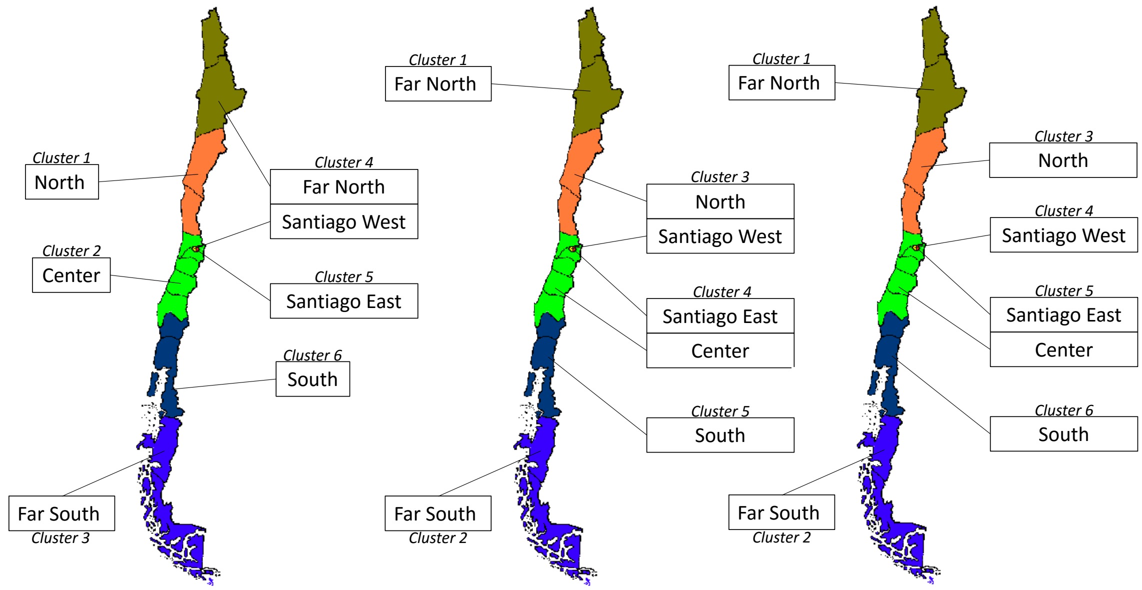

FIGURE 3.6: Illustration of clusters (under the most likely mapping) for women from age groups $[0, 29]$ (left), $[30, 39]$ (middle), and $[40, 99]$ (right) based on customers' location in Chile.

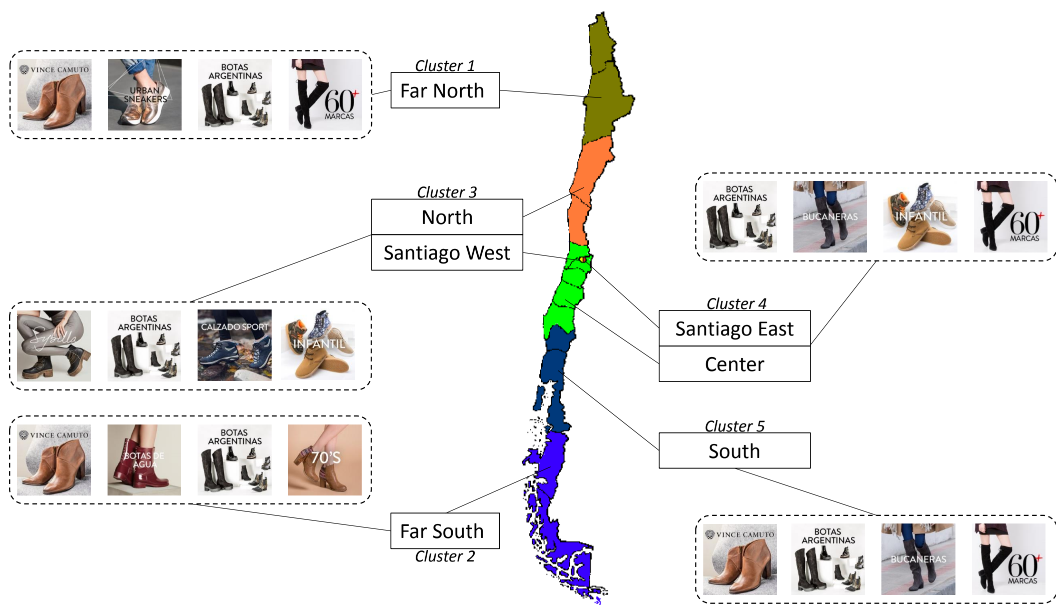

FIGURE 3.7: Illustration of optimal assortments for women from age group $[30, 39]$.

of these policies, which separate exploration from exploitation. In the separation-based experiments, we randomly generate a sample of transaction data with random assortment offerings and random customer arrivals based on the estimated MNL parameters and arrival distribution of profiles from the dataset. All policies use the same sample to estimate the MNL parameters (exploration stage). Each policy then finds the optimal assortment for each customer profile. We then generate another random sample of customer arrivals with the same size as that used in the exploration phase (according to the estimated arrival distribution) over which each policy offers its personalized optimal assortment to each arriving customer (exploitation stage). We consider the cumulative regret and revenue of all policies only for the exploitation phase as all policies use the same sample (and thus incur the same regret) in the exploration phase. As a result, any difference in performance is due to the quality of estimation. We consider two scenarios in terms of the underlying demand model.

The first scenario is based on the original transaction dataset, which exhibits a non-linear dependence of mean utilities on customer attributes. We experiment with different sample sizes T. The dynamic clustering policy outperforms both the data-intensive and linear-utility policies in all instances. Table 3.3 below reports the percentage improvement of the dynamic clustering policy over the other policies, both in terms of regret and revenue (i.e., expected number of transactions). The revenue improvements can be as high as 50% and 25.4% compared to the data-intensive and linear-utility policies, respectively. Moreover, the improvements are generally higher for smaller samples sizes, as the dynamic clustering policy performs particularly well in settings with limited transaction data by pooling information. Table 3.4 reports the average estimation times (i.e., computation time of MCMC and MLE for the dynamic clustering and linear-utility policies, respectively). As expected, the data-intensive policy has the lowest computation time as the estimation can be done independently for each customer profile. The linear-utility model is faster than the dynamic clustering policy for smaller sample sizes. However, its computation time increases significantly for larger sample sizes, suggesting that the linear-utility policy may be better suited for an offline setting.

Table 3.3: Average percentage improvement of dynamic clustering policy over data-intensive and linear-utility policies in separation-based experiments using the original dataset.

	Regret		Revenue	
	Data-Intensive	Linear-Utility	Data-Intensive	Linear-Utility
$T = 500$	34.2%	24.6%	47.8%	25.4%
$T = 1000$	42.6%	28.4%	50.0%	21.6%
$T = 5000$	39.0%	26.6%	20.7%	10.7%
$T = 10000$	24.3%	12.9%	8.9%	3.9%
$T = 20000$	6.7%	3.6%	1.5%	0.8%

Table 3.4: Average estimation time (in seconds) in separation-based experiments based on original dataset.

	$T = 500$	$T = 1000$	$T = 5000$	$T = 10000$	$T = 20000$
Dynamic Clustering	1.214	1.301	2.307	3.861	7.010
Data-Intensive	0.003	0.006	0.025	0.048	0.096
Linear-Utility	0.508	0.922	5.872	13.877	31.168

While the original dataset does not exhibit a linear dependence on customer attributes, we generate a second set of (synthetic) experiments in which the underlying demand model is linear as in (3.7). To that end, we estimate β_j in model (3.7) for each product j from the dataset and use such estimates to randomly generate synthetic (linear) transaction data for simulation. We find that the dynamic clustering policy outperforms the linear-utility approach in all instances except for that with the largest transaction sample (i.e., $T = 20000$) – please refer to Table B.1 in Appendix B.2 for a summary of the results of this numerical study. This suggests that the benefit of pooling information achieved by clustering can lead to better performance even when the underlying demand model is, in fact, linear. Such benefits are more pronounced in the short-term, i.e., for small and moderate sample sizes.

To conclude, the dynamic clustering policy outperforms the linear-utility approach in the case study based on the Chilean retailer's dataset. Each approach has its advantages and may be deemed more appropriate depending on the specific setting. For example, the retailer that provided the dataset is interested in identifying customer segments (i.e., clusters of customers with similar preferences). These segments are part of the output of the dynamic clustering policy and can be interpreted based on customers' attributes (as in the examples in Figures 3.6 and 3.7).

3.6 Value of Pooling Information

In this section, we provide analytical support for the insights derived in the case study in Section 3.5. More specifically, this section explores the impact of pooling information about customers' preferences on the retailer's revenue by considering a stylized version of the dynamic assortment personalization problem. To this end, we focus on three policies that differ in the extent by which they aggregate information across customers. The data-intensive policy, described in Section 3.5.2, treats customer profiles independently to estimate preferences and make assortment decisions. We further introduce a *semi-oracle* policy that knows upfront the underlying mapping of profiles to clusters but not the customer preferences for each cluster. The semi-oracle policy reflects the key element of the dynamic clustering policy – in that it pools transaction information across customers with similar preferences – but it bypasses the estimation of the mapping of profiles to clusters by assuming that it is known to the retailer. Working with the dynamic clustering policy analytically is not possible, as it requires a Bayesian update of the mapping that cannot be done in closed-form. In the other extreme, we consider a *pooling* policy that aggregates transaction data across *all* customer profiles (regardless of whether the customers have similar preferences or not). We show that the semi-oracle outperforms the data-intensive policy. Moreover, we analytically characterize settings in which the pooling policy outperforms the data-intensive policy.

All policies we consider in this section – the data-intensive policy ($\pi^{d\text{-}int}$), the semi-oracle policy ($\pi^{s\text{-}orc}$), and the pooling policy (π^{pool}) – follow the same bandit algorithm to determine what assortment to offer each arriving customer. The policies, however, differ in how they use the available information to estimate the customers' preferences and make assortment decisions. For ease of analysis, we focus on the *independent* demand model in this section.[8] We also assume, for tractability, that $C = 1$ and $r_j = 1$ for all products $j \in \mathcal{N}$.

Let $R^{\pi^{d\text{-}int}}$, $R^{\pi^{s\text{-}orc}}$, and $R^{\pi^{pool}}$ denote the regrets associated with the data-intensive, semi-oracle, and pooling policies, respectively. To simplify notation, we denote them as $R_{d\text{-}int}$, $R_{s\text{-}orc}$, and R_{pool} hereafter. We further define the gap functions

$$G_1 := R_{d\text{-}int} - R_{s\text{-}orc} \quad \text{and} \quad G_2 := R_{d\text{-}int} - R_{pool}.$$

Our goal is to determine conditions under which these gaps are non-negative. Because characterizing the regret functions in closed-form is not possible, we use upper bounds on the regret for the semi-oracle and pooling polices, denoted by $U_{s\text{-}orc}$ and U_{pool}, respectively, and a lower bound on the regret for the data-intensive policy, denoted by $L_{d\text{-}int}$. Therefore, $L_{d\text{-}int} - U_{s\text{-}orc}$ provides a lower bound for the gap function G_1 and $L_{d\text{-}int} - U_{pool}$ provides a lower bound for the gap function G_2. Hence, we focus on characterizing settings in which these lower bounds are non-negative, which in turn implies that $G_1, G_2 \geqslant 0$.[9] Lai and Robbins (1985*a*) prove an asymptotic lower bound (for large T) on the achievable performance of any *consistent* policy in the classic bandit setting (i.e., with a homogeneous population of customers).[10] Roughly speaking, the long-run number of mistakes (associated with pulling suboptimal arms) under any consistent policy is smaller than T^a for large T and every $a > 0$. In particular, it is smaller than a linear function of T, which corresponds to making

[8] One can obtain similar results for the MNL demand model as well.

[9] While the lower bounds are not always tight, the goal is to show the non-negativity of the gap functions. As a result, working with the bounds enables the analysis and leads to the desired results.

[10] An admissible policy π is consistent if, for any distribution of preferences F (that satisfies certain regularity conditions), $\frac{R^\pi(T)}{T^a} \to 0$, as $T \to \infty$, for every $a > 0$. That is, if $R^\pi(T) = o(T^a)$. See Lai and Robbins (1985*a*).

mistakes for every customer. Let $\mathcal{P}' \subseteq \mathcal{P}$ denote the set of consistent admissible policies. We restrict attention to consistent policies $\pi \in \mathcal{P}'$ and use Lai and Robbins' lower bound to derive $L_{d\text{-}int}$. More specifically, we derive a lower bound on the regret associated with each profile and define $L_{d\text{-}int}$ as the sum of these lower bounds.

The upper bounds on the regrets for the semi-oracle and pooling policies depend on the specific bandit algorithm used for selecting the product to offer each arriving customer. We focus here on the celebrated upper confidence bound (UCB1) policy of Auer et al. (2002). After an initialization phase during which each product is offered once, UCB1 offers customer t a product j with the highest index $\bar{\mu}_j + \sqrt{2\ln(t-1)/k_j(t-1)}$, where $\bar{\mu}_j$ is the sample mean of the number of purchases for product j, and $k_j(t-1)$ is the number of times that product j has been offered up to (and including) time $t-1$. The UCB1 policy is easy to implement and its regret admits a finite-time upper bound which is simple to use. We extend the results of Sections 3.6.1 and 3.6.2 for Thompson Sampling in Appendix B.2.

3.6.1 Semi-Oracle

In this section, we compare the performance of the data-intensive policy to that of the semi-oracle policy. As expected, the semi-oracle outperforms the data-intensive policy in terms of regret (i.e., revenue). This result emphasizes the benefit of estimating the mapping of profiles to clusters (as in the dynamic clustering policy) as it helps expedite the learning process by pooling transaction information across customer profiles within a cluster. (We provide a formal statement and proof of the result in Theorem 33 in Appendix B.1.)

We next show that there are diminishing marginal returns to pooling information from an increasing number of customer profiles. To that end, consider a general market with K clusters where $1 \leqslant K < I$. We assume, without loss of generality, that $K < I$, since if $K = I$, then both the semi-oracle and data-intensive policies incur the same regret and therefore $G_1 = 0$. Let $\mathcal{I}_k$ denote the set of profiles belonging to cluster k and $I_k := |\mathcal{I}_k|$. Also, let $\mathcal{I}' := (\mathcal{I}_1, \ldots, \mathcal{I}_K)$. This vector summarizes the mapping of profiles to clusters. We assume, without loss of generality, that product 1 has the highest purchase probability for each profile, i.e., $\mu_j^i < \mu_1^i$ for $j = 2, \ldots, N$ and all $i \in \mathcal{I}$. Computing the lower bound $L_{\textit{d-int}} - U_{\textit{s-orc}}$ for the gap function G_1 requires an additional approximation as the setting studied in Lai and Robbins (1985a) considers a homogeneous population of customers. This additional approximation involves a first-order Taylor expansion and, as such, the resulting approximate lower bound is very close to $L_{\textit{d-int}} - U_{\textit{s-orc}}$. We denote by $G_{1l}(T, \mathcal{I}')$ the approximation to the lower bound for the gap function G_1. This approximate lower bound depends on the total number of customer arrivals T and on the vector $\mathcal{I}'$ which encodes the mapping of profiles to clusters. We provide a detailed derivation of $G_{1l}(T, \mathcal{I}')$ in Appendix B.1.

Theorem 13. *Consider the case of uniform arrivals within each cluster, i.e.,* $p_i = P_k/I_k$ *for all* $i \in \mathcal{I}_k$ *where* $P_k := \sum_{i \in \mathcal{I}_k} p_i$ *is constant. We then have that* $G_{1l}(T, \mathcal{I}')$ *is increasing in* I_k *for* $T > eI_k/P_k$ *(*e *is the Euler's number) and concave in* I_k *for* $T \geqslant 1$*.*

Theorem 13 shows the first- and second-order effects of the number of customer profiles I_k on the approximate lower bound. We find that, for sufficiently large T, $G_{1l}(T, \mathcal{I}')$ is increasing in the number of profiles I_k. That is, the benefit of pooling information increases with the number of profiles within any cluster k, as it becomes increasingly more time-consuming for the data-intensive policy to learn the preferences of each customer profile when I_k increases. In addition, the result shows that $G_{1l}(T, \mathcal{I}')$ is concave in I_k. That is, there are diminishing marginal returns to pooling information from an increasing number of customer profiles within any cluster.

We also explore whether the result in Theorem 13 applies to the dynamic clustering policy and MNL demand model by considering an example based on the dataset from the Chilean retailer. Specifically, we estimate the MNL parameters for the customer profile (female, $[40, 99]$, Center) from the dataset and assume that a cluster's demand follows such MNL model. We then increase the number of profiles in that cluster and evaluate the gap between the regrets of the data-intensive and dynamic clustering policies in a market with $T = 5000$ customers. Figure 3.8 shows the result of this experiment, where $R_{d\text{-}int}$ and R_{dc} denote the regrets of data-intensive and dynamic clustering policies, respectively. As noted in the graph, and consistent with Theorem 13, the (actual) gap between the regrets of the two policies is increasing and concave in the number of customer profiles within the cluster.

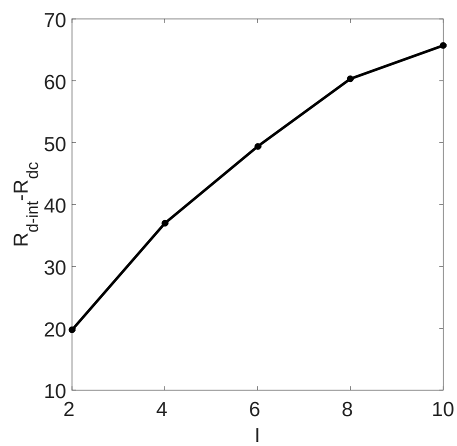

FIGURE 3.8: Gap between the regrets of data-intensive and dynamic clustering policies as a function of number of profiles.

3.6.2 Pooling in the Short-Term

In this section, we consider a setting with heterogeneous customers and a *pooling* policy that aggregates information across *all* customer profiles. As one would expect, pooling information across all customer profiles is not necessarily beneficial for the retailer in a heterogeneous market as it could lead to erroneous estimates. However, we show that, under some conditions, the pooling policy tends to outperform the data-intensive policy in the short-term even if customer preferences are heterogeneous. This, in turn, allows us to examine the key drivers of efficiency gains derived by pooling information.

Consider a market with $K \geqslant 2$ clusters. Without loss of generality, we assume that $K = N$, where N is the number of products. We also assume that cluster k's customers have the highest purchase probability for product k, for $k = 1, \ldots, K$. Furthermore, we assume that $\mu_k^k - \mu_j^k = \Delta$ for some $\Delta > 0$ and for all $k = 1, \ldots, K$ and $j \neq k$, where, to simplify notation, μ_j^k denotes the purchase probability of product j for all profiles in cluster k. Let $P := (P_1, P_2, \ldots, P_K)$ where $P_k = \sum_{i \in \mathcal{I}_k} p_i$ is the proportion of profiles belonging to cluster k. We also define $P' := (P_2, \ldots, P_{K-1})$. We assume, without loss of generality, that $P_k < P_1 \leqslant 1$ for all $k = 2, \ldots, K$. Note that P_1 is a measure of heterogeneity in this setting and a smaller P_1 leads to a more heterogeneous customer population. In the extreme, $P_1 = 1$ reduces this setting to one with a homogeneous market.

As in Section 3.6.1, we consider an approximate lower bound on the regret of the data-intensive policy, based on Lai and Robbins (1985*a*). Existing upper bounds in the bandit literature (including that of the UCB1 policy) assume a homogeneous market. This introduces additional complexity in the computation of the upper bound on the regret of the pooling policy. This upper bound is derived in Appendix B.1. Let $G_{2l}(t, I, P)$ be the approximation of the lower bound $L_{\textit{d-int}} - U_{\textit{pool}}$ to the gap function G_2 at any time period t. We provide a detailed derivation of $G_{2l}(t, I, P)$ in Appendix B.1. The next result provides conditions under which the pooling policy outperforms the data-intensive policy (subject to the approximations), that is, $G_{2l}(t, I, P) \geqslant 0$.

Theorem 14. *Consider the case of uniform arrivals, i.e.,* $p_i = 1/I$ *for all* $i \in \mathcal{I}$. *There exist thresholds* $\tilde{I}_l(P)$ *and* $\tilde{P}_1(I, P')$ *such that if* $I \geqslant \tilde{I}_l(P)$ *and* $\tilde{P}_1(I, P') < P_1 \leqslant 1$, *then*

$$G_{2l}(t, I, P) \geqslant 0 \quad \text{for} \quad \tilde{t}_l(I, P) \leqslant t \leqslant \tilde{t}_u(I, P),$$

with $1 < \tilde{t}_l(I, P) \leqslant \tilde{t}_u(I, P) \leqslant \infty$. *Moreover,* $\tilde{I}_l(P)$ *and* $\tilde{P}_1(I, P')$ *are non-increasing in* P_1 *and* I, *respectively.*

The result in Theorem 14 shows that, under some conditions, the pooling policy tends to outperform the data-intensive policy for a range of customer arrivals, even if the retailer learns the customers' preferences inaccurately in a heterogeneous market under the pooling policy. This is the result of faster learning under the pooling policy achieved by aggregating information across all customer profiles. In particular, Theorem 14 illustrates the benefit of pooling information in the short-term, when transaction data is limited. Moreover, Theorem 14 implies that three key factors favor the performance of the pooling policy over the data-intensive policy:

- *Heterogeneity* (P_1): If the population is not too heterogeneous (i.e., if $P_1 > \tilde{P}_1(I, P')$), then the pooling policy tends to outperform the data-intensive policy for a range of customer arrivals. This is because, under such condition, the benefit associated with faster learning by aggregating information outweighs the cost associated with the errors the pooling policy makes by not differentiating between clusters (and therefore offering suboptimal assortments). Moreover, the threshold $\tilde{P}_1(I, P')$ decreases as the number of profiles increases.

- *Number of profiles* (I): An increase in the number of profiles impacts negatively on the performance of the data-intensive policy. As I increases, the average number of customer arrivals per profile decreases and thus it takes longer for the data-intensive policy to learn the preferences for each profile. On the other hand, the pooling policy aggregates information across all customers and thus its performance does not degrade as long as $P_1 > \tilde{P}_1(I, P')$. As the population becomes more homogeneous in terms of preferences (i.e., as P_1 increases), the pooling policy tends to outperform the data-intensive policy for an even smaller number of customer profiles.

- *Number of Customers* (t): Although a relatively more homogeneous market and a large number of profiles can favor the performance of the pooling policy, the *key* factor is the amount of transaction information available to the retailer. When the number of customer arrivals is still relatively small, the data-intensive policy does not have enough sample points to accurately learn the preference of each customer profile. On the other hand, the pooling policy aggregates information and therefore tends to outperform the data-intensive policy as long as the population is not too heterogeneous with respect to their product preferences. As more customers arrive, the performance of the data-intensive policy prevails. In particular, $G_{2l}(t, I, P) \to -\infty$ (and $G_2(t, I, P) \to -\infty$) as $t \to \infty$ if $P_1 < 1$.

The result in Theorem 14 is consistent with the observations about the dynamic clustering policy illustrated in the right panel of Figure 3.4. Early on in the selling season, when only a limited number of transactions have been observed, the average number of clusters that emerge from the dynamic clustering policy is small. This echoes the preceding discussion – when limited information is available, the retailer might be better off pooling all available data (even if they correspond to profiles with different preferences) to speed up the learning process. The number of clusters then increases as more data becomes available (as can be noted in the right panel of Figure 3.4) and the retailer is able to personalize the assortment offerings by better matching customer preferences.

3.7 Conclusion

This paper considers a retailer endowed with multiple products that dynamically personalizes the assortment offerings over a finite selling season. Customers are assigned to different profiles based on their observable personal attributes. Their preferences are unknown to the retailer and must be learned over time. The primary goal of the paper is to explore the efficient use of data in retail operations and its benefits (in terms of revenue) for assortment personalization. To that end, we propose the *dynamic clustering* policy as a prescriptive approach for assortment personalization in an online setting. We take advantage of existing tools from the literature and introduce a policy that adaptively combines estimation (by estimating customer preferences through dynamic clustering) and optimization (by making dynamic personalized assortment decisions using a bandit policy) in an online setting. The dynamic clustering policy adaptively adjusts the composition of customer segments (i.e., mapping of profiles to clusters) based on the observed customers' purchasing decisions. The policy exploits the similarity in preferences of customers in the same cluster by aggregating their transaction information and expediting the learning process. Using the estimated mapping and preferences, the policy uses existing bandit algorithms to make assortment decisions.

To illustrate the practical value of the dynamic clustering policy in a realistic setting, we apply the policy to a dataset from a large Chilean retailer. We compare the performance of the dynamic clustering policy with two alternatives: a data-intensive policy that treats each customer profile independently and a linear-utility policy that estimates product mean utilities as linear functions of customer attributes. The case study suggests that the dynamic clustering policy can significantly increase the average number of transactions relative to the other policies. We also demonstrate the scalability and efficiency of the dynamic clustering policy in terms of computation time.

We then study a simplified version of the problem in which the retailer offers a single product to each arriving customer. We show that a semi-oracle policy that knows upfront the mapping of profiles to clusters (but not the customers' preferences) outperforms the data-intensive policy, indicating that pooling information is beneficial for the retailer. We also demonstrate that there are decreasing marginal returns to pooling information as the number of customer profiles increases. Finally, we characterize conditions under which a policy that pools information across all customer profiles outperforms the data-intensive policy even when customer preferences are heterogeneous. This result emphasizes the benefit of pooling information in the short-term, when there is insufficient data to accurately estimate preferences for each customer profile.

In this work, we have made some simplifying assumptions for tractability. Future work can take into consideration the presence of inventory constraints in the model. Moreover, we have assumed that prices are constant throughout the selling season. Incorporating pricing decisions is another direction for future research. Because the proposed approach relies on MCMC for estimation, the computational cost is a potential concern for implementing the policy in a high-dimensional setting in real-time; nonetheless, the paper discusses several approaches to expedite the computation. Furthermore, the proposed approach does not assume any particular relation between the products' mean utilities and customer attributes. In a high-dimensional setting, imposing further structure on the utility model (e.g., similar to the linear-utility approach) is a potential alternative to expedite the estimation.

4

Uncertainty in Choice: The Role of Return Policies

4.1 Introduction

Merchandise returns account for a considerable portion of annual retail sales. According to the National Retail Federation (Anonymous, 2017), total merchandise returns in 2017 reached $351 billion for US retailers. This amount is equivalent to 10% of the total US retail sales in 2017. The size of returns is overwhelming as it is close to the estimated 2017 federal budget deficit of $400 billion. Returns are costly for the retailers as they result in reverse logistic costs for handling the returns. Product returns cost manufacturer and retailers more than $100 billion per year, or an average loss per company of about 3.8% in profit (Petersen and Kumar, 2010). According to AlixPartners, items returned to the store cost a retailer roughly $3 to process and are available for resale within a day. Items shipped back to a distribution center or third-party logistics provider cost $6 or more to process and take at least four days before they are available for resale (Phillips, 2017).

Despite its handling costs, retailers recognize the value of offering a return option to customers as it allows them to return unwanted products. This is especially important for online shoppers as they are unable to fully evaluate the product at the time of purchase (e.g., the exact fit of a piece of clothing). As a result, many customers rely on product returns as part of their shopping process.

To reduce the cost and frequency of product returns, some companies create product-return disincentives such as limited time frames for returns (e.g., within 30 days after purchase) or charging a return fee (e.g., a restocking fee). However, several recent studies have shown the potential positive impacts of returns. For example, Petersen and Kumar (2010) find that when a company has a lenient return policy, customers are more willing to make other purchases. Thus, the question of how lenient the retailers should be in offering their return policies still remains somewhat unanswered: on one hand, the retailer incurs reverse logistic costs to handle returned products; on the other hand, however, customers value the option of return and this, in turn, could have positive impacts on retailer's (future) sales.

In this paper, we study the interplay between a retailer's return and pricing policies and customers' purchasing decisions. More specifically, we model the customers' purchasing and returns decisions using a discrete choice model and study how such decisions are impacted by the retailer's pricing and return policies. We study three main research questions in this work: (*i*) how the pricing decision of the retailer is impacted by product returns; (*ii*) what factors make the return option favorable for the retailer (as opposed to not offering the return policy); (*iii*) what is the impact of accounting for product returns on demand estimation.

We study the seller's revenue maximization problem in the cases with and without product returns. In the returns case, we consider a policy in which the seller fully reimburses the customer for a returned product. We then model the customers' purchasing and returns decisions using a discrete choice model. More specifically, in the case without product returns, we model the customers' purchasing decisions as a Multinomial Logit (MNL) model. In the case with product returns, on the other hand, we use a Nested MNL (N-MNL) to model customer's purchasing and returns decisions. In such case, customer's choice can be viewed as a sequential (two-stage) process in which the customer first decides between purchasing a product from the offered assortment or choosing the outside option. Conditional on a product purchase, the customer then decides whether to keep or return the purchased product.

We first focus on the single-product case and characterize the seller's optimal price in the cases with and without product returns. The results show that offering the return option results in different (but not necessarily higher) optimal prices compared to the case with no returns. Moreover, we find that offering the return option is not always favorable for the retailer. In addition, we characterize conditions under which offering the return option to customers increases the retailer's revenue.

We also numerically study the impact of accounting for product returns on demand estimation. We experiment with a dataset from a large Chilean retailer (this dataset was used in the second essay in this dissertation). The preliminary numerical results suggest that our N-MNL model, which accounts for product returns, increases demand estimation accuracy compared to models that do not consider product returns in their estimation.

The remainder of this paper is organized as follows. Section 4.2 reviews the relevant literature. Section 4.3 describes the customer's choice model and Section 4.4 models the seller's revenue maximization problem. Section 4.5 discusses the preliminary analytical results. Section 4.6 numerically studies the impact of accounting for product returns on demand estimation. Section 4.7 provides concluding remarks and future research directions. All proofs are relegated to Appendix C.1.

4.2 Literature Review

Product returns and their impact on customers and retailers have been extensively studied in the literature from both theoretical and empirical perspectives. We first briefly discuss the analytical work and then review the empirical research.

There is an extensive body of literature on return policies. The classic work of Pasternack (1985) studies the optimal pricing and return policies of a manufacturer in a newsvendor setting. Davis et al. (1995) develop a simple model to characterize conditions under which money-back guarantees are optimal. Che (1996) shows that the seller adopts the (full-refund) return policy (as opposed to a no-return policy) when consumers are highly risk averse or retail costs are high. Moorthy and Srinivasan (1995) find that full refunds may signal high quality. Several other papers focus on preventing inappropriate returns when customers have no intention of keeping the purchased products. For example, Hess et al. (1996) show that in the context of mail-order catalogs, inappropriate returns can be controlled by imposing nonrefundable charges that increase with the value of the merchandise ordered. Davis et al. (1998) develop an analytical model to identify potential causes of variation among retailers' return policies and support their insights empirically.

More recently, Shulman et al. (2009) develop an analytical model to characterize the seller's optimal pricing and restocking fee policy while taking into account the customers' strategic behavior. They also identify conditions under which it is (or not) optimal to provide product fit information to customers. Su (2009) studies the optimal price, quantity, and refund amount decisions of a seller in a newsvendor setting and investigates the impact of different return policies on supply chain performance. Altug and Aydinliyim (2016) study the impact of consumer's strategic purchase deferrals on an online retailer's return policy. Nageswaran et al. (2017) study the pricing and return policy decisions of an omnichannel retailer.

In contrast to the papers discussed above, we model the customer's purchasing and returns decisions using a discrete choice model. Closest to ours in terms of the model is the work by Alptekinoğlu and Grasas (2014) which uses the N-MNL framework to model the customer's choice. The authors study how the optimal assortment decision of a retailer is impacted by its return policy. We, however, focus on the pricing decision of the retailer. Moreover, the return policy is a decision variable in our model while in Alptekinoğlu and Grasas (2014) the return policy (and prices) are exogenous. Wang and Sahin (2017) study the impact of consumer search cost on assortment and pricing decisions of a retailer. They model the consumer choice as a two-stage "consider-then-choose" process by incorporating the search cost into the consumer choice model. In a similar setting, Wagner and Martınez-de Albéniz (2017) study pricing and assortment decisions of a retailer when customers learn about products' value in a sequential product exchange process.

Product returns have also been widely studied using empirical studies. We provide a few examples here. Petersen and Kumar (2009) empirically determine the factors that help explain product return behavior and the consequences of product returns on future customer and firm behavior. The paper shows that increases in product return increase future customer purchases up to a threshold. Anderson et al. (2009) develop a structural model to measure the value of the return option to consumers. Petersen and Kumar (2015) conduct a large-scale field experiment and show that the firm can increase its profit by accounting for the customer's perceived risk related to product returns in addition to managing product return costs. In our paper, we aim at understanding the impact of accounting for product returns on demand estimation accuracy using a discrete choice model. To the best of our knowledge, this issue has not been studied in the existing literature.

4.3 Customer's Choice Model

In this section, we model the customer's choice given the seller's return policy. Suppose that a customer arrives at the store and faces a category of substitutable products. The full assortment consists of n products, but the seller offers a subset or *assortment* S. We model the customer's purchasing decision using a discrete choice model. Because offering the return option to the customer could impact the customer's purchasing decision, we consider the following two scenarios.

4.3.1 Non-Returnable Products

In this case, the products are non-returnable, and thus a purchase means a *final sale*. We model customer's purchasing decision using a Multinomial Logit (MNL) model. More specifically, let product 0 denote the outside option (e.g., the no-purchase option or purchasing the product from a competing retailer). The utility of choosing product $i \in S \cup \{0\}$ is given by a linear-in-parameter function of the form[1]

$$U_i^{NR} := w_i + \mu_1 \epsilon_i, \quad \text{where } w_i := \beta_i - p_i, \tag{4.1}$$

and where β_i and p_i denote the attractiveness (i.e., nominal mean utility) and price of product i, respectively. The noise terms ϵ_i are unobservable by the seller and are i.i.d. random variables following a standard Gumbel distribution. Moreover, $\mu_1 > 0$ is the degree of *heterogeneity* between products[2]. Without loss of generality, we set the mean utility of the outside option $w_0 = 0$.

The customer purchases the product (including the outside option) with the highest utility U_i^{NR}. According to the MNL model, the purchase probability of product $i \in S \cup \{0\}$ is given by

$$\mathbb{P}_i^{NR} := \frac{\exp(w_i/\mu_1)}{1 + \sum_{h \in S} \exp(w_h/\mu_1)}.$$

(Note that $\mathbb{P}_i^{NR}$ depends on the assortment S, but we drop such dependence to simplify the notation.)

---

[1] The superscript NR in U_i^{NR} indicates the case with no product returns.

[2] Note that having the noise term $\mu_1 \epsilon_i$ in (4.1) is equivalent to having a noise term that follows a Gumbel distribution with location 0 and scale μ_1.

4.3.2 Returnable Products

In this case, we model the customer's purchasing and returns decisions using the Nested Multinomial Logit (N-MNL) framework in order to capture the impact of a return option on the customer's decision. In the N-MNL model, customer's choice can be viewed as a sequential (two-stage) process in which the customer first decides between purchasing a product from the offered assortment or choosing the outside option. Conditional on a product purchase, the customer then decides whether to keep or return the purchased product. We discuss the model in further detail in what follows, starting from the second stage.

Second Stage. Suppose that the customer has purchased product $i \in S$ in the first stage. In the second stage, the customer revises her (perceived) utility for the product after trying it out and evaluating other factors which were bypassed at the time of purchase (e.g., similarity with other comparable items that she already has, change of mind, others' opinions, etc.). At this point, the customer decides whether to keep or return the purchased product. (Note that if the customer chooses the outside option in the first stage, the second stage is no longer relevant.) The utility that the customer obtains from keeping or returning the product is given by

$$U_i^{keep} := w_i + \mu_2 \ \epsilon_i^{keep}, \quad U_i^{return} := -c + \mu_2 \ \epsilon_i^{return},$$

where w_i is the mean utility of product i minus its price and $c > 0$ is the hassle cost that the customer incurs in case she decides to return the product (e.g., the additional trip to the store). Moreover, ϵ_i^{keep} and ϵ_i^{return} denote the customer's post-purchase uncertainty which are i.i.d. random variables following a standard Gumbel distribution, and $\mu_2 > 0$ is the degree of heterogeneity between the two options (i.e., a measure of post-purchase heterogeneity). Note that for simplicity, we assume that in case of product returns, the seller fully reimburses the customer. This represents a lenient return policy which seems to be common practice, at least for fashion products (one can modify the model to represent the case in which the seller only partially reimburses the customer).

Under the N-MNL model, the conditional probability of returning the product $i \in S$ given that it was purchased in the first stage follows an MNL model and is given by:

$$\mathbb{P}_i^{return} := \frac{\exp\left(\frac{-c}{\mu_2}\right)}{\exp\left(\frac{-c}{\mu_2}\right) + \exp\left(\frac{w_i}{\mu_2}\right)}.$$

First Stage. In the first stage, the customer decides between purchasing a product from the offered assortment or choosing the outside option. At this point, the customer does not know for certain if she would keep or return the product in the second stage (i.e., she cannot observe the realization of the post-purchase uncertainties ϵ_i^{keep} and ϵ_i^{return}) if she decides to purchase the product in the first stage. Therefore, the customer chooses an option with the highest utility while taking into account the uncertainty in the second stage. More specifically, the utility that the customer obtains from choosing product $i \in S \cup \{0\}$ is given by[3]

$$U_i^R := A_i + \mu_1 \epsilon_i, \tag{4.2}$$

where A_i denotes the expected utility of purchasing product $i \in S$ (i.e., after re-evaluating the product and making the final return/keep decision in the second stage). More specifically, we have that $A_i := \mathbb{E}\left(\max\left\{w_i + \mu_2\ \epsilon_i^{keep}, -c + \mu_2\ \epsilon_i^{return}\right\}\right)$ for $i \in S$ where $\mathbb{E}$ denotes the expectation. It can be shown that (see, e.g., Anderson and De Palma (1992)) A_i is:

$$A_i = \mu_2 \ln\left(\exp\left(\frac{-c}{\mu_2}\right) + \exp\left(\frac{w_i}{\mu_2}\right)\right), \text{ for } i \in S, \tag{4.3}$$

[3] The superscript R in U_i^R indicates the case with product returns.

up to an additive constant, and we set $A_0 = k(c) \leqslant 0$ for the outside option (i.e., product 0), where $k(c)$ is the expected utility of the outside option (in the case of returnable products) which is an increasing function of the hassle cost c. (We discuss this function in more detail in what follows.) Moreover, ϵ_i in (4.2) denote the customer's pre-purchase uncertainty which are i.i.d. random variables following a standard Gumbel distribution, and $\mu_1 > 0$ is a measure of pre-purchase heterogeneity – this is the same constant as that in (4.1). (We need the technical condition of $\mu_1 \geqslant \mu_2$ for the validity of the N-MNL model (Anderson and De Palma 1992). This assumption has been used in the literature and is reasonable – see Alptekinoğlu and Grasas (2014).)

We then have that the probability of choosing product $i \in S \cup \{0\}$ in the first stage is given by

$$\mathbb{P}_i^R := \frac{\exp(A_i/\mu_1)}{\exp(k/\mu_1) + \sum_{h \in S} \exp(A_h/\mu_1)}.$$

Similar to the case with no product returns, we drop the dependence of $\mathbb{P}_i^R$ on the assortment S to simplify the notation. We also denote $k(c)$ by k for simplicity.

Model Discussion. As mentioned before, the flexibility of being able to return a product may impact the customer's purchasing decision. This is accounted for in our model by formulating the returns case with a N-MNL as opposed to the MNL model for the no-returns case. However, the return option could also impact the customer's evaluation of the outside option. In particular, in the case of non-returnable products, the customer probably values her outside option more than that in the case with returnable products (e.g., this could capture the existence of competition between retailers). This is captured in our model: the mean utility of the outside option in the case with no returns is $w_0 = 0$, while in case with returns, the mean utility of the outside option is $A_0 = k(c) \leqslant 0$. Moreover, the customer's evaluation of the outside option (e.g., purchasing the product from a competing retailer) probably depends on the customer's hassle cost c (e.g., the cost of making an additional trip to the store to return the product). To that end, we make the following assumptions about the function $k(c)$ and provide our justification for them:

- $k(c) \leqslant 0$ for all $c > 0$ and $k(0) < 0$: as discussed in above, it may be reasonable to assume that the outside option is less valuable for the customer in the case with returns; therefore, we assume that $k(c) \leqslant 0$ for all $c > 0$. Moreover, if the hassle cost is $c = 0$, the outside option in the case with returns should be strictly less valuable than that in the no-returns case. This is captured by $k(0) < 0$.

- $k(c)$ is an increasing function of the hassle cost c: as the hassle cost increases in the case with returns, the customer would value her outside option more as it becomes increasingly more costly for the customer to return a purchased product.

- $k(c) \to 0$ as $c \to \infty$: if the hassle cost is extremely large, the customer, in principle, does not care about the return option as it would be extremely costly to return a product. This reduces the case with returns to that with no returns. This is also evident in the purchase and return probabilities: $\mathbb{P}_i^R \to \mathbb{P}_i^{NR}$ and $\mathbb{P}_i^{return} \to 0$ as $c \to \infty$.

A function that satisfies the assumptions above is $k(c) = -k'/(c+1)$ for some positive constant $k' > 0$.

4.4 Seller's Revenue Maximization Problem

We assume that the seller decides on the price vector $p := (p_1, \ldots, p_n)$, the assortment S to offer, and whether to offer the return option to customers in order to maximize total expected revenue. We consider the following cases.

Non-Returnable Products. If products are non-returnable, the seller's revenue maximization problem can be represented as follows:

$$\max_{S,p} \pi_0(S,p) := \sum_{i \in S} \left(\mathbb{P}_i^{NR} p_i \right),$$

where π_0 denotes the seller's expected revenue in the case with no returns.

Returnable Products. If products are returnable, the seller's revenue maximization problem can be represented as follows:

$$\max_{S,p} \pi_1(S,p) := \sum_{i \in S} \mathbb{P}_i^{R} p_i \left(1 - \mathbb{P}_i^{return} \right),$$

where π_1 denotes the seller's expected revenue in the case with returns.

4.5 Analysis and Results

In this section, we discuss some preliminary analytical results. In particular, we focus on the price optimization problem for a simplified version of the seller's problem. In Section 4.5.1, we consider the price optimization problem for the single-product case. In Section 4.5.2, we consider the optimization of a single uniform price for the multi-product case.

4.5.1 Price Optimization for Single-Product Case

Because we only have one product in this case, we drop the subscript i (for products) in this section. We first study the case with no returns.

Non-Returnable Product. In the no-returns case, the revenue function becomes:

$$\pi_0(p) = p \, \frac{\exp\left(\frac{\beta - p}{\mu_1}\right)}{1 + \exp\left(\frac{\beta - p}{\mu_1}\right)}.$$

We define $\phi_0(p) := \mu_1 \left(1 + \exp\left((\beta - p)/\mu_1\right)\right)$. The following result characterizes the optimal price and revenue in the no-returns case.

Proposition 15 (**Optimality**). *$\pi_0(p)$ is unimodal with a maximum at p_0^* where p_0^* is the unique solution to $p_0^* = \phi_0(p_0^*)$ (first-order condition). Moreover, the optimal revenue is given by:*

$$\pi_0^* = \pi_0(p_0^*) = p_0^* - \mu_1 = \mu_1 \exp\left(\frac{\beta - p_0^*}{\mu_1}\right).$$

The next result shows the monotonicity of optimal price and revenue in the product's mean utility.

Proposition 16. *π_0^* and p_0^* are increasing in β.*

Returnable Product. The revenue function in this case is also unimodal. Proposition 17 characterizes the optimal price p_1^* in the case with returns.

Proposition 17 (**Optimality**). *$\pi_1(p)$ is unimodal with a maximum at p_1^* where p_1^* is the unique solution to the first-order condition. Moreover, the optimal revenue is given by:*

$$\pi_1^* = \pi_1(p_1^*) = \frac{\mu_1\mu_2 \exp\left(\frac{\beta+c}{\mu_2}\right)\left(\exp\left(\frac{-c}{\mu_2}\right) + \exp\left(\frac{\beta-p_1^*}{\mu_2}\right)\right)^{\mu_2/\mu_1}}{\mu_1 \exp\left(\frac{p_1^*}{\mu_2}\right)\left(\left(\exp\left(\frac{-c}{\mu_2}\right) + \exp\left(\frac{\beta-p_1^*}{\mu_2}\right)\right)^{\mu_2/\mu_1} + \exp(k/\mu_1)\right) + \mu_2 \exp(k/\mu_1)\exp\left(\frac{\beta+c}{\mu_2}\right)}.$$

The next result shows the monotonicity of optimal price and revenue in problem parameters.

Proposition 18. *p_1^* and π_1^* are: (i) increasing in β, and (ii) decreasing in k.*

Proposition 18 shows that as the product's mean utility increases, the retailer can charge a higher optimal price which, in turn, increases the optimal revenue. The intuition for this finding is as follows: a higher product's mean utility results in higher purchase and smaller return probabilities which allow the retailer to charge a higher optimal price. Moreover, Proposition 18 shows that as the outside option becomes more attractive to customers, the optimal price and revenue decrease.

The dependence of π_1^* on the hassle cost c relies on the specific form of the function $k(c)$ and other problem parameters. However, unlike β and k, it might not be the case that a larger c would always increase (or decrease) π_1^*. The intuition for this finding is as follows: although a larger hassle cost decreases the probability of return, it also decreases the probability of purchase in the first period, because the outside option becomes more favorable and customers take into account their expected utility from purchase in the first stage – a lager hassle cost means a smaller expected utility of purchase for customers (this can be seen from (4.3)). Therefore, the overall impact of changing c on π_1^* is not clear. Figure 4.1 illustrates this point for a setting with $k(c) = -2/(c+1)$, $\beta = 2, \mu_1 = 1$, and $\mu_2 = 0.5$. However, we can prove the following result.

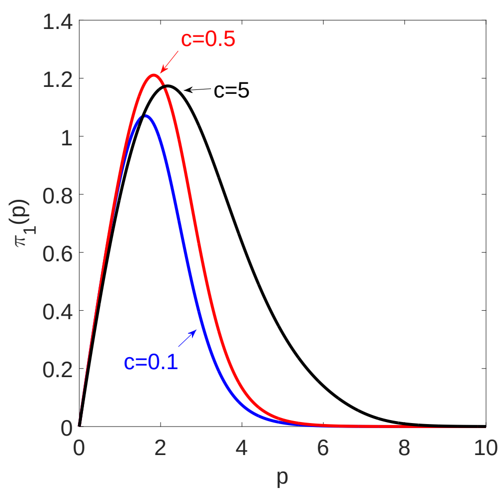

FIGURE 4.1: Revenue function for different hassle costs in the case with a single returnable product.

Proposition 19. *Suppose that $k(c) = -k'/(c+1)$ for some $k' > 0$. There exists a threshold $\tilde{c}$ such that π_1^* is decreasing in c for all $c \geqslant \tilde{c}$.*

Proposition 19 shows that in the case with $k(c) = -k'/(c+1)$, the optimal revenue is decreasing in the hassle cost when such cost is large enough. Thus, if the hassle cost is relatively large, the decrease in the purchase probability outweighs the decrease in the return probability and as a result, the optimal revenue decreases.

For the special case in which the mean utility of the outside option does not depend on the hassle cost, we can also show that the optimal price and revenue are increasing in the hassle cost.

Proposition 20. *Suppose that $k(c) = 0, \forall c$. We then have that p_1^* and π_1^* are increasing in c.*

Returns versus No-Returns. In this section, we compare the optimal price and revenue in the case with and without product returns. In particular, we characterize conditions under which offering (or not) the return option is favorable for the retailer.

We first make a numerical observation.

Remark. We can numerically check that the optimal price p_1^* in the case with returns could be smaller or larger than p_0^*, the optimal price in the case with no returns. It might be possible to analytically characterize conditions under which p_1^* is larger (or smaller) than p_0^*. This is, however, a subject of future research.

We can show that for the case of $k(c) = -k'/(c+1)$, offering the return option is favorable if the hassle cost is sufficiently large.

Proposition 21. *Suppose that $k(c) = -k'/(c+1)$ for some $k' > 0$. There exists a threshold $\tilde{c}$ such that $\pi_1^* \geqslant \pi_0^*$ for all $c \geqslant \tilde{c}$.*

For the special case in which the mean utility of the outside option does not depend on the hassle cost, we can show that not offering the return option is always preferred by the retailer.

Proposition 22. *Suppose that $k(c) = 0, \forall c$. We then have that $\pi_1^* \leqslant \pi_0^*$.*

4.5.2 Single Price Optimization for Multi-Product Case

In this section, we consider the price optimization problem given the full assortment of products. For simplicity, we consider the single price optimization problem, that is, we assume that all products are priced equally – given that products are substitutable, this assumption may be reasonable, and has been used in the literature: see for example Alptekinoğlu and Grasas (2014).

Non-Returnable Products. Similar to the single product case, the revenue function is unimodal in this case.

Proposition 23 (**Optimality**). *$\pi_0(p)$ is unimodal with a maximum at p_0^* where p_0^* is the unique solution to the first-order condition. Moreover, the optimal revenue is given by:*

$$\pi_0^* = \pi_0(p_0^*) = p_0^* - \mu_1 = \mu_1 \left(\sum_{i=1}^{n} \exp\left(\frac{\beta_i - p_0^*}{\mu_1} \right) \right).$$

Returnable Products. The revenue function in this case is not necessarily unimodal as illustrated in Figure 4.2 for an example with two products, and $\beta_1 = 50, \beta_2 = 10, c = 20, \mu_1 = 20, \mu_2 = 0.5$, and $k = -0.2$. It might be possible to analytically characterize conditions under which the revenue function $\pi_1(p)$ is unimodal in the multi-product case, however, this is a subject of future research.

4.6 Impact of Product Returns on Demand Estimation

In this section, we numerically study the impact of accounting for product returns on demand estimation. More specifically, we compare the prediction power of models that account for product returns with those that ignore product returns. The preliminary numerical results based on a real dataset suggest that accounting for product returns increases the demand estimation accuracy.

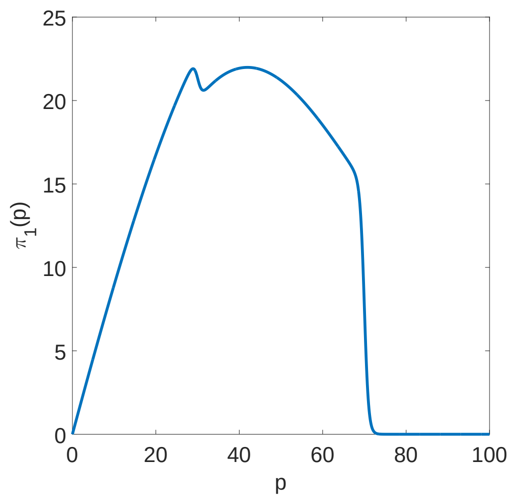

FIGURE 4.2: An example of non-unimodality of the revenue function in the case with two returnable products.

Product returns indeed have an impact on demand estimation as, for example, ignoring the returns could result in overestimation of customer demand for products. In order to study such an impact, we assume that the underlying demand model is unknown to the retailer and must be estimated from historical data. We compare the demand estimation accuracy of the N-MNL model in Section 4.3.2, which directly models product returns, with that of two MNL benchmarks that do not explicitly model product returns. We refer to such benchmarks as *MNL-Discard* and *MNL-No-Purchase* and denote them by MNL-Disc and MNL-NP for simplicity. Both benchmarks follow an MNL model as described in Section 4.3.1, but they differ in how they use product returns transactions for demand estimation. The MNL-Disc model ignores the returns transactions and considers a purchase transaction as an actual purchase regardless of whether that product was later returned or not. In other words, the MNL-Disc model discards the returns transactions for demand estimation and thus, is expected to overestimate the demand. The MNL-NP model, on the other hand, considers the returns transactions as no-purchases, that is, this model considers the purchase transaction of a product that was later returned as a no-purchase transaction. The N-MNL model, however, directly uses the returns transactions to estimate the return probabilities (in the second stage of the model) in addition to the purchase probabilities (in the first stage).

We experiment with a dataset from a large Chilean retailer (this dataset was used in the second essay in this dissertation). The description of this dataset is provided in Section 3.5.1 of this dissertation. In a nutshell, the dataset provides the assortment history (shown to customers on the retailer's website) as well as the click/no-click records for a set of 19 pairs of shoes. This dataset, however, does not include any returns transactions and thus, cannot be directly used for our purposes. We augment the dataset with returns transactions by randomly selecting a subset of purchase (i.e., click) transactions to be labeled as returned. Moreover, because the number of no-purchases (i.e., no-clicks) is significantly more than the number of purchases in the dataset, we randomly discard and replicate some of no-purchase and purchase transactions, respectively.

In order to compare the prediction power of different models, we partition the dataset into train and test transactions. We randomly select 80% of transactions for training different models and estimate the demand parameters using the maximum likelihood estimation. We then use the estimated demand models to predict the outcome of the test (out-of-sample) transactions. We compare the demand estimation accuracy of different models using the hard root mean squared error (RMSE) metric which we define in what follows.

Because the two MNL benchmarks do not model product returns explicitly, they cannot predict whether a purchased product will be returned (while the N-MNL model can). Therefore, when testing different models, we study two scenarios in which we consider (or not) product returns as possible outcomes and calculate two hard RMSE metrics. More specifically, given an assortment S that is shown to a customer, we consider two scenarios with: (i) $2\,|S|+1$ possible outcomes which comprise the no-purchase outcome, the purchase-then-keep outcome for each product, and the purchase-then-return outcome for each product; (ii) $|S|+1$ possible outcomes which comprise the no-purchase outcome, and the purchase outcome for each product. In the latter scenario, we consider the returns transactions (i.e., purchase-then-return transactions) as no-purchases when testing different models. We calculate the hard RMSE in each scenario and denote them by $RMSE^{hard}$ and $RMSE_r^{hard}$, respectively. The first scenario favors the N-MNL model as the MNL benchmarks do not account for product returns outcomes (i.e., $|S|$ out of $2\,|S|+1$ possible outcomes) and assign zero probabilities to such outcomes. The second scenario, however, favors the MNL benchmarks. Given an assortment S, let S^R denote the set of possible purchase-then-keep and purchase-then-return outcomes, that is $S^R := \cup_{j\in S}\{(j, keep), (j, return)\}$. The $RMSE^{hard}$ metric can be calculated as follows:

$$RMSE^{hard} := \sqrt{\frac{\sum_{t=1}^{T}\sum_{j\in S_t^R\cup\{0\}}\left(\mathbf{1}\{j=o_t\}-\mathbb{P}_M\left(j|S_t\right)\right)^2}{\sum_{t=1}^{T}\left(2\,|S_t|+1\right)}},$$

where T denotes the total number of test (out-of-sample) transactions, S_t denotes the displayed assortment in transaction t, $\mathbf{1}\{\cdot\}$ is the indicator function, o_t denotes the actual outcome of transaction t, and $\mathbb{P}_M\left(j|S_t\right)$ is the probability of outcome j given the assortment S_t under a model M (e.g., the N-MNL model). Similarly, $RMSE_r^{hard}$ can be calculated as:

$$RMSE_r^{hard} := \sqrt{\frac{\sum_{t=1}^{T}\sum_{j\in S_t\cup\{0\}}\left(\mathbf{1}\{j=o_t\}-\mathbb{P}_M\left(j|S_t\right)\right)^2}{\sum_{t=1}^{T}\left(|S_t|+1\right)}}.$$

Table 4.1 shows the hard RMSE of different models for the out-of-sample transactions given a dataset with 32554 total number of transactions (including both the train and test transactions) and 30% returns rate. We also show in parenthesis the percentage improvements of the N-MNL model over different benchmarks. To obtain a clearer picture of the relative performance of different models, we also consider two other benchmarks: an MNL model with identical exponentiated mean utilities of products set to $1/n$ (where n is the total number of products). We call this model the *MNL-Uni*; We also compare to a uniform random model that assigns uniform probabilities to all possible outcomes. For example, in a scenario with $|S|+1$ possible outcomes, this model assigns the probability $1/(|S|+1)$ to each possible outcome. We refer to this model as *RAND*.

Table 4.1: Hard RMSE of different models on a dataset with 30% returns rate.

	N-MNL	MNL-Disc	MNL-NP	MNL-Uni	RAND
$RMSE^{hard}$	0.3062	0.3136 (2.37%)	0.3245 (5.64%)	0.4189 (26.91%)	0.3143 (2.57%)
$RMSE_r^{hard}$	0.3917	0.4171 (6.09%)	0.3917 (0%)	0.4725 (17.10%)	0.4 (2.08%)

Several observations based on Table 4.1 are in order. We first note that the N-MNL model outperforms the benchmarks using both metrics. However, the percentage improvements of the N-MNL model over the MNL benchmarks are not very significant. This is partially explained by the fact that the ratio μ_2/μ_1 in the N-MNL model is estimated as one using this dataset. To this end, we note that when $\mu_2 = \mu_1$, the N-MNL model reduces to an MNL. Thus, the two models become essentially the same except for the number of possible outcomes that they account for (i.e., $2|S|+1$ versus $|S|+1$). The percentage improvements of the N-MNL over MNL benchmarks are generally larger using the $RMSE^{hard}$ metric. The intuition behind this finding is that the N-MNL model accounts for and predicts some possible outcomes (i.e., the returns outcomes) which are not accounted for by the MNL models. However, using the $RMSE_r^{hard}$ metric, the MNL-NP performs as well as the N-MNL as in such case, product returns are considered as no-purchase transactions. However, using the $RMSE^{hard}$ metric, the MNL-NP model is outperformed by the N-MNL (and MNL-Disc) as the MNL-NP model underestimates the demand in such case. We finally note that the RAND model performs relatively well using both metrics. This further pronounces the better performance of the N-MNL model relative to the benchmarks. (In our experiments, the performance of the RAND model is independent of the number of transactions and only depends on the number of possible outcomes.).

We also experimented with datasets with the same number of transactions but with smaller rates of returns (e.g., 10%). We observed consistent performance, but the percentage improvements of the N-MNL model over the MNL benchmarks were smaller. This observation shows that the performance of the N-MNL model relative to the MNL benchmarks improves as the number of returns transactions increases, because the main advantage of the N-MNL model over the MNL benchmarks stems from accounting for product returns.

Finally, we note that the main insight from the numerical experiments is that accounting for product returns indeed has an impact on demand estimation. In particular, models that account for product returns (e.g., the studied N-MNL model) increase the demand estimation accuracy over those that do not explicitly consider product returns. This point is illustrated by the comparison between the prediction power of the N-MNL and MNL models as discussed above.

4.7 Conclusion and Future Research

In this paper, we study the interplay between a retailer's return and pricing policies and customers' purchasing decisions. We model the customers' purchasing and returns decisions using a discrete choice model and study the seller's revenue maximization problem in the cases with and without product returns. For the single-product case, we characterize the seller's optimal prices and numerically show that the return option results in different (but not necessarily higher) optimal prices compared to the case with no returns. Moreover, we find that offering the return option is not always favorable for the retailer. In addition, we characterize conditions under which offering the return option to customers increases the retailer's revenue.

We also numerically study the impact of accounting for product returns on demand estimation. The preliminary numerical results based on a real dataset suggest that our model, which accounts for product returns, increases demand estimation accuracy compared to models that do not consider product returns in their estimation.

There are several directions for future research that we still aim at exploring. In terms of the analytical results, we plan to study conditions under which the optimal price in the case with returns is larger (or smaller) than that in the no-returns case. Moreover, further examining the conditions under which offering the return option increases the retailer's revenue is another direction to pursue. In addition, we aim at characterizing the optimal prices in the general case with multiple returnable products. Finally, we plan to expand the numerical experiments to further explore the potential benefits of our model in improving the demand estimation accuracy in a setting with product returns.

5

Conclusion

Data-driven approaches to decision-making under uncertainty is at the center of many operational problems. These are problems in which there is an element of uncertainty that needs to be *estimated* (*learned*) from data in order to make *online* (*dynamic*) operational decisions. An example of such problems is estimating customer demand from transaction data to make dynamic assortment decisions.

This dissertation adopts a data-driven active learning approach to study various operational problems under uncertainty with a focus on retail operations. The first two essays in this dissertation study the classic *exploration* (i.e., parameter estimation) versus *exploitation* (i.e., optimization) trade-off from different perspectives. The first essay takes a theoretical approach and studies such trade-off in a combinatorial optimization setting. Our paper is the first to prove a lower bound on the asymptotic performance of any admissible policy. We also propose near-optimal policies that are implementable in real-time. The second essay studies the dynamic assortment personalization problem of an online retailer facing heterogeneous customers with unknown product preferences. We propose a prescriptive approach, called the *dynamic clustering* policy, for dynamic estimation of customer preferences and optimization of personalized assortments. Further focusing on retail operations, the final essay studies the interplay between a retailer's return and pricing policies and customers' purchasing decisions. We characterize the retailer's optimal prices in the cases with and without product returns and determine conditions under which offering the return option to customers increases the retailer's revenue. We also explore the impact of accounting for product returns on demand estimation.

Appendix A

Appendices for Chapter 2

A.1 Omitted Proofs and Materials from Section 2.4

A.1.1 A Limit on Achievable Performance

In this section we prove Proposition 3 and Theorem 5. We begin with some preliminaries.

Definition 24 (Regularity)**.** *The distribution F is regular if $\mu \in \mathcal{B}$ and the density of $B(a)$: (i) can be parametrized by its mean $\mu(a)$, and thus we denote it by $f_a(\cdot;\mu(a))$; (ii) $0 < I_a(\mu(a), \lambda(a)) < \infty$ for all $l(a) < \lambda(a) < \mu(a) < u(a)$, $a \in A$, where $I_a(\mu(a), \lambda(a))$ denotes the Kullback-Leibler divergence (see e.g., Cover and Thomas (2006)) between $f_a(\cdot;\mu(a))$ and $f_a(\cdot;\lambda(a))$; and (iii) $I_a(\mu(a), \lambda(a))$ is continuous in $\lambda(a) < \mu(a)$ for all $\mu(a) \in (l(a), u(a))$.*

Definition 25 (Consistency)**.** *A policy π is said to be consistent if $R^{\pi}(F, N) = o(N^{\alpha})$ for all $\alpha > 0$, for every regular F.*

Define $\Theta_a := (l(a), u(a))$. For $\lambda(a) \in \Theta_a$, we have that

$$I_a(\mu(a), \lambda(a)) = \int_{-\infty}^{\infty} [\ln (f_a(x_a;\mu(a))/f_a(x_a;\lambda(a)))] f_a(x_a;\mu(a))\, dx_a.$$

Define $\lambda := (\lambda(a) : a \in A)$ and let $\mathbb{E}_\lambda$ and P_λ denote the expectation and probability induced when each f_a receives the parameter $\lambda(a) \in \Theta_a$ for all $a \in A$.

Finally, define $\tilde{T}_{N+1}(S)$ as the number of times that the decision-maker has implemented solution $S \in \mathcal{S}$ prior to period $N+1$, that is, $\tilde{T}_{N+1}(S) := |m < N+1 : S_m = S|$. We can then rewrite the regret function as

$$R^\pi(F, N) = \sum_{S \in \mathcal{S}} \Delta_S^\mu \, \mathbb{E}_F \left\{ \tilde{T}_{N+1}(S) \right\}.$$

Next, We prove Proposition 3.

Proof of Proposition 3. Consider $D \in \mathcal{D}$ as defined in Section 2.4, and take $\lambda \in \mathcal{B} = \prod_{a \in A} (l(a), u(a))$ so that $\lambda(a) = \mu(a)$ for $a \notin D$, and that $D \subseteq S^*$ for all $S^* \in \mathcal{S}^*(\lambda)$. By the consistency of π, one has that

$$\mathbb{E}_\lambda \left\{ N - \sum_{S^* \in \mathcal{S}^*(\lambda)} \tilde{T}_{N+1}(S) \right\} = o(N^\alpha),$$

for any $\alpha > 0$. By construction, each optimal solution under λ includes each $a \in D$. Thus, one has that $\sum_{S^* \in \mathcal{S}^*(\lambda)} \tilde{T}_{N+1}(S) \leqslant \max\{T_{N+1}(a) : a \in D\}$, and therefore

$$\mathbb{E}_\lambda \{N - \max\{T_{N+1}(a) : a \in D\}\} \leqslant \mathbb{E}_\lambda \left\{ N - \sum_{S^* \in \mathcal{S}^*(\lambda)} \tilde{T}_{N+1}(S) \right\} = o(N^\alpha). \quad \text{(A.1)}$$

We focus on $0 < \alpha < 1$ and take ϵ such that $0 < \alpha < \epsilon < 1$. Define $I(D, \lambda) := |D| \max\{I_a(\mu(a), \lambda(a)) : a \in D\}$, $D \in \mathcal{D}$. We then have that

$$\begin{aligned} \mathbb{P}_\lambda \left\{ \max\{T_{N+1}(a) : a \in D\} < \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\} &= \mathbb{P}_\lambda \left\{ N - \max\{T_{N+1}(a) : a \in D\} > N - \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\} \\ &\overset{(a)}{\leqslant} \frac{\mathbb{E}_\lambda \{N - \max\{T_{N+1}(a) : a \in D\}\}}{N - \frac{(1-\epsilon)\ln N}{I(D,\lambda)}}, \end{aligned}$$

where (a) follows from Markov's inequality. Note that for N large enough, we have that $N - ((1-\epsilon)\ln N / I(D,\lambda)) > 0$, and because $(1-\epsilon)\ln N/I(D,\lambda) = O(\ln N)$, from (A.1) we have that

$$(N - O(\ln N))\ \mathbb{P}_\lambda \left\{ \max\left\{T_{N+1}(a) : a \in D\right\} < \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\} = o(N^\alpha),$$

where in above, $(N - O(\ln N))$ refers to $N - ((1-\epsilon)\ln N/I(D,\lambda))$. The above can be re-written as

$$\mathbb{P}_\lambda \left\{ \max\left\{T_{N+1}(a) : a \in D\right\} < \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\} = o(N^{\alpha-1}). \tag{A.2}$$

For $a \in D$ and $n \in \mathbb{N}$ define

$$L_n(a) := \sum_{k=1}^{n} \ln \left(f_a(\hat{b}_a^k; \mu(a)) / f_a(\hat{b}_a^k; \lambda(a)) \right),$$

where $\hat{b}_a^k$ denotes the k-th cost realization for $a \in D$ when policy π is implemented. Also, define the event

$$\Xi(N) := \left\{ L_{T_{N+1}(a)}(a) \leqslant \frac{(1-\alpha)\ln N}{|D|} \text{ for all } a \in D\,,\ \max\left\{T_{N+1}(a) : a \in D\right\} < \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\},$$

and note that

$$\mathbb{P}_\lambda \left\{\Xi(N)\right\} \leqslant \mathbb{P}_\lambda \left\{ \max\left\{T_{N+1}(a) : a \in D\right\} < \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\}.$$

Next, we relate the probability of the event $\Xi(N)$ under the two parameter configurations.

$$\begin{aligned}
\mathbb{P}_\lambda\{\Xi(N)\} &= \int_{\omega\in\Xi(N)} d\mathbb{P}_\lambda(\omega) \\
&\overset{(a)}{=} \int_{\omega\in\Xi(N)} \prod_{a\in D} \exp(-L_{T_{N+1}(a)}(a))\ d\mathbb{P}_\mu(\omega) \\
&\overset{(b)}{\geqslant} \int_{\omega\in\Xi(N)} \exp(-(1-\alpha)\ln N)\ d\mathbb{P}_\mu(\omega) \\
&= N^{\alpha-1}\mathbb{P}_\mu\{\Xi(N)\},
\end{aligned}$$

where (a) follows from noting that probabilities under λ and μ differ only in that cost realizations in D have different probabilities under λ and μ, and (b) follows from noting that $L_{T_{N+1}(a)}(a) \leqslant (1-\alpha)\ln N/|D|$ for all $\omega \in \Xi(N)$.

From above and (A.2) we have that

$$\lim_{N\to\infty} \mathbb{P}_\mu\{\Xi(N)\} \leqslant \lim_{N\to\infty} N^{1-\alpha}\ \mathbb{P}_\lambda\{\Xi(N)\} = 0. \tag{A.3}$$

Now, fix $a \in D$. By the Strong Law of Large Numbers (see page 8 of Lai and Robbins (1985b)) we have that

$$\lim_{n\to\infty} \max_{m\leqslant n} L_m(a)/n = I_a(\mu(a),\lambda(a)), \quad \text{a.s.}[\mathbb{P}_\mu], \quad \forall a \in D.$$

From above, we have that

$$\lim_{N\to\infty} \max\left\{ \frac{L_m(a)}{\frac{(1-\epsilon)\ln N}{|D|I_a(\mu(a),\lambda(a))}} : m < \frac{(1-\epsilon)\ln N}{|D|\,I_a(\mu(a),\lambda(a))} \right\} = I_a(\mu(a),\lambda(a)), \quad \text{a.s.}[\mathbb{P}_\mu], \quad \forall a \in D.$$

From above and seeing that $1-\alpha > 1-\epsilon$, we have for all $a \in D$ that

$$\lim_{N\to\infty} \mathbb{P}_\mu\left\{ L_m(a) > \tfrac{(1-\alpha)\ln N}{|D|} \text{ for some } m < \tfrac{(1-\epsilon)\ln N}{|D|I_a(\mu(a),\lambda(a))} \right\} =$$

$$\lim_{N\to\infty} \mathbb{P}_\mu\left\{ \max\left\{ \tfrac{L_m(a)}{\frac{(1-\epsilon)\ln N}{|D|I_a(\mu(a),\lambda(a))}} : m < \tfrac{(1-\epsilon)\ln N}{|D|I_a(\mu(a),\lambda(a))} \right\} > \left(\tfrac{1-\alpha}{1-\epsilon}\right) I_a(\mu(a),\lambda(a)) \right\} = 0.$$

Because $I(D,\lambda) \geqslant |D|\, I_a(\mu(a),\lambda(a))$, we further have that

$$\lim_{N\to\infty} \mathbb{P}_\mu \left\{ L_m(a) > \frac{(1-\alpha)\ln N}{|D|} \text{ for some } m < \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\} = 0 \quad \forall a \in D.$$

Then, in particular by taking $m = T_{N+1}(a)$ we have that

$$\lim_{N\to\infty} \mathbb{P}_\mu \left\{ L_{T_{N+1}(a)}(a) > \frac{(1-\alpha)\ln N}{|D|}, \quad T_{N+1}(a) < \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\} = 0 \quad \forall a \in D,$$

which in turn implies

$$\lim_{N\to\infty} \mathbb{P}_\mu \left\{ L_{T_{N+1}(a)}(a) > \frac{(1-\alpha)\ln N}{|D|}, \quad \max\left\{T_{N+1}(a) : a \in D\right\} < \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\} = 0 \quad \forall a \in D.$$

Finally, by taking the union of events over $a \in D$ we have that

$$\lim_{N\to\infty} \mathbb{P}_\mu \left\{ L_{T_{N+1}(a)}(a) > \frac{(1-\alpha)\ln N}{|D|} \text{ for some } a \in D\,,\ \max\left\{T_{N+1}(a) : a \in D\right\} < \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\} = 0. \tag{A.4}$$

Thus, by (A.3), (A.4), and the definition of $\Xi(N)$ we have that

$$\lim_{N\to\infty} \mathbb{P}_\mu \left\{ \max\left\{T_{N+1}(a) : a \in D\right\} < \frac{(1-\epsilon)\ln N}{I(D,\lambda)} \right\} = 0.$$

The result follows from letting ϵ and α approach zero, and taking $K_D := I(D,\lambda)^{-1}$.

□

Theorem 5. *The regret of any consistent policy π is such that*

$$\liminf_{N\to\infty} \frac{R^\pi(F,N)}{\ln N} \geqslant z^*_{LBP}. \tag{2.4}$$

Proof of Theorem 5. For any consistent policy π, define

$$\zeta^\pi(F,N) := \sum_{S\in\mathcal{S}} \Delta^\mu_S \tilde{T}_{N+1}(S)$$

to be the total additional cost (relative to an oracle) associated with that policy. Note that $\mathbb{E}_F\{\zeta^\pi(F,N)\} = R^\pi(F,N)$. The next lemma ties the asymptotic bounds in (2.2) to the solution to LBP and establishes an asymptotic bound on the regret of any consistent policy.

Lemma 26. *For any consistent policy π and regular F we have that*

$$\lim_{N\to\infty} \mathbb{P}_F\Big(\zeta^\pi(F,N) \geqslant z^*_{LBP}\ln N\Big) = 1.$$

Proof of Lemma 26. Define the event $\Upsilon_N := \bigcap_{D\in\mathcal{D}}\{\max\{T_{N+1}(a) : a\in D\} \geqslant K_D\ln N\}$ and let Υ_N^c denote the complement of the event Υ_N. Note that $\zeta^\pi(F,N) \geqslant z^*_{LBP}\ln N$ when Υ_N occurs, because $(x(a) = T_{N+1}(a)/\ln N, a\in A)$ and $(y(S) = \tilde{T}_{N+1}(S)/\ln N, S\in\mathcal{S})$ are feasible to LBP. Thus, one has that

$$\begin{aligned}\mathbb{P}_F\left\{\frac{\zeta^\pi(F,N)}{\ln N} < z^*_{LBP}\right\} &= \mathbb{P}_F\left\{\frac{\zeta^\pi(F,N)}{\ln N} < z^*_{LBP}\,,\,\Upsilon_N\right\} + \mathbb{P}_F\left\{\frac{\zeta^\pi(F,N)}{\ln N} < z^*_{LBP}\,,\,\Upsilon_N^c\right\}\\ &\leqslant \mathbb{P}_F\{\Upsilon_N^c\}. \end{aligned} \tag{A.5}$$

From Proposition 3 and the union bound, we have that

$$\lim_{N\to\infty}\mathbb{P}_F\{\Upsilon_N^c\} \leqslant \sum_{D\in\mathcal{D}}\lim_{N\to\infty}\mathbb{P}_F\{\max\{T_{N+1}(a) : a\in D\} < K_D\ln N\} = 0,$$

because $|\mathcal{D}| < \infty$. Thus, taking the limit in (A.5) we have that

$$\lim_{N\to\infty}\mathbb{P}_F\{\zeta^\pi(F,N) < z^*_{LBP}\ln N\} = 0.$$

□

Note that Lemma 26 establishes convergence in probability (hence it can be used to bound $\zeta^\pi(F,N)$, rather than just its expectation, which is the regret). Theorem 5 then follows directly from Lemma 26 and Markov's inequality.

□

A.1.2 Family of Instances with Finite Regret

Proposition 27. *If the combinatorial problem in* (2.1) *corresponds to a shortest path, minimum-cost spanning tree, minimum-cost perfect matching, generalized Steiner tree or knapsack problem, then there exists a family of instances where* $z^*_{LBP} = 0$ *while the minimum-size cover of* A *is arbitrarily large.*

Proof of Proposition 27. The family for the shortest path problem is that based on Example 2 (which is parametrized by an integer k), and described after Theorem 5 in Section 2.4.

For minimum-cost spanning tree, consider a complete graph $G = (V, A)$ with $|V| = k$ nodes, $\mu(a) = \epsilon$ and $l(a) = 0$ for all $a \in \{(i, i+1) : i < k\}$, and $l(a) = M > 0$ for all $a \notin \{(i, i+1) : i < k\}$ with $k\,\epsilon < M$. One can check that any cover of A is of size at least $(k-2)/2$. In contrast, $\mathcal{D} = \varnothing$, independent of k, thus $z^*_{LBP} = 0$. Note that the Steiner tree problem generalizes the minimum-cost spanning tree problem, thus this instance covers the Steiner tree case as well.

For minimum-cost perfect matching, consider a complete graph $G = (V, A)$ with $|V| = 2k$ nodes, $\mu(a) = \epsilon$ and $l(a) = 0$ for all $a \in \{(2i+1, 2i+2) : i < k\}$, and $l(a) = M > 0$ for all $a \notin \{(2i+1, 2i+2) : i < k\}$ with $k\,\epsilon < M$. One can check that any cover of A is of size at least $2(k-1)$. In contrast, $\mathcal{D} = \varnothing$, independent of k, thus $z^*_{LBP} = 0$.

Finally, for the knapsack problem, consider the items $A := \{0, 1, \ldots, Ck\}$, where $C \in \mathbb{N}$ denotes the knapsack capacity, and weights $w \in \mathbb{R}^{Ck+1}$ so that $w(0) = C$, and $w(i) = 1$ for $i > 0$. In addition, set $u(0) = 0$ and $\mu(0) = \epsilon$ and $u(i) = -M < 0$ for $i > 0$ (where $u(a)$ denotes the upper bound on the range of the "utility" distribution of ground element a), with $\epsilon < M$. Note that in this case the problem is of utility maximization. One can check that any cover of A is of size at least $k+1$. In contrast, $\mathcal{D} = \varnothing$, independent of k, thus $z^*_{LBP} = 0$.

$\square$

A.1.3 LBP-based policy

In this section we make the dependence of the LBP problem (and the set $\mathcal{D}$ and K_D) on the cost vector explicit. That is, we rewrite the $LBP(\nu)$ problem in (2.3) for a given cost vector ν as follows:

$$\begin{aligned} z^*_{LBP}(\nu) := \min \quad & \sum_{S \in \mathcal{S}} \Delta^{\nu}_S \, y(S) \\ s.t. \quad & \max\{x(a) : a \in D\} \geqslant K_D(\nu), \quad D \in \mathcal{D}(\nu) \\ & x(a) \leqslant \sum_{S \in \mathcal{S} : a \in S} y(S), \quad a \in A \\ & x(a),\, y(S) \in \mathbb{R}_+, \quad a \in A, S \in \mathcal{S}. \end{aligned}$$

We define $\Gamma_{LBP}(\nu)$ as the set of optimal solutions to $LBP(\nu)$.

To match the lower bound of Theorem 5, given the construction of $LBP\,(\mu)$, it is natural to try allocating exploration efforts only to the solutions prescribed by a solution to $LBP\,(\mu)$ (i.e., those $S \in \mathcal{S}$ with $y(S) > 0$). Unfortunately, said solution is not readily available in practice, as it depends on the mean cost vector which is only partially estimated at any given time. Nonetheless, one might still attempt to focus exploration on the solution to $LBP\,(\widehat{\mu}_n)$ (where $\widehat{\mu}_n = (\widehat{\mu}_n(a) : a \in A)$ and $\widehat{\mu}_n(a)$ is the sample mean of cost realizations of ground element a up to (and excluding) period n) hoping that said solution converges to that of $LBP\,(\mu)$. While this is indeed the case when $\widehat{\mu}_n \to \mu$, collecting information only on solutions prescribed by the solution to $LBP\,(\widehat{\mu}_n)$ does not suffice (in general) to accurately estimate the full mean cost vector, as the following example illustrates.

Example 28. *Let $G = (V, A)$ be the digraph depicted in Figure A.1 and let $\mathcal{S}$ be composed of all paths from node s to node t. Set $l(a) = 0$ and $u(a) = \infty$ for every arc $a \in A$, and F be such that $\mu\,(e) = c$, $\mu\,(d) = \mu\,(p_1) = \mu\,(q_1) = \epsilon/6$,*

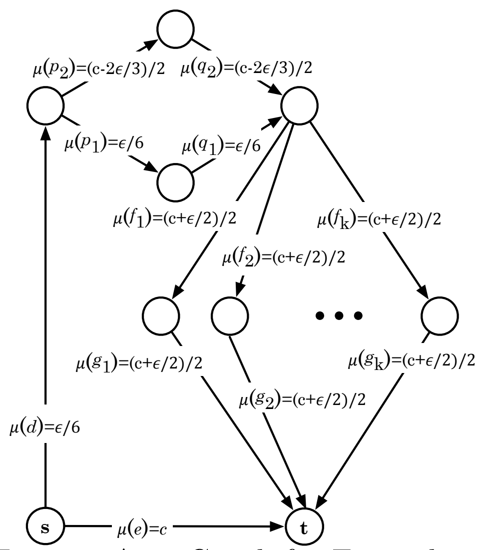

FIGURE A.1: Graph for Example 28.

$\mu(p_2) = \mu(q_2) = \frac{c-2\epsilon/3}{2}$, *and* $\mu(f_i) = \mu(g_i) = \frac{c+\epsilon/2}{2}$ *for all* $i \in \{1, \ldots, k\}$ *where* $0 < \epsilon \ll c$. *The shortest (expected) path in this digraph is* $\{e\}$.

For every $i \in \{1, \ldots, k\}$, define $S_i := \{d, p_1, q_1, f_i, g_i\}$ and $\tilde{S}_i := \{d, p_2, q_2, f_i, g_i\}$. In Example 28 we have that $\mathcal{D} = \{\{f_1\}, \{f_2\}, \ldots, \{f_k\}, \{g_1\}, \{g_2\}, \ldots, \{g_k\}\}$. This, in turn, implies that the minimum-regret solution-cover (i.e., exploration set) induced by the optimal solution to $LBP(\mu)$ is $\{S_i\}_{i=1}^k$ with a regret of $k\epsilon$.

Suppose that we implement a policy that initially draws samples of the costs of p_1 and q_1 that are extremely high, so that the solution to $LBP(\widehat{\mu}_n)$ consists of solutions $\{\tilde{S}_i\}_{i=1}^k$. There on, focusing exploration on the solution to $LBP(\widehat{\mu}_n)$ might imply that no further samples of p_1 and q_1 are needed, thus $\widehat{\mu}_n \to \nu' = (\nu'(a) : a \in A)$, with $\nu'(a) = \mu(a)$ for all a in A except $a \in \{p_1, q_1\}$. One can see that in such a case, the exploration set (solution-cover) that $LBP(\widehat{\mu}_n)$ could converge to is $\{\tilde{S}_i\}_{i=1}^k$ with a regret of ck which is not an optimal solution to $LBP(\mu)$.

Example 28 shows that convergence of $LBP(\widehat{\mu}_n)$ to $LBP(\mu)$ (and even $z^*_{LBP}(\widehat{\mu}_n)$ to $z^*_{LBP}(\mu)$) is not guaranteed if exploration is restricted to the solution to $LBP(\widehat{\mu}_n)$. Thus, to assure convergence of $z^*_{LBP}(\widehat{\mu}_n)$ to $z^*_{LBP}(\mu)$ (so as to attain the asymptotic performance in the lower bound result in Theorem 5), one must collect samples on a subset of A that might contain more elements than those explored by the solution to $LBP(\mu)$, and do so at a small but positive frequency.

While one might be able to formulate the problem of finding a subset of the ground set whose exploration incurs the least regret while guaranteeing the convergence of $LBP(\widehat{\mu}_n)$ to $LBP(\mu)$, we instead choose to expand the exploration efforts to the whole ground set. By maintaining exploration frequencies on these additional elements small, the overall regret should still be driven by the cost of exploring the solution to $LBP(\widehat{\mu}_n)$.

Following the discussion above, next we propose a policy that focuses exploration on the solution to $LBP(\widehat{\mu}_n)$, but also at a lesser (tunable) degree on a solution-cover of the ground set. Such an approach ensures the convergence of the solution to $LBP(\mu)$ by guaranteeing that $\widehat{\mu}_n \to \mu$ (see below for a more detailed discussion). To simplify the reconstruction of the aforementioned proxy (and the exposition), we make the following technical assumption, needed for proving a performance guarantee.

Assumption 1. *F is regular and the density function $f_a(\cdot;\cdot)$ is known by the decision-maker for all $a \in A$, and there exists a known finite constant K such that $K_D(\mu) \leqslant K$ for all $D \in \mathcal{D}(\mu)$. In addition, there is no set $D \subseteq A$ such that $z^*_{Comb}(\mu) = z^*_{Comb}((\mu \wedge l)(D))$ with $\mathcal{S}^*(\mu) \neq \mathcal{S}^*((\mu \wedge l)(D))$.*

Knowing the parametric form of the cost density function for each $a \in A$ reduces the burden of estimating $K_D(\mu)$ as this can be performed by simply estimating μ (as is also the case for Δ^{μ}_{S} and the set $\mathcal{D}(\mu)$). The last part of Assumption 1 is necessary to correctly reconstruct the set of constraints (2.3b), and holds with probability one when, for example, mean costs are random draws from an absolutely continuous distribution: this suits most practical settings where mean costs are unknown and no particular structure for them is anticipated (note that any additional prior structural information on the mean cost vector might be taken advantage of).

Under Assumption 1, convergence of $z^*_{LBP}(\widehat{\mu}_n)$ to $z^*_{LBP}(\mu)$ is assured if $\widehat{\mu}_n$ converges to μ. As discussed in Example 1, this can be achieved by exploring a cover of A. We formalize the problem of finding a minimum-regret cover of A in the following definition.

Definition 29. *For a cost vector $\nu \in \mathcal{B}$, define the cover problem $Cover(\nu)$ as*

$$z^*_{Cover}(\nu) := \min \quad \sum_{S \in \mathcal{S}} \Delta^{\nu}_S \; y(S) \tag{A.7a}$$

$$s.t. \qquad 1 \leqslant \sum_{S \in \mathcal{S}: a \in S} y(S), \quad a \in A \tag{A.7b}$$

$$y(S) \in \{0,1\}, S \in \mathcal{S}, \tag{A.7c}$$

where $z^*_{Cover}(\nu)$ denotes the optimal objective value of the $Cover(\nu)$ problem. Also, define $\Gamma_{Cover}(\nu)$ as the set of optimal solutions to $Cover(\nu)$.

The proposed policy, which we refer to as the *LBP-based* policy and denote by π^*, is described by Algorithm 2. The LBP-based policy formulates and solves $LBP(\widehat{\mu}_n)$ and $Cover(\widehat{\mu}_n)$, and focuses exploration efforts (at different degrees) on the solutions to said problems. To enforce the logarithmic exploration frequency found in Theorem 5, the policy uses the doubling trick. This approach also allows us to minimize the number of times that the underlying combinatorial problem and auxiliary exploration problems $LBP(\widehat{\mu}_n)$ and $Cover(\widehat{\mu}_n)$ need to be solved. Recall that the doubling trick divides the horizon into cycles of growing length so that cycle i starts at time n_i where $\{n_i\}_{i \in \mathbb{N}}$ is a strictly increasing sequence of positive integers such that $n_1 = 1$ and $n_{i+2} - n_{i+1} \geqslant n_{i+1} - n_i$ for all $i \in \mathbb{N}$. Within each cycle, we first solve $Comb(\widehat{\mu}_n)$, $LBP(\widehat{\mu}_n)$ and $Cover(\widehat{\mu}_n)$, and then ensure that the appropriate exploration frequencies are achieved (in expectation). The frequency of exploration can then be controlled by varying the increment in length of the cycles (e.g., to achieve exploration frequencies proportional to $\ln N/N$, we can use cycles of exponentially increasing lengths). In Algorithm 2, we choose $n_i := \max\{\lfloor e^{i^{1/(1+\varepsilon)}} \rfloor, n_{i-1}+1\}$ for all $i \geqslant 2$, given a tunig parameter ε. For $S \in \mathcal{S} \backslash \mathcal{S}^*(\widehat{\mu}_n)$, we define

$$p_S := \begin{cases} y(S)/(n_{i+1}-n_i) & \text{if} \quad \sum_{S' \in \mathcal{S}} y(S') \leqslant (n_{i+1}-n_i) \\ y(S)/\sum_{S' \in \mathcal{S}} y(S') & \text{otherwise} \end{cases}$$

where $y(S)$ (in the definition of p_S) is from the solution to the LBP problem (see Algorithm 2). We also define $p_{S*} := (1 - \sum_{S \in \mathcal{S} \backslash \mathcal{S}^*(\widehat{\mu}_n)} p_S)/|\mathcal{S}^*(\widehat{\mu}_n)|$ for $S^* \in \mathcal{S}^*(\widehat{\mu}_n)$. Note that p_S is a probability measure that enforces the right exploration frequency (as prescribed by the LBP problem) for solution $S \in \mathcal{S}$. Also, in Algorithm 2, γ is a tuning parameter that controls the cover-based exploration frequency.

Algorithm 2 LBP-based policy $\pi^*(\gamma, \varepsilon)$

```
Set i = 0, and draw (b_1(a) : a ∈ A) randomly from 𝓑
for n = 1 to N do
  if n = n_i then
    Set i = i + 1
    Set S* ∈ 𝓢* (μ̂_n)                                   [Update exploitation set]
    Set 𝓔 ∈ Γ_Cover (μ̂_n)                          [Update Cover-exploration set]
    Set (x, y) ∈ Γ_LBP (μ̂_n)                          [Update LBP-exploration set]
  end if
  if T_n(a) < γ i for some a ∈ A then
    Set S_n = S for any S ∈ 𝓔 such that a ∈ S            [Cover-based exploration]
  else
    Set S_n = S with probability p_S, S ∈ 𝓢 [LBP-based exploration/Exploitation]
  end if
end for
```

A.1.4 Performance Guarantee of the LBP-based policy

Suppose that Assumption 1 holds. The following result provides a performance guarantee for the LBP-based policy. There, we denote $z^*_{LBP}(\mu)$ and $z^*_{Cover}(\mu)$ by z^*_{LBP} and z^*_{Cover} for simplicity.

Theorem 6. *Consider $\gamma \in (0,1)$ and $\varepsilon > 0$ arbitrary. The LBP-based policy $\pi^*(\gamma, \varepsilon)$ is such that*

$$\lim_{N \to \infty} \frac{R^{\pi^*(\gamma,\varepsilon)}(F,N)}{(\ln N)^{1+\varepsilon}} \leqslant z^*_{LBP} + \gamma\, z^*_{Cover}, \tag{2.5}$$

where z^*_{Cover} denotes the optimal objective value of the Cover problem

Proof of Theorem 6. The regret of the policy π^* (we drop the dependence of π^* on γ and ε for simplicity) stems from two sources: exploration efforts and exploitation errors. That is,

$$R^{\pi^*}(F,N) = R_1(F,N) + R_2(F,N),$$

where $R_1(F,N)$ is the exploration-based regret, i.e., that incurred at period n during cycle i if $T_n(a) < \gamma\, i$ for some $a \in A$, or alternatively when sampling a solution, picking $S_n \neq S^*$ with $S^* \in \mathcal{S}^*(\widehat{\mu}_{n_i})$, and $R_2(F,N)$ is the exploitation-based regret, i.e., that incurred when $T_n(a) \geqslant \gamma\, i$ for all $a \in A$ and we sample $S_n = S^*$. We prove the result by bounding each term above separately. (We dropped the dependence of $R_1(F,N)$ and $R_2(F,N)$ on the policy π^* to simplify notation.)

In the remainder of this proof, $\mathbb{E}$ and $\mathbb{P}$ denote expectation and probability when costs are distributed according to F and policy π^* is implemented.

Step 1 (Exploitation-based regret). Exploitation-based regret during cycle i is due to implementing suboptimal solutions when minimum cover-based exploration requirements are met.

Let i' denote a finite upper bound on the first cycle in which one is sure to randomize a solution on at least one period, e.g., $i' := 1 + \inf\{i \in \mathbb{N}, i \geqslant 2 : n_i \geqslant i\,|A|\,, n_{i+1} - n_i > |A|\}$. (Note that i' does not depend on N).

Fix $i \geqslant i'$ and note that when cover-based exploration requirements are met for $n \in [n_i, n_{i+1} - 1]$, one may exploit, that is, one may implement $S_n = S^*$ for some $S^* \in \mathcal{S}^*(\widehat{\mu}_{n_i})$. We use the event $\{S_n \in \mathcal{S}^*(\widehat{\mu}_{n_i})\}$ to denote exploitation. We also define

$\Delta^{\mu}_{max} := \max_{S \in \mathcal{S}} \{\Delta^{\mu}_{S}\}$. We then have that

$$\begin{aligned} R_2(F,N) &\leqslant n_{i'}\Delta^{\mu}_{max} + \sum_{i=i'}^{\lceil (\ln N)^{1+\varepsilon}\rceil} \sum_{n=n_i}^{n_{i+1}-1} \mathbb{E}\left\{\mathbf{1}\left\{T_n(a) \geqslant \gamma(i-1)\,,\, \forall a \in A\,, S_n \in \mathcal{S}^*(\widehat{\mu}_{n_i})\right\} \Delta^{\mu}_{S_n}\right\} \\ &\leqslant n_{i'}\Delta^{\mu}_{max} + \sum_{i=i'}^{\infty} (n_{i+1}-n_i)\,\mathbb{P}\left\{\mathcal{S}^*(\widehat{\mu}_{n_i}) \nsubseteq \mathcal{S}^*(\mu), T_{n_i}(a) \geqslant \gamma(i-1), \forall a \in A\right\} \Delta^{\mu}_{max}. \end{aligned} \quad \text{(A.8)}$$

Next, we find an upper bound for the probability inside the sum in (A.8). For this, note that

$$\left\{\mathcal{S}^*(\widehat{\mu}_{n_i}) \nsubseteq \mathcal{S}^*(\mu)\right\} \subseteq \bigcup_{a \in A} \left\{ |\widehat{\mu}_{n_i}(a) - \mu(a)| \geqslant \frac{\Delta^{\mu}_{min}}{2s} \right\}, \quad \text{(A.9)}$$

where $s := \max\{|S| : S \in \mathcal{S}\}$ and $\Delta^{\mu}_{min} := \min\{\Delta^{\mu}_{S} : S \in \mathcal{S} \backslash \mathcal{S}^*(\mu)\}$ denote the maximum solution size and minimum optimality gap for the full-information problem, respectively. (We assume, without loss of generality, that Δ^{μ}_{max} and Δ^{μ}_{min} are both positive, since otherwise, the problem is trivial.) Indeed, note that

$$\left\{ |\widehat{\mu}_{n_i}(a) - \mu(a)| < \frac{\Delta^{\mu}_{min}}{2s}\,,\, \forall a \in A \right\} \subseteq \left\{ \sum_{a \in S^*} \widehat{\mu}_{n_i}(a) < \sum_{a \in S} \widehat{\mu}_{n_i}(a)\,,\, \forall S^* \in \mathcal{S}^*(\mu)\,,\, S \in \mathcal{S} \backslash \mathcal{S}^*(\mu) \right\}.$$

The next proposition, whose proof can be found in Appendix A.6, allows us to bound (A.8) using the observation above.

Proposition 30. *For any fixed $a \in A$, $n \in \mathbb{N}$, $k \in \mathbb{N}$, and $\epsilon > 0$ we have that*

$$\mathbb{P}\left\{|\widehat{\mu}_n(a) - \mu(a)| \geqslant \epsilon, T_n(a) \geqslant k\right\} \leqslant 2\exp\left\{-\frac{2\epsilon^2 k}{\mathcal{L}^2}\right\},$$

where $\mathcal{L} := \max\{u(a) - l(a) : a \in A\}$.

Using the above, the union bound, and (A.9), we have that

$$\begin{aligned} &\mathbb{P}\left\{\mathcal{S}^*(\widehat{\mu}_{n_i}) \nsubseteq \mathcal{S}^*(\mu)\,,\, T_{n_i}(a) \geqslant \gamma(i-1)\,,\, \forall a \in A\right\} \leqslant \\ &\sum_{a \in A} \mathbb{P}\left\{|\widehat{\mu}_{n_i}(a) - \mu(a)| \geqslant \frac{\Delta^{\mu}_{min}}{2s}, T_{n_i}(a) \geqslant \gamma(i-1)\right\} \leqslant 2\,|A| \exp\left\{-\frac{(\Delta^{\mu}_{min})^2 \gamma(i-1)}{2s^2\mathcal{L}^2}\right\}. \end{aligned} \quad \text{(A.10)}$$

Now, for $i \geqslant i'$, one has that $n_{i+1} \leqslant e^{(i+1)^{1/(1+\varepsilon)}}$ and $n_i \geqslant e^{(i-1)^{1/(1+\varepsilon)}}$. Hence, $n_{i+1} - n_i \leqslant e^{(i+1)^{1/(1+\varepsilon)}}$. Using this, (A.8) and (A.10) we conclude that

$$R_2(F,N) \leqslant \Delta^{\mu}_{max} \left(n_{i'} + \sum_{i=i'}^{\infty} 2\,|A| \exp\left\{ (i+1)^{1/(1+\varepsilon)} - \frac{\left(\Delta^{\mu}_{min}\right)^2 \gamma (i-1)}{2s^2 \mathcal{L}^2} \right\} \right).$$

Because $(i+1)^{1/(1+\varepsilon)} < i \frac{\left(\Delta^{\mu}_{min}\right)^2 \gamma}{2s^2 \mathcal{L}^2}$ for i large enough, we conclude that $R_2(F,N) \leqslant C_1$, for a positive finite constant C_1, independent of N.

Step 2 (Exploration-based regret). We separate the exploration-based regret into cover-based and LBP-based regrets. The former arises at period n when there exists $a \in A$ such that $T_n(a) < \gamma\, i$. The latter arises when the cover-based exploration requirements are met and one samples $S_n \neq S^*$ for $S^* \in \mathcal{S}^*(\widehat{\mu}_{n_i})$. Let $R_1^{Cover}(F,N)$ and $R_1^{LBP}(F,N)$ denote the cover-based and LBP-based exploration regrets, respectively, so that

$$R_1(F,N) := R_1^{Cover}(F,N) + R_1^{LBP}(F,N).$$

Step 2.1 (Cover-based exploration regret). We first bound the cover-based exploration regret. Let $\mathbf{C}$ denote the set of minimal covers of A, and Δ^{C}_{min} denote the minimum optimality gap for the $Cover\,(\mu)$ problem in (A.7), i.e.,

$$\Delta^{C}_{min} := \min \left\{ \left(\sum_{S \in \mathcal{E}} \Delta^{\mu}_{S} \right) - z^*_{Cover}(\mu) : \mathcal{E} \in \mathbf{C} \backslash \Gamma_{Cover}\,(\mu) \right\}.$$

We assume that $\Delta^{C}_{min} > 0$, since otherwise, the cover problem is trivial. Consider $i > i'$ and let $\mathcal{E}_i \in \Gamma_{Cover}\,(\widehat{\mu}_{n_i})$ denote the cover-based exploration set for any period $n \in [n_i, n_{i+1} - 1]$. Define $c := \max\{|\mathcal{E}| : \mathcal{E} \in \mathbf{C}\}$ as the maximum size of a minimal cover of A and let $I := \left\{ i \leqslant (\ln N)^{1+\varepsilon} : i > i'\,, \lceil \gamma(i-1) \rceil < \lceil \gamma\, i \rceil \right\}$ denote the set of cycles in which cover-based exploration requirements are increased. Noting that

$T_{n_i}(a) \geqslant \gamma(i-1)$ for all $a \in A$ when $i > i'$, we have that

$$\begin{aligned} R_1^{Cover}(F,N) &\leqslant c\, i' \Delta^{\mu}_{max} + \sum_{i \in I} \mathbb{E}\left\{ \mathbf{1}\left\{ T_{n_i}(a) \geqslant \gamma(i-1)\ \forall a \in A\,,\, \mathcal{E}_i \in \Gamma_{Cover}\left(\mu\right)\right\} \sum_{S \in \mathcal{E}_i} \Delta^{\mu}_S \right\} \\ &\quad + \sum_{i \in I} \mathbb{E}\left\{ \mathbf{1}\left\{ T_{n_i}(a) \geqslant \gamma(i-1)\ \forall a \in A\,,\, \mathcal{E}_i \notin \Gamma_{Cover}\left(\mu\right)\right\} \sum_{S \in \mathcal{E}_i} \Delta^{\mu}_S \right\} \\ &\leqslant c\, i' \Delta^{\mu}_{max} + \left(\gamma \left(\ln N\right)^{1+\varepsilon} + 1 \right) z^*_{Cover}(\mu) \\ &\quad + \Delta^{\mu}_{max}\, c \sum_{i \in I} \mathbb{P}\left\{ T_{n_i}(a) \geqslant \gamma(i-1)\ \forall a \in A\,,\, \mathcal{E}_i \notin \Gamma_{Cover}\left(\mu\right)\right\}. \end{aligned} \tag{A.11}$$

Next, we bound the probability inside the sum in (A.11). For that, observe

$$\left\{ \Gamma_{Cover}(\widehat{\mu}_{n_i}) \nsubseteq \Gamma_{Cover}(\mu) \right\} \subseteq \bigcup_{a \in A} \left\{ \left| \widehat{\mu}_{n_i}(a) - \mu(a) \right| \geqslant \frac{\Delta_1}{4cs} \right\}, \tag{A.12}$$

where $\Delta_1 := \min\left\{ \Delta^C_{min}, \Delta^{\mu}_{min} \right\}$. Indeed, note that

$$\begin{aligned} \left\{ \left| \widehat{\mu}_{n_i}(a) - \mu(a) \right| < \frac{\Delta_1}{4cs}\,,\, \forall a \in A \right\} &\subseteq \left\{ \left| \Delta^{\widehat{\mu}_{n_i}}_S - \Delta^{\mu}_S \right| < \frac{\Delta_1}{2c}\,,\, \forall S \in \mathcal{S} \right\} \\ &\subseteq \left\{ \left| \sum_{S \in \mathcal{E}} \left(\Delta^{\widehat{\mu}_{n_i}}_S - \Delta^{\mu}_S \right) \right| < \frac{\Delta_1}{2}\,,\, \forall \mathcal{E} \in \mathbf{C} \right\} \\ &\subseteq \left\{ \sum_{S \in \mathcal{E}} \Delta^{\widehat{\mu}_{n_i}}_S > \sum_{S \in \mathcal{E}^*} \Delta^{\widehat{\mu}_{n_i}}_S\,,\, \forall \mathcal{E}^* \in \Gamma_{Cover}\left(\mu\right)\,,\, \mathcal{E} \in \mathbf{C} \backslash \Gamma_{Cover}\left(\mu\right) \right\}, \end{aligned}$$

where we remember that for a cost vector $\nu \in \mathcal{B}$, $\Delta^{\nu}_S = \sum_{a \in S} \nu(a) - z^*_{Comb}(\nu)$. We note that as discussed in (A.9) in Step 1, by taking $\Delta_1 \leqslant \Delta^{\mu}_{min}$, we also ensure that $\{\mathcal{S}^*(\widehat{\mu}_{n_i}) \subseteq \mathcal{S}^*(\mu)\}$.

Using Proposition 30, the union bound, and (A.12), we have that

$$\begin{aligned} &\mathbb{P}\left\{ \mathcal{E}_i \notin \Gamma_{Cover}(\mu)\,,\, T_{n_i}(a) \geqslant \gamma(i-1)\,,\, \forall\, a \in A \right\} \leqslant \\ &\sum_{a \in A} \mathbb{P}\left\{ \left| \widehat{\mu}_{n_i}(a) - \mu(a) \right| \geqslant \frac{\Delta_1}{4sc}, T_{n_i}(a) \geqslant \gamma(i-1) \right\} \leqslant 2\left|A\right| \exp\left\{ -\frac{\left(\Delta_1\right)^2 \gamma(i-1)}{8 s^2 c^2 \mathcal{L}^2} \right\}. \end{aligned} \tag{A.13}$$

Using the above and (A.11) we obtain that

$$R_1^{Cover}(F,N) \leqslant \gamma\,(\ln N)^{1+\varepsilon}\, z^*_{Cover}(\mu) + C_2,$$

for a positive finite constant C_2, independent of N.

Step 2.2 (LBP-based exploration regret). Consider now the LBP-based exploration regret $R_1^{LBP}(F,N)$. Let $\Delta^{\mathcal{D}}$ denote a uniform upper bound on the precision of each mean cost estimate necessary to *approximately* reconstruct the set $\mathcal{D}(\mu)$. That is, $\Delta^{\mathcal{D}} := \min\left\{\Delta^{\mu}_{min}, \Delta^{\mathcal{D}}_2, \Delta^{\mathcal{D}}_3\right\}/(2s)$, where

$$\begin{aligned}
\Delta_2^{\mathcal{D}} &:= \min\left\{\min\left\{\Delta_S^{(\mu\wedge l)(D)} : S \notin \mathcal{S}^*\left(\left(\mu \wedge l\right)\left(D\right)\right)\right\} : D \subseteq A\backslash H,\, \mathcal{S}^*\left(\mu\right) = \mathcal{S}^*\left(\left(\mu\wedge l\right)\left(D\right)\right)\right\}, \\
\Delta_3^{\mathcal{D}} &:= \min\left\{z^*_{Comb}(\mu) - z^*_{Comb}\left(\left(\mu\wedge l\right)\left(D\right)\right) : D \subseteq A\backslash H,\, \mathcal{S}^*\left(\mu\right) \neq \mathcal{S}^*\left(\left(\mu\wedge l\right)\left(D\right)\right)\right\},
\end{aligned}$$

Δ^{μ}_{min} is as defined in Step 1, $H := \bigcup_{S^*\in\mathcal{S}^*(\mu)}\bigcup_{a\in S^*}\{a\}$, and $\left(\mu\wedge l\right)\left(D\right) = \left(\mu(a),\, a \in A\backslash D\right) \cup \left(l(a) : a \in D\right)$. The first threshold Δ^{μ}_{min} ensures that $\mathcal{S}^*(\widehat{\mu}_n) \subseteq \mathcal{S}^*(\mu)$. This is supported by Step 1 (see (A.9)). The second threshold $\Delta^{\mathcal{D}}_2$ ensures that

$$\mathcal{D}(\widehat{\mu}_n) \subseteq \mathcal{D}(\mu) \cup 2^H.$$

This follows from noting that: (i) for $D \notin \mathcal{D}\left(\mu\right)$,

$$\bigcap_{a\in A}\left\{\left|\widehat{\mu}_n(a) - \mu(a)\right| < \Delta^{\mathcal{D}}\right\} \subseteq \left\{z^*_{Comb}(\widehat{\mu}_n) = z^*_{Comb}\left(\left(\widehat{\mu}_n\wedge l\right)\left(D\right)\right)\right\},$$

implying that $D \notin \mathcal{D}\left(\widehat{\mu}_n\right)$; and (ii) not all solutions in $\mathcal{S}^*(\mu)$ are necessarily optimal in the approximate problem (i.e., using the average costs), therefore, some of their ground elements might belong to $\mathcal{D}(\widehat{\mu}_n)$. The third threshold $\Delta^{\mathcal{D}}_3$ ensures that $\mathcal{D}(\mu) \subseteq \mathcal{D}(\widehat{\mu}_n)$. This follows from noting that for $D \in \mathcal{D}(\mu)$,

$$\bigcap_{a\in A}\left\{\left|\widehat{\mu}_n(a) - \mu(a)\right| < \Delta^{\mathcal{D}}\right\} \subseteq \left\{z^*_{Comb}(\widehat{\mu}_n) > z^*_{Comb}\left(\left(\widehat{\mu}_n\wedge l\right)\left(D\right)\right)\right\},$$

implying that $D \in \mathcal{D}\left(\widehat{\mu}_n\right)$. We conclude that

$$\bigcap_{a\in A}\left\{\left|\widehat{\mu}_n(a) - \mu(a)\right| < \Delta^{\mathcal{D}}\right\} \subseteq \left\{\mathcal{D}\left(\widehat{\mu}_n\right) = \mathcal{D}\left(\mu\right) \cup H_o\right\},$$

for some $H_o \in 2^H$. While we assume, without loss of generality, that Δ^{μ}_{min} and $\Delta^{\mathcal{D}}_2$ are positive (since otherwise, the problem is trivial), Assumption 1 implies that $\Delta^{\mathcal{D}}_3 > 0$. Thus, we have that $\Delta^{\mathcal{D}} > 0$.

Consider now the issue of approximating the K_D constants. We denote such estimates by $\hat{K}_D$. By the continuity of $I_a(\cdot,\cdot)$ for all $a \in A$, we have that $K_D(\nu)$ is also continuous for all $D \in \mathcal{D}(\mu)$. In addition, because it is known that $K_D(\mu) \leqslant K$, there exists a finite constant $\kappa > 0$ such that

$$\left|\hat{K}_D(\widehat{\mu}_n) - K_D(\mu)\right| \leqslant \kappa \sum_{a \in A} \left|\widehat{\mu}_n(a) - \mu(a)\right|,$$

for $\widehat{\mu}_n$ in a neighborhood of μ (specifically, we consider a ball -using infinite norm- of radious lower than $\varrho/\left(|A|\,\kappa\right)$ centered at μ for $\varrho > 0$ arbitrary). Note that we make use of the uniform bound and use the approximation

$$\hat{K}_D(\nu) := K_D(\nu) \wedge K.$$

This, in turn, implies that $\hat{K}_D(\nu) \leqslant K$.

Define $\Delta^K := \varrho/\left(|A|\,\kappa\right)$ for $\varrho > 0$ arbitrary. We conclude that

$$\bigcap_{a \in A} \left\{\left|\widehat{\mu}_n(a) - \mu(a)\right| < \Delta^K\right\} \subseteq \left\{\left|\hat{K}_D(\widehat{\mu}_n) - K_D(\mu)\right| < \varrho,\ D \in \mathcal{D}(\mu)\right\}.$$

Let $(x^n, y^n) \in \Gamma_{LBP}(\widehat{\mu}_n)$, and consider $(x^*, y^*) \in \Gamma_{LBP}(\mu)$, augmented so that $y^*\left(S^*\right) = K$ for all $S^* \in \mathcal{S}^*(\mu)$ (note that because $\Delta^{\mu}_{S*} = 0$ for all $S^* \in \mathcal{S}^*(\mu)$, one can make this augmentation without affecting the objective value of $LBP(\mu)$). Suppose that $\|\widehat{\mu}_n - \mu\|_\infty < \delta/(2s)$ for some $0 < \delta < \min\left\{\Delta^K, \Delta^{\mathcal{D}}, \varrho\right\}$, then we have that

$$\begin{aligned}
\max\left\{x^n(a) + \delta : a \in D\right\} &\geqslant K_D(\mu),\ D \in \mathcal{D}(\mu) \\
\max\left\{x^*(a) + \delta : a \in D\right\} &\geqslant \hat{K}_D(\widehat{\mu}_n),\ D \in \mathcal{D}(\widehat{\mu}_n).
\end{aligned} \tag{A.14}$$

For $z \in \mathbb{R}^k$ and $\delta > 0$, we define z^δ so that $z^\delta(j) := z(j) + \delta \mathbf{1}\{z(j) > 0\}$, $j \leqslant k$, where $z(j)$ is the j-th element of z. From (A.14) we conclude that $(x^{*,\delta}, y^{*,\delta})$ is feasible to $LBP(\widehat{\mu}_n)$. Seeing that $\|\widehat{\mu}_n - \mu\|_\infty < \delta/(2s)$, we have $\left|\Delta_S^\mu - \Delta_S^{\widehat{\mu}_n}\right| < \delta$ for all $S \in \mathcal{S}$. Therefore, we have that

$$\begin{aligned}
\sum_{S \in \mathcal{S}} y^n(S)\Delta_S^\mu &\overset{(a)}{\leqslant} \sum_{S \in \mathcal{S}} y^n(S)\Delta_S^{\widehat{\mu}_n} + |\mathcal{S}|\, K\delta \\
&\overset{(b)}{\leqslant} \sum_{S \in \mathcal{S}} y^{*,\delta}(S)\Delta_S^{\widehat{\mu}_n} + |\mathcal{S}|\, K\delta \\
&\overset{(c)}{\leqslant} \sum_{S \in \mathcal{S}} y^*(S)\Delta_S^\mu + \delta\, |\mathcal{S}|\, (\delta + \Delta_{max}^\mu + K) + |\mathcal{S}|\, K\delta \\
&= z_{LBP}^*(\mu) + \delta\, |\mathcal{S}|\, (\delta + \Delta_{max}^\mu + 2K)\,,
\end{aligned}$$

where (a) follows from the fact that $y^n(S) \leqslant K$ for all $S \in \mathcal{S}$ (this because $\hat{K}_D(\widehat{\mu}_n) \leqslant K$), (b) comes from that $(x^{*,\delta}, y^{*,\delta})$ is feasible to $LBP(\widehat{\mu}_n)$, and (c) follows from that $\left|\Delta_S^\mu - \Delta_S^{\widehat{\mu}_n}\right| < \delta$ and $y^{*,\delta}(S) \leqslant y^*(S) + \delta$ for all $S \in \mathcal{S}$, and $y^*(S) \leqslant K$ for all $S \in \mathcal{S}$. Seeing that $\delta < \Delta^{\mathcal{D}} < \Delta_{min}^\mu$, taking $\delta \leqslant \varrho\, z_{LBP}^*(\mu)/\left(|\mathcal{S}|\,(\Delta_{min}^\mu + \Delta_{max}^\mu + 2K)\right)$, we have that

$$\sum_{S \in \mathcal{S}} y^n(S)\Delta_S^\mu \leqslant (1 + \varrho) z_{LBP}^*(\mu).$$

Consider $i > i'$ and let $(x_i, y_i) \in \Gamma_{LBP}(\widehat{\mu}_{n_i})$ be the solution used for LBP-based exploration for $n \in [n_i, n_{i+1} - 1]$. In what follows, with abuse of notation, we use the event $\{S_n \in \Gamma_{LBP}(\widehat{\mu}_n)\}$ to denote the LBP-based exploration. We have that

$$\begin{aligned}
R_1^{LBP}(F,N) \quad &\leqslant \quad n_{i'}\Delta^{\mu}_{max} + \sum_{i=i'}^{\lceil (\ln N)^{1+\varepsilon}\rceil} \sum_{n=n_i}^{n_{i+1}-1} \mathbb{E}\left\{\mathbf{1}\left\{T_{n_i}(a) \geqslant \gamma(i-1)\ \forall a \in A\,,\ S_n \in \Gamma_{LBP}(\widehat{\mu}_n)\right\}\Delta^{\mu}_{S_n}\right\} \\
&\leqslant n_{i'}\Delta^{\mu}_{max} + \sum_{i=i'}^{\lceil (\ln N)^{1+\varepsilon}\rceil} \sum_{n=n_i}^{n_{i+1}-1} \mathbb{E}\left\{\mathbf{1}\left\{T_{n_i}(a) \geqslant \gamma(i-1), |\widehat{\mu}_n(a)-\mu(a)| < \delta/(2s), \forall a \in A, S_n \in \Gamma_{LBP}(\widehat{\mu}_n)\right\}\Delta^{\mu}_{S_n}\right\} \\
&+ \sum_{i=i'}^{\lceil (\ln N)^{1+\varepsilon}\rceil} \sum_{n=n_i}^{n_{i+1}-1} \mathbb{E}\left\{\mathbf{1}\left\{T_{n_i}(a) \geqslant \gamma(i-1), \forall a \in A, \cup_{a\in A}\left\{|\widehat{\mu}_n(a)-\mu(a)| \geqslant \delta/(2s)\right\}, S_n \in \Gamma_{LBP}(\widehat{\mu}_n)\right\}\Delta^{\mu}_{S_n}\right\} \\
&\leqslant \quad n_{i'}\Delta^{\mu}_{max} + \sum_{i=i'}^{\lceil (\ln N)^{1+\varepsilon}\rceil} (1+\varrho)\ z^*_{LBP}(\mu) \\
&\quad + \sum_{i=i'}^{\infty} (n_{i+1}-n_i)\,\Delta^{\mu}_{max} \sum_{a\in A} \mathbb{P}\left\{|\widehat{\mu}_n(a)-\mu(a)| \geqslant \delta/(2s), T_{n_i}(a) \geqslant \gamma(i-1)\right\}. \qquad \text{(A.15)}
\end{aligned}$$

Using Proposition 30 to bound the probability in (A.15), we have that

$$R_1^{LBP}(F,N) \leqslant n_{i'}\Delta^{\mu}_{max} + (\ln N)^{1+\varepsilon}(1+\varrho)\ z^*_{LBP}(\mu) + \sum_{i=i'}^{\infty} e^{(i+1)^{1/(\varepsilon+1)}}\Delta^{\mu}_{max} 2\,|A| \exp\left\{-\frac{\delta^2\ \gamma(i-1)}{2s^2\mathcal{L}^2}\right\}.$$

Because $(i+1)^{1/(1+\varepsilon)} < i\frac{\delta^2\gamma}{2s^2\mathcal{L}^2}$ for i large enough, we conclude that

$$R_1^{LBP}(F,N) \leqslant (\ln N)^{1+\varepsilon}(1+\varrho)\ z^*_{LBP}(\mu) + C_3$$

for a positive finite constant C_3, independent of N. Putting all the above together, we conclude that

$$R^{\pi^*}(F,N) \leqslant ((1+\varrho)\, z^*_{LBP}(\mu) + \gamma\, z^*_{Cover}(\mu))\,(\ln N)^{1+\varepsilon} + C_4,$$

for a finite positive constant C_4, independent of N.

We finally note that the optimal solutions to the $Cover(\widehat{\mu}_{n_i})$ and $LBP(\widehat{\mu}_{n_i})$ problems converge a.s. to optimal and ϱ-optimal solutions to $Cover(\mu)$ and $LBP(\mu)$, respectively. For this, note that Proposition 30, (A.13) and (A.15) imply (via Borel-Cantelli) that $\mathbb{P}\{\mathcal{E}_i \in \Gamma_{Cover}(\mu) \text{ e.v.}\} = 1$ and

$$\mathbb{P}\left\{(x^{\varrho}_i, y^{\varrho}_i) \text{ is a } \varrho\text{-optimal solution to } LBP(\mu) \text{ e.v.}\right\} = 1.$$

The result follows from noting that one can choose ϱ arbitrarily small.

$\square$

A.1.5 Performance Gap Analysis

In this section we provide a discussion on the gap between the lower and upper bounds in Theorems 5 and 6.

Optimal Scaling with Respect to $\boldsymbol{N}$. While it is possible to achieve the optimal $\ln N$ dependence in the upper bound in Theorem 6 (through a different definition of cycles n_i and the introduction of additional tunable parameters), this comes at the price of additional constants in front of the second term in the right-hand side of (2.5). We introduce an additional sub-logarithmic term to the optimal $\ln N$ scaling, so as to avoid introducing terms that emanate in part from the proof techniques, and so as to have a bound that reflects a fundamental insight about the result: asymptotic regret arises from suboptimal exploration which in the near-optimal policy (i.e., the LBP-based policy) is distributed between the solution to LBP problem and, at a lower frequency, the solution to $Cover$ problem.

Improved Upper Bounds. By setting γ arbitrarily close to zero, one can set the leading constant in the right-hand side of (2.5) arbitrarily close to that in Theorem 5 up to sub-logarithmic terms. However, it is not possible to set $\gamma = 0$ in general, as illustrated in Example 28, as this would not guarantee convergence on the solution to LBP.

It is possible, however, to reduce the gap between the leading constants in Theorems 5 and 6. For that, instead of complementing exploration on the solution to LBP with the solution to $Cover$, one can find a minimum-regret solution set that fulfills condition (2.2) and is robust to changes in the mean cost of unexplored ground elements. That is, one can design a policy whose regret admits a bound of the form

$$\lim_{N\to\infty} \frac{R^{\pi^*(\gamma,\varepsilon)}(F,N)}{(\ln N)^{1+\varepsilon}} \leqslant z^*_R(\mu,\gamma),$$

for $\gamma > 0$, where $z^*_R(\nu,\gamma)$ is the optimal solution to a "robust" variation of $LBP(\nu)$ for a given cost vector $\nu \in \mathcal{B}$ (this formulation is presented in Appendix A.1.6), and such that

$$z^*_{LBP}(\nu) \leqslant z^*_R(\nu,\gamma) \leqslant z^*_{LBP}(\nu) + \gamma\, z^*_{Cover}(\nu).$$

While we do not prove such bounds here (this requires more convoluted, lengthier arguments), the insight derived from it remains the same: regret emanates from sub-optimal exploration. There are settings, however, where samples from exploring the solution to LBP suffice to guarantee the optimality of said solution. For example, one can show that if $Comb(\cdot)$ is a *weighted basis* or *independent set matroid* minimization problem, feedback from the solution to LBP suffices to guarantee its optimality. In such cases one has that $z^*_{LBP}(\nu) = z^*_R(\nu,\gamma)$ for all $\gamma > 0$ and $\nu \in \mathcal{B}$, thus a variation of the LBP-based policy focusing exploration exclusively on a ϱ-optimal solution to the robust LBP problem (and not a cover) admits a performance bound of

$$\lim_{N\to\infty} \frac{R^{\pi^*(\gamma,\varepsilon,\varrho)}(F,N)}{(\ln N)^{1+\varepsilon}} \leqslant (1+\varrho) z^*_{LBP}(\mu),$$

for $\varrho > 0$ and $\varepsilon > 0$ arbitrary (see Remark 7 for the role of ϱ in said algorithm).

Improved Lower Bounds. As shown above, in general it is not possible to improve the leading constant in (2.5) as finding and validating an optimal solution to $LBP(\mu)$ might require knowledge of the mean costs of ground elements that are not explored by said solution. Hence, to find an optimal solution of $LBP(\mu)$ we may need complementary exploration through a cover or a robust version of $LBP(\mu)$. In contrast, our theoretical lower bound assumes advance knowledge of these unexplored costs, which allows it to bypass this complementary exploration. This difference is precisely the source of the gap between the leading constants in (2.4) and (2.5). It may be possible to derive an improved lower bound by not assuming such an advance knowledge. Unfortunately, it is not clear how to derive such a bound using the techniques in this paper or previous work on bandits.

A.1.6 Adjoint Formulation for Tighter Upper Bound

The following formulation is a variation of LBP that is robust with respect to changes in the mean cost vector of elements that are not "covered" by its optimal solution. For that, we introduce an additional variable $w(a)$ indicating whether one would impose additional exploration (beyond that required in the lower bound result - the parameter γ indicates the frequency of such exploration) on a ground element $a \in A$, and variable $r(a)$ indicates the degree at which element $a \in A$ is covered in a solution. The variable $\underline{z}$ computes the minimum cost attainable if one were to change the mean cost of an unexplored ground element. Finally, the formulation imposes that the optimal cost is not lower than such an alternative minimum cost. In the formulation below, we define $\nu' := (\nu \wedge l)\left(\{a \in A : r(a) = 0\}\right)$ where for a set

D, $(\nu \wedge l)(D) = (\mu(a)\mathbf{1}\{a \notin D\} + l(a)\mathbf{1}\{a \in D\} : a \in A)$.

$$
\begin{aligned}
z^*_R(\nu,\gamma) := \min \quad & \sum_{S\in\mathcal{S}} \Delta^{\nu}_S \, y(S) \\
s.t. \qquad & \underline{z} = \min_{y'\in\mathbb{R}^{|\mathcal{S}|}_+} \left\{ \sum_{S\in\mathcal{S}} \Delta^{\nu'}_S \, y'(S) : r(a) \leqslant \sum_{S\in\mathcal{S}:a\in S} y'(S),\ a \in A \right\} \\
& \sum_{S\in\mathcal{S}} \Delta^{\nu}_S \, y(S) \leqslant \underline{z} \\
& r(a) \leqslant \sum_{S\in\mathcal{S}:a\in S} y(S),\ a \in A \\
& r(a) = x(a) + \gamma\, w(a),\ a \in A \\
& \max\{x(a) : a \in D\} \geqslant K_D(\nu), \quad D \in \mathcal{D}(\nu) \\
& w(a) \in \{0,1\},\ x(a),\ r(a), y(S) \in \mathbb{R}_+, \quad a \in A,\ S \in \mathcal{S}.
\end{aligned}
$$

A.2 Omitted Proofs and Materials from Section 2.5

A.2.1 Equivalence of LBP and OCP

In this section we prove the equivalence of LBP and OCP problems in Lemma 9.

Proof of Lemma 9. We prove the result by contradiction. Let $R\text{-}OCP(\mu)$ denote the linear relaxation of $OCP(\mu)$. Without loss of generality, we assume that $K = 1$. Let (x, y) be a feasible solution to $R\text{-}OCP(\mu)$, and suppose that there exists a $D \in \mathcal{D}$ such that $\max\{x(a) : a \in D\} = 0$. By the definition of $\mathcal{D}$, one has that $z^*_{Comb}((\mu \wedge l)(D)) < z^*_{Comb}(\mu)$, thus for $S^* \in \mathcal{S}^*((\mu \wedge l)(D))$, we have that

$$
\begin{aligned}
z^*_{Comb}((\mu \wedge l)(D)) \quad &= \quad \sum_{a\in S^*\setminus D} \mu(a) + \sum_{a\in D} l(a) \\
&\overset{(a)}{\geqslant} \quad \sum_{a\in S^*} \left(l(a)(1 - x(a)) + \mu(a)\, x(a)\right) \\
&\overset{(b)}{\geqslant} \quad z^*_{Comb}(\mu),
\end{aligned}
$$

where (a) follows from the fact that $l(a) = (l(a)(1-x(a)) + \mu(a)\,x(a))$, for $a \in D$, and $\mu(a) \geqslant (l(a)(1-x(a)) + \mu(a)\,x(a))$, for $a \notin D$, and (b) follows from the fact that (x, y) satisfies constraints (2.7c) (because it is feasible to $R\text{-}OCP(\mu)$). The last inequality above contradicts $z^*_{Comb}((\mu \wedge l)\,(D)) < z^*_{Comb}(\mu)$, thus we have that $\max\{x(a) : a \in D\} = 1$ for all $D \in \mathcal{D}$, therefore (x, y) is feasible to $LBP\,(\mu)$.

Now, let (x, y) be a feasible solution to $LBP\,(\mu)$ such that $x(a) \in \{0, 1\}$ for all $a \in A$ (note that when $K_D = 1$ for all $D \in \mathcal{D}$, one can restrict attention only to feasible solutions to $LBP\,(\mu)$ with binary x), and that $x(a) = 1$ and $y(S^*) = 1$ for all $a \in S^*$ and $S^* \in \mathcal{S}^*(\mu)$ (because $\Delta^{\mu}_{S^*} = 0$ for all $S^* \in \mathcal{S}^*(\mu)$, this extra requirement does not affect the optimal solution to $LBP\,(\mu)$). Suppose (x, y) is not feasible to $R\text{-}OCP(\mu)$, i.e., there exists some $S \in \mathcal{S}$ such that

$$\sum_{a \in S} (l(a)(1-x(a)) + \mu(a)\,x(a)) < z^*_{Comb}(\mu). \tag{A.17}$$

Let S_0 be one such S that additionally minimizes the left-hand side in (A.17) (in case of ties we pick any minimizing solution S_0 with smallest value of $|\{a \in S_0 : x(a) = 0\}|$). Then $D := \{a \in S_0 : x(a) = 0\}$ (or a subset of D) belongs to $\mathcal{D}$. This contradicts the feasibility of (x, y) to $LBP\,(\mu)$, because if (x, y) is feasible to $LBP\,(\mu)$, then we must have $\max\{x(a) : a \in D\} \geqslant 1$ for all $D \in \mathcal{D}$. Thus, we conclude that (x, y) is feasible to $R\text{-}OCP(\mu)$.

Summarizing, when $K_D = 1$ for all $D \in \mathcal{D}$, feasible solutions to $R\text{-}OCP(\mu)$ are feasible to $LBP\,(\mu)$, and feasible solutions to $LBP\,(\mu)$ that cover all optimal elements in A are feasible to $R\text{-}OCP(\mu)$. The result follows from noting that there always exists an optimal solution to $LBP\,(\mu)$ such that x is binary, and $x(a) = 1$ and $y(S^*) = 1$ for all $a \in S^*$ and $S^* \in \mathcal{S}^*(\mu)$. $\square$

A.2.2 Modified OCP-Based Policy

In this section we discuss a variant of the OCP-based policy for which we can prove a performance guarantee. We obtain this policy based on OCP by simply exploring the elements of $\Gamma_{OCP}(\widehat{\mu}_n)$ with frequency $(\ln N)^{1+\varepsilon}/N$ (more precisely, by implementing $\mathcal{G}$ for some $(C, \mathcal{G}) \in \Gamma_{OCP}(\widehat{\mu}_n)$). Enforcing this exploration frequency and updating $\Gamma_{OCP}(\widehat{\mu}_n)$ can again be achieved through the doubling trick where we define the cycles as $n_i := \max\{\lfloor e^{i^{1/(1+\varepsilon)}} \rfloor, n_{i-1}+1\}$ with $\varepsilon > 0$ arbitrary, for all $i \geqslant 2$. However, to obtain an asymptotic performance bound on this policy, we need additional modifications similar to those for the LBP-based policy in Algorithm 2. The resulting policy, which we refer to as the *modified OCP-based* policy and denote by π'_{OCP}, is depicted in Algorithm 3.

To prove an asymptotic performance bound on the policy described by Algorithm 3, we need a relaxed version of Assumption 1.

Assumption 2. *There is no set $D \subseteq A$ such that $z^*_{Comb}(\mu) = z^*_{Comb}((\mu \wedge l)(D))$ with $\mathcal{S}^*(\mu) \neq \mathcal{S}^*((\mu \wedge l)(D))$.*

Note that Assumption 2 ensures that Constraint (2.7c) is not active for any $S \notin \mathcal{S}^*(\mu)$ and any vectors x and y satisfying (2.7b) and (2.7d).

As discussed in Section A.1.3, this assumption holds when, for example, mean costs are randomly drawn from an absolutely continuous distribution. This suits most practical settings where mean costs are unknown and no particular structure for them is anticipated.

Remark 7. Note that parameter ϱ in Algorithm 3 allows the policy to converge to an optimal solution to $OCP(\mu)$ (because there might exist multiple optimal solutions, the "Update OCP-exploration set" step ensures that the policy settles on one of them). Also, as in Algorithm 2, γ is a tuning parameter that controls the cover-based exploration frequency.

Algorithm 3 Modified OCP-based policy $\pi'_{OCP}(\gamma, \varepsilon, \varrho)$

Set $i = 0$, $C = A$, $\mathcal{E}$ a minimal cover of A, $\mathcal{G} = \mathcal{E}$, and draw $(b_1(a) : a \in A)$ randomly from $\mathcal{B}$
for $n = 1$ to N **do**
  if $n = n_i$ **then**
    Set $i = i + 1$
    Set $S^* \in \mathcal{S}^*(\widehat{\mu}_n)$ [Update exploitation set]
    Set $\mathcal{E} \in \Gamma_{Cover}(\widehat{\mu}_n)$ [Update Cover-exploration set]
    if $(C, \mathcal{G})$ is not a ϱ-optimal solution to $OCP(\widehat{\mu}_n)$ **then**
      Set $(C, \mathcal{G}) \in \Gamma_{OCP}(\widehat{\mu}_n)$ [Update OCP-exploration set]
    end if
  end if
  if $T_n(a) < \gamma\, i$ for some $a \in A$ **then**
    Set $S_n = S$ for any $S \in \mathcal{E}$ such that $a \in S$ [Cover-based exploration]
  else if $\gamma < 1$ and $T_n(a) < i$ for some $a \in C$ **then**
    Set $S_n = S$ for any $S \in \mathcal{G}$ such that $a \in S$ [OCP-based exploration]
  else
    Set $S_n = S^*$ [Exploitation]
  end if
end for

Suppose that Assumption 2 holds. We then obtain the following performance bound for the policy $\pi'_{OCP}(\gamma, \varepsilon, \varrho)$. There, z^*_{Cover} refers to $z^*_{Cover}(\mu)$.

Theorem 10. *Consider $\gamma \in (0, 1)$, $\varrho > 0$, and $\varepsilon > 0$ arbitrary. We then have that for ϱ sufficiently small*

$$\lim_{N \to \infty} \frac{R^{\pi'_{OCP}(\gamma,\varepsilon,\varrho)}(F, N)}{(\ln N)^{1+\varepsilon}} \leqslant z^*_{OCP}(\mu) + \gamma\, z^*_{Cover}.$$

Proof of Theorem 10. As in the case of the LBP-based policy π^*, the regret of policy π'_{OCP} (we again ignore the dependence of the policy on γ, ε, and ϱ to simplify the notation) stems from three sources: Cover-based and OCP-based exploration efforts, and exploitation errors. That is,

$$R^{\pi'_{OCP}}(F, N) = R_1^{Cover}(F, N) + R_1^{OCP}(F, N) + R_2(F, N), \tag{A.18}$$

where $R_1^{Cover}(F, N)$ is the Cover-based exploration regret, i.e., that incurred at period n during cycle i if $T_n(a) < \gamma\, i$ for some $a \in A$, $R_1^{OCP}(F, N)$ is the OCP-based exploration regret, i.e., that incurred at period n during cycle i if $T_n(a) < i$ for some $a \in C$, and $R_2(F, N)$ is the exploitation-based regret, i.e., that incurred when exploration conditions are met and one implements solution $S_n = S^*$ with $S^* \in \mathcal{S}^*(\widehat{\mu}_n)$.

We prove the result by bounding each term in (A.18) separately. It turns out that the bounds for $R_1^{Cover}(F, N)$ and $R_2(F, N)$ in Step 1 and Step 2.1 in the proof of Theorem 6 apply to this setting unmodified, thus we omit them here. Next, we bound the OCP-based exploration regret $R_1^{OCP}(F, N)$.

As in the proof of the LBP-based policy, in the remainder of this proof, $\mathbb{E}$ and $\mathbb{P}$ denote expectation and probability when costs are distributed according to F and policy π'_{OCP} is implemented.

Step 2.2' (OCP-based exploration regret).

Following the arguments in Step 2.2 of the proof of Theorem 6, we first define the minimum precision threshold on the accuracy of mean cost estimates necessary to reconstruct the solution to $OCP(\mu)$. For that, we define

$$\Delta^{\mathcal{D}} := \min \left\{ \Delta^{\mu}_{min}, \Delta^{\mathcal{D}}_2, \Delta^{\mathcal{D}}_3, \Delta^{\mathcal{D}}_4 \right\} / (8sc),$$

where

$$\begin{aligned}
\Delta^{\mathcal{D}}_2 &:= \min \left\{ \min \left\{ \Delta_S^{(\mu \wedge l)(D)} : S \notin \mathcal{S}^* \left((\mu \wedge l) (D) \right) \right\} : D \subseteq A \backslash H, \, \mathcal{S}^* (\mu) = \mathcal{S}^* \left((\mu \wedge l) (D) \right) \right\}, \\
\Delta^{\mathcal{D}}_3 &:= \min \left\{ z^*_{Comb}(\mu) - z^*_{Comb} \left((\mu \wedge l) (D) \right) : \, D \subseteq A \backslash H, \, \mathcal{S}^* (\mu) \neq \mathcal{S}^* \left((\mu \wedge l) (D) \right) \right\}, \\
\Delta^{\mathcal{D}}_4 &:= \min \left\{ \left(\sum_{S \in \mathcal{G}} \Delta^{\mu}_S \right) - z^*_{OCP}(\mu) : (C, \mathcal{G}) \in \mathbf{G} \backslash \Gamma_{OCP}(\mu) \right\},
\end{aligned}$$

and $\mathbf{G}$ denotes the set of all feasible solutions $(C, \mathcal{G})$ to $OCP(\mu)$ problem. Note that as in the proof of Theorem 6, $\Delta^{\mu}_{min} = \min \{ \Delta^{\mu}_S : S \in \mathcal{S} \backslash \mathcal{S}^* (\mu) \}$, $s = \max \{ |S| : S \in \mathcal{S} \}$, $c = \max \{ |\mathcal{E}| : \mathcal{E} \in \mathbf{C} \}$, i.e., the maximum size of a minimal cover of A, and $H = \bigcup_{S^* \in \mathcal{S}^*(\mu)} \bigcup_{a \in S^*} \{a\}$. Also note that $\Delta^{\mathcal{D}}_4$ denotes the minimum optimality gap of problem $OCP(\mu)$. Note that thresholds Δ^{μ}_{min}, $\Delta^{\mathcal{D}}_2$ and $\Delta^{\mathcal{D}}_4$ are always positive, while $\Delta^{\mathcal{D}}_3 > 0$ by Assumption 2.

We now check that having mean cost estimates with enough precision allows to reconstruct the feasible set $\mathbf{G}$. Consider (x, y) satisfying (2.7b) and (2.7d). We first note that as discussed in Step 1 of the proof of Theorem 6, $\{\|\widehat{\mu}_n - \mu\|_\infty < \Delta^{\mu}_{min}/(2s)\}$ ensures that $\{\mathcal{S}^*(\widehat{\mu}_n) \subseteq \mathcal{S}^*(\mu)\}$. One then has that

$$\left\{\|\widehat{\mu}_n - \mu\|_\infty < \Delta^{\mathcal{D}}\right\} \subseteq \left\{\left|\sum_{a\in S} x(a)\left(\widehat{\mu}_n(a) - \mu(a)\right)\right| < \Delta^{\mathcal{D}} s,\ \forall S \in \mathcal{S}\right\} \cap \left\{\left|z^*_{Comb}(\widehat{\mu}_n) - z^*_{Comb}(\mu)\right| < \Delta^{\mathcal{D}} s\right\}$$

$$\subseteq \left\{\left|\left(\sum_{a\in S}(x(a)\widehat{\mu}_n(a) + (1-x(a))l(a)) - z^*_{Comb}(\widehat{\mu}_n)\right) - \left(\sum_{a\in S}(x(a)\mu(a) + (1-x(a))l(a)) - z^*_{Comb}(\mu)\right)\right| < 2\Delta^{\mathcal{D}} s,\ \forall S \in \mathcal{S}\right\}.$$

We conclude that, because $2\Delta^{\mathcal{D}} s < \Delta^{\mathcal{D}}_2 \wedge \Delta^{\mathcal{D}}_3$,

$$\sum_{a\in S}(x(a)\mu(a) + (1-x(a))l(a)) \geqslant z^*_{Comb}(\mu) \text{ iff } \sum_{a\in S}(x(a)\widehat{\mu}_n(a) + (1-x(a))l(a)) \geqslant z^*_{Comb}(\widehat{\mu}_n).$$

Having the same feasible region for both $OCP(\mu)$ and $OCP(\widehat{\mu}_n)$ problems, we now show that ϱ-optimal solutions to the latter problem corresponds to an optimal solution to the former. Indeed, we have that

$$\begin{aligned}
\left\{\|\widehat{\mu}_n - \mu\|_\infty < \Delta^{\mathcal{D}}\right\} &\subseteq \left\{\left|\Delta^{\widehat{\mu}_n}_S - \Delta^{\mu}_S\right| < \frac{\Delta^{\mathcal{D}}_4}{4c},\ \forall S \in \mathcal{S}\right\} \\
&\subseteq \left\{\left|\sum_{S\in\mathcal{G}}\left(\Delta^{\widehat{\mu}_n}_S - \Delta^{\mu}_S\right)\right| < \frac{\Delta^{\mathcal{D}}_4}{4},\ \forall (C,\mathcal{G}) \in \mathbf{G}\right\}
\end{aligned}$$

$$\subseteq \left\{\sum_{S\in\mathcal{G}}\Delta^{\widehat{\mu}_n}_S > \Delta^{\mathcal{D}}_4/2 + \sum_{S\in\mathcal{G}^*}\Delta^{\widehat{\mu}_n}_S,\ \forall (C^*,\mathcal{G}^*) \in \Gamma_{OCP}(\mu),\ (C,\mathcal{G}) \in \mathbf{G}\backslash\Gamma_{OCP}(\mu)\right\}.$$

The above not only implies that $\Gamma_{OCP}(\widehat{\mu}_n) \subseteq \Gamma_{OCP}(\mu)$, but also that ϱ-optimal solutions to $OCP(\widehat{\mu}_n)$ are also optimal to $OCP(\mu)$, as long as $\varrho < \Delta^{\mathcal{D}}_4/2$. Letting $\Gamma^{\varrho}_{OCP}(\nu)$ denote the set of ϱ-optimal solutions to $OCP(\nu)$, the above implies that for $\varrho < \Delta^{\mathcal{D}}_4/2$,

$$\left\{\|\widehat{\mu}_n - \mu\|_\infty < \Delta^{\mathcal{D}}\right\} \subseteq \left\{\Gamma^{\varrho}_{OCP}(\widehat{\mu}_n) \subseteq \Gamma_{OCP}(\mu)\right\}.$$

We are now ready to provide a bound on $R_1^{OCP}(F,N)$. Similar to the proof of Theorem 6, let i' be a finite upper bound on a cycle in which one is sure to conduct all OCP-based explorations (e.g., $i' := 1 + \inf\{i \in \mathbb{N}, i \geqslant 2 : n_{i+1} - n_i > i\,|A|\}$). Fix $i > i'$ and let $(C_i, \mathcal{G}_i)$ denote the OCP-based exploration set for any period $n \in [n_i, n_{i+1}-1]$. Define the events $\Xi_i^1 := \{(C_i,\mathcal{G}_i) \in \Gamma_{OCP}(\mu)\}$ and $\Xi_i^2 := \{\mathcal{G}_i = \mathcal{G}_{i-1}\}$. We then have that

$$
\begin{aligned}
R_1^{OCP}(F,N) \quad \leqslant \quad & n_{i'+1}\Delta^{\mu}_{max} + \sum_{i=i'+1}^{\lceil (\ln N)^{1+\varepsilon}\rceil} \mathbb{E}\left\{\mathbf{1}\left\{T_{n_{i-1}}(a) \geqslant \gamma(i-2)\,,\, \forall a \in A\,,\, (\Xi_i^1 \cap \Xi_i^2)\right\} \sum_{S \in \mathcal{G}_i} \Delta^{\mu}_S\right\} \\
& + \sum_{i=i'+1}^{\lceil (\ln N)^{1+\varepsilon}\rceil} i\; \mathbb{E}\left\{\mathbf{1}\left\{T_{n_{i-1}}(a) \geqslant \gamma(i-2)\,,\, \forall a \in A\,,\, (\Xi_i^1 \cap \Xi_i^2)^c\right\} \sum_{S \in \mathcal{G}_i} \Delta^{\mu}_S\right\} \\
\leqslant \quad & n_{i'+1}\Delta^{\mu}_{max} + \left((\ln N)^{1+\varepsilon} + 1\right) z^*_{OCP}(\mu) \\
& + \Delta^{\mu}_{max}\, c \sum_{i=i'+1}^{\infty} i\; \mathbb{P}\left\{T_{n_{i-1}}(a) \geqslant \gamma(i-2)\,,\, \forall a \in A\,,\, (\Xi_i^1 \cap \Xi_i^2)^c\right\}, \qquad \text{(A.19)}
\end{aligned}
$$

where $(\Xi_i^1 \cap \Xi_i^2)^c$ denotes the complement of the event $(\Xi_i^1 \cap \Xi_i^2)$. Next, we bound the probability inside the sum in (A.19). For that, observe that

$$
\begin{aligned}
\left\{\|\widehat{\mu}_{n_{i-1}} - \mu\|_\infty \vee \|\widehat{\mu}_{n_i} - \mu\|_\infty < \Delta^{\mathcal{D}}\right\} \quad \subseteq \quad & \left\{\Gamma^{\varrho}_{OCP}(\widehat{\mu}_{n_{i-1}}) \subseteq \Gamma_{OCP}(\mu)\right\} \\
& \cap\ \left\{\Gamma_{OCP}(\mu) \subseteq \Gamma^{\varrho}_{OCP}(\widehat{\mu}_{n_i})\right\} \\
& \subseteq \left(\Xi_i^1 \cap \Xi_i^2\right).
\end{aligned}
$$

Using above and Proposition 30, we conclude that

$$
\mathbb{P}\left\{T_{n_{i-1}}(a) \geqslant \gamma(i-2)\,,\, \forall a \in A\,,\, \left(\Xi_i^1 \cap \Xi_i^2\right)^c\right\} \leqslant 4\,|A| \exp\left\{-\frac{2\left(\Delta^{\mathcal{D}}\right)^2 \gamma(i-2)}{\mathcal{L}^2}\right\}. \qquad \text{(A.20)}
$$

Using the above and (A.19), we have that

$$
R_1^{OCP}(F,N) \leqslant (\ln N)^{1+\varepsilon}\, z^*_{OCP}(\mu) + C_5,
$$

for a finite positive constant C_5, independent of N. Putting the results from Steps 1 and 2.1 (from the proof of Theorem 6), and Step 2.2' together, we conclude that

$$R^{\pi'_{OCP}}(F, N) \leqslant (z^*_{OCP}(\mu) + \gamma\, z^*_{Cover}(\mu))\, (\ln N)^{1+\varepsilon} + C_6,$$

for a finite positive constant C_6, independent of N.

We finally note that the optimal solutions to the $Cover(\widehat{\mu}_{n_i})$ and $OCP(\widehat{\mu}_{n_i})$ problems converge a.s. to optimal solutions to $Cover(\mu)$ and $OCP(\mu)$, respectively. For this, note that Proposition 30, (A.13) and (A.20) imply (via Borel-Cantelli) that $\mathbb{P}\{\mathcal{E}_i \in \Gamma_{Cover}(\mu) \text{ e.v.}\} = 1$ and

$$\mathbb{P}\{(C_i, \mathcal{G}_i) \in \Gamma_{OCP}(\mu) \text{ e.v.}\} = 1.$$

□

A.3 Omitted Proofs and Materials from Section 2.6

A.3.1 General Complexity of OCP

To prove Theorem 11 and Proposition 12, we will use the following lemma.

Lemma 31. *We may restrict the OCP or $Cover$ problems to have at most $|A|$ non-zero $y(S)$ variables without changing the problems.*

Proof of Lemma 31. For the OCP problem, the result follows from noting that any critical set C can be covered by at most $|A|$ solutions (i.e., by a solution-cover $\mathcal{G}$ of at most size $|A|$). Hence, if an optimal solution for OCP has $|\mathcal{G}| > |A|$, we may remove one solution from it while preserving feasibility. If the removed solution is suboptimal for $Comb$, we would obtain a solution with lower objective value contradicting the optimality for OCP. If the removed solution is optimal for $Comb$, we obtain an alternate optimal solution for OCP.

For the $Cover$ problem, the result follows by noting that A can be covered by at most $|A|$ solutions.

□

Theorem 11. *If Comb is in P, then OCP is in NP.*

Proof of Theorem 11. By Lemma 31, optimal solutions to OCP and $Cover$ have sizes that are polynomial in $|A|$ and their objective function can be evaluated in polynomial time. Checking the feasibility of these solutions for OCP can be achieved in polynomial time, because checking (2.7c) can be achieved by solving $Comb(\nu_x)$ where $\nu_x := (\nu_x(a) : a \in A)$ for $\nu_x(a) := l(a)(1 - x(a)) + \nu(a)x(a)$. This problem is polynomially solvable by assumption. $\square$

A.3.2 Critical Sets for Matroids

Lemma 32. *Let $Comb(\nu)$ be a weighted basis or independent set matroid minimization problem. Then there exists a unique critical set that can be found in polynomial time.*

Proof of Lemma 32. To simplify the exposition, we assume that $\mathcal{S}^*(\nu) = \{S^*\}$ is a singleton. Also, for $S \in \mathcal{S}$, we let e^S denote the incidence vector associated with S (i.e., $e^S := (e^S(a) : a \in A)$ with $e^S(a) \in \{0,1\}$, $a \in A$, is such that $e^S(a) = 1$ if $a \in S$ and $e^S(a) = 0$ otherwise).

Let $P := \text{conv}\left(\left\{e^S\right\}_{S \in \mathcal{S}}\right) \subseteq \mathbb{R}^{|A|}$ be the independent set (base) polytope of $\mathcal{S}$. Then, for ν feasible, $S^* \in \mathcal{S}^*(\nu)$ if and only if $\sum_{a \in S^*} \nu(a) \leqslant \sum_{a \in S} \nu(a)$ for any $S \in \mathcal{S}$ such that e^{S^*} and e^S are adjacent vertices in P. Furthermore, each adjacent vertex to e^{S^*} can be obtained from S^* by: removing (denoted by "R"), adding (denoted by "A"), or exchanging (denoted by "E") a single element of S^* (Schrijver, 2003, Theorem 40.6). Thus, we construct the set C so that S^* is always optimal if and only if the cost of all elements of C are at their expected value. The construction procedure starts with $C = S^*$. In some steps we distinguish between $\mathcal{S}$ corresponding to independent sets or bases.

R. (for the independent set case) From the optimality of S^*, removing an element never leads to optimality.

A. (for the independent set case) For each $a \in A \backslash S^*$ such that $S^* \cup \{a\}$ is an independent set, if $l(a) < 0$, then add a to C.

E. (for both cases) For each $a \in A \backslash S^*$, add a to C if

$$l(a) < \max \{\nu(a') : a' \in S^*,\, S^* \cup \{a\} \setminus \{a'\} \text{ is an independent set (base)}\}.$$

By construction, covering all elements in C guarantees optimality of S^*, and not covering some guarantees that S^* is no longer optimal. Note that the set C is unique. For the case of multiple optimal solutions we simply repeat this procedure for each one. Finally, the only computationally non-trivial step in the construction of C is checking that this set is an independent set or a base, which can be done in polynomial time. □

A.3.3 Basic MIP Formulation for OCP

In this section we prove Proposition 12.

Proof of Proposition 12. For any feasible solution (x, y) to (2.8), we have that x is the incidence vector of a critical set. This, because (2.8d) enforces dual feasibility of w when elements with $x(a) = 0$ are not covered, and (2.8e) forces the objective value of the dual of $Comb(\nu')$ to be greater than or equal to $z^*_{Comb}(\nu)$, where $\nu' = \operatorname{diag}(l)(\mathbf{1}-x)+\operatorname{diag}(\nu)x$. With this, the optimal objective value of $Comb(\nu')$ is greater than or equal to $z^*_{Comb}(\nu)$. On the other hand, any y^i is the incidence vector of some $S \in \mathcal{S}$ because of (2.8c) and the assumptions on M and d. Finally, (2.8b) ensures that the critical set is covered by the solution-cover (i.e., y^i's) induced by OCP. Lemma 31 ensures that the $|A|$ variables y^i are sufficient for an optimal solution to OCP. If less than $|A|$ solutions are needed for the cover, then the optimization can pick the additional y^i variables to be the incidence vector of an optimal solution to

$Comb(\nu)$ so that they do not increase the objective function value. The extension for $Cover$ is straightforward. $\square$

A.3.4 IP Formulation for OCP when Comb(ν) Admits a Compact IP Formulation

Suppose $Comb(\nu)$ admits a compact IP formulation such that $\{y^S\}_{S\in\mathcal{S}} = \{y \in \{0,1\}^{|A|} : My \leqslant d\}$ for some $M \in \mathbb{R}^{m\times|A|}$ and $d \in \mathbb{R}^m$, where y^S denotes the incidence vector of $S \in \mathcal{S}$. For simplicity, we assume that $A = \{1, \ldots, |A|\}$. Then an IP formulation of $OCP(\nu)$ is given by

$$\min \quad \sum_{i\in A}\left(\sum_{a\in A}\nu(a)y^i(a) - z^*_{Comb}(\nu)\right) \tag{A.21a}$$

$$s.t. \qquad x(a) \leqslant \sum_{i\in A} y^i(a), \quad a \in A \tag{A.21b}$$

$$My^i \leqslant d, \qquad i \in A \tag{A.21c}$$

$$\sum_{a\in S}(l(a)(1-x(a)) + \nu(a)x(a)) \geqslant z^*_{Comb}(\nu), \quad S \in \mathcal{S} \tag{A.21d}$$

$$x(a),\, y^i(a) \in \{0,1\}, \qquad a, i \in A. \tag{A.21e}$$

As in formulation (2.8), a feasible solution (x, y) to (A.21) is such that x is the incidence vector of a critical set (this is enforced by (A.21d)), and the y^i's are a cover of such set, due to (A.21b), (A.21c), and the assumptions on M and d. Note that an efficient cover includes at most $|A|$ solutions (the optimization can pick the additional y^i to be the incidence vector of an optimal solution).

Formulation (A.21) has a polynomial number of variables, but the number of constraints described by (A.21d) is in general exponential. However, the computational burden of separating these constraints is the same as solving $Comb(\nu)$ (finding a violated inequality (A.21d) or showing that it satisfies all these inequalities can be done by solving $Comb(\nu')$ for $\nu'(a) = l(a)(1-x(a)) + \nu(a)x(a))$. Hence, if we can solve $Comb(\nu)$ sufficiently fast (e.g., when the problem is in P, or it is a practically solvable NP-hard problem) we should be able to effectively solve (A.21) with a branch-and-cut algorithm that dynamically adds constraints (A.21d) as needed. Finally, note that a formulation for *Cover* is obtained by setting $x(a) = 1$ for all $a \in A$ and removing (A.21d).

A.3.5 Linear-sized Formulation of OCP for Shortest Path Problem

Let $Comb(\nu)$ correspond to a shortest $s - t$ path problem in a digraph $G = (V, A)$. Define $\hat{A} = A \cup \{(t, s)\}$ and let $\hat{\delta}_{out}$ and $\hat{\delta}_{in}$ denote the outbound and inbound arcs in digraph $\hat{G} = (V, \hat{A})$. An optimal solution (x, p, w) to

$$\min \quad \left(\sum_{a \in A} \nu(a)\, p(a)\right) - z^*_{Comb}(\nu)\, p\left((t, s)\right) \tag{A.22a}$$

$$s.t. \quad x(a) \leqslant p(a), \quad a \in A \tag{A.22b}$$

$$\sum_{a \in \hat{\delta}_{out}(v)} p(a) - \sum_{a \in \hat{\delta}_{in}(v)} p(a) = 0, \quad v \in V \tag{A.22c}$$

$$w(u) - w(v) \leqslant l\left((u, v)\right)\left(1 - x\left((u, v)\right)\right) + \nu\left((u, v)\right) x\left((u, v)\right), \quad (u, v) \in A \tag{A.22d}$$

$$w(s) - w(t) \geqslant z^*_{Comb}(\nu) \tag{A.22e}$$

$$p(a) \in \mathbb{Z}_+, \quad a \in \hat{A} \tag{A.22f}$$

$$x(a) \in \{0, 1\}, \; w(v) \in \mathbb{R}, \quad a \in A, \; v \in V, \tag{A.22g}$$

is such that $(C, \mathcal{G})$ is an optimal solution to $OCP(\nu)$, where $C = \{a \in A : x(a) = 1\}$ and $\mathcal{G} \subseteq \mathcal{S}$ is a set of paths for which $p(a) = |\{S \in \mathcal{G} : a \in S\}|$. Such a set $\mathcal{G}$ can be constructed from p in time $O(|A||V|)$.

The first difference between formulations (A.22) and (2.8) is the specialization of the LP duality constraints to the shortest path setting. The second one is the fact that the paths in cover $\mathcal{G}$ are aggregated into an integer circulation in augmented graph $\hat{G}$, which is encoded in variables p. Indeed, using known properties of circulations (Schrijver, 2003, pp. 170-171), we have that $p = \sum_{S \in \mathcal{G}} y^{\hat{S}}$, where $y^{\hat{S}}$ is the incidence vector of the circulation obtained by adding (t, s) to path S. Furthermore, given a feasible p we can recover the paths in $\mathcal{G}$ in time $O(|A||V|)$. To obtain a formulation for $Cover$, we simply set $x(a) = 1$ for all $a \in A$ and remove (A.22d)–(A.22e).

It is possible to construct similar formulations for other problems with the well-known integer decomposition property (Schrijver, 2003).

A.3.6 A Time-Constrained Asynchronous Policy

Depending on the application, real-time implementation might require choosing a solution $S_n \in \mathcal{S}$ prior to the *exogenous* arrival of the cost vector B_n. However, the solution times for the problems $OCP(\cdot)$ or even $Comb(\cdot)$ could be longer than the time available to the executing policy. For example, most index-based policies must solve an instance of $Comb(\cdot)$ at each period, which might not be possible in practice. Fortunately, a key feature of our proposed OCP-based policies is that the frequency at which the problems $Comb(\cdot)$ and $OCP(\cdot)$ need to be solved decreases exponentially. Indeed, such problems are solved at the beginning of each cycle and the length of cycle i is $\Theta(\exp(i/H))$ for a fixed tuning parameter $H > 0$. Hence, as cycles elapse, there will be eventually enough time to solve these problems.

Nonetheless, the policy cannot proceed until the problems $Comb(\cdot)$ and $OCP(\cdot)$ are solved. However, one can easily modify the policy so that it begins solving $Comb(\cdot)$ and $OCP(\cdot)$ at the beginning of a cycle, but continues to implement incumbent solutions while these problems are being solved (such solutions might be computed either upfront or in previous cycles). Solutions to these problems update incumbent solutions as they become available, which for long cycles would be at the beginning of the next one. Algorithm 4 presents one such possible modification for the OCP-based policy.

Algorithm 4 Basic Time-Constrained Asynchronous OCP-based policy $\pi^{A}_{OCP}(H)$

```
Set i = 0, C = A, and 𝒢 a minimal cover of A
Let S* ∈ 𝒮 be an arbitrary solution and μ̂_Comb = μ̂_OCP be an initial cost estimate
Asynchronously begin solving Comb(μ̂_Comb) and OCP(μ̂_OCP)
for n = 1 to N do
  if n = n_i then
    Set i = i + 1
    if Asynchronous solution to Comb(μ̂_Comb) has finished then
      Set S* ∈ 𝒮*(μ̂_Comb)                                  [Update exploitation set]
      Set μ̂_Comb = μ̂_n
      Asynchronously begin solving Comb(μ̂_Comb)
    end if
    if Asynchronous solution to OCP(μ̂_OCP) has finished then
      Set (C, 𝒢) ∈ Γ_OCP(μ̂_OCP)                         [Update OCP-exploration set]
      Set μ̂_OCP = μ̂_n
      Asynchronously begin solving OCP(μ̂_OCP)
    end if
  end if
  if T_n(a) < i for some a ∈ C then
    Set S_n = S for any S ∈ 𝒢 such that a ∈ S               [OCP-based exploration]
  else
    Set S_n = S*                                                   [Exploitation]
  end if
end for
```

A.3.7 Greedy Oracle Polynomial-Time Heuristic

To further illustrate the potential practicality of policies based on OCP, we develop a greedy heuristic for solving OCP that only requires a polynomial number of queries to an oracle for $Comb(\cdot)$ (plus a polynomial number of additional operations). This heuristic always returns a solution that is equal and possibly arbitrarily better than a minimal cover of A.

We begin by describing the heuristic for solving $OCP(\nu)$ in Algorithm 5. Given a cost vector ν, the heuristic first sets all costs to their lowest possible values, and successively solves instances of $Comb$, each time incorporating the incumbent solution into the solution-cover $\mathcal{G}$, adding its ground elements to the (critical) set C, and updating the cost vector accordingly. The procedure stops when the feedback from C *suffices* to guarantee the optimality of the best solution (i.e., when $z^*_{Comb}(\nu') \geqslant z^*_{Comb}(\nu)$). To achieve *efficiency* of such a feedback, the heuristic then prunes elements in C that are not required to guarantee sufficiency of the feedback.

Algorithm 5 Oracle Polynomial-Time Heuristic

> Set $\nu' := (\nu'(a) : a \in A) = (l(a) : a \in A)$, $\mathcal{G} = \varnothing$, $C = \varnothing$.
> **while** $z^*_{Comb}(\nu') < z^*_{Comb}(\nu)$ **do**
>   Select $S \in \mathcal{S}^*(\nu')$ and set $\nu'(a) = \nu(a)$ for all $a \in S$
> $\mathcal{G} \leftarrow \mathcal{G} \cup \{S\}$ and $C \leftarrow C \cup S$
> **end while**
> **for** $a \in C$ **do**
>   **if** $z^*_{Comb}((\nu' \wedge l)(\{a\})) \geqslant z^*_{Comb}(\nu)$ **then**
> $C \leftarrow C \backslash \{a\}$ and $\nu'(a) \leftarrow l(a)$
>   **end if**
> **end for**

Note that in each iteration of the first loop, Algorithm 5 calls an oracle for *Comb* and adds at least one ground element to C. Similarly, in the second loop, the heuristic calls such an oracle once for every element in C. Hence, the procedure calls such an oracle at most $2|A|$ times. Thus, the heuristic makes a linear number of calls to the oracle for *Comb*. In particular, if *Comb* is in P, then the heuristic runs in *polynomial time*.

The performance of the heuristic ultimately depends on the specifics of a setting. For instance, in the setting of Example 1, the heuristic returns, in the worst case, a solution with $|\mathcal{G}| = k$, which is of the order of a cover of A. In the setting of Example 2 on the other hand, the heuristic returns a solution with $|\mathcal{G}| = 2$ (in such a setting a cover of A is of order k). It is not hard to identify settings where the heuristic performs arbitrarily better than any cover of A.

We finally note that the heuristic in Algorithm 5 can be modified as follows for solving the *Cover* problem: the first loop should be implemented while $A \nsubseteq C$ and the second loop is no longer needed. The resulting set $\mathcal{G}$ provides a cover of A.

A.4 Additional Computational Results

In this section we provide the computational results for Examples 1, 2 and 28. Figure A.2 depicts the average performance of different policies on Examples 1 (left), 2 (center) and 28 (right), respectively.

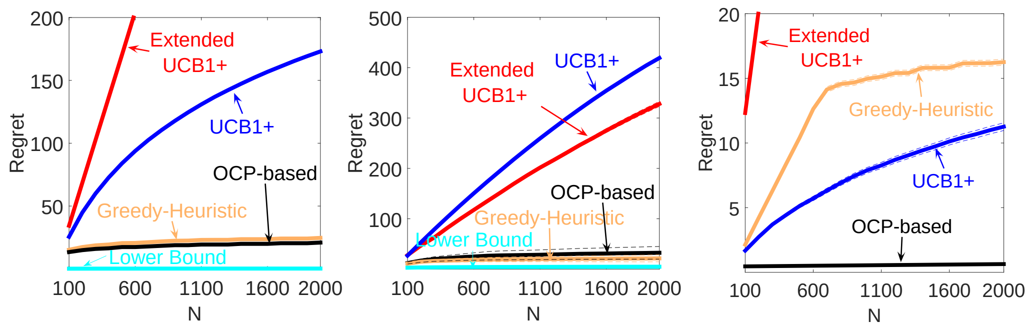

FIGURE A.2: Average performance of different policies on Examples 1 (left), 2 (center) and 28 (right).

On Example 1, the OCP-based and Greedy-Heuristic policies perform significantly better than the benchmark policies. The situation is essentially the same on Example 2, only that this time Extended UCB1+ outperforms the UCB1+ policy. There, the solution to $OCP(\mu)$ is only of size 2, which helps our policies achieve the best performance. (Note that for this setting, the Greedy-Heuristic tends to find the actual optimal solution to $OCP(\mu)$ even with unreliable estimates.) On Example 28, the heuristic solution to OCP coincides with the minimum-regret cover of $\mathcal{S}$, thus the Greedy-Heuristic policy is outperformed by UCB1+ (note that this latter policy rarely uses the arcs p_2 and q_2, since the costs of p_1 and q_1 are close to 0).

As we discussed before, the lower bound in Theorem 5 is asymptotic, so it is not clear whether the lower bound is meaningful in the finite time. However, we plot the lower bound for the three shortest path examples in Figure A.2. As can be noted from the graph, in Examples 1 and 2, the lower bound is in fact meaningful and the regret of the OCP-based and Greedy-Heuristic policies is much closer to the lower bound than the other benchmark policies. In Example 28, however, the lower bound is not meaningful, that is, the lower bound is larger than the regret of all policies as it only provides an asymptotic lower bound on regret.

In terms of efficient information collection, one can divide the set of ground elements (arcs) into three classes: those that are part of the optimal solution (called the "Optimal arcs"), those that are covered by at least one optimal solution to $OCP(\mu)$ (called the "Exploration arcs"), and the rest (called the "*Uninformative* arcs"). Table A.1 shows the average number of times that each type of arc (shown in columns called "Opt.", "Exp.", and "Uninf.", respectively) is tested up to period $N = 2000$ by each policy. Note that the OCP-based and Greedy-Heuristic policies spend significantly less time exploring uninformative arcs. Table A.1 also shows the average length of implemented solutions (i.e., the average number of arcs in the implemented solutions) for different policies (the column called "Length").

Table A.1: Average number of arc trials up to period $N = 2000$, and average solution size for different policies on Examples 1, 2 and 28.

	Example 1				Example 2				Example 28			
	Opt.	Exp.	Uninf.	Length	Opt.	Exp.	Unin.	Length	Opt.	Exp.	Unin.	Length
OCP-based	1958.93	470.67	2.25	3.06	1858.25	548.12	4.55	1.19	140.03	214.50	1.00	4.72
Greedy-Heuristic	1951.62	472.18	3.38	3.07	1918.43	524.20	3.32	1.11	106.83	215.94	35.71	4.79
UCB1+	1660.75	533.35	42.12	3.51	474.31	929.80	66.61	3.19	92.45	217.75	24.61	4.82
Ext. UCB1+	791.31	684.36	364.72	4.81	870.88	795.78	53.76	2.67	14.87	219.02	151.79	4.97

Figure A.3 depicts box plots of the 100 different cumulative regrets at the final time period $N = 2000$ (i.e., sample path final regrets) for OCP-based, UCB1+ and Extended UCB1+ policies in Examples 1, 2 and 28.. We observe that the OCP-based

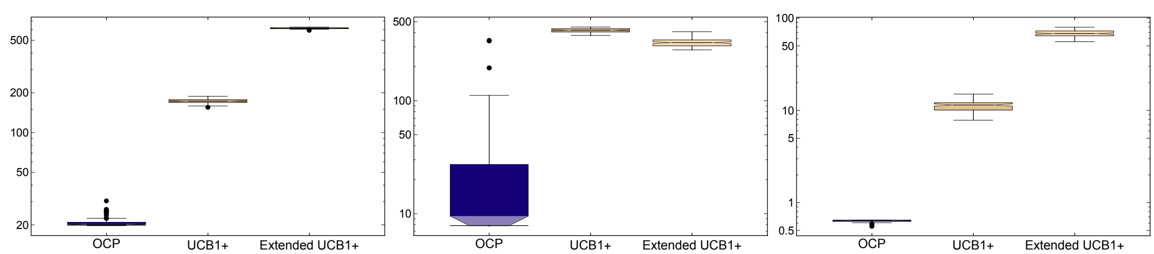

FIGURE A.3: Box plots of performance for OCP-based and benchmark policies on Examples 1 (left), 2 (center) and 28 (right).

policy significantly outperforms UCB1+ and Extended UCB1+ not only on average, but also for (almost) all replications.

A.5 Alternative Feedback Setting

The flexibility of the OCP-based policies allows them to be easily extended or combined with other techniques that consider similar what-and-how-to-explore questions. For instance, the OCP-based policy can be easily combined with the "barycentric spanner" of Awerbuch and Kleinberg (2004) to extend our results from element-level observations to set- or solution-level observations as follows. For a particular application, it might be the case that the decision-maker only has access, for example, to the *total* cost incurred by implementing solution S_n. We begin by showing how a cover-based policy (i.e., a policy that conducts exploration by implementing solutions in a cover) can be adapted to this last setting. For a set of ground elements $S \subseteq A$, let $I_S := (I_S(a) : a \in A) \in \{0,1\}^{|A|}$ denote the incidence vector of the ground set (so that $S = \{a : I_S(a) = 1, a \in A\}$). We say a solution set $\mathcal{E}$ *recovers* a set $E \subseteq A$ if for each $a \in E$, there exists a vector $\gamma(a) := (\gamma_S(a), S \in \mathcal{E}) \in \mathbb{R}^{|\mathcal{E}|}$ such that

$$\sum_{S \in \mathcal{E}} \gamma_S(a) I_S = I_{\{a\}}. \tag{A.23}$$

Without loss of generality, one can assume that each ground element is recovered by at least one solution set. Let $\mathcal{E}$ be a solution set that recovers A, and let $\gamma := (\gamma(a), a \in A)$ be such that $\sum_{S \in \mathcal{E}} \gamma_S(a) I_S = I_{\{a\}}$, for all $a \in A$. One can implement a cover-based policy with $\mathcal{E}$ playing the role of a cover while using the estimate mean cost vector $\widehat{\mu}_n = (\widehat{\mu}_n(a) : a \in A)$, where

$$\widehat{\mu}_n(a) := \sum_{S \in \mathcal{E}} \frac{\gamma_S(a)}{|m < n : S_m = S|} \sum_{m<n:S_m=S} \sum_{a \in S} b_m(a), \quad a \in A. \tag{A.24}$$

The estimate above reconstructs the expected cost of each solution in $\mathcal{E}$ and uses (A.23) to translate such estimates to the ground-element level. Implementing this modification requires precomputing a solution set $\mathcal{E}$ recovering A. Such a set can be selected so that $|\mathcal{E}| \leqslant |A|$, and computed by solving $O(|A|)$ instances of $Comb(\cdot)$ (see e.g., the algorithm in Awerbuch and Kleinberg (2004)).

The idea above can also be used to extend the OCP-based policy to this new setting. For that we could consider the estimates in (A.24) and $(C, \mathcal{E})$ to be a solution to an alternative version of $OCP(\nu)$, denoted by $OCP'(\nu)$, where in addition to (2.7b)-(2.7d), one imposes that $\mathcal{E}$ recovers C, that is, $OCP'(\nu)$ is given by

$$\min \quad \sum_{S \in \mathcal{S}} \Delta_S^{\nu} \, y(S) \tag{A.25a}$$

$$s.t. \quad \sum_{S \in \mathcal{S}} \gamma_S(a) I_S = x(a) I_{\{a\}}, \quad a \in A \tag{A.25b}$$

$$\gamma_S(a) \leqslant Q \, y(S), \quad S \in \mathcal{S}, \, a \in A \tag{A.25c}$$

$$-\gamma_S(a) \leqslant Q \, y(S), \quad S \in \mathcal{S}, \, a \in A \tag{A.25d}$$

$$\sum_{a \in S} \left(l(a)(1 - x(a)) + b(a) x(a) \right) \geqslant z^*_{Comb}(\nu), \quad S \in \mathcal{S} \tag{A.25e}$$

$$x(a), \, y(S) \in \{0, 1\}, \, \gamma_S(a) \in \mathbb{R}, \quad a \in A, S \in \mathcal{S}, \tag{A.25f}$$

where Q is an instance-dependent constant, whose size is polynomial in the size of the instance. The additional constraints(A.25b)-(A.25d) in $OCP'(\nu)$ ensure that the solution-cover $\mathcal{E}$ recovers the critical set C. Like OCP, the formulation above can be specialized to accommodate the combinatorial structure of $Comb$. The performance guarantee in Theorem 10 would remain valid with the constants associated with OCP'. We anticipate that the challenge of solving OCP' effectively is comparable to that of solving OCP.

A.6 Auxiliary Result for the Proof of Theorem 6 and Theorem 10

In this section we prove Proposition 30 which we used in the proof of Theorem 6 and Theorem 10.

Proof of Proposition 30. For $m \in \mathbb{N}$, define $t_m(a) := \inf\{n \in \mathbb{N} : T_n(a) = m\} - 1$. Indexed by m, one has that $B_{t_{m(a)}}(a) - \mu(a)$ is a bounded martingale difference sequence, thus one has that

$$\begin{aligned}
\mathbb{P}\{|\hat{\mu}_n(a) - \mu(a)| \geqslant \epsilon\,,\, T_n(a) \geqslant k\} &= \mathbb{P}\left\{\left|\sum_{m=1}^{T_n(a)} \left(B_{t_{m(a)}}(a) - \mu(a)\right)\right| \geqslant \epsilon\, T_n(a)\,,\, T_n(a) \geqslant k\right\} \\
&\leqslant \sum_{h=k}^{\infty} \mathbb{P}\left\{\left|\sum_{m=1}^{h} \left(B_{t_{m(a)}}(a) - \mu(a)\right)\right| \geqslant \epsilon\, h\,,\, T_n(a) = h\right\} \\
&\overset{(a)}{\leqslant} 2 \sum_{h=k}^{\infty} \exp\left\{\frac{-2\, h\epsilon^2}{\mathcal{L}^2}\right\} \mathbb{P}\{T_n(a) = h\} \\
&\leqslant 2\exp\left\{\frac{-2\, k\epsilon^2}{\mathcal{L}^2}\right\},
\end{aligned}$$

where (a) follows from the Hoeffding-Azuma Inequality (see, for example, Lemma A.7 in Cesa-Bianchi and Lugosi (2006)). □

Appendix B

Appendices for Chapter 3

B.1 Proofs

Detailed Derivation of the approximate lower bound $G_{1l}(T, \mathcal{I}')$. We define product j's optimality gap for customers with profile i as $\Delta_j^i := \max_{\{1 \leqslant l \leqslant N\}}\{\mu_l^i\} - \mu_j^i$ for $j \in \mathcal{N}$ and $i \in \mathcal{I}$. Let $k_j^i(t)$ denote the number of times that product j has been offered to a customer with profile i up to (and including) time t. One can rewrite the retailer's regret associated with a policy π, defined in (3.3), as $R^\pi(T, I) = \sum_{i=1}^{I} R_i^\pi(T)$ where $R_i^\pi(T)$ is the regret associated with profile i and policy π:

$$R_i^\pi(T) := \sum_{j=1}^{N} \Delta_j^i \, \mathbb{E}_\pi \left[k_j^i(T)\right].$$

The regret is proportional to the number of times that suboptimal assortments are offered to customers over the selling season.

In their seminal work, Lai and Robbins (1985a) proved that the regret of any consistent policy in a classic bandit setting grows asymptotically with order at least

$\ln(T)$. Formally, for any consistent policy π, we have that

$$\liminf_{T\to\infty} \frac{R^{\pi}(T)}{\ln(T)} \geqslant \sum_{j\in\mathcal{N}:\Delta_j>0} \frac{\Delta_j}{D_j}, \tag{B.1}$$

where Δ_j is the optimality gap and D_j is the Kullback-Leibler divergence associated with arm $j \in \mathcal{N}$.[1]

Because all profiles within a cluster have the same distribution of preferences, with some abuse of notation, we let Δ_j^k denote the optimality gap of product j for all profiles in cluster k; that is, $\Delta_j^i = \Delta_j^k$ for all $i \in \mathcal{I}_k$ where recall that $\mathcal{I}_k$ denotes the set of profiles belonging to cluster k. Similarly, we let $D_j^i = D_j^k$ for all $i \in \mathcal{I}_k$. Building on Lai and Robbins' result in (B.1), we consider the following (asymptotic) lower bound on the regret for the data-intensive policy:

$$L_{d\text{-}int}(T,\mathcal{I}') := \sum_{k=1}^{K} \left(\sum_{j=2}^{N} \frac{\Delta_j^k}{D_j^k} \right) \sum_{i\in\mathcal{I}_k} \mathbb{E}\left[\ln(T_i)\right],$$

where T_i is the (random) number of customers from profile i that arrive during the selling season. Hence, $\sum_{i=1}^{I} T_i = T$ almost surely.[2] Note that $L_{d\text{-}int}$ is the sum of the lower bound on the regret associated with each profile (or cluster). Unlike the setting studied in Lai and Robbins (1985*a*), which considers a homogeneous population of customers, the number of customer arrivals from each profile (i.e, T_i) is random. This introduces additional complexity in the computation of the lower bound, which we address by using a first-order Taylor expansion of the term $\ln(T_i)$.[3] We therefore

[1] The Kullback-Leibler divergence measures the difference between two probability distributions. For the Bernoulli distribution, we have $D_j := \mu_j \ln(\mu_j/\mu^*) + (1-\mu_j)\ln((1-\mu_j)/(1-\mu^*))$, where $\mu^* := \max_{j\in\mathcal{N}}\{\mu_j\}$.

[2] Although the lower bound in (B.1) is attained when $T \to \infty$, one can obtain a similar lower bound on the regret associated with each profile i, $R_i^{\pi}(T)$, as $T_i \to \infty$. This is because $T_i \to \infty$ almost surely as $T \to \infty$, as long as $p_i > 0$ for each profile $i \in \mathcal{I}$.

[3] In deriving the bounds, we use a first-order Taylor expansion which, as one may expect, provides a reasonable approximation for smooth functions – we also verified this numerically.

approximate $L_{d\text{-}int}$ by replacing $\ln(T_i)$ by its first-order Taylor expansion around the mean $\mathbb{E}(T_i) = p_i T$. To this end, we have that $\ln(T_i) \approx \ln(p_i T) + (1/(p_i T))(T_i - p_i T)$ and taking expectation on both sides, we have that $\mathbb{E}\left[\ln(T_i)\right] \approx \ln(p_i T)$ as $\mathbb{E}(T_i) = p_i T$. Thus, we approximate the lower bound function $L_{d\text{-}int}$ with $L^{apx}_{d\text{-}int}$ where

$$L_{d\text{-}int}(T, \mathcal{I}') \approx L^{apx}_{d\text{-}int}(T, \mathcal{I}') := \sum_{k=1}^{K} \left(\sum_{j=2}^{N} \frac{\Delta_j^k}{D_j^k} \right) \sum_{i \in \mathcal{I}_k} \ln(p_i T).$$

Using the results in Auer et al. (2002) for the UCB1 policy, an upper bound on the regret for the semi-oracle policy is given by:

$$U_{s\text{-}orc}(T, \mathcal{I}') := \sum_{k=1}^{K} \left(\sum_{j=2}^{N} \left(\frac{8}{\Delta_j^k} \right) \mathbb{E}\left[\ln(T'_k)\right] + \left(1 + \frac{\pi^2}{3}\right) \sum_{j=2}^{N} \Delta_j^k \right),$$

where T'_k denotes the (random) number of customers from cluster k. Using a similar first-order Taylor expansion as in $L^{apx}_{d\text{-}int}$, we approximate the upper bound function $U_{s\text{-}orc}$ with $U^{apx}_{s\text{-}orc}$ where

$$U_{s\text{-}orc}(T, \mathcal{I}') \approx U^{apx}_{s\text{-}orc}(T, \mathcal{I}') := \sum_{k=1}^{K} \left(\sum_{j=2}^{N} \left(\frac{8}{\Delta_j^k} \right) \ln(P_k T) + \left(1 + \frac{\pi^2}{3}\right) \sum_{j=2}^{N} \Delta_j^k \right),$$

where $P_k := \sum_{i \in I_k} p_i$.

We therefore approximate the lower bound function $L_{d\text{-}int}(T, \mathcal{I}') - U_{s\text{-}orc}(T, \mathcal{I}')$ by

$$G_{1l}(T, \mathcal{I}') := L^{apx}_{d\text{-}int}(T, \mathcal{I}') - U^{apx}_{s\text{-}orc}(T, \mathcal{I}').$$

Since the gap function can be defined as the gap associated with each cluster, we further define $G_{1l}(T, \mathcal{I}') = \sum_{k=1}^{K} G^k_{1l}(T, \mathcal{I}_k)$ where

$$G^k_{1l}(T, \mathcal{I}_k) := \left[\left(\sum_{j=2}^{N} \frac{\Delta_j^k}{D_j^k} \right) \sum_{i \in \mathcal{I}_k} \ln(p_i T) \right] - \left[\sum_{j=2}^{N} \left(\frac{8}{\Delta_j^k} \right) \ln(P_k T) + \left(1 + \frac{\pi^2}{3}\right) \sum_{j=2}^{N} \Delta_j^k \right].$$

$\square$

The next result provides conditions under which $G_{1l}(T, \mathcal{I}') \geqslant 0$.

Theorem 33. *For each $k \in \{1, \ldots, K\}$, suppose that $I_k \geqslant I_l^k$ for some I_l^k independent of T and of the distribution of customer arrivals. Then, (i) there exist thresholds $t_l^k(I_k)$ such that $G_{1l}(T, \mathcal{I}') \geqslant 0$ for $T \geqslant \max_k \left\{t_l^k(I_k)\right\}$; and (ii) $G_{1l}(T, \mathcal{I}')$ is increasing and concave in T.*

Proof. (i) To prove that $G_{1l}(T, \mathcal{I}') \geqslant 0$, we first characterize conditions under which $G_{1l}^k(T, \mathcal{I}_k) \geqslant 0$. After some algebra, we obtain that

$$G_{1l}^k(T, \mathcal{I}_k) = A \ln(T) - B,$$

where $A := I_k \left(\sum_{j=2}^N \Delta_j^k / D_j^k\right) - \sum_{j=2}^N 8/\Delta_j^k$,

$$B := -\left(\sum_{j=2}^N \Delta_j^k / D_j^k\right) \ln\left(\prod_{i \in \mathcal{I}_k} p_i\right) + \left(\sum_{j=2}^N 8/\Delta_j^k\right) \ln(P_k) + \left(1 + \pi^2/3\right) \sum_{j=2}^N \Delta_j^k,$$

$P_k = \sum_{i \in \mathcal{I}_k} p_i$, and D_j^k is the Kullback-Leibler divergence number. Let I_l^k be the smallest integer greater than $\left(\sum_{j=2}^N 8/\Delta_j^k\right) / \left(\sum_{j=2}^N \Delta_j^k / D_j^k\right)$ and set $I_k \geqslant I_l^k$. Note that $A > 0$ for $I_k \geqslant I_l^k$. We therefore have that $G_{1l}^k(T, \mathcal{I}_k) \geqslant 0$ if

$$T \geqslant t_l^k(I_k) := \lceil \exp(B/A) \rceil.$$

The result follows from noting that $G_{1l}(T, \mathcal{I}') = \sum_{k=1}^K G_{1l}^k(T, \mathcal{I}_k)$.
(ii) This part follows immediately as $G_{1l}(T, \mathcal{I}') = \sum_{k=1}^K G_{1l}^k(T, \mathcal{I}_k)$ where $G_{1l}^k(T, \mathcal{I}_k) = A \ln(T) - B$ (where A and B are as defined in part (i)), and $A > 0$ for $I_k \geqslant I_l^k$. □

As expected, the semi-oracle outperforms the data-intensive policy as it learns about customers' preferences faster by pooling information across customer profiles within any cluster. The requirement of sufficiently large I_k in Theorem 33 is mostly technical and emerges from the mismatch between the leading constants in the lower and upper bounds to the regret functions. On the other hand, the requirement on T ensures that the number of customers in each profile is not too small with high probability. Theorem 33(ii) shows that $G_{1l}(T, \mathcal{I}')$ is increasing (and, in fact, grows to infinity) in the number of arriving customers T.

Proof of Theorem 13. We assume for simplicity that I_k is continuous. The results follow directly for the discrete case. We prove the results by taking the first- and second-order partial derivatives. We first prove that $\partial G_{1l}(T, \mathcal{I}')/\partial I_k > 0$. We have that $\ln\left(\prod_{i \in \mathcal{I}_k} p_i\right) = I_k \ln(P_k) - I_k \ln(I_k)$. Then,

$$\frac{\partial G_{1l}(T, \mathcal{I}')}{\partial I_k} = \left(\sum_{j=2}^{N} \frac{\Delta_j^k}{D_j^k}\right) (\ln(T) + \ln(P_k) - \ln(I_k) - 1)\,.$$

Note that $\partial G_{1l}(T, \mathcal{I}')/\partial I_k > 0$ if $T > eI_k/P_k$. To prove concavity, note that

$$\frac{\partial^2 G_{1l}(T, \mathcal{I}')}{\partial {I_k}^2} = -\left(\sum_{j=2}^{N} \frac{\Delta_j^k}{D_j^k}\right) \left(\frac{1}{I_k}\right) < 0.$$

$\square$

Detailed Derivation of the approximate lower bound $G_{2l}(T, I, P)$. We assume that $\Delta_j^k = \Delta > 0$ for all $k = 1, \ldots, K$ and $j \neq k$, where recall that Δ_j^k denotes the optimality gap of product j for all profiles in cluster k. Similarly, we let $D_j^k = D > 0$ for all $k = 1, \ldots, K$ and $j \neq k$, where D_j^k denotes the Kullback-Leibler divergence associated with product j for all profiles in cluster k. As in derivation of the approximate lower bound $G_{1l}(T, \mathcal{I}')$, we consider the (asymptotic) lower bound on the regret of the data-intensive policy:

$$L_{\textit{d-int}}(T, I, P) := \frac{(K-1)\Delta}{D} \sum_{i=1}^{I} \mathbb{E}\left[\ln(T_i)\right].$$

Existing upper bounds in the bandit literature (including that of the UCB1 policy) assume a homogeneous market. This introduces additional complexity in the computation of the upper bound on the regret of the pooling policy, which we address in the following result.

Lemma 34. *The regret for the pooling policy in this setting is at most*

$$U_{pool}(T, I, P) = \left[\frac{8\ln(T)}{\Delta} \sum_{k=2}^{K} \left(\frac{1}{P_1 - P_k}\right) + \left(1 + \frac{\pi^2}{3}\right)(KP_1 - 1)\Delta\right] + T(1 - P_1)\Delta.$$

Note that U_{pool} in Lemma 34 has an additional term compared to its counterpart in the case of a homogeneous market. The term inside square brackets is the upper bound on the regret of the pooling policy relative to a "weak" oracle. The "weak" oracle assumes that the market is homogeneous and therefore always offers product 1 (as it has a higher purchase probability averaged over all customer profiles) to any arriving customer. However, the regret has to also account for the pooling policy's inability to differentiate between clusters. The term outside square brackets incorporates such a penalty. Note that this penalty term disappears as $P_1 \uparrow 1$.

Let $G_{2l}(T, I, P)$ be the approximation of the lower bound $L_{d\text{-}int} - U_{pool}$, obtained by replacing $\ln(T_i)$ with its first-order Taylor expansion. That is,

$$G_{2l}(T,I,P) := \left[\frac{(K-1)\Delta}{D}\sum_{i=1}^{I}\ln(p_i T)\right] - \left[\frac{8\ln(T)}{\Delta}\sum_{k=2}^{K}\left(\frac{1}{P_1-P_k}\right) + T(1-P_1)\Delta + \left(1+\frac{\pi^2}{3}\right)(KP_1-1)\Delta\right].$$

$\square$

Proof of Lemma 34. We construct U_{pool} in two steps. First, consider a "weak" oracle that assumes that the market is homogeneous, and therefore, presumes that the purchase probabilities are $P_1\mu_j^1 + P_2\mu_j^2 + \cdots + P_K\mu_j^K$ for each product j, where μ_j^k denotes the purchase probability of product j for cluster k. Because the optimality gap of products in all clusters is the same (i.e., $\Delta_j^k = \Delta$ for any cluster k and product $j \neq k$) and $P_1 > \max\{P_2, \ldots, P_K\}$, the weak oracle offers product 1 to all arriving customers regardless of their profiles (or clusters). As a result, the optimality gap of product $k = 2, \ldots, K$ (averaged over all profiles) for the weak oracle is:

$$\Delta'_k := (P_1\mu_1^1 + P_2\mu_1^2 + \cdots + P_K\mu_1^K) - (P_1\mu_k^1 + P_2\mu_k^2 + \cdots + P_K\mu_k^K) = (P_1 - P_k)\Delta > 0.$$

If the pooling policy follows the UCB1 algorithm of Auer et al. (2002) (which assumes a homogeneous market), its regret relative to the weak oracle is at most

$$\sum_{k=2}^{K}\left[\frac{8\ln(T)}{\Delta'_k} + \left(1+\frac{\pi^2}{3}\right)\Delta'_k\right] = \frac{8\ln(T)}{\Delta}\sum_{k=2}^{K}\left(\frac{1}{P_1-P_k}\right) + \left(1+\frac{\pi^2}{3}\right)(KP_1-1)\Delta.$$

The pooling policy's regret, however, has to also account for its inability to differentiate between clusters. Now, consider a "strong" oracle that knows the purchase probabilities of all customer profiles (and thus the mapping of profiles to clusters). The strong oracle offers product k to customers from cluster k for $k = 1, \ldots, K$, while the weak oracle offers product 1 to all arriving customers regardless of their profiles (or clusters). As a result, the weak oracle incurs an additional cost (regret) relative to the strong oracle whenever a customer from a cluster other than cluster 1 arrives. Hence, the weak oracle's regret relative to the strong oracle due to the lack of knowledge of the mapping of profiles to clusters is $T(1-P_1)\Delta$. Therefore, we obtain

$$U_{pool}(T,I,P) = \left[\frac{8\ln(T)}{\Delta}\sum_{k=2}^{K}\left(\frac{1}{P_1-P_k}\right)+\left(1+\frac{\pi^2}{3}\right)(KP_1-1)\Delta\right]+T(1-P_1)\Delta.$$

$\square$

Proof of Theorem 14. Note that if $P_1 = 1$, this setting coincides with a homogeneous market and thus the results follow from Theorem 33. Therefore, we prove the results for $P_1 < 1$.

We first characterize conditions under which $G_{2l}(t,I,P) \geqslant 0$. For simplicity we assume that t is continuous, but similar arguments apply to the discrete case. After some algebra, we obtain that

$$G_{2l}(t,I,P) = A'\ln(t) - B't - C',$$

where $A' := (K-1)I\Delta/D - 8\beta/\Delta$, $B' := (1-P_1)\Delta$,

$$C' := (-(K-1)\Delta/D)\ln\left(\prod_{i=1}^{I} p_i\right) + \left(1+\pi^2/3\right)(KP_1-1)\Delta,$$

$\beta := \sum_{k=2}^{K} 1/(P_1-P_k)$, and D is the Kullback-Leibler divergence. Note that if $A' \leqslant 0$, then $G_{2l}(t,I,P) < 0$ for all $t \geqslant 1$ as $B' > 0$ and $C' > 0$. Thus, we need $A' > 0$ which requires that

$$I > \frac{8\beta D}{(K-1)\Delta^2}. \tag{B.2}$$

Having $A' > 0$ implies that $G_{2l}(t,I,P)$ is a concave function of t, and this function has a maximum at $t = A'/B'$. Let $G_l^{max} := \max_{t>0}\{G_{2l}(t,I,P)\} = G_{2l}(A'/B',I,P)$.

Note that $\partial G_{2l}(t, I, P)/\partial t \geqslant (<)\ 0$ for $t \leqslant (>)\ A'/B'$, and $G_{2l}(t, I, P) \to -\infty$ as $t \to \infty$ (since $B' > 0$). Therefore, if $G_l^{max} > 0$, we have that $G_{2l}(t, I, P) \geqslant 0$ for $\tilde{t}_l(I, P) \leqslant t \leqslant \tilde{t}_u(I, P)$ for some thresholds $1 < \tilde{t}_l(I, P) \leqslant \tilde{t}_u(I, P) \leqslant \infty$. In what follows, we provide conditions that guarantee that $G_l^{max} > 0$. We have that

$$G_l^{max} = A' \ln(A'/B') - A' - C'.$$

It follows that $G_l^{max} > 0$ if

$$\frac{A'}{B'} > \exp\left(\frac{C'}{A'} + 1\right). \tag{B.3}$$

Considering the fact that arrivals are uniform, i.e., $p_i = 1/I, \forall i$, we have that $\ln\left(\prod_{i=1}^{I} p_i\right) = -I\ln(I)$. This implies that

$$\begin{aligned}
\frac{C'}{A'} &= \frac{((K-1)\Delta/D)\, I \ln(I) + (1+\pi^2/3)\,(KP_1 - 1)\Delta}{(K-1)I\Delta/D - 8\beta/\Delta} \\
&= \ln(I) + \frac{(8\beta/\Delta)\ln(I) + (1+\pi^2/3)\,(KP_1 - 1)\Delta}{(K-1)I\Delta/D - 8\beta/\Delta} \\
&= \ln(I) + f(I, P),
\end{aligned}$$

where

$$f(I, P) := \frac{(8\beta/\Delta)\ln(I) + (1+\pi^2/3)\,(KP_1 - 1)\Delta}{(K-1)I\Delta/D - 8\beta/\Delta}.$$

Therefore, to ensure that (B.3) holds, we provide conditions that guarantee that

$$\frac{A'}{B'} > \exp(\ln(I) + f(I, P) + 1) = (\exp(f(I, P) + 1))\, I. \tag{B.4}$$

After some algebra, we obtain that (B.4) is true if

$$\left(\frac{K-1}{(1-P_1)D} - \exp(f(I, P) + 1)\right) I > \frac{8\beta}{(1-P_1)\Delta^2}. \tag{B.5}$$

Note that for the above inequality to hold, the left-hand side of (B.5) must be positive. This requires that $P_1 > \Pi_1(I, P)$ where

$$\Pi_1(I, P) := 1 - \frac{K-1}{D \exp(f(I, P) + 1)}.$$

For any given I and P', we let $\tilde{P}_1(I, P') := \inf\{P_1 : P_1 > \Pi_1(I, P), \max\{P_2, \ldots, P_K\} < P_1 < 1\}$ where $P_K = 1 - (P_1 + P_2 + \cdots + P_{K-1})$ and recall that $P' = (P_2, \ldots, P_{K-1})$. Note that $\Pi_1(I, P) < 1$. Moreover, in what follows (Lemma 35), we show that $f(I, P)$ (and thus $\Pi_1(I, P)$) is decreasing in P_1 for $I > e$. Therefore, such $\tilde{P}_1(I, P')$ exists for $I > e$. It follows that, if $P_1 > \tilde{P}_1(I, P')$ and $I > e$, then after some algebra we have that (B.5) is true if

$$I > \frac{8\beta D}{\Delta^2 \left((K-1) - \exp(f(I, P) + 1)(1 - P_1) D\right)}. \tag{B.6}$$

We now present the intermediate result.

Lemma 35. *$f(I, P)$ is decreasing in I and P_1 (assuming that the vector P' is constant) for $I > e$.*

Proof. We prove the result by taking the first-order partial derivatives. We have that

$$\begin{aligned}
\frac{\partial f}{\partial I} &= \frac{8(K-1)\beta/D - (8\beta/\Delta)^2 (1/I) - ((K-1)\Delta/D)\left[(8\beta/\Delta)\ln(I) + \left(1 + \pi^2/3\right)(KP_1 - 1)\Delta\right]}{((K-1)I\Delta/D - 8\beta/\Delta)^2} \\
&= \frac{\left[(8(K-1)\beta/D)(1 - \ln(I))\right] - (8\beta/\Delta)^2 (1/I) - \left(1 + \pi^2/3\right)(K-1)(KP_1 - 1)\Delta^2/D}{((K-1)I\Delta/D - 8\beta/\Delta)^2}.
\end{aligned}$$

Note that $(8(K-1)\beta/D)(1 - \ln(I)) < 0$ guarantees that $\partial f/\partial I < 0$. The former inequality holds if $I > e$.

We note that $P_K = 1 - (P_1 + P_2 + \cdots + P_{K-1})$. We then have that

$$\frac{\partial \beta}{\partial P_1} = -\left[\frac{1}{(P_1 - P_2)^2} + \frac{1}{(P_1 - P_3)^2} + \cdots + \frac{1}{(P_1 - P_{K-1})^2} + \frac{2}{(2P_1 + P_2 + P_3 + \cdots + P_{K-1} - 1)^2}\right] < 0.$$

Thus, β is decreasing in P_1 which implies that the denominator of function f is increasing in P_1. Therefore, to show that function f is decreasing in P_1, we provide conditions that guarantee that its numerator is decreasing in P_1. Let f_{num} denote the numerator of function f. We then have that

$$\frac{\partial f_{num}}{\partial P_1} = -\left(8\ln(I)/\Delta\right)\gamma + K\left(1+\pi^2/3\right)\Delta,$$

where $\gamma := -\partial\beta/\partial P_1$. Therefore, $\partial f_{num}/\partial P_1 < 0$ if

$$\ln(I) > \frac{K\left(1+\pi^2/3\right)\Delta^2}{8\gamma}.$$

Since $\gamma \geqslant K$, we have that $I > \exp\left(\left(1+\pi^2/3\right)\Delta^2/8\right)$ guarantees that $\partial f/\partial P_1 < 0$. Because $\exp(1) = e > \exp\left(\left(1+\pi^2/3\right)\Delta^2/8\right)$ as $0 < \Delta \leqslant 1$, we conclude that $I > e$ guarantees that $\partial f_{num}/\partial P_1 < 0$.

$\square$

We conclude that in order to have $G_l^{max} > 0$, we need that $P_1 > \tilde{P}_1(I, P')$ and the inequalities (B.2) and (B.6) to hold. Seeing that the right-hand side of inequality (B.6) is larger than that for inequality (B.2), for any given P, we set $\tilde{I}_l(P)$ to be the smallest integer I greater than e that satisfies the inequality (B.6) and take $I \geqslant \tilde{I}_l(P)$. From Lemma 35, we know that $f(I, P)$ is decreasing in I for $I > e$. Because the right-hand side of the inequality (B.6) is increasing in $f(I, P)$ (and thus decreasing in I for $I > e$), such $\tilde{I}_l(P)$ exists (note that the right-hand side of the inequality (B.6) converges to $8\beta D/\left(\Delta^2((K-1)-(1-P_1)eD)\right)$ as $I \to \infty$). This completes the proof of $G_{2l}(t, I, P) \geqslant 0$.

We next prove that $\tilde{I}_l(P)$ is non-increasing in P_1 (assuming that the vector P' is constant). We know that $\tilde{I}_l(P)$ is the smallest integer I greater than e that satisfies the inequality in (B.6). The result follows from noting that the right-hand side of inequality (B.6) is decreasing in P_1 for $I > e$ as a result of Lemma 35.

We finally prove that $\tilde{P}_1(I, P')$ is non-increasing in I. We have from above that

$$\tilde{P}_1(I, P') = \inf \{P_1 : P_1 > \Pi_1(I, P), \max \{P_2, \ldots, P_K\} < P_1 < 1\},$$

where $\Pi_1(I, P) = 1 - (K-1)/(D \exp(f(I, P) + 1))$. The result follows from noting that $\Pi_1(I, P)$ is decreasing in I for $I > e$ as a result of Lemma 35.

$\square$

B.2 Additional Proofs and Numerical Results

Extension of Theoretical Results for Thompson Sampling. Consider a classic bandit setting with N arms and Bernoulli rewards and suppose, without loss of generality, that arm 1 is the unique optimal arm. Let Δ_j denote the optimality gap of product j. Agrawal and Goyal (2011) prove the following upper bound on the regret of Thompson Sampling policy in this setting:

$$R^{\pi}(T) \leqslant C_1 \ln(T) + C_2,$$

where π denotes the Thompson Sampling policy,

$$C_1 := 1152\left(\sum_{j=2}^{N}\frac{1}{\Delta_j^2}\right)^2 + 288\sum_{j=2}^{N}\frac{1}{\Delta_j^2} + 72\sum_{j=2}^{N}\frac{1}{\Delta_j}$$

and

$$C_2 := 192N\sum_{j=2}^{N}\frac{1}{\Delta_j^2} + 104(N-1).$$

Using the upper bound above, extending the results of Section 3.6.1 is immediate. In what follows, we discuss the changes to the proof of Lemma 34 and Theorem 14. Following the same arguments as in the proof of Lemma 34, we have in this setting that

$$U_{pool}(T, I, P) = (C_1' \ln(T) + C_2') + T(1 - P_1)\Delta,$$

where

$$C_1' := 1152\left(\sum_{k=2}^{K}\frac{1}{{\Delta'_k}^2}\right)^2 + 288\sum_{k=2}^{K}\frac{1}{{\Delta'_k}^2} + 72\sum_{k=2}^{K}\frac{1}{\Delta'_k}$$

and

$$C_2' := 192N\sum_{k=2}^{K}\frac{1}{{\Delta'_k}^2} + 104(N-1),$$

and $\Delta'_k = (P_1 - P_k)\Delta$ is as defined in the proof of Lemma 34. After some algebra, we obtain that

$$G_{2l}(t, I, P) = A' \ln(t) - B't - C',$$

where $A' := (K-1)I\Delta/D - C'_1$, $B' := (1-P_1)\Delta$, $C' := (-(K-1)\Delta/D)\ln\left(\prod_{i=1}^{I} p_i\right) + C'_2$. Following the same arguments as in the proof of Theorem 14, we need $A' > 0$ which requires that

$$I > \frac{C'_1 D}{(K-1)\Delta}. \tag{B.7}$$

The only changes to the rest of the proof are the definitions of the functions $f(I, P)$ and $\tilde{I}_l(P)$, and inequality (B.6). In this setting, we have that

$$f(I, P) := \frac{C'_1 \ln(I) + C'_2}{(K-1)I\Delta/D - C'_1}.$$

After some algebra, we have that the inequality (B.6) in this setting becomes

$$I > \frac{C'_1 D}{\Delta\left((K-1) - \exp(f(I, P) + 1)(1 - P_1)D\right)}. \tag{B.8}$$

Similar to the proof of Theorem 14, we note that the right-hand side of inequality (B.8) is larger than that for inequality (B.7). Thus, we define $\tilde{I}_l(P)$ as the smallest integer I greater than e that satisfies the inequality in (B.8).

To conclude the extension, we only need to prove Lemma 35 in this setting. Following the same arguments as in the proof of Lemma 35, we have that

$$\frac{\partial f}{\partial I} = \frac{\left[(C'_1(K-1)\Delta/D)\left(1 - \ln(I)\right)\right] - {C'_1}^2/I - ((K-1)\Delta C'_2/D)}{((K-1)I\Delta/D - C'_1)^2}.$$

Note that $I > e$ guarantees that $\partial f/\partial I < 0$. To show that $f(I, P)$ is decreasing in P_1, note that Δ'_k is increasing in P_1 which, in turn, implies that C'_1 and C'_2 are decreasing in P_1.

The non-increasing property of $\tilde{I}_l(P)$ and $\tilde{P}_1(I, P')$ in P_1 and I, respectively, follows from similar arguments as in the proof of Theorem 14 and the discussion above.

$\square$

Details of Numerical Study in Section 3.5.5. The separation-based experiments are discussed at the end of Section 3.5.5. Table B.1 below reports the percentage improvement of the dynamic clustering policy over the other policies, both in terms of regret and revenue (i.e., expected number of transactions) in the separation-based experiments using a synthetic linear dataset.

Table B.1: Average percentage improvement of dynamic clustering policy over data-intensive and linear-utility policies in separation-based experiments using a synthetic linear dataset.

	Regret		Revenue	
	Linear-Utility	Data-Intensive	Linear-Utility	Data-Intensive
$T = 500$	31.2%	42.4%	26.2%	51%
$T = 1000$	33.4%	50%	20.3%	51%
$T = 5000$	36.4%	56%	11.1%	28.5%
$T = 10000$	16.7%	47.1%	3.3%	16.7%
$T = 20000$	-21.3%	34.4%	-2.5%	8.5%

Appendix C

Appendices for Chapter 4

C.1 Proofs

Proof of Proposition 15. Considering the first-order condition (FOC) of $\pi_0(p)$ with respect to p, after some algebra we have that

$$\frac{\partial \pi_0(p)}{\partial p} = \frac{\exp\left((\beta + p)/\mu_1\right)\left(\phi_0(p) - p\right)}{\left(\exp(\beta/\mu_1) + \exp(p/\mu_1)\right)^2 \mu_1} = 0,$$

where $\phi_0(p) = \mu_1\left(1 + \exp\left((\beta - p)/\mu_1\right)\right)$. Root of the above equation can be found by either setting $p = \infty$, or by setting $\phi_0(p) - p = 0$. One can easily see that $\phi_0(p)$ is a decreasing function of p, and that $\phi_0(p = 0) > 0$ and $\phi_0(p) \to \mu_1$ as $p \to \infty$. Thus, the equation $\phi_0(p_0^*) = p_0^*$ has a unique solution at a finite price $0 < p_0^* < \infty$. Therefore, the FOC only holds at $p = p_0^*$ and $p = \infty$. We also note that $\partial\pi_0(p)/\partial p > 0$ for $0 \leqslant p < p_0^*$ and $\partial\pi_0(p)/\partial p < 0$ for $p > p_0^*$. Thus, $\pi_0(p)$ is increasing (decreasing) in p for p smaller than (larger than) p_0^*. Finally, we note that $\pi_0(p = 0) = 0$, $\pi_0(p) \to 0$ as $p \to \infty$, and $\pi_0(p)$ is finite for all finite values of $0 < p < \infty$. These imply that $\pi_0(p)$ is a unimodal function of p with a maximum at the optimal price $0 < p_0^* < \infty$.

Replacing p_0^* by $\phi_0(p_0^*)$ in $\pi_0(p)$, we obtain that

$$\pi_0^* = \pi_0(p_0^*) = p_0^* - \mu_1 = \mu_1 \exp\left(\frac{\beta - p_0^*}{\mu_1}\right).$$

$\square$

Proof of Proposition 16. From the Envelope Theorem and after some algebra, we have that

$$\frac{\partial \pi_0^*}{\partial \beta} = \left.\frac{\partial \pi_0}{\partial \beta}\right|_{p=p_0^*} = \frac{1}{1+\exp\left(\frac{-\beta+p_0^*}{\mu_1}\right)} > 0.$$

Thus, π_0^* is increasing in β. To see that p_0^* is increasing in β, we note from the proof of Proposition 15 that $\phi_0(p) = \mu_1(1+\exp((\beta-p)/\mu_1))$, which is increasing in β. Thus, the optimal price p_0^* which is the unique solution to $\phi_0(p_0^*) = p_0^*$ is increasing in β. $\square$

Proof of Proposition 17. Considering the FOC of $\pi_1(p)$ with respect to p, after some algebra we have that $\partial\pi_1(p)/\partial p = 0$ by either setting $p = \infty$, or by setting $\phi_1(p) - p = 0$ where

$$\phi_1(p) = \frac{\left(\exp\left(\frac{\beta+c}{\mu_2}\right)+\exp\left(\frac{p}{\mu_2}\right)\right)\left(\exp\left(\frac{k}{\mu_1}\right)+\left(\exp\left(\frac{-c}{\mu_2}\right)+\exp\left(\frac{\beta-p}{\mu_2}\right)\right)^{\mu_2/\mu_1}\right)\mu_1\mu_2}{\exp\left(\frac{p}{\mu_2}\right)\left(\exp\left(\frac{k}{\mu_1}\right)+\left(\exp\left(\frac{-c}{\mu_2}\right)+\exp\left(\frac{\beta-p}{\mu_2}\right)\right)^{\mu_2/\mu_1}\right)\mu_1+\exp\left(\frac{k}{\mu_1}+\frac{\beta+c}{\mu_2}\right)\mu_2}.$$

Let $C_1 := \left(\exp\left(\frac{-c}{\mu_2}\right)+\exp\left(\frac{\beta-p}{\mu_2}\right)\right)^{\mu_2/\mu_1}$. We then have that

$$\frac{\partial\phi_1(p)}{\partial p} = -\frac{\exp\left(\frac{b+c+p}{\mu_2}\right)\left(\exp\left(\frac{k}{\mu_1}\right)+C_1\right)\mu_1\left(C_1\mu_1+\exp\left(\frac{k}{\mu_1}\right)(\mu_1-\mu_2)\right)+\exp\left(\frac{k}{\mu_1}+\frac{2(\beta+c)}{\mu_2}\right)C_1\mu_2^2}{\left(\exp\left(\frac{p}{\mu_2}\right)\left(\exp\left(\frac{k}{\mu_1}\right)+C_1\right)\mu_1+\exp\left(\frac{k}{\mu_1}+\frac{\beta+c}{\mu_2}\right)\mu_2\right)^2}.$$

Because $\mu_1 \geqslant \mu_2$, we can see from above that $\partial\phi_1(p)/\partial p < 0$ for all $p \geqslant 0$, that is, the function $\phi_1(p)$ is decreasing in p. We can also see that $\phi_1(p=0) > 0$ and $\phi_1(p) \to \mu_2$ as $p \to \infty$. Thus, the equation $\phi_1(p_1^*) = p_1^*$ has a unique solution at a finite price

$0 < p_1^* < \infty$. Therefore, the FOC only holds at $p = p_1^*$ and $p = \infty$. One can also check that $\partial\pi_1(p)/\partial p > 0$ for $0 \leqslant p < p_1^*$ and $\partial\pi_1(p)/\partial p < 0$ for $p > p_1^*$. Thus, $\pi_1(p)$ is increasing (decreasing) in p for p smaller than (larger than) p_1^*. Finally, we note that $\pi_1(p = 0) = 0$, $\pi_1(p) \to 0$ as $p \to \infty$, and $\pi_1(p)$ is finite for all finite values of $0 < p < \infty$. These imply that $\pi_1(p)$ is a unimodal function of p with a maximum at the optimal price $0 < p_1^* < \infty$. Replacing p_1^* by $\phi_1(p_1^*)$ in $\pi_1(p)$, we obtain that

$$\pi_1^* = \pi_1(p_1^*) = \frac{\mu_1\mu_2 \exp\left(\frac{\beta+c}{\mu_2}\right)\left(\exp\left(\frac{-c}{\mu_2}\right) + \exp\left(\frac{\beta-p_1^*}{\mu_2}\right)\right)^{\mu_2/\mu_1}}{\mu_1 \exp\left(\frac{p_1^*}{\mu_2}\right)\left(\left(\exp\left(\frac{-c}{\mu_2}\right) + \exp\left(\frac{\beta-p_1^*}{\mu_2}\right)\right)^{\mu_2/\mu_1} + \exp(k/\mu_1)\right) + \mu_2 \exp(k/\mu_1)\exp\left(\frac{\beta+c}{\mu_2}\right)}.$$

□

Proof of Proposition 18. From the Envelope Theorem and after some algebra, we have that

$$\frac{\partial\pi_1^*}{\partial\beta} = \frac{\partial\pi_1}{\partial\beta}\bigg|_{p=p_1^*} = \frac{\exp\left(\frac{\beta+c}{\mu_2}\right)\left(\exp\left(\frac{-c}{\mu_2}\right) + \exp\left(\frac{\beta-p_1^*}{\mu_2}\right)\right)^{\mu_2/\mu_1}}{\left(\exp\left(\frac{\beta+c}{\mu_2}\right) + \exp\left(\frac{p_1^*}{\mu_2}\right)\right)\left(\exp\left(\frac{k}{\mu_1}\right) + \left(\exp\left(\frac{-c}{\mu_2}\right) + \exp\left(\frac{\beta-p_1^*}{\mu_2}\right)\right)^{\mu_2/\mu_1}\right)} > 0.$$

Thus, π_1^* is increasing in p_1^*. Moreover, we have that

$$\frac{\partial\phi_1}{\partial\beta} = \frac{\exp\left(\frac{b+c+p}{\mu_2}\right)\left(\exp\left(\frac{k}{\mu_1}\right) + C_1\right)\mu_1\left(C_1\mu_1 + \exp\left(\frac{k}{\mu_1}\right)(\mu_1 - \mu_2)\right) + \exp\left(\frac{k}{\mu_1} + \frac{2(\beta+c)}{\mu_2}\right)C_1\mu_2^2}{\left(\exp\left(\frac{p}{\mu_2}\right)\left(\exp\left(\frac{k}{\mu_1}\right) + C_1\right)\mu_1 + \exp\left(\frac{k}{\mu_1} + \frac{\beta+c}{\mu_2}\right)\mu_2\right)^2},$$

where ϕ_1 is as defined in the proof of Proposition 17 and

$$C_1 = \left(\exp\left(\frac{-c}{\mu_2}\right) + \exp\left(\frac{\beta - p}{\mu_2}\right)\right)^{\mu_2/\mu_1}.$$

Because $\mu_1 \geqslant \mu_2$, we can see from above that $\partial\phi_1/\partial\beta > 0$. Thus, the optimal price p_1^* which is the unique solution to $\phi_1(p_1^*) = p_1^*$ is increasing in β.

To prove the results for k, note that

$$\pi_1(p) = \frac{\exp\left(\frac{\beta+c}{\mu_2}\right)\left(\exp\left(\frac{-c}{\mu_2}\right)+\exp\left(\frac{\beta-p}{\mu_2}\right)\right)^{\mu_2/\mu_1} p}{\left(\exp\left(\frac{\beta+c}{\mu_2}\right)+\exp\left(\frac{p}{\mu_2}\right)\right)\left(\exp\left(\frac{k}{\mu_1}\right)+\left(\exp\left(\frac{-c}{\mu_2}\right)+\exp\left(\frac{\beta-p}{\mu_2}\right)\right)^{\mu_2/\mu_1}\right)}.$$

From above, we can easily see that π_1 is decreasing in k which implies that π_1^* is also decreasing in k. In addition,

$$\frac{\partial\phi_1}{\partial k} = -\frac{\exp\left(\frac{k}{\mu_1}+\frac{\beta+c}{\mu_2}\right) C_1\left(\exp\left(\frac{\beta+c}{\mu_2}\right)+\exp\left(\frac{p}{\mu_2}\right)\right)\mu_2^2}{\left(\exp\left(\frac{p}{\mu_2}\right)\left(\exp\left(\frac{k}{\mu_1}\right)+C_1\right)\mu_1+\exp\left(\frac{k}{\mu_1}+\frac{\beta+c}{\mu_2}\right)\mu_2\right)^2} < 0.$$

Thus, p_1^* which is the unique solution to $\phi_1(p_1^*) = p_1^*$ is decreasing in k. $\square$

Proof of Proposition 19. From the Envelope Theorem and after some algebra, we have that the sign of $\partial\pi_1^*/\partial c = \partial\pi_1/\partial c|_{p=p_1^*}$ is determined by the sign of

$$(1+c)^2\exp\left(\frac{k'}{\mu_1+c\,\mu_1}+\frac{p_1^*}{\mu_2}\right)C_2\,\mu_1 - \exp\left(\frac{\beta+c}{\mu_2}\right)k'\mu_2 + \exp\left(\frac{p_1^*}{\mu_2}\right)\left((1+c)^2(\mu_1-\mu_2)-k'\mu_2\right),$$

where $C_2 := \left(\exp\left(\frac{-c}{\mu_2}\right)+\exp\left(\frac{\beta-p_1^*}{\mu_2}\right)\right)^{\mu_2/\mu_1}$. From above, we can see that for sufficiently large c, the term $-\exp((\beta+c)/\mu_2)\,k'\mu_2$ dominates the function above, thus, $\partial\pi_1^*/\partial c < 0$ for large enough c. We can also check that

$$\lim_{c\to\infty}\frac{\partial\phi_1}{\partial c} = \lim_{c\to\infty}\left(\frac{\mu_1\left(1+\exp\left(\frac{\beta-p}{\mu_1}\right)\right)}{\mu_2} - \frac{\exp\left(\frac{\beta+c}{\mu_2}\right)\mu_2\left(\mu_1(1+c)^2\exp\left(\frac{\beta+c}{\mu_2}\right)\left(1+\exp\left(\frac{\beta-p}{\mu_1}\right)\right)\right)}{(1+c)^2\mu_2^2\exp\left(\frac{2(\beta+c)}{\mu_2}\right)}\right) = 0,$$

where ϕ_1 is as defined in the proof of Proposition 17. Seeing that the optimal price p_1^* is the unique solution to $\phi_1(p_1^*) = p_1^*$, we conclude that there exists a threshold c' such that p_1^* is roughly constant for $c \geqslant c'$ (we know that as $c\to\infty$, p_1^* converges to p_0^* which is the optimal price for the no-returns case). The discussion above implies that there exists a threshold $\tilde{c}$ such that π_1^* is decreasing in c for $c \geqslant \tilde{c}$. $\square$

Proof of Proposition 20. In case that $k(c) = 0, \forall c$, we have that

$$\frac{\partial \pi_1}{\partial c} = \frac{\exp\left(\frac{\beta+c+p}{\mu_2}\right) C_1 \left(\mu_1 + C_1\mu_1 - \mu_2\right)}{\left(\exp\left(\frac{\beta+c}{\mu_2}\right) + \exp\left(\frac{p}{\mu_2}\right)\right)^2 \left(1+C_1\right)^2 \mu_1\mu_2},$$

where $C_1 = \left(\exp\left(\frac{-c}{\mu_2}\right) + \exp\left(\frac{\beta-p}{\mu_2}\right)\right)^{\mu_2/\mu_1}$. Because $\mu_1 \geqslant \mu_2$, we can see from above that $\partial\pi_1/\partial c > 0$ and thus, π_1 is increasing in c which, in turn, implies that π_1^* is increasing in c. Moreover, we have that

$$\frac{\partial \phi_1}{\partial c} = \frac{\exp\left(\frac{\beta+c+p}{\mu_2}\right) \left(C_1^2\mu_1^2 + \mu_1(\mu_1 - \mu_2) - C_1(-2\mu_1^2 + \mu_1\mu_2 + \mu_2^2)\right)}{\left(\exp\left(\frac{p}{\mu_2}\right)\left(1+C_1\right)\mu_1 + \exp\left(\frac{\beta+c}{\mu_2}\right)\mu_2\right)^2},$$

where ϕ_1 is as defined in the proof of Proposition 17. Because $\mu_1 \geqslant \mu_2$, we have that $(-2\mu_1^2 + \mu_1\mu_2 + \mu_2^2) \leqslant 0$ which then implies that $\partial\phi_1/\partial c > 0$. Thus, p_1^* which is the unique solution to $\phi_1(p_1^*) = p_1^*$ is increasing in c.

$\square$

Proof of Proposition 21. We note that the case with product returns reduces to that without returns as $c \to \infty$. (This is evident from the purchase and return probabilities: $\mathbb{P}_i^R \to \mathbb{P}_i^{NR}$ and $\mathbb{P}_i^{return} \to 0$ as $c \to \infty$.) That is, $\pi_1^* \to \pi_0^*$ as $c \to \infty$. From Proposition 19, we know that for the case of $k(c) = -k'/(c+1)$, there exists a threshold $\tilde{c}$ such that π_1^* is decreasing in c for all $c \geqslant \tilde{c}$. The result follows from above and seeing that π_0^* is independent of c.

$\square$

Proof of Proposition 22. As discussed in the proof of Proposition 21, $\pi_1^* \to \pi_0^*$ as $c \to \infty$. From Proposition 20, we know that for the case of $k(c) = 0, \forall c$, we have that π_1^* is increasing in c. The result follows from above and seeing that π_0^* is independent of c.

$\square$

Proof of Proposition 23. The result follows from a similar line of arguments as in the proof of Proposition 15, but with the difference that $\phi_0(p) = \mu_1(1 + \sum_{i=1}^{n} \exp(\frac{\beta_i - p_0^*}{\mu_1}))$.

$\square$

Bibliography

Abernethy, J., Hazan, E. and Rakhlin, A. (2008). Competing in the dark: An efficient algorithm for bandit linear optimization.. 'COLT'. pp. 263–274.

Achterberg, T. and Wunderling, R. (2013). *Mixed Integer Programming: Analyzing 12 Years of Progress*. Springer Berlin Heidelberg. Berlin, Heidelberg. pp. 449–481.

Agrawal, R. (1995). The continuum-armed bandit problem. *SIAM J. Control Optim.* **33**(6), 1926–1951.

Agrawal, R., Hegde, M. and Teneketzis, D. (1990). Multi-armed bandit problems with multiple plays and switching cost. *Stochastics: An International Journal of Probability and Stochastic Processes* **29**(4), 437–459.

Agrawal, S., Avadhanula, V., Goyal, V. and Zeevi, A. (2016). MNL-bandit: A dynamic learning approach to assortment selection. Working paper, Columbia University.

Agrawal, S. and Goyal, N. (2011). Analysis of thompson sampling for the multi-armed bandit problem. *arXiv preprint arXiv:1111.1797* .

Alptekinoğlu, A. and Grasas, A. (2014). When to carry eccentric products? optimal retail assortment under consumer returns. *Production and Operations Management* **23**(5), 877–892.

Altug, M. S. and Aydinliyim, T. (2016). Counteracting strategic purchase deferrals: The impact of online retailers return policy decisions. *Manufacturing & Service Operations Management* **18**(3), 376–392.

Anantharam, V., Varaiya, P. and Walrand, J. (1987). Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays-part i: IID rewards. *Automatic Control, IEEE Transactions on* **32**(11), 968–976.

Anderson, E. T., Hansen, K. and Simester, D. (2009). The option value of returns: Theory and empirical evidence. *Marketing Science* **28**(3), 405–423.

Anderson, S. P. and De Palma, A. (1992). Multiproduct firms: A nested logit approach. *The Journal of Industrial Economics* pp. 261–276.

Anonymous (2017). 2017 consumer returns in the retail industry. *Appriss Retail* .

Ansari, A. and Mela, C. F. (2003). E-customization. *J. Marketing Res.* **40**(2), 131–145.

Antoniak, C. E. (1974). Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *The annals of statistics* pp. 1152–1174.

Applegate, D., Bixby, R., Chvátal, V. and Cook, W. (2011). *The Traveling Salesman Problem: A Computational Study*. Princeton Series in Applied Mathematics. Princeton University Press.

Arora, N., Dreze, X., Ghose, A., Hess, J. D., Iyengar, R., Jing, B., Joshi, Y., Kumar, V., Lurie, N., Neslin, S. et al. (2008). Putting one-to-one marketing to work: Personalization, customization, and choice. *Marketing Lett.* **19**(3-4), 305–321.

Auer, P., Cesa-Bianchi, N. and Fischer, P. (2002). Finite-time analysis of the multi-armed bandit problem. *Machine Learn.* **47**(2-3), 235–256.

Auer, P., Cesa-bianchi, N., Freund, Y. and Schapire, R. E. (2003). The non-stochastic multi-armed bandit problem. *SIAM Journal on Computing* **32**, 48–77.

Awerbuch, B. and Kleinberg, R. D. (2004). Adaptive routing with end-to-end feedback: distributed learning and geometric approaches. 'Proceedings of the thirty-sixth annual ACM symposium on Theory of computing'. STOC '04. ACM. New York, NY, USA. pp. 45–53.

Balas, E. and Carrera, M. C. (1996). A dynamic subgradient-based branch-and-bound procedure for set covering. *Operations Research* **44**, 875–890.

Bernstein, F., Kök, A. G. and Xie, L. (2015). Dynamic assortment customization with limited inventories. *Manufacturing Service Oper. Management* **17**(4), 538–553.

Bernstein, F., Modaresi, S. and Sauré, D. (2017). A dynamic clustering approach to data-driven assortment personalization. Forthcoming at *Management Sci.*

Berry, D. and Fristedt, B. (1985). *Bandit Problems*. Chapman and Hall. London, UK.

Besbes, O. and Sauré, D. (2016). Product assortment and price competition under multinomial logit demand. *Production Oper. Management* **25**(1), 114–127.

Bixby, R. E. (2012). A brief history of linear and mixed-integer programming computation. *Documenta Mathematica* pp. 107–121.

Bubeck, S., Munos, R., Stoltz, G. and Szepesvári, C. (2011). X-armed bandits. *Journal of Machine Learning Research* **12**, 1655–1695.

Burda, M., Harding, M. and Hausman, J. (2008). A bayesian mixed logit–probit model for multinomial choice. *Journal of Econometrics* **147**(2), 232–246.

Caro, F. and Gallien, J. (2007). Dynamic assortment with demand learning for seasonal consumer goods. *Management Sci.* **53**(2), 276–292.

Carvajal, R., Constantino, M., Goycoolea, M., Vielma, J. P. and Weintraub, A. (2013). Imposing connectivity constraints in forest planning models. *Operations Research* **61**(4), 824–836.

Cesa-Bianchi, N. and Lugosi, G. (2006). *Prediction, Learning, and Games*. Cambridge University Press.

Cesa-Bianchi, N. and Lugosi, G. (2012). Combinatorial bandits. *Journal of Computer and System Sciences* .

Che, Y.-K. (1996). Customer return policies for experience goods. *The Journal of Industrial Economics* pp. 17–24.

Chen, W., Wang, Y. and Yuan, Y. (2013). Combinatorial multi-armed bandit: General framework, results and applications. 'Proceedings of the 30th International Conference on Machine Learning (ICML-13)'. pp. 151–159.

Chen, X., Owen, Z., Pixton, C. and Simchi-Levi, D. (2015). A statistical learning approach to personalization in revenue management. *Available at SSRN 2579462* .

Ciocan, D. F. and Farias, V. F. (2014). Fast demand learning for display advertising revenue management. Working paper, MIT.

Cook, W. J., Cunningham, W. H., Pulleyblank, W. R. and Schrijver, A. (1998). *Combinatorial optimization*. John Wiley & Sons, Inc.. New York, NY, USA.

Cover, T. and Thomas, J. (2006). *Elements of Information theory*. John Wiley & Sons, Inc.. Hoboken, NJ.

Dani, V., Hayes, T. P. and Kakade, S. M. (2008). Stochastic linear optimization under bandit feedback.. 'COLT'. pp. 355–366.

Davis, S., Gerstner, E. and Hagerty, M. (1995). Money back guarantees in retailing: Matching products to consumer tastes. *Journal of Retailing* **71**(1), 7–22.

Davis, S., Hagerty, M. and Gerstner, E. (1998). Return policies and the optimal level of hassle. *Journal of Economics and Business* **50**(5), 445–460.

eMarketer (2017). Worldwide retail and ecommerce sales: emarketer's estimates for 2016-2021. https://www.emarketer.com/Report/Worldwide-Retail-Ecommerce-Sales-eMarketers-Estimates-20162021/2002090.

Etcheberry, J. (1977). The set-covering problem: A new implicit enumeration algorithm. *Operations research* **25**, 760–772.

Ferguson, T. (1973). A bayesian analysis of some nonparametric problems. *The annals of statistics* pp. 209–230.

Ferguson, T. S. (1983). Bayesian density estimation by mixtures of normal distributions. *Recent advances in statistics* **24**, 287–302.

Fischetti, M. and Lodi, A. (2011). 'Heuristics in mixed integer programming, volume 3 of wiley encyclopedia of operations research and management science, jj cochran edt'.

Fisher, M. and Vaidyanathan, R. (2014). A demand estimation procedure for retail assortment optimization with results from implementations. *Management Sci.* **60**(10), 2401–2415.

Gai, Y., Krishnamachari, B. and Jain, R. (2012). Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations. *IEEE/ACM Transactions on Networking (TON)* **20**(5), 1466–1478.

Gallego, G., Li, A., Truong, V.-A. and Wang, X. (2015). Online resource allocation with customer choice. *arXiv preprint arXiv:1511.01837* .

Gamrath, G., Fischer, T., Gally, T., Gleixner, A. M., Hendel, G., Koch, T., Maher, S. J., Miltenberger, M., Müller, B., Pfetsch, M. E., Puchert, C., Rehfeldt, D., Schenker, S., Schwarz, R., Serrano, F., Shinano, Y., Vigerske, S., Weninger, D., Winkler, M., Witt, J. T. and Witzig, J. (2016). The scip optimization suite 3.2. Technical Report 15-60. ZIB. Takustr.7, 14195 Berlin.

Gelman, A., Carlin, J. B., Stern, H. S. and Rubin, D. B. (2014). *Bayesian data analysis*. Vol. 2. Taylor & Francis.

Gittins, J. (1979). Bandit processes and dynamic allocation rules. *Journal of the Royal Statistical Society* **41**, 148–177.

Gleixner, A., Eifler, L., Gally, T., Gamrath, G., Gemander, P., Gottwald, R. L., Hendel, G., Hojny, C., Koch, T., Miltenberger, M., Müller, B., Pfetsch, M. E., Puchert, C., Rehfeldt, D., Schlösser, F., Serrano, F., Shinano, Y., Viernickel, J. M., Vigerske, S., Weninger, D., Witt, J. T. and Witzig, J. (2017). The scip optimization suite 5.0. Technical Report 17-61. ZIB. Takustr.7, 14195 Berlin.

Golrezaei, N., Nazerzadeh, H. and Rusmevichientong, P. (2014). Real-time optimization of personalized assortments. *Management Sci.* **60**(6), 1532–1551.

Heller, K. A. and Ghahramani, Z. (2005). Bayesian hierarchical clustering. 'Proceedings of the 22nd international conference on Machine learning'. ACM. pp. 297–304.

Hess, J. D., Chu, W. and Gerstner, E. (1996). Controlling product returns in direct marketing. *Marketing Letters* **7**(4), 307–317.

Hoffman, K. L. and Padberg, M. (1993). Solving airline crew scheduling problems by branch-and-cut. *Management Science* **39**, 657–682.

Jasin, S. and Kumar, S. (2012). A re-solving heuristic with bounded revenue loss for network revenue management with customer choice. *Mathematics of Operations Research* **37**(2), 313–345.

Jünger, M., Liebling, T., Naddef, D., Nemhauser, G., Pulleyblank, W., Reinelt, G., Rinaldi, G. and Wolsey, L. (2010). Springer-Verlag. New York.

Kallus, N. and Udell, M. (2016). Dynamic assortment personalization in high dimensions. *arXiv preprint arXiv:1610.05604* . Working paper, Cornell University.

Kleinberg, R., Slivkins, A. and Upfal, E. (2008). Multi-armed bandits in metric spaces. *CoRR* **abs/0809.4882**.

Koch, T. and Martin, A. (1998). Solving steiner tree problems in graphs to optimality. *Networks* **32**(3), 207–232.

Kök, A. G., Fisher, M. L. and Vaidyanathan, R. (2015). Assortment planning: Review of literature and industry practice. *Retail Supply Chain Management* pp. 175–236.

Kulkarni, S. and Lugosi, G. (1997). Minimax lower bounds for the two-armed bandit problem. 'Decision and Control, 1997., Proceedings of the 36th IEEE Conference on'. Vol. 3. IEEE. pp. 2293–2297.

Lai, T. L. (1987). Adaptive treatment allocation and the multi-armed bandit problem. *The Annals of Statistics* pp. 1091–1114.

Lai, T. L. and Robbins, H. (1985*a*). Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics* **6**(1), 4–22.

Lai, T. L. and Robbins, H. (1985*b*). Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics* **6**(1), 4–22.

Linden, G., Smith, B. and York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. *Internet Computing, IEEE* **7**(1), 76–80.

Liu, K., Vakili, S. and Zhao, Q. (2012). Stochastic online learning for network optimization under random unknown weights. Working paper.

Magnanti, T. L. and Wolsey, L. A. (1995). *Optimal trees*. Vol. 7 of *Handbooks in Operational Research and Management Science*. North-Holland, Amsterdam. pp. 503–615.

Maher, S. J., Fischer, T., Gally, T., Gamrath, G., Gleixner, A., Gottwald, R. L., Hendel, G., Koch, T., Lübbecke, M. E., Miltenberger, M., Müller, B., Pfetsch, M. E., Puchert, C., Rehfeldt, D., Schenker, S., Schwarz, R., Serrano, F., Shinano, Y., Weninger, D., Witt, J. T. and Witzig, J. (2017). The scip optimization suite 4.0. Technical Report 17-12. ZIB. Takustr.7, 14195 Berlin.

Martin, R. K. (1991). Using separation algorithms to generate mixed integer model reformulations. *Operations Research Letters* **10**, 119–128.

Mersereau, A., Rusmevichientong, P. and Tsitsiklis, J. (2009). A structured multi-armed bandit problem and the greedy policy. *IEEE Transactions on Automatic Control* **54**(12), 2787–2802.

Modaresi, S., Sauré, D. and Vielma, J. P. (2014). Learning in combinatorial optimization: What and how to explore. Working paper, MIT.

Moorthy, S. and Srinivasan, K. (1995). Signaling quality with a money-back guarantee: The role of transaction costs. *Marketing Science* **14**(4), 442–466.

Nageswaran, L., Cho, S.-H. and Scheller-Wolf, A. A. (2017). Consumer return policies in omnichannel operations. Working paper, Carnegie Mellon University.

Neal, R. (2000). Markov chain sampling methods for dirichlet process mixture models. *Journal of Computational and Graphical Statistics* **9**(2), 249–265.

Niño-Mora, J. (2011). Computing a classic index for finite-horizon bandits. *INFORMS Journal on Computing* **23**(2), 254–267.

Pasternack, B. A. (1985). Optimal pricing and return policies for perishable commodities. *Marketing science* **4**(2), 166–176.

Petersen, J. A. and Kumar, V. (2009). Are product returns a necessary evil? antecedents and consequences. *Journal of Marketing* **73**(3), 35–51.

Petersen, J. A. and Kumar, V. (2010). Can product returns make you money?. *MIT Sloan Management Review* **51**(3), 85.

Petersen, J. A. and Kumar, V. (2015). Perceived risk, product returns, and optimal resource allocation: evidence from a field experiment. *Journal of Marketing Research* **52**(2), 268–285.

Phillips, E. E. (2017). Retailers offer myriad returns options to retain customers. *Wall Street Journal* .

Robbins, H. (1952). Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society* **58**, 527–535.

Robbins, H. (1985). Some aspects of the sequential design of experiments. *Herbert Robbins Selected Papers* pp. 169–177.

Rothvoß, T. (2013). Some 0/1 polytopes need exponential size extended formulations. *Mathematical Programming* **142**, 255–268.

Rothvoß, T. (2017). The matching polytope has exponential extension complexity. *Journal of the ACM (JACM)* **64**(6), 41.

Rusmevichientong, P., Shen, Z.-J. M. and Shmoys, D. B. (2010). Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Oper. Res.* **58**(6), 1666–1680.

Rusmevichientong, P. and Tsitsiklis, J. (2010). Linearly parameterized bandits. *Mathematics of Operations Research* **35**(2), 395–411.

Ryzhov, I. O. and Powell, W. B. (2009). The knowledge gradient algorithm for online subset selection. 'Proceedings of the 2009 IEEE International Symposium on Adaptive Dynamic Programming and Reinforcement Learning'. pp. 137–144.

Ryzhov, I. O. and Powell, W. B. (2011). Information collection on a graph. *Operations Research* **59**(1), 188–201.

Ryzhov, I. O., Powell, W. B. and Frazier, P. I. (2012). The knowledge gradient algorithm for a general class of online learning problems. *Operations Research* **60**(1), 180–195.

Sauré, D. and Zeevi, A. (2013). Optimal dynamic assortment planning with demand learning. *Manufacturing Service Oper. Management* **15**(3), 387–404.

Schrijver, A. (2003). *Combinatorial Optimization - Polyhedra and Efficiency.* Springer.

Shulman, J. D., Coughlan, A. T. and Savaskan, R. C. (2009). Optimal restocking fees and information provision in an integrated demand-supply model of product returns. *Manufacturing & Service Operations Management* **11**(4), 577–594.

Stanley, R. (1999). *Enumerative combinatorics, Volume 2.* Cambridge studies in advanced mathematics. Cambridge University Press.

Su, X. (2009). Consumer returns policies and supply chain performance. *Manufacturing & Service Operations Management* **11**(4), 595–612.

Thompson, W. R. (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* pp. 285–294.

Toriello, A. and Vielma, J. P. (2012). Fitting piecewise linear continuous functions. *European Journal of Operational Research* **219**, 86 – 95.

Ulu, C., Honhon, D. and Alptekinoğlu, A. (2012). Learning consumer tastes through dynamic assortments. *Oper. Res.* **60**(4), 833–849.

Ventura, P. and Eisenbrand, F. (2003). A compact linear program for testing optimality of perfect matchings. *Operations Research Letters* **31**(6), 429–434.

Vielma, J. P. (2015). Mixed integer linear programming formulation techniques. *SIAM Review* **57**, 3–57.

Wagner, L. and Martınez-de Albéniz, V. (2017). Pricing and assortment strategies with product exchanges. Working paper, IESE Business School.

Wang, R. and Sahin, O. (2017). The impact of consumer search cost on assortment planning and pricing. *Management Science* .

Wedel, M. and Kamakura, W. A. (2012). *Market segmentation: Conceptual and methodological foundations*. Vol. 8. Springer Science & Business Media.

Whittle, P. (1982). *Optimization over time: Vol I*. John Wiley and Sons Ltd.

Williamson, D. P. and Shmoys, D. B. (2011). *The Design of Approximation Algorithms*. Cambridge University Press.

Wood, M. (2014). A new kind of e-commerce adds a personal touch. *New York Times* .

Biography

Sajad Modaresi received his Bachelor's degree in Industrial Engineering in 2010 from Sharif University of Technology in Tehran, Iran, and his Master of Science in Industrial Engineering from University of Pittsburgh in 2012. Upon receiving his Ph.D. in Business Administration from Duke University, he will join the Kenan-Flagler Business School at the University of North Carolina in Chapel Hill as an Assistant Professor of Operations Management.

The first essay of this dissertation (Modaresi et al., 2014) is under revision at *Operations Research*. An earlier version of this paper won the second prize in the INFORMS JFIG Paper Competition in 2013. The second essay (Bernstein et al., 2017) is accepted for publication at *Management Science*. The final essay is a working paper.