Mathematics of Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Optimal Policy for Dynamic Assortment Planning Under Multinomial Logit Models

Xi Chen, Yining Wang, Yuan Zhou

To cite this article:
Xi Chen, Yining Wang, Yuan Zhou (2021) Optimal Policy for Dynamic Assortment Planning Under Multinomial Logit Models. Mathematics of Operations Research

Published online in Articles in Advance 13 May 2021

. https://doi.org/10.1287/moor.2021.1133 http://pubsonline.informs.org/journal/moor

MATHEMATICS OF OPERATIONS RESEARCH
Articles in Advance, pp. 1–19
ISSN 0364-765X (print), ISSN 1526-5471 (online)

Optimal Policy for Dynamic Assortment Planning Under Multinomial Logit Models

Xi Chen,[a] Yining Wang,[b] Yuan Zhou[c]

[a] Stern School of Business, New York University, New York, New York 10012; [b] Warrington College of Business, University of Florida, Gainesville, Florida 32611; [c] Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801

Contact: xc13@stern.nyu.edu, https://orcid.org/0000-0002-9049-9452 (XC); yining.wang@warrington.ufl.edu (YW); yuanz@illinois.edu (YZ)

Received: February 1, 2019
Revised: October 17, 2019
Accepted: November 15, 2020
Published Online in Articles in Advance: May 13, 2021

MSC2000 Subject Classification: Primary: 90B50; secondary: 90C59
OR/MS Subject Classification: Primary: analysis of algorithms; secondary: inventory/production

https://doi.org/10.1287/moor.2021.1133

Abstract. We study the dynamic assortment planning problem, where for each arriving customer, the seller offers an assortment of substitutable products and the customer makes the purchase among offered products according to an uncapacitated multinomial logit (MNL) model. Because all the utility parameters of the MNL model are unknown, the seller needs to simultaneously learn customers' choice behavior and make dynamic decisions on assortments based on the current knowledge. The goal of the seller is to maximize the expected revenue, or, equivalently, to minimize the expected regret. Although dynamic assortment planning problem has received an increasing attention in revenue management, most existing policies require the estimation of mean utility for each product and the final regret usually involves the number of products N. The optimal regret of the dynamic assortment planning problem under the most basic and popular choice model—the MNL model—is still open. By carefully analyzing a revenue potential function, we develop a trisection-based policy combined with adaptive confidence bound construction, which achieves an *item-independent* regret bound of $O(\sqrt{T})$, where T is the length of selling horizon. We further establish the matching lower bound result to show the optimality of our policy. There are two major advantages of the proposed policy. First, the regret of all our policies has no dependence on N. Second, our policies are almost assumption-free: there is no assumption on mean utility nor any "separability" condition on the expected revenues for different assortments. We also extend our trisection search algorithm to capacitated MNL models and obtain the optimal regret $\widetilde{O}(\sqrt{NT})$ (up to logrithmic factors) without any assumption on the mean utility parameters of items.

Funding: Support from the National Science Foundation [Grants IIS-1845444 and CCF-2006526] is gratefully acknowledged.
Supplemental Material: The online supplement is available at https://doi.org/10.1287/moor.2021.1133.

Keywords: dynamic assortment optimization • multinomial logit choice model • trisection algorithm • regret analysis

1. Introduction

Assortment planning has a wide range of applications in retailing and online advertising. Given a large number of substitutable products, the assortment planning problem refers to the selection of a subset of products (a.k.a., an assortment) offered to a customer such that the expected revenue is maximized. To model customers' choice behavior when facing a set of offered products, discrete choice models have been widely used, which capture demand for each product as a function of the entire assortment. One of the most popular discrete choice models is the *multinomial logit* (*MNL*) *model*, which naturally results from the random utility theory where a customer's preference of a product is represented by the mean utility of the product with a random factor (McFadden [26]). In many scenarios, customers' choice behavior (e.g., mean utilities of products) may not be given a priori and cannot be easily estimated well because of the insufficiency of historical data (e.g., fast fashion sales or online advertising). To address this challenge, dynamic assortment planning that simultaneously learns choice behavior and makes decisions on the assortment has received a lot of attention (Agarwal et al. [1, 2], Caro and Gallien [7], Rusmevichientong et al. [28], Saure and Zeevi [29]). More specifically, in a dynamic assortment planning problem, the seller offers an assortment to each arriving customer in a finite time horizon of length T. The goal of the seller is to maximize the cumulative expected revenue over T periods or, equivalently, to minimize the *regret*, which is defined as the gap between the expected revenue generated by the policy and the oracle expected revenue when the mean utility for each product is known a priori.

Despite a lot of research in the area of dynamic assortment planning under various choice models (see Section 2), the optimal policy for the most fundamental uncapacitated MNL model still remains open in the literature. A natural idea to tackle this problem is to conduct some form of maximum likelihood estimation (MLE) of mean utilities of different products on the fly, and then select the assortment that maximizes the expected revenue based on the current estimate of mean utilities. However, when the number of products N is large compared with the horizon length T, accurate estimation of mean utilities is extremely difficult, if not impossible, without additional assumptions. In terms of regret analysis, this approach usually incurs a regret that is polynomial in N, which is suboptimal according to our lower bound result (i.e., $\Omega(\sqrt{T})$). Therefore, the following question naturally arises: Can we design dynamic assortment policies without explicit estimation of mean utilities and achieve the optimal regret that is independent of N?

In this paper, we provide affirmative answers to this question under the most fundamental and popular uncapacitated multinominal logit model. As mentioned above, the estimation of utility parameters will be inaccurate when N is large and thus existing methods based on maximum likelihood estimation cannot be directly used. We design several new techniques to address this challenge. Under an MNL model, we leverage the structure of the optimal assortment in static problems and convert the problem into a *dynamic optimization* of a carefully designed *potential function*. In particular, the seminal results by Gallego et al. [19], Liu and van Ryzin [23], and Talluri and van Ryzin [30] show that the optimal assortment belongs to the set of revenue-ordered assortments. More precisely, assuming that N products are revenue ordered with the revenues $r_1 \geqslant r_2 \geqslant \cdots \geqslant r_N$, the optimal assortment must belong to the set $\{\{1\}, \{1,2\}, \ldots, \{1, \ldots, N\}\}$. Therefore, it suffices to consider only the following level sets of products: for each cutoff parameter $\theta \geqslant 0$, we define the *level set* to be the products whose revenue is greater than or equal to θ. Furthermore, motivated by Rusmevichientong et al. [28], we can define the potential function $F(\theta)$ to be the expected revenue when this level set is offered as an assortment.

To construct our policy, we first establish a set of important properties of the potential function $F(\theta)$, including (1) showing that the fixed point of $F(\theta)$ is the maximizer θ^* and leads to the optimal assortment, and (2) setting up a reference line and comparing $F(\theta)$ with the reference line to decide whether F is increasing or decreasing locally at θ. Based on these properties, we propose a trisection search policy that dynamically searches the maximizer θ^* of the potential function and achieves an optimal regret up to logarithmic factors in T. Then we further develop an approach with adaptive confidence levels to remove the logarithmic factor in T. The matching lower bound result has also been established, which shows the optimality of the proposed policy. By exploring the structure of the potential function, we no longer need to estimate N parameters of mean utilities; instead, we only estimate the expected revenue of level sets at a few cutoff points. Before we present an overview of our technical result in Section 1.1, we briefly highlight two important advantages of the proposed policies:

1. First, the regrets of our policies have no dependence on the number of products N. This property makes our result more favorable for scenarios when a large number of potential items is available, for example, online sales or online advertisement. And a key message behind this result is that by exploring the structure of the problem, the explicit estimation of utility parameters could be avoided in dynamic assortment planning.
2. Second, our policy is almost assumption-free: we require only that the revenue for each product is upper bounded by a constant and the knowledge of total selling horizon T, which is usually available in practice. We have no assumption on the mean utilities, for example, an assumption that no purchase is the most frequent outcome (Agarwal et al. [1]). Moreover, we do not have any "separation condition" on the expected revenue between a pair of candidate assortments, which has been assumed in the existing literature (Rusmevichientong et al. [28], Saure and Zeevi [29]).

Finally, we extend our proposed trisection algorithm to the capacitated setting, in which the sizes of assortments provided are constrained to be no larger than $K < N$. Our algorithm and analysis show an $\widetilde{O}(\sqrt{NT})$ regret that matches previous upper and lower bounds (Agarwal et al. [1, 2], Chen et al. [8]). The additional $\sqrt{N}$ dependency in the regret arises because of the need to estimate individual preference parameters v_i for each product and cannot be avoided in general (Chen and Wang [8]).

Our proposed algorithm is different from the ones in Agarwal et al. [1, 2] in that a trisection framework is employed on top of assortment optimization and exploration subroutines, whereas in Agarwal et al. [1, 2] the assortment decisions are made from upper confidence bounds (UCBs) or posterior distribution of individual product preference parameters. Please refer to Remark 1 for more detailed comparisons with the existing work.

1.1. Our Results and Techniques

The main contribution of this paper is an optimal characterization of the worst-case regret for dynamic assortment planning under the MNL model. More specifically, we have the following informal statement of the main results in this paper.

Theorem 1. (Informal). *There exists a policy whose worst-case regret over T time periods is upper bounded by $C_1\sqrt{T}$ for some universal constant $C_1 > 0$. Furthermore, there exists another universal constant $C_2 > 0$ such that no policy can achieve a worst-case regret smaller than $C_2\sqrt{T}$.*

To enable such an N-independent regret, we provide a refined analysis of a certain *unimodal* revenue potential function first studied in Rusmevichientong and Topaloglu [27] and consider a trisection algorithm on revenue levels, extending some ideas on unimodal bandits to either discrete or continuous arm domains (Agarwal et al. [3], Combes and Proutiere [17], Yu and Mannor [33]). An important challenge in our problem is that the revenue potential function (defined in Equation (5)) does not satisfy convexity or local Lipschitz growth, and therefore previous results on unimodal bandits cannot be directly applied (see the related work section, Section 2, for details). Moreover, it is a simple exercise that mere unimodality in multiarmed bandits cannot lead to regret smaller than $\sqrt{NT}$, because the worst-case constructions in the classical lower bound in multiarmed bandits are based on unimodal arms (see, e.g., Bubeck and Cesa-Bianchi [5], Bubeck et al. [6]).

To overcome these difficulties, we establish additional properties of the revenue potential function that are different from classical convexity or Lipschitz growth properties. In particular, we prove connections between the potential function and the straight line $F(\theta) = \theta$, which are then used as guidelines in our updated rules of trisection. Also, because the potential function behaves differently on $F(\theta) \leqslant \theta$ and $F(\theta) \geqslant \theta$, our trisection algorithm is *asymmetric* in the treatments of the two trisection midpoints, which is in contrast to previous trisection-based methods for unimodal bandits (Combes and Proutiere [17], Yu and Mannor [33]) that treat both trisection midpoints symmetrically.

We also remark that the trisection search policy leads to a regret $O(\sqrt{T \ln T})$, where the optimal regret should be $\Theta(\sqrt{T})$. The removal of additional $\ln T$ terms in dynamic assortment selection and unimodal bandit problems is quite nontrivial, which requires new technical development. In fact, most previous results on dynamic assortment selection (Agarwal et al. [1, 2], Rusmevichientong and Topaloglu [27]) and unimodal/convex bandits (Agarwal et al. [3], Combes and Proutiere [17], Yu and Mannor [33]) have additional $\ln T$ terms in regret upper bounds. The removal of this $\ln(T)$ term is achieved by using confidence bounds with adaptively chosen confidence levels corresponding to different amounts of data collected. At a higher level, our strategy shares a spirit similar to that of the minimax optimal strategy in the stochastic case (MOSS) algorithm for multiarmed bandits (Audibert and Bubeck [4]). On the other hand, the analysis is quite different from the analysis of the MOSS algorithm, involving new concentration inequalities and induction arguments tailored specifically to our model and proposed policy.

We note that a preliminary version of this paper appeared in the 2018 conference proceedings *Advances in Neural Information Processing Systems* (Wang et al. [32]). The journal version (1) develops a new adaptive trisection search policy, which gets rid of the logarithmic dependence on T as compared with the policies in the conference version (see Section 6); (2) extends the trisection search idea to capacitated MNL models and obtains the optimal regret bound (see Section 9); and (3) provides the corresponding numerical results in Section 8.

The rest of this paper is organized as follows. Section 2 discusses the related work from both revenue management and bandit learning fields. We introduce the model and notations in Section 3. We further define the revenue potential function and investigate its properties in Section 4. The policy and regret analysis will be provided in Section 5, and the lower bound results are developed in Section 7. In Section 8, we provide some simulation studies to illustrate the performance of the proposed policies. Extension to capacitated models is given in Section 9, and conclusion and discussions follow in Section 10. Some technical proofs are relegated to the online supplement.

2. Related Work

There are two lines of related work—dynamic assortment planning and unimodal bandits. We will provide a brief review of both fields and highlight some closely related work.

2.1. Dynamic Assortment Planning

Static assortment planning with known choice behavior has been an active research area since the seminal work by van Ryzin and Mahajan [31] and Mahajan and van Ryzin [25]. When the customer makes the choice according to the MNL model, Talluri and van Ryzin [30] and Gallego et al. [19] prove that the optimal assortment will belong to revenue-ordered assortments (see Lemma 1 in Section 4). An alternative proof is provided in Liu and van Ryzin [23]. This important structural result enables efficient computation of static assortment planning under the

MNL model, which reduces the number of candidate assortments from 2^N to N and will also be used in our policy development.

Motivated by large-scale online retailing, researchers have started to relax the assumption on prior knowledge of customers' choice behavior. The question of *dynamic* optimization of assortments, where the mean utilities of products are unknown and have to be learnt on the fly, has received increasing attention in both the machine learning and operations management communities (Agarwal et al. [1, 2], Caro and Gallien [7], Rusmevichientong et al. [28], Saure and Zeevi [29]). Motivated by fast-fashion retailing, the work by Caro and Gallien [7] was the first to study the dynamic assortment planning problem, which assumes that the demands for products are independent of each other. The work Rusmevichientong et al. [28] and Saure and Zeevi [29] incorporate MNL choice models into dynamic assortment planning and formulate the problem into a online regret minimization problem.

The work by Rusmevichientong et al. [28] is closely related to our paper, and analyzes the same revenue potential function and proposes a golden ratio search algorithm based on the unimodal property of the potential function. However, using only the unimodal property leads a regret bound involving $\ln(N)$ (Rusmevichientong et al. [28]), which is not N-independent. Moreover, the golden ratio search algorithm imposes a strong "separability assumption" (see Rusmevichientong et al. [28, equation (8)]), which assumes a constant gap between the expected revenues of any pair of candidate assortments, which may fail when the number of items N is large. In this work, we relax the gap assumption and also remove the additional $\ln N$ dependency by a more refined analysis of properties of the revenue potential function.

Our paper is also closely related to recent works by Agarwal et al. [1, 2]. These works develop variants of UCB and Thompson sampling type methods for *capacitated* MNL assortment models, where the size of each assortment is not allowed to exceed a prespecified parameter K. Here the capacity limit K is usually much smaller than N. For the capacitated MNL model, the paper by Chen and Wang [8] further establishes a lower bound result, which shows an $\Omega(\sqrt{NT})$ regret lower bound exists provided that $K \leqslant N/4$. By comparing this result with our result described in Theorem 1, it is interesting to see that the regret behavior in capacitated and uncapacitated MNL models is significantly different (see Table 1). Whereas the dependence on N in regret is unavoidable in the capacitated case, this paper shows that it can be got rid of in the uncapacitated case. We remove this dependence on N by designing a novel policy that does not explicitly estimate utility parameters.

2.2. Unimodal Bandits

Another relevant line of research is *unimodal bandits* (Agarwal et al. [3], Combes and Proutiere [17], Cope [18], Yu and Mannor [33]), in which discrete or continuous multiarmed bandit problems are considered with additional unimodality constraints on the means of the arms. Apart from unimodality, additional structures such as "inverse Lipschitz continuity" (e.g., $|\mu(i)-\mu(j)| \geqslant L|i-j|$ for some constant L, where $\mu(i)$ denotes the mean reward of the i th arm) or convexity are imposed to ensure smaller regret compared with unstructured multiarmed bandits. However, both conditions fail to hold for the revenue potential function arising from uncapacitated MNL-based assortment planning problems. In addition, under the gap-free setting where an $O(\sqrt{T})$ regret is to be expected, most previous works have additional $\ln T$ terms in their regret upper bounds (except for the work of Cope [18], which introduces additional strong regularity conditions on the underlying functions). In Cohen-Addad and Kanade [16], a more general problem of optimizing piecewise-constant function is considered, without assuming a unimodal structure of the function. Consequently, a weaker $\widetilde{O}(T^{2/3})$ regret is derived.

2.3. Other Related Works

The works of Cohen et al. [15], Leme and Schneider [22], and Lobel et al. [24] considered sequential *contextual* decision-making problems, with applications in healthcare management and dynamic pricing. They consider

Table 1. Summary of the state-of-the-art worst-case regrets for dynamic assortment planning under uncapacitated MNL and capacitated MNL models, where T and N denote the length of the horizon and the number of products, respectively. We also provide the reference for each result, either the theorem number (when the result is first derived in this paper) or the reference. Here, the tilde-O notation, $\widetilde{O}$, is used as a variant of the standard big-O notation but hides logarithmic factors.

Worst-case regret	Uncapacitated MNL	Capacitated MNL ($K \leqslant N/4$)
Upper bound	$O(\sqrt{T})$ (Theorem 3)	$\widetilde{O}(\sqrt{NT}+N)$ (Agarwal et al. [1, 2])
Lower bound	$\Omega(\sqrt{T})$ (Theorem 4)	$\Omega(\sqrt{NT})$ (Chen and Wang [8])

sequential cuts to a convex body with hyperplanes and employ novel volume- and surface-based arguments to upper bound the number of cuts, eventually leading to a generalization of the one-dimensional binary search idea to multiple dimensions. On the other hand, although we adopt only a single-dimensional searching scheme, our algorithm fully explores and leverages the structure of the multinomial logit models.

In addition to MNL models, there are some recent works studying dynamic assortment under more complicated choice models, such as nested logit models (Chen et al. [11]) and contextual MNL models (Chen et al. [9, 10], Cheung and Simchi-Levi [13]). We also note that to highlight our key idea and focus on the balance between information collection and revenue maximization, we study stylized dynamic assortment planning problems following some existing literature (Agarwal et al. [1, 2], Rusmevichientong et al. [28], Saure and Zeevi [29]) that ignores operational considerations such as price decisions and inventory replenishment. It is also worthwhile noting that there are recent works studying pricing decisions and inventory planning under the context of assortment optimization (Chen et al. [12], Cheung et al. [14], Golrezaei et al. [20]). In particular, the works by Golrezaei et al. [20] and Chen et al. [12] study assortment optimization under inventory constraints with known choice functions (no learning component), and thus adopt the competitive ratio (instead of regret) as the performance measure. In contrast, our main focus is to effectively learn underlying utility parameters in an MNL choice model. The recent work by Cheung et al. [14] studies the resource allocation problem, where the context vectors and arrival sequence are adversarially chosen but the combination of context vector and action is drawn from a fixed unknown distribution.

3. Model Specification

Let $\mathcal{N}$ be a finite set of all products/items with $|\mathcal{N}| = N$, and each item $i \in \mathcal{N}$ is associated with a revenue parameter $r_i > 0$ and a utility parameter (a.k.a., preference parameter) $v_i > 0$.[1] Throughout this paper, we conveniently label all items in $\mathcal{N}$ as $1, 2, \ldots, N$. The revenue parameters $r_1, \ldots, r_N$ are known to the retailer, who has full knowledge of each items' price/cost, whereas the utility parameters $v_1, \ldots, v_N$ are unknown. Let $\mathbb{S} = 2^{\mathcal{N}}$ be the set of all possible assortments. At every time t, a retailer picks an assortment $S_t \in \mathbb{S}$ ($S_t \neq \emptyset$) and observes a purchasing action $i_t \in S_t \cup \{0\}$, where $i_t = 0$ means no purchase occurs at time t. If a purchasing action is made (i.e., $i_t \neq 0$), the corresponding revenue r_{i_t} is collected. It is worthy noting that because items are substitutable, a typical setting of assortment planning usually restricts each purchase to be a single item.

The distribution of i_t is modeled by the following MNL model:

$$\Pr\left[i_t = j\right] = \begin{cases} v_j \big/ \left(1 + \sum_{i \in S_t} v_i\right) & j \in S_t; \\ 1 \big/ \left(1 + \sum_{i \in S_t} v_i\right) & j = 0. \end{cases} \tag{1}$$

Define also $R(S_t)$ as the *expected* revenue by supplementing S_t to a customer; more specifically,

$$R(S_t) := \sum_{j \in S_t} \Pr\left[i_t = j\right] \cdot r_j = \frac{\sum_{j \in S_t} r_j v_j}{1 + \sum_{j \in S_t} v_j}. \tag{2}$$

For normalization purposes, the utility parameter for the "no-purchase" action is assumed to be $v_0 = 1$. Apart from that, the rest of the preference parameters $\{v_i\}_{i=1}^N$ are *unknown* to the retailer and have to be either explicitly or implicitly learnt from customers' purchasing actions $\{i_t\}_{t=1}^T$.

The retailer's objective is to maximize the expected revenue over the T time periods. Such an objective is equivalent to the "regret minimization," in which the retailer's assortment sequence is compared against the optimal assortment. More specifically, the goal of the retailer is to design a policy π that generates $\{S_t\}_{t=1}^T$ to minimize the following cumulative regret:

$$\mathrm{Regret}\left(\{S_t\}_{t=1}^T\right) := \sum_{t=1}^T R(S^*) - \mathbb{E}^\pi\left[R(S_t)\right], \quad \text{where } S^* \in \arg\max_{S \in \mathbb{S}} R(S). \tag{3}$$

Here, $R(S_t) = \mathbb{E}[r_{i_t} | S_t]$ is the expected revenue the retailer collects on assortment S_t. For notational convenience, we define $r_0 = 0$ corresponding to the no-purchase action.

Finally, throughout this paper, we make only the following standard assumption on the revenue parameters (see, e.g., Agarwal et al. [2, theorem 1]):

Assumption 1. $r_\infty := max_{i \in \mathcal{N}} r_i \leqslant 1$.

We note that upper bound on the maximum revenue is assumed to be one without loss of generality, because one can always normalize the revenues.

4. The Revenue Potential Function and Its Properties

The set $\mathbb{S}$ consists of 2^N different assortments, which poses a significant challenge on both regret minimization (treating each assortment in $\mathbb{S}$ independently results in exponentially large regret) and computation (as it is intractable to enumerate all assortments in $\mathbb{S}$). To address the challenge, we can reduce the number of candidate assortments in $\mathbb{S}$ by constraining such assortment selections to "level sets." In particular, for a given real number $\theta \geqslant 0$, define the θ-level set to be

$$\mathcal{L}_\theta(\mathcal{N}) := \{i \in \mathcal{N} : r_i \geqslant \theta\},$$

that is, as all items whose revenues are not smaller than θ. For notational simplicity, we will use $\mathcal{L}_\theta$ (omitting $\mathcal{N}$ in the parentheses) when the context is clear. Furthermore, let

$$\mathbb{P} := \left\{\mathcal{L}_\theta(\mathcal{N}) : \theta \geqslant 0\right\} \subseteq \mathbb{S} \tag{4}$$

be the class of all candidate assortments in $\mathbb{S}$ that can be expressed as level sets. It is easy to verify that $|\mathbb{P}| \leqslant N$, which is significantly smaller than $|\mathbb{S}| = 2^N$.

It is well known that the optimal expected revenue for the static assortment optimization problem will remain the same when reducing the candidate assortments from $\mathbb{S}$ to $\mathbb{P}$. More precisely, the following lemma is a classical result in revenue management (Gallego et al. [19], Liu and van Ryzin [23], Talluri and van Ryzin [30]) that shows the optimal expected revenue can be achieved by only considering the restricted level-set class $\mathbb{P}$ under the MNL model.

Lemma 1. (Gallego et al. [19], Liu and van Ryzin [23], Talluri and van Ryzin [30]). *Under the MNL model, there exists an subset $S^* \subseteq \mathcal{N}$ such that $R(S^*) = \max_{S \in \mathbb{S}} R(S) = \max_{S \in \mathbb{P}} R(S)$.*

In other words, Lemma 1 suggests that it suffices to consider level-set-type assortments $\mathcal{L}_\theta$ and to find $\theta \in [0,1]$ that gives rises to the largest $R(\mathcal{L}_\theta)$.

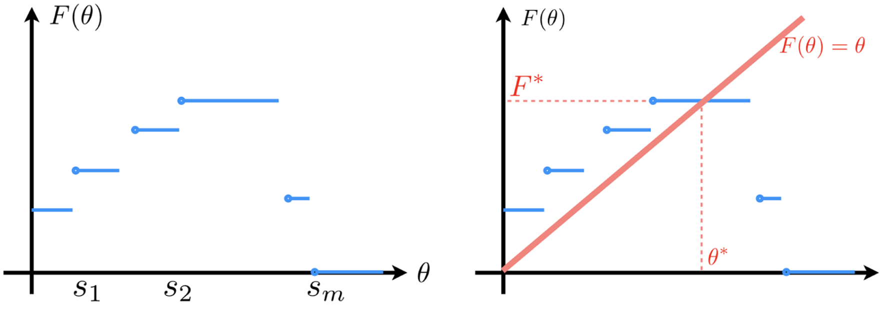
This motivates the following "potential" function, which takes a revenue threshold θ as input and outputs the expected revenue of its corresponding level-set assortments:

$$\text{The revenue potential function:} \quad F(\theta) := R(\mathcal{L}_\theta), \ \theta \in [0,1]. \tag{5}$$

Intuitively, $F(\theta)$ is the expected revenue obtained by providing the assortment consisting of all items whose revenues exceed or are equal to θ. The potential function plays a central role in the development of our dynamic trisection search algorithm and item-independent regret bounds. The similar idea of studying the expected revenue of revenue-ordered items was also considered in Rusmevichientong and Topaloglu [27]. But we will derive a more comprehensive list of properties of the potential function F to facilitate our algorithmic development and analysis. The derived properties in this section could also be potentially useful for solving other assortment planning problems under the MNL model.

Because item revenues r_i are discrete, F is a piecewise-constant function, as illustrated in the left panel of Figure 1, where $\mathcal{S} = \{s_1, \ldots, s_m\}$ are the changing points of F. More specifically, we have the following proposition, and its verification is easy from the definition and the discretized nature of F.

Figure 1. (Color online) Illustration of the potential function $F(\theta)$, the important quantities F^* and θ^*, and their properties.



Proposition 1. *There exists $c_0, \ldots, c_m \geqslant 0$ satisfying $c_i \neq c_{i+1}$ for all $i = 0, \ldots, m-1$, and $\mathcal{S} = \{s_1, \ldots, s_m\} \subseteq \{r_i\}_{i=1}^N$, such that*

$$F(\theta) = c_0 \cdot \mathbf{1}[\theta \leqslant s_1] + \sum_{i=1}^{m-1} c_i \cdot \mathbf{1}[s_i < \theta \leqslant s_{i+1}] + c_m \cdot \mathbf{1}[\theta > s_m], \tag{6}$$

where $c_m = 0$.

Define $F^* := \max_{0 \leqslant i \leqslant m} c_i = \sup_{\theta \geqslant 0} F(\theta)$ as the maximum value of F. By Lemma 1, we have the following corollary saying that F^* equals the expected revenue of the optimal assortment.

Corollary 1. $F^* = R(S^*)$.

We further establish some more refined structural properties of F. For notational simplicity, let $F(x^+) := \lim_{y \to x^+} F(y)$ and $F(x^-) := \lim_{y \to x^-} F(y)$.

Lemma 2. *There exists $\theta^* > 0$ such that $\theta^* = F(\theta^*) = F^*$.*

Lemma 3. *For any $\theta \geqslant \theta^*$, we have $F(\theta) \leqslant \theta$ and $F(\theta) \geqslant F(\theta^+)$.*

Lemma 4. *For any $\theta \leqslant \theta^*$, we have $F(\theta) \geqslant \theta$ and $F(\theta) \leqslant F(\theta^+)$.*

The proofs of the above lemmas are given in the supplemental material. Lemmas 2–4 provide a complete picture of the structure of the potential function F, and most importantly the relationship between F and the central straight line $F(\theta) = \theta$, as depicted in the right panel of Figure 1. In particular, F intersects with the $y = x$ line at θ^* that attains the maximum function value F^*, and monotonically decreases as one moves away from θ^*, meaning that F is *unimodal*. Furthermore, Lemmas 3 and 4 show that (1) F is left continuous, and (2) F^* lies below the $y = x$ line to the right of θ^* and above the $y = x$ line to the left of θ^*. This helps us judge the positioning of a particular revenue level θ by simply comparing the expected revenue of $R(\mathcal{L}_\theta)$ with θ itself, motivating an asymmetric trisection algorithm, which we describe in the next section. It is worthwhile to note that the unimodality and properties in Lemmas 2–4 have already been sufficient for the development of the optimal trisection algorithm. The piecewise constant property is not directly used in the algorithmic development because each piece can be very small, which makes this property difficult to utilize.

5. Trisection and Regret Analysis

We propose an algorithm based on trisections of the potential function F in order to locate level θ^* at which the maximum expected revenue $F^* = F(\theta^*)$ is attained. Our algorithm avoids explicitly estimating individual items' mean utilities $\{v_i\}_{i=1}^N$, and subsequently yields a regret independent of the number of items N. We first give a simplified algorithm (pseudocode description in Algorithm 1) with an additional $O(\sqrt{\ln T})$ term in the regret upper bound and outline its proofs. We further show how the additional logarithmic dependency on T can be removed by using more advanced techniques.

Algorithm 1 (The Trisection Algorithm)

```
Input: revenue parameters $r_1, \ldots, r_n \in [0,1]$, time horizon $T$
Output: sequence of assortment selections $S_1, S_2, \ldots, S_T \subseteq \mathcal{N}$
Initialization: $a_0 = 0, b_0 = 1$;
for $\tau = 0, 1, \ldots$ do
    $x_\tau = \frac{2}{3}a_\tau + \frac{1}{3}b_\tau, y_\tau = \frac{1}{3}a_\tau + \frac{2}{3}b_\tau$ ;                ▷ trisection
    $\ell_0(x_\tau) = \ell_0(y_\tau) = 0, u_0(x_\tau) = u_0(y_\tau) = 1$ ;     ▷ initialization of confidence intervals
    $\rho_0(x_\tau) = \rho_0(y_\tau) = 0$ ;                      ▷ initialization of accumulated rewards
    for $t = 1$ to $16\lceil (y_\tau - x_\tau)^{-2} \ln(T) \rceil$ † do
        if $\ell_{t-1}(y_\tau) \leqslant y_\tau \leqslant u_{t-1}(y_\tau)$ then
            $\rho_t(y_\tau), \ell_t(y_\tau), u_t(y_\tau) \leftarrow$ EXPLORE$(y_\tau, t, 1/T^2)$
        else
            $\rho_t(y_\tau), \ell_t(y_\tau), u_t(y_\tau) \leftarrow \rho_{t-1}(y_\tau), \ell_{t-1}(y_\tau), u_{t-1}(y_\tau)$
        Exploit the left end point $a_\tau$: pick assortment $S = \mathcal{L}_{a_\tau}$ ;
    ▷ Update trisection parameters
    if $u_t(y_\tau) < y_\tau$ then $a_{\tau+1} = a_\tau, b_{\tau+1} = y_\tau$
    else $a_{\tau+1} = x_\tau, b_{\tau+1} = b_\tau$
```

† Stop whenever the maximum number of iterations T is reached.

Algorithm 2 (Explore Subroutine: Exploring a Certain Revenue Level θ)

```
Input: revenue level $\theta$, time $t$, confidence level $\delta$
Output: accumulated revenue $\rho_t(\theta)$, confidence intervals $\ell_t(\theta)$ and $u_t(\theta)$
Pick assortment $S = \mathcal{L}_\theta(\mathcal{N})$ and observe purchasing action $j \in S \cup \{0\}$;
Update accumulated reward: $\rho_t(\theta) = \rho_{t-1}(\theta) + r_j$;        $\triangleright\ r_0 := 0$
Update confidence intervals: $[\ell_t(\theta), u_t(\theta)] = \frac{\rho_t(\theta)}{t} \pm \sqrt{\frac{\ln(1/\delta)}{2t}}$.
```

To assist with readability, below we list notations used in the algorithm description together with their meanings:

- a_τ and b_τ are left and right boundaries that contain θ^*; it is guaranteed that $a_\tau \leqslant \theta^* \leqslant b_\tau$ with high probability, and the regret incurred on failure events is strictly controlled.
- x_τ and y_τ are trisection points; x_τ is closer to a_τ and y_τ is closer to b_τ.
- $\ell_t(y_\tau)$ and $u_t(y_\tau)$ are lower and upper confidence bounds for $F(y_\tau)$ established at iteration t; it is guaranteed that $\ell_t(y_\tau) \leqslant F(y_\tau) \leqslant u_t(y_\tau)$ with high probability, and the regret incurred on failure events is strictly controlled.
- $\rho_t(y_\tau)$ is the accumulated reward from exploring level set $\mathcal{L}_{y_\tau}$ up to iteration t.

With these notations in place, we provide a detailed description of Algorithm 1 to facilitate the understanding. The algorithm operates in epochs (outer iterations) $\tau = 1, 2, \ldots$ until a total of T assortment selections are made. The objective of each outer iteration τ is to find the relative position between trisection points (x_τ, y_τ) and the "reference" location θ^*, after which the algorithm either moves a_τ to x_τ or b_τ to y_τ, effectively shrinking the length of the interval $[a_\tau, b_\tau]$ that contains θ^* to two-thirds. Furthermore, to avoid a large cumulative regret, the level set corresponding to the left end point a_τ is exploited in each time period within the epoch τ to offset potentially large regret incurred by exploring y_τ.

In Step 8 of Algorithm 1, lower and upper confidence bounds $[\ell_t(y_\tau), u_t(y_\tau)]$ for $F(y_\tau)$ are constructed using concentration inequalities (e.g., Hoeffding's [21] inequality).

These confidence bounds are updated until the relationship between y_τ and $F(y_\tau)$ is clear, or a prespecified number of inner iterations for outer iteration τ has been reached (set to $n_\tau := \lceil 16(y_\tau - x_\tau)^{-2} \ln(T^2) \rceil$ in Step 6). Algorithm 2 gives detailed descriptions on how such confidence intervals are built, based on repeated exploration of level set $\mathcal{L}_{y_\tau}$.

After sufficiently many explorations of $\mathcal{L}_{y_\tau}$, a decision is made on whether to advance the left boundary (i.e., $a_{\tau+1} \leftarrow x_\tau$) or the right boundary (i.e., $b_{\tau+1} \leftarrow y_\tau$). Below we give high-level intuitions on how such decisions are made, with rigorous justifications presented later as part of the proof of the main regret theorem for Algorithm 1:

1. If there is sufficient evidence that $F(y_\tau) < y_\tau$ (e.g., $u_t(y_\tau) < y_\tau$), then y_τ must be to the right of θ^* (i.e., $y_\tau \geqslant \theta^*$) because of Lemma 3. Therefore, we will shrink the value of right boundary by setting $b_{\tau+1} \leftarrow y_\tau$.

2. On the other hand, when $u_t(y_\tau) \geqslant y_\tau$, we can conclude that x_τ *must be to the left of* θ^* (i.e., $x_\tau \leqslant \theta^*$). We show this by contradiction. Assuming that $x_\tau > \theta^*$, because y_τ is always greater than x_τ (and thus $y_\tau > \theta^*$) and the gap between y_τ and $F(y_\tau)$ is at least $y_\tau - x_\tau$,[2] the gap will be detected by the confidence bounds, and thus we will have $u_t(y_\tau) < y_\tau$ with high probability. This leads to a contradiction. Because x_τ is to the left of θ^*, we should increase the value of the left boundary by setting $a_{\tau+1} \leftarrow x_\tau$.

We remark that the lengths of epochs $O(\ln(T)(b_\tau - a_\tau))$ increase as the lower and upper bounds of θ^*, a_τ and b_τ, get close to each other. Additionally, the $O(\ln T)$ term reflects the union bounds of concentration of confidence intervals used in the algorithm.

The following theorem is our main upper bound result for the (worst-case) regret incurred by Algorithm 1.

Theorem 2. (Regret Upper Bound). *There exists a universal constant $C_1 > 0$ such that for all parameters $\{v_i\}_{i=1}^N$ and $\{r_i\}_{i=1}^N$ satisfying $r_i \in [0,1]$, the regret incurred by Algorithm 1 satisfies*

$$\mathrm{Regret}\left(\{S_t\}_{t=1}^T\right) = \mathbb{E}\left[\sum_{t=1}^T R(S^*) - R(S_t)\right] \leqslant C_1 \sqrt{T \ln T}. \tag{7}$$

5.1. Proof Sketch

In the rest of the section, we sketch key steps and lemmas toward the proof of Theorem 2. The proofs of technical lemmas are provided in the supplemental material. We first state a simple lemma showing that the confidence bounds $\ell_t(y_\tau)$ and $u_t(y_\tau)$ constructed in Algorithm 1 contain $F(y_\tau)$ with high probability.

Lemma 5. *With probability $1 - O(T^{-1})$, $\ell_t(\theta) \leqslant F(\theta) \leqslant u_t(\theta)$ for all t.*

The following lemma, based on properties of the potential function F and Lemma 5, establishes that (with high probability) the shrinkage of a_τ or b_τ is "consistent"; that is, θ^* is always contained in $[a_\tau, b_\tau]$. Its proof is based on the intuitive two-case analysis discussed before Theorem 2 and will be provided in the supplemental material.

Lemma 6. *With probability $1 - O(T^{-1})$, $a_\tau \leqslant \theta^* \leqslant b_\tau$ for all $\tau = 1, 2, \ldots, \tau_0$, where τ_0 is the last outer iteration of Algorithm 1.*

Using Lemmas 5 and 6, we are able to prove the following lemma that upper bounds the regret incurred at each outer iteration τ using the distance between the trisection points x_τ and y_τ.

Lemma 7. *For $\tau = 0, 1, \ldots$, let $\mathcal{T}(\tau)$ denote the set of all indices of inner iterations at outer iteration τ. Conditioned on the success events in Lemma 5 and 6, it holds that*

$$\mathbb{E} \sum_{t \in \mathcal{T}(\tau)} R(S^*) - R(S_t) \leqslant 206 \varepsilon_\tau^{-1} \ln T, \tag{8}$$

where $\varepsilon_\tau = y_\tau - x_\tau$.

We are now ready to prove Theorem 2.

Proof of Theorem 2. Recall the definition that $\varepsilon_\tau = y_\tau - x_\tau$ for outer iterations $\tau = 0, 1, \ldots$. Because after each outer iteration we either set $b_{\tau+1} = y_\tau$ or $a_{\tau+1} = x_\tau$, it is easy to verify that $\varepsilon_\tau = (2/3) \cdot \varepsilon_{\tau-1}$. Subsequently, invoking Lemma 6 and using summation of geometric series, we have

$$\begin{aligned} \mathbb{E}\sum_{t=1}^{T} R(S^*) - R(S_t) &\leqslant 206 \sum_{\tau=0}^{\tau_0} \varepsilon_\tau^{-1} \ln T \leqslant 206 \sum_{\tau=0}^{\tau_0} (2/3)^{-\tau} \ln T \\ &\leqslant 206 \times \frac{(3/2)^{\tau_0} - 1}{(3/2) - 1} \times \ln T \leqslant 412 \times \varepsilon_{\tau_0}^{-1} \ln T, \end{aligned} \tag{9}$$

where τ_0 is the total number of outer iterations executed by Algorithm 1. On the other hand, because at each outer iteration τ the revenue level a_τ is exploited exactly $n_\tau = 16\lceil (y_\tau - x_\tau)^{-2} \ln(T^2) \rceil$ times, we have

$$T \geqslant n_{\tau_0} \geqslant 32 \varepsilon_{\tau_0}^{-2} \ln T. \tag{10}$$

Combining Equations (9) and (10), we conclude that $\cdots$ $\text{Regret}(\{S_t\}_{t=1}^T) \leqslant 75\sqrt{T \ln T} = O(\sqrt{T \ln T})$. □

6. Improved Regret with Adaptive Confidence Levels

In this section, we consider a variant of Algorithm 1 that achieves an improved regret of $O(\sqrt{T})$. The key idea is to use an adaptive allocation of confidence levels, by allowing larger failure probability as more data are collected. This is because later failures result in smaller accumulated regret. Such a strategy is motivated by the MOSS algorithm (Audibert and Bubeck [4]) for multiarmed bandits. However, our analysis is quite different from Audibert and Bubeck's [4], involving new concentration inequalities and induction arguments tailored specifically to our model and proposed policy.

We start with a new uniform concentration inequality for adaptively chosen confidence levels.

Lemma 8. *Let $X_1, \ldots, X_L$ be independent and identically distributed random variables with mean μ and satisfy $a \leqslant X_i \leqslant b$ almost surely for all $\ell \in [L]$. For any $\delta \in (0, 1]$, it holds that*

$$\Pr\left[\forall \ell \in [L], \left|\frac{1}{\ell}\sum_{i=1}^{\ell} X_i - \mu\right| \leqslant \sqrt{\frac{2(b-a)^2 \ln(8/(\delta\ell))}{\ell}}\right] \geqslant 1 - L\delta. \tag{11}$$

The proof of Lemma 8 is placed in the supplemental material, based on a careful doubling argument with Hoeffding's [21] maximal inequality. Compared with the classical Hoeffding inequality with the union bound, one notable difference is the increasing "failure probability" as ℓ increases (effectively $\ell\delta$ in $\sqrt{\frac{2\ln(8/(\delta\ell))(b-a)^2}{\ell}}$ instead of δ). This allows the confidence intervals to be much shorter for large ℓ.

With Lemma 8, we are ready to describe the variant of Algorithm 1, which attains the tight regret bound. Most steps in Algorithms 1 and 2 remain unchanged, and the changes are summarized below:

- Step 3 in Algorithm 2 is replaced with

$$[\ell_t(\theta), u_t(\theta)] = \frac{\rho_t(\theta)}{t} \pm \sqrt{\frac{2\ln[8/(\delta t)]}{t}}. \tag{12}$$

- In Step 8 of Algorithm 1 is replaced with $\mathrm{Explore}(y_\tau, t, 1/T)$; correspondingly, the number of inner iterations is changed to $n_\tau = 8\lceil (y_\tau - x_\tau)^{-2} \ln(8T(y_\tau - x_\tau)^2)\rceil$.

The first change for improving the regret is the way confidence intervals $[\ell_t(\theta), u_t(\theta)]$ of $F(\theta)$ are constructed. Instead of using fixed confidence level $1/T^2$ as in the baseline policy, in the revised policy, *varying* confidence levels are employed, with "effective" failure probabilities increasing as the algorithm collects more data.

We also remark that similar confidence parameter choices were also adopted in Audibert and Bubeck [4] to remove additional $\ln(T)$ factors in multiarmed bandit problems.

The following theorem shows that the algorithm variant presented above achieves an asymptotic regret of $O(\sqrt{T})$, considerably improving Theorem 2 with an $O(\sqrt{T\ln T})$ regret bound. Its proof is rather technical and involves careful analysis of failure events at each outer iteration τ of the trisection algorithm. To highlight the main idea behind the proof, we provide a sketch of the proof in Section 6.1 and defer the entire proof of Theorem 3 to the online supplement.

Theorem 3. (Rate-Optimal Regret Upper Bound). *There exists a universal constant $C_2 > 0$ such that for all parameters $\{v_i\}_{i=1}^N$ and $\{r_i\}_{i=1}^N$ satisfying $r_i \in [0,1]$, the regret incurred by the variant of Algorithm 1 described above satisfies*

$$\mathrm{Regret}\Big(\{S_t\}_{t=1}^T\Big) = \mathbb{E}\sum_{t=1}^{T} R(S^*) - R(S_t) \leqslant C_2\sqrt{T}. \tag{13}$$

Comparing Theorem 3 with Theorem 2, we observe that the additional $O(\sqrt{\ln T})$ term is shaved off. Such improvement is made possible mainly by the adaptive choices of confidence levels (more specifically, $O(1/\delta t)$ instead of $O(1/T^2)$) in Equation (12), which, coupled with a more refined uniform concentration result (Lemma 8) and more careful inductive/recurrence analysis (Lemmas 9, 10, and 11 in Section 6.1), delivers the desired improvement in regret bounds. The refined uniform concentration result (Lemma 8), which plays a central role in the improvement of an $O(\sqrt{\ln T})$ term, is proved using the Hoeffding's maximal inequality coupled with a "doubling" type argument.

We also remark that our proposed algorithm can be made "anytime" (i.e., without prior knowledge of the time horizon T) by using the standard technique of doubling. More specifically, consider geometrically increasing "metaepochs" $j = 0, 1, 2, \ldots$, with metaepoch j consisting of $T_j = 2^j$ consecutive time periods. Within metaepoch j, the proposed Algorithm 1 is run from scratch with $T = T_j$ as the time horizon. By Theorem 3, the cumulative regret of such an algorithm is $O(\sum_{j=0}^{j_0}\sqrt{T_j}) = O(\sum_{j=0}^{j_0} 2^{j/2}) = O(2^{j_0/2})$, where j_0 is the last metaepoch before the algorithm reaches T time periods. On the other hand, we have that $2^{j_0-1} \leqslant T$. Consequently, the cumulative regret of such a doubling-based algorithm is upper bounded by $O(\sqrt{T})$.

6.1. Proof Sketch

We sketch key steps and lemmas toward the proof of Theorem 2. The proofs of technical lemmas are provided in the supplemental material. We first define some notations. Let $\tau = 0, 1, \ldots$ be the number of outer iterations in Algorithm 1, $\varepsilon_\tau = (y_\tau - x_\tau)$ be the distance between the two trisection points at outer iteration τ, and $n_\tau = 8\lceil \varepsilon_\tau^{-2}\ln(8T\varepsilon_\tau^2)\rceil$ be the prespecified number of inner iterations. Recall also that $\theta^* = F(\theta^*) = F^*$ is the optimal revenue value suggested by Lemma 2.

Define the following three disjoint events that partition the entire probabilistic space:

- Event $\mathcal{E}_1(\tau)$: $\theta^* < a_\tau < b_\tau$;
- Event $\mathcal{E}_2(\tau)$: $a_\tau \leqslant \theta^* \leqslant b_\tau$;
- Event $\mathcal{E}_3(\tau)$: $a_\tau < b_\tau < \theta^*$.

Let $\tau_0 \in \mathbb{N}$ be the last outer iteration in Algorithm 1. Let also $\mathcal{T}(\tau) \subseteq [T]$ be the indices of inner iterations in outer iteration τ, satisfying $|\mathcal{T}(\tau)| \leqslant 2n_\tau$ almost surely. For $\omega \in \{1, 2, 3\}$, $\tau \in \mathbb{N}$, and $\alpha, \beta \in \mathbb{R}^+$, define

$$\psi_\tau^\omega(\alpha, \beta) := \mathbb{E}\left[\sum_{\tau'=\tau}^{\tau_0}\sum_{t\in\mathcal{T}(\tau')} R(S^*) - R(S_t)\,\middle|\, \mathcal{E}_\omega(\tau), |a_\tau - \theta^*| = \alpha, |F(a_\tau) - a_\tau| = \beta\right]. \tag{14}$$

Intuitively, $\psi_\tau^\omega(\alpha,\beta)$ is the expected regret Algorithm 1 incurs for outer iterations $\tau,\tau+1,\ldots,\tau_0$, conditioned on the event $\mathcal{E}_\omega(\tau)$ and other boundary conditions at the left margin a_τ.

The following three lemmas are the central steps in our proof, which establish recurrence relationships among $\psi_\tau^\omega(\alpha,\beta)$, for $\omega\in\{1,2,3\}$. The proofs are technically involved and, as we have mentioned, deferred to the supplemental material. To simplify notations, we write $a_n \lesssim b_n$ or $b_n \gtrsim a_n$ if there exists a *universal* constant $C>0$ such that $|a_n|\leqslant C|b_n|$ for all $n\in\mathbb{N}$.

Lemma 9. (Regret in Case 1). $\psi_\tau^1(\alpha,\beta)\leqslant\beta T+\sum_{\tau'=\tau+1}^{\tau_0}\sup_{\Delta>\varepsilon_{\tau'}}\Delta T\exp\{-n_\tau\Delta^2\}+O(\varepsilon_{\tau'}^{-1}\ln(T\varepsilon_{\tau'}^2))$.

Lemma 10. (Regret in Case 2). $\psi_\tau^2(\alpha,\beta)=O(\varepsilon_\tau^{-1}\ln(T\varepsilon_\tau^2))+\psi_{\tau+1}^2(\alpha'_2,\beta'_2)+\psi_{\tau+1}^3(\alpha'_3,\beta'_3)\cdot O(\ln(T\varepsilon_\tau^2)/(T\varepsilon_\tau^2))+\sup_{\Delta>\varepsilon_\tau}\psi_{\tau+1}^1(\alpha'_1,\beta'_1(\Delta))\exp\{-n_\tau\Delta_\tau^2\}$ *for parameters* $\alpha'_1,\beta'_1(\Delta),\alpha'_2,\beta'_2,\alpha'_3,\beta'_3$ *that satisfy* $\beta'_1(\Delta)\leqslant\Delta$ *and* $\alpha'_3\leqslant 3\varepsilon_\tau$.

Lemma 11. (Regret in Case 3). $\psi_\tau^3(\alpha,\beta)\leqslant\alpha T$.

We are now ready to complete the proof of Theorem 3 by combining Lemmas 10, 9 and 11.

Proof of Theorem 3. We first get a cleaning expression of $\psi_\tau^1(\alpha,\beta)$ using Lemma 9. First note that $\Delta\mapsto\Delta\exp\{-n_\tau\Delta^2\}$ attains its maximum on $\Delta>0$ at $\Delta=\sqrt{1/2n_\tau}$. Also note that $n_\tau=\lceil 8\varepsilon_\tau^{-2}\ln(8T\varepsilon_\tau^2)\rceil$, and therefore $\sqrt{1/2n_\tau}\leqslant\varepsilon_\tau$. Subsequently,

$$\begin{aligned}\sum_{\tau'=\tau}^{\tau_0}\sup_{\Delta>\varepsilon_\tau}\Delta T\exp\{-n_\tau\Delta^2\}&\leqslant\sum_{\tau'=\tau}^{\tau_0}\varepsilon_\tau T\exp\{-n_\tau\varepsilon_\tau^2\}\leqslant\sum_{\tau'=\tau}^{\tau_0}\varepsilon_\tau T\exp\left\{-\ln\left(T\varepsilon_\tau^2\right)\right\}\\&\leqslant\sum_{\tau'=\tau}^{\tau_0}\varepsilon_\tau^{-1}=O\left(\varepsilon_{\tau_0}^{-1}\right),\end{aligned}\tag{15}$$

where the last asymptotic holds because $\{\varepsilon_\tau\}$ forms a geometric series. Subsequently,

$$\psi_\tau^1(\alpha,\beta)\leqslant\beta T+\sum_{\tau'=\tau}^{\tau_0}O\left(\varepsilon_{\tau'}^{-1}\ln\left(T\varepsilon_\tau^2\right)\right).\tag{16}$$

It remains to bound the summation term on the right-hand side of the above inequality. Let $s_{\tau'}=\varepsilon_{\tau'}^{-1}\ln(T\varepsilon_{\tau'}^2)=\rho^{-\tau'}\ln(T\rho^{2\tau'})$, where $\rho=2/3$. We then have $s_{\tau'}=\rho^{\tau_0-\tau'}[1+\ln\rho^{-2(\tau_0-\tau')}]s_{\tau_0}\leqslant 2(\tau_0-\tau'+1)\rho^{\tau_0-\tau'}\ln(1/\rho)$ for all $\tau'\leqslant\tau_0$. Subsequently,

$$\sum_{\tau'=\tau}^{\tau_0}s_{\tau'}\leqslant\sum_{\tau'=0}^{\tau_0}2(\tau_0-\tau'+1)\rho^{\tau_0-\tau'}\ln(1/\rho)\cdot s_{\tau_0}\leqslant C\cdot s_{\tau_0}.\tag{17}$$

Therefore,

$$\psi_\tau^1(\alpha,\beta)\leqslant\beta T+O\left(\varepsilon_{\tau_0}^{-1}\ln\left(T\varepsilon_{\tau_0}^2\right)\right).\tag{18}$$

We are now ready to derive the final regret upper bound by analyzing $\psi_0^2(\alpha,\beta)$, because the event $\mathcal{E}_2(0)$ always holds because $0\leqslant\theta^*\leqslant 1$. Applying Lemma 10 with Lemma 11 and Equation (18), we have for all $\tau\in\{0,1,\ldots,\tau_0\}$ that

$$\begin{aligned}\psi_\tau^2(\alpha,\beta)&\leqslant\psi_{\tau+1}^2(\alpha'_2,\beta'_2)+O\left(\varepsilon_\tau^{-1}\ln\left(T\varepsilon_\tau^2\right)\right)+O(\varepsilon_\tau T)\cdot\frac{\ln\left(T\varepsilon_\tau^2\right)}{T\varepsilon_\tau^2}\\&\quad+\sup_{\Delta>\varepsilon_\tau}\left(\Delta T+O\left(\varepsilon_{\tau_0}^{-1}\ln\left(T\varepsilon_{\tau_0}^2\right)\right)\right)\exp\{-n_\tau\Delta^2\}\\&\leqslant\psi_{\tau+1}^2(\alpha'_2,\beta'_2)+O\left(\varepsilon_\tau^{-1}\ln\left(T\varepsilon_\tau^2\right)\right)+\sup_{\Delta>\varepsilon_\tau}\Delta T\exp\{-n_\tau\Delta^2\}\\&\quad+O\left(\varepsilon_{\tau_0}^{-1}\ln\left(T\varepsilon_{\tau_0}^2\right)\right)\cdot\exp\{-n_\tau\varepsilon_\tau^2\}.\end{aligned}\tag{19}$$

Using the same analysis as in Equation (15), we know $\sup_{\Delta>\varepsilon_\tau}\Delta T\exp\{-n_\tau\Delta^2\}=O(\varepsilon_\tau^{-1})$ and $\exp\{-n_\tau\varepsilon_\tau^2\}\leqslant 1/(T\varepsilon_\tau^2)$. Subsequently, summing all terms $\tau=0,1,\ldots,\tau_0$ together, we have

$$\begin{aligned}\psi_0^2(\alpha,\beta)&\leqslant\sum_{\tau=0}^{\tau_0}O\left(\varepsilon_\tau^{-1}\ln\left(T\varepsilon_\tau^2\right)\right)+O\left(\varepsilon_{\tau_0}^{-1}\ln\left(T\varepsilon_{\tau_0}^2\right)\right)\cdot\frac{1}{T\varepsilon_\tau^2}\\&\lesssim\varepsilon_{\tau_0}^{-1}\ln\left(T\varepsilon_{\tau_0}^2\right)\cdot\left(1+1/\left(T\varepsilon_{\tau_0}^2\right)\right).\end{aligned}\tag{20}$$

Finally, note that $n_{\tau_0} \gtrsim \varepsilon_{\tau_0}^{-2}$ and $n_{\tau_0} \leqslant T$, implying that $\varepsilon_{\tau_0} \gtrsim \sqrt{1/T}$. Plugging the lower bound on ε_{τ_0} into the above inequality, we have $\psi_0^2(\alpha,\beta) \lesssim \sqrt{T}$, which completes the proof of Theorem 3. □

7. Lower Bound

We prove the following theorem showing that no policy can achieve an accumulated regret smaller than $\Omega(\sqrt{T})$ in the worst case.

Theorem 4. (Regret Lower Bound). *Let N and T be the number of items and the time horizon that can be arbitrary. There exists revenue parameters* $r_1, \ldots, r_N \in [0,1]$ *such that for any policy* π,

$$\sup_{v_1,\ldots,v_N \geqslant 0} \operatorname{Regret}\left(\{S_t\}_{t=1}^T\right) \geqslant \frac{\sqrt{T}}{384}. \tag{21}$$

Theorem 4 shows that our regret upper bounds in Theorems 2 and 3 are tight up to $\sqrt{\ln T}$ factors and numerical constants.

7.1. Proof Sketch of Theorem 4

We next give a sketch of the proof of Theorem 4. Because of space constraints, we present only an outline of the proof and defer proofs of all technical lemmas to the online supplement. Without loss of generality, we assume the number of items N is even, because an odd number of items can be easily handled by setting $r_N = v_N = 0$ for the last item.

We first describe the underlying parameter values on which our lower bound proof is built. Fix revenue parameters $\{r_i\}_{i=1}^N$ as $r_i = 1$ for i odd and $r_i = 1/2$ for i even, which are known a priori. We then consider two constructions of the unknown utility parameters $\{v_i\}_{i=1}^N$:

$$P_0\!: \quad v_i = \left(1 - 1/64\sqrt{T}\right)/(0.5N) \quad \text{for } i \text{ odd}, \; v_i = 1 \text{ for } i \text{ even};$$
$$P_1\!: \quad v_i = \left(1 + 1/64\sqrt{T}\right)/(0.5N) \quad \text{for } i \text{ odd}, \; v_i = 1 \text{ for } i \text{ even}.$$

We note that P_0 and P_1 also give the probability distributions that characterize the customer random purchasing actions, and thus we will use $P_j[A]$ to denote the probability of event A under the utility parameters specified by P_j for $j \in \{0,1\}$.

The first lemma shows that there does not exist estimators that can identify P_0 from P_1 with high probability with only T observations of random purchasing actions. Its proof involves careful calculation of the Kullback–Leibler divergence between the two hypothesized distributions and subsequent application of Le Cam's lemma to the testing question between P_0 and P_1.

Lemma 12. *For any estimator* $\hat{\psi} \in \{0,1\}$ *whose inputs are T random purchasing actions* $i_1, \ldots, i_T$, *it holds that* $\max_{j\in\{0,1\}} P_j[\hat{\psi} \neq j] \geqslant 1/3$.

On the other hand, the following lemma shows that if the policy π can achieve a small regret under both P_0 and P_1, then one can construct an estimator based on π such that with large probability, the estimator can distinguish between P_0 and P_1 from observed customers' purchasing actions.

Lemma 13. *Suppose a policy* π *satisfies* $\operatorname{Regret}(\{S_t\}_{t=1}^T) < \sqrt{T}/10^4$ *for both* P_0 *and* P_1. *Then there exists an estimator* $\hat{\psi} \in \{0,1\}$ *such that* $P_j[\hat{\psi} \neq j] \leqslant 1/4$ *for both* $j = 0$ *and* $j = 1$.

Lemma 13 is proved by explicitly constructing a classifier (tester) $\hat{\psi}$ from any sequence of low regret. In particular, for any assortment sequence $\{S_t\}_{t=1}^T$, we construct $\hat{\psi}$ as $\hat{\psi} = 0$ if $\frac{1}{T}\frac{2}{N}\sum_{t=1}^T \sum_{j=0}^{\lfloor (N-1)/2 \rfloor} \mathbf{1}\{2j+2 \in S_t\} \geqslant 1/2$ and $\hat{\psi} = 1$ otherwise. Using Markov's inequality and the construction of $\{r_i, v_i\}$, it can be shown that if $\operatorname{Regret}(\{S_t\}_{t=1}^T) > \sqrt{T}/10^4$, then $\hat{\psi}$ is a good tester with small testing error. Detailed calculations and the complete proof are deferred to the online supplement.

Combining Lemmas 12 and 13, we proved our lower bound result in Theorem 4.

8. Simulation Results

We present numerical results of our proposed trisection (and its improved variant) algorithm and compare their performance with several competitors on synthetic data.

8.1. Experimental Setup

We generate each of the revenue parameters $\{r_i\}_{i=1}^{N}$ independently and identically from the uniform distribution on $[0.4, 0.5]$. For the preference parameters $\{v_i\}_{i=1}^{N}$, they are generated independently and identically from the uniform distribution on $[10/\mathrm{N}, 20/\mathrm{N}]$, where N is the total number of items available.

To motivate our parameter setting, consider the following three types of assortments: the "single assortment" $S = \{i\}$ for some $i \in \mathcal{N}$, the "full assortment" $S = \{1, 2, \ldots, N\}$, and the "appropriate" assortment $S = \{i \in \mathcal{N} : r_i \geqslant 0.42\}$. For the single assortment $S = \{i\}$, because the preference parameter for each item is rather small ($v_i \leqslant 20/N$), no single assortment can produce an expected revenue exceeding $0.5 \times (20/N)/(1 + 20/N) = 10/(20 + N)$. For the full assortment $S = \{1, 2, \ldots, N\}$, because $\sum_{i=1}^{N} r_i v_i \xrightarrow{p} 0.45 \times 15/N \times N = 6.75$ and $\sum_{i=1}^{N} v_i \xrightarrow{p} 15$ by the law of large numbers, the expected revenue of S is around $6.75/(1 + 15) = 0.422$. Finally, for the appropriate assortment $S = \{i \in \mathcal{N} : r_i \geqslant 0.42\}$, we have $\sum_{i\in S} r_i v_i \xrightarrow{p} 0.46 \times 15/N \times 0.8N = 5.52$ and $\sum_{i\in S} v_i \xrightarrow{p} 15/N \times 0.8N = 12$. Therefore, the expected revenue of S is around $5.52/(1 + 12) = 0.425 > 0.422$. The above discussion shows that a revenue threshold $r^* \in (0.4, 0.5)$ is mandatory to extract a portion of the items $\{i \in \mathcal{N} : r_i \geqslant r^*\}$ that attain the optimal expected revenue, which is highly nontrivial for a dynamic assortment selection algorithm to identify.

8.2. Comparative Methods

Our trisection algorithm with $O(\sqrt{T}\ln T)$ regret is denoted by trisec, and its improved adaptive variant (with regret $O(\sqrt{T})$) is denoted by adap-trisec. The other methods we compare against include the upper confidence bound algorithm of Agarwal et al. [2] (denoted by ucb), the Thompson sampling algorithm of Agarwal et al. [1] (denoted by thompson), and the golden ratio search algorithm of Rusmevichientong et al. [28] (denoted by grs). Note that both ucb and thompson proposed in Agarwal et al. [1, 2] were initially designed for the *capacitated* MNL model, in which the number of items each assortment contains is restricted to be at most $K < N$. In our experiments, we operate both ucb and thompson under the uncapacitated setting, simply by removing the constraint set when performing each assortment optimization.

In our improved adaptive trisection algorithm (adap-trisec), we replace the $\sqrt{\frac{2\ln(8/(\delta\ell))}{\ell}}$ confidence interval configuration with $\sqrt{\frac{0.1\ln(8/(\delta\ell))}{\ell}}$. We observe that a smaller constant value leads to better empirical performance. We clarify that the 0.1 numerical constant is not "fine tuned" as in, for example, cross-validation practices. Instead, we simply choose a reasonably small numerical constant (without any tuning), and other smaller constants lead to similar performance.

Another modification is the grs algorithm: in Rusmevichientong et al. [28], the number of exploration iterations is set to $34\ln(2N)/\beta^2$, where $\beta = \min_{j\neq j'}|R(\mathcal{L}_{r_j}) - R(\mathcal{L}_{r_{j'}})|$, which is inappropriate for our "gap-free" synthetic setting in which $\beta = 0$. Instead, we use the common choice of $\sqrt{T}$ exploration iterations in typical gap-independent bandit problems for grs.

8.3. Results

In Table 2, we report the mean and maximum regret from 20 independent runs of each algorithm on our synthetic data, with different settings of N (number of items) and T (time horizon length). We observe that as the number of items (N) becomes large, our algorithms (Trisec and Adap-Trisec) achieve smaller mean and maximum regret compared with their competitors, and Adap-Trisec consistently outperforms Trisec in all settings. Unlike Ucb or

Table 2. Average (mean) and worst-case (max) regret of our trisection (Trisec) and adaptive trisection (Adap-Trisec) algorithms and their competitors on synthetic data, where N is the number of items, and T is the time horizon.

	Ucb		Thompson		Grs		Trisec		Adap-Trisec	
(N, T)	Mean	Max	Mean	Max	Mean	Max	Mean	Max	Mean	Max
(100, 500)	34.9	38.1	1.28	2.97	10.9	22.4	7.68	7.68	1.99	1.99
(250, 500)	54.3	56.2	2.81	4.95	7.93	34.2	7.57	7.57	2.23	2.23
(500, 500)	73.4	75.5	4.90	4.95	7.02	43.4	7.43	7.43	2.23	2.23
(1,000, 500)	90.3	93.5	8.17	10.7	5.34	45.1	7.44	7.44	2.25	2.25
(100, 1,000)	73.1	78.2	1.36	2.79	139.9	175.0	8.69	8.69	3.90	3.90
(250, 1,000)	113.7	119.3	3.36	5.17	90.1	110.1	8.69	8.69	4.13	4.14
(500, 1,000)	136.8	140.3	5.65	7.64	65.7	113.9	9.38	9.38	3.80	3.80
(1,000, 1,000)	160.8	165.4	9.31	12.4	8.43	22.8	9.77	9.77	3.97	3.97

Thompson, whose regret depends polynomially on N, our Trisec and Adap-Trisec algorithms have no dependency on N, and hence their regret does not increase with N. Moreover, the separate exploration and exploitation structure in Grs makes its performance somewhat unstable, which leads to a larger gap between mean and maximum regrets.

9. Generalization to Capacitated Models

In this section, we show how our trisection-based method could be generalized to *capacitated* models, achieving performance guarantees comparable to, and actually even more general than, existing results in the literature.

In a capacitated assortment optimization model, a *capacity parameter* $K < N$ is prespecified, and capacity constraints $|S_t| \leqslant K$ are imposed for all assortments $\{S_t\}_{t=1}^{T}$ supplied throughout the T time periods. The other parts of the model remain the same as specified in Section 3. The regret is then defined as

$$\mathrm{Regret}\Big(\{S_t\}_{t=1}^{T}\Big) = \sum_{t=1}^{T} R(S^*) - R(S_t), \quad \text{where } S^* = \arg\max_{S \subseteq \mathcal{N}, |S| \leqslant K} R(S).$$

9.1. A Revised Potential Function and Its Properties

Because of the capacity constraint $|S_t| \leqslant K$, the potential function $F(\cdot)$ defined in Section 4 is no longer sufficient, as the optimal assortment might not simply contain products with the highest revenue parameters. This also means that the level sets $\mathcal{L}_\theta(\mathcal{N})$ might not be optimal. Instead, under the capacitated setting, for every $\theta \in [0,1]$, we define

$$\mathcal{M}_{\theta,K}(\mathcal{N}) := \arg\max_{S \subseteq \mathcal{N}, |S| \leqslant K} \sum_{i \in S} (r_i - \theta) v_i.$$

More specifically, $\mathcal{M}_{\theta,K}(\mathcal{N})$ consists of at most K items from $\mathcal{N}$ with the largest nonzero values of $(r_i - \theta)v_i$. The following result is well known in the literature of static optimization of assortments with capacity constraints (Rusmevichientong et al. [28]), which we also prove in the supplemental material for completeness.

Lemma 14. *There exists* $\theta \in [0,1]$ *such that* $R(S^*) = \max_{S \subseteq \mathcal{N}, |S| \leqslant K} R(S) = R(\mathcal{M}_{\theta,K}(\mathcal{N}))$.

A modified potential function $G : [0,1] \to \mathbb{R}^+$ is then defined as

$$*G(\theta) := R\big(\mathcal{M}_{\theta,K}(\mathcal{N})\big), \quad \theta \in [0,1].$$

Note that graphical illustration of $G(\theta)$ is virtually the same as the plots in Figure 1 (piecewise constant and the same unimodality properties), with the exception for more than N discontinuity points.

In the rest of this subsection, we establish several important properties of G that will be used in algorithm development and analysis later. The proofs of these properties are relegated to the supplemental material.

Lemma 15. *There exists a unique* $\theta^* \in [0,1]$ *such that the following hold:*

1. $\theta^* = G(\theta^*) = \max_{\theta \in [0,1]} G(\theta) = \max_{S \subseteq \mathcal{N}, |S| \leqslant K} R(S)$.
2. *For all* $\theta \leqslant \theta^*$, $G(\theta) \geqslant \theta$.
3. *For all* $\theta \geqslant \theta^*$, $G(\theta) \leqslant \theta$.

9.2. A Trisection Algorithm for the Capacitated Model

The potential function G constructed in the previous subsection has some notable differences from the potential function F for uncapacitated models. For example, F is unimodal and piecewise constant, whereas G is monotonically decreasing and not a constant on most of its domain. Nevertheless, an algorithm similar to the trisection method presented in Algorithm 1 can still be designed for the capacitated setting and potential function G.

Algorithm 3 gives a pseudocode description of the proposed method for the capacitated model. Compared with Algorithm 1 for the uncapacitated model, there are two significant differences:

1. In the uncapacitated model, we obtain estimates and confidence intervals directly on the values of $F(y_\tau)$, and later compare them with the $F(\theta) = \theta$ reference line. In capacitated models, however, because of the complexity of the underlying model, we can no longer estimate the values of $G(y_\tau)$ accurately; instead, we focus on testing the relationship between $G(y_\tau)$ and y_τ directly, which suffices for the sake of the trisection algorithms and is much easier.
2. In the uncapacitated model, estimates of $F(y_\tau)$ can be obtained by repetitively offering the same level-set assortment $\mathcal{L}_{y_\tau}(\mathcal{N})$; in capacitated models, on the other hand, the estimation of $G(y_\tau)$ becomes much more involved because $\mathcal{M}_{\theta,K}(\mathcal{N})$ cannot be directly computed without knowledge of $\{v_i\}_{i \in \mathcal{N}}$. To overcome this issue, we use an UCB-type algorithm to estimate v_i and (approximately) compute $\mathcal{M}_{\theta,K}(\mathcal{N})$ at the same time.

Algorithm 3 (The Trisection Algorithm for the Capacitated Model)

```
Input: revenue parameters $r_1, \dots, r_n \in [0,1]$, time horizon $T$, capacity constraint $K$,
       numerical constants $\gamma_1, \gamma_2, \gamma_3 > 0$
Output: sequence of assortment selections $S_1, S_2, \dots, S_T \subseteq \mathcal{N}, |S_t| \leqslant K$
Initialization: $a_0 = 0$, $b_0 = 1$;
for $\tau = 0, 1, \dots$ do
    $x_\tau = \frac{2}{3}a_\tau + \frac{1}{3}b_\tau$, $y_\tau = \frac{1}{3}a_\tau + \frac{2}{3}b_\tau$;                ▷ trisection
    $n_i^{\mathrm{y}} = m_i^{\mathrm{y}} = 0$ for all $i \in \mathcal{N}$;                ▷ initialization of cumulative statistics for $y_\tau$
    $n_i^{\mathrm{a}} = m_i^{\mathrm{a}} = 0$ for all $i \in \mathcal{N}$;                ▷ initialization of cumulative statistics for $a_\tau$
    $t^{\mathrm{y}} = t^{\mathrm{a}} = 0, s = 0$;                ▷ initialization of time period counters
    $t_{\max} = \max\{3\gamma_1(y_\tau - x_\tau)^{-1}, 9\gamma_2^2(y_\tau - x_\tau)^{-2}\} \times N \ln^3(NT+1)$;
    $\rho^{\mathrm{y}} = \rho^{\mathrm{a}} = 0$;                ▷ accumulated rewards
    while $\max\{t^{\mathrm{y}}, t^{\mathrm{a}}\} < t_{\max}$ † do
        $\hat{S}^{\mathrm{y}} \leftarrow$ OptimizeCap$(K, y_\tau, \{n_i^{\mathrm{y}}, m_i^{\mathrm{y}}\})$;
        $\hat{S}^{\mathrm{a}} \leftarrow$ OptimizeCap$(K, a_\tau, \{n_i^{\mathrm{a}}, m_i^{\mathrm{a}}\})$;
        $\bar{\rho}^{\mathrm{y}} \leftarrow \min\left\{1, \frac{\rho^{\mathrm{y}}}{t^{\mathrm{y}}} + \gamma_1 \frac{N\ln^3(NT+1)}{t^{\mathrm{y}}} + \gamma_2\sqrt{\frac{N\ln^3(NT+1)}{t^{\mathrm{y}}}} + \gamma_3\sqrt{\frac{\ln T}{t^{\mathrm{y}}}}\right\}$;
        if $\bar{\rho}^{\mathrm{y}} \geqslant y_\tau$ then
            $\{\hat{n}^{\mathrm{y}}, \hat{m}_i{}^{\mathrm{y}}\}, \Delta t^{\mathrm{y}}, \Delta\rho^{\mathrm{y}} \leftarrow$ ExploreCap$(\{n_i^{\mathrm{y}}, m_i^{\mathrm{y}}\}, \hat{S}^{\mathrm{y}}, t_{\max} - t^{\mathrm{y}})$;
            $t^{\mathrm{y}} \leftarrow t^{\mathrm{y}} + \Delta t^{\mathrm{y}}$, $\rho^{\mathrm{y}} \leftarrow \rho^{\mathrm{y}} + \Delta\rho^{\mathrm{y}}$;
            $\{\hat{n}^{\mathrm{a}}, \hat{m}_i{}^{\mathrm{a}}\} \leftarrow \{\hat{n}^{\mathrm{a}}, \hat{m}_i{}^{\mathrm{a}}\}$;
        else
            $\{\hat{n}^{\mathrm{a}}, \hat{m}_i{}^{\mathrm{a}}\}, \Delta t^{\mathrm{a}}, \Delta\rho^{\mathrm{a}} \leftarrow$ ExploreCap$(\{n_i^{\mathrm{a}}, m_i^{\mathrm{a}}\}, \hat{S}^{\mathrm{a}}, t_{\max} - t^{\mathrm{a}})$;
            $t^{\mathrm{a}} \leftarrow t^{\mathrm{a}} + \Delta t^{\mathrm{a}}$;
            $\{\hat{n}^{\mathrm{y}}, \hat{m}_i{}^{\mathrm{y}}\} \leftarrow \{\hat{n}^{\mathrm{y}}, \hat{m}_i{}^{\mathrm{y}}\}$;
        Update counter: $s \leftarrow s + 1$;
    ▷ Update trisection ExploreCap parameters
    if $\bar{\rho}^{\mathrm{y}} < y_\tau$ then $a_{\tau+1} = a_\tau$, $b_{\tau+1} = y_\tau$ else $a_{\tau+1} = x_\tau$, $b_{\tau+1} = b_\tau$
† Stop whenever the maximum number of iterations $T$ is reached.
```

The exploration in capacitated models is accomplished by two subroutines. The first subroutine, OptimizeCap, outputs an assortment with size at most K that approximates $\mathcal{M}_{\theta,K}(\mathcal{N})$ in terms of the objective $\sum_{i\in S}(r_i - \theta)v_i$, for $\theta \in \{a_\tau, y_\tau\}$. The second subroutine, ExploreCap, explores the approximately optimal assortment $\hat{S}^{\mathrm{y}}{}_{,}(s)$ or $\hat{S}^{\mathrm{a}}{}_{,}(s)$ and collects data (i.e., purchasing activities of arriving customers) in order to refine the estimates of v_i or $G(\theta)$. Note that instead of providing the assortment for one time period, the ExploreCap routine provides the assortment repetitively until a no-purchase event occurs, which is inspired by the MNL-bandit (Agarwal et al. [1, 2]) approaches in the previous literature.

To further give insights into the design of Algorithm 3, we first explain some important notations in the pseudocode description:

- y and a indicate whether the statistic/estimate is for the right trisection point y_τ or the left end point a_τ. With an y superscript, the related notation is for y_τ, whereas with an a superscript, the related notation is for a_τ.
- m_i^{y} and m_i^{a} are the number of times product i is *offered* in an assortment explored by subroutine ExploreCap.
- n_i^{y} and n_i^{a} are the number of times product i is *purchased* within subroutine ExploreCap.
- ρ^{y} and $\bar{\rho}^{\mathrm{y}}$ are the estimates of the potential function $G(y_\tau)$ and are used to determine the relative positions of (x_τ, y_τ) and θ^*.

At a higher level, Algorithm 3 uses the same trisection framework as in the uncapacitated case to locate the optimal revenue level θ^* (at which $G(\theta^*) = \theta^*$) by comparing (estimates) of $G(y_\tau)$ with y_τ. The OptimizeCap and ExploreCap subroutines, additionally, provide more refined controls over which assortments are to be selected (at certain revenue levels y_τ) and how these assortments are explored, in order to control regret and better estimate $G(y_\tau)$.

In the rest of this subsection, we give more details of the OptimizeCap and ExploreCap subroutines and establish several theoretical properties for them.

9.2.1. The OptimizeCap Subroutine. A pseudocode description of the OptimizeCap subroutine is given in Algorithm 4.

Step 3 of Algorithm 4 can be computed efficiently by sorting the items according to their score $(r_i-\theta)\bar{v}_i$ and choosing the top K items with the largest nonnegative scores. The time complexity of the algorithm is $O(N\log N)$ for each execution of Step 3.

Additionally, the original combinatorial optimization question $\max_{S\subseteq[N],|S|\leqslant K}R(S)$ can be viewed from a joint optimization perspective. More specifically, the question $\max_{S\subseteq[N],|S|\leqslant K}R(S)$ can be equivalently written as $\max_{\theta\in[0,1]}\max_{S\subseteq[N],|S|\leqslant K}\varphi(\theta,S)$, where $\varphi(\theta,S)=\min\{\theta,\sum_{i\in S}(r_i-\theta)v_i\}$. From this perspective, the variable θ is similar to the "dual variable" of S from an optimization perspective, and the $\varphi(\theta,S)$ joint function is similar to the Lagrangian multipliers.

Algorithm 4 (The OptimizeCap Subroutine)

```
Input: capacity constraint $K$, parameter $\theta$, statistics $\{n_i,m_i\}_{i\in\mathcal{N}}$
Output: assortment $\hat{S}$
1  For $i\in\mathcal{N}$ compute $\hat{v}_i=n_i/m_i$ and $\delta_i=\min\left\{T,\max\{\sqrt{\hat{v}_i},\hat{v}_i\}\sqrt{\frac{24\ln(NT+1)}{m_i}+\frac{48\ln(NT+1)}{m_i}}\right\}*$;
2  Define $\bar{v}_i=\min\{T,\hat{v}_i+\delta_i\}$;
3  Compute $\hat{S}\leftarrow\arg\max_{S\subseteq\mathcal{N},|S|\leqslant K}\sum_{i\in S}(r_i-\theta)\bar{v}_i$;
* If $m_i<48\ln(NT+1)$ then set $\hat{v}_i=0$ and $\delta_i=T$.
```

Algorithm 5 (The ExploreCap Subroutine)

```
Input: statistics $\{n_i,m_i\}_{i\in\mathcal{N}}$, candidate assortment $\hat{S}$, maximum time periods $t_{\max}$
Output: updated statistics $\{\widetilde{n}_i,\widetilde{m}_i\}_{i\in\mathcal{N}}$, $\Delta t$, $\Delta\rho$
Offer assortment $\hat{S}$ repetitively until a customer makes no purchases or a total of $t_{\max}$ time periods are reached;
$\Delta t\leftarrow$ the number of time periods $\hat{S}$ is offered;
$\Delta\rho\leftarrow$ the total rewards collected in the offering time periods;
$\widetilde{m}_i\leftarrow m_i+1$ if $i\in\hat{S}$ and $\widetilde{m}_i\leftarrow m_i$ if $i\notin\hat{S}$;
$\widetilde{n}_i\leftarrow\widetilde{n}_i+$ no. of time periods in which $i$ is purchased if $i\in\hat{S}$ and $\widetilde{n}_i\leftarrow n_i$ if $i\notin\hat{S}$;
```

The following lemma shows that, with high probability, the estimated preference parameters $\{\hat{v}_i\}_{i\in\mathcal{N}}$ are very close to the true values $\{v_i\}_{i\in\mathcal{N}}$, and furthermore, the constructed upper estimates $\{\bar{v}_i\}_{i\in\mathcal{N}}$ are valid with high probability. Its proof is deferred to the supplemental material.

Lemma 16. *With probability $1-O(T^{-1})$ uniformly over all calls of* OptimizeCap *and all $i\in\mathcal{N}$, it holds that $v_i\leqslant\bar{v}_i$ and $|\bar{v}_i-v_i|\leqslant 2\delta_i\leqslant 2\min\left\{T,2\max\{\sqrt{v_i},v_i\}\sqrt{\frac{24\ln(NT+1)}{m_i}+\frac{82\ln(NT+1)}{m_i}}\right\}$.*

9.2.2. The ExploreCap Subroutine. A pseudocode description of the OptimizeCap subroutine is given in Algorithm 5.

In the rest of this subsection, recall that $\hat{S}$ is obtained by finding the top K products associated with the largest nonnegative values of $(r_i-\theta)\bar{v}_i$, where $\bar{v}_i$ are upper estimates of v_i obtained in the OptimizeCap subroutine, and θ, being either y_τ or a_τ, is the partition point on which ExploreCap is invoked.

Lemma 17. *With probability $1-O(T^{-1})$ uniformly over all calls of* ExploreCap, *it holds that $\Delta t\leqslant(1+2\ln(T))\left(1+\sum_{i\in\hat{S}}v_i\right)$.*

Lemma 18. *Suppose $G(\theta)\geqslant\theta$, for $\theta\in\{a_\tau,y_\tau\}$. Conditioned on the events of* Lemmas 16 *and* 17, *it holds at any time period in τ that $\theta t-\rho\leqslant 800\,N\ln^3(NT+1)+4\sqrt{6\,N\,t\ln^3(NT+1)}+2\sqrt{t\ln T}$ with probability $1-O(T^{-1})$, where $t\in\{t^{\mathrm{y}},t^{\mathrm{a}}\}$ and $\rho\in\{\rho^{\mathrm{y}},\rho^{\mathrm{a}}\}$ correspond to $\theta\in\{y_\tau,a_\tau\}$.*

The proofs of both lemmas are deferred to the supplemental material. Lemma 17 upper bounds the number of time periods elapsed in each call of ExploreCap, using the preference parameters v_i of products offered in the assortment $\hat{S}$. This lemma shows that lengthy exploration of suboptimal assortments $\hat{S}$ is rare, thereby upper bounding the regret accumulated in the ExploreCap subroutine. Lemma 18 shows that, to the left of the "critical point" θ^* (at which $G(\theta^*)=\theta^*$), the deviation between $\bar{\rho}^{\mathrm{y}}$ in Algorithm 3 and θ can be upper bounded. With such upper bounds, the proposed algorithm can safely detect the relative position of x_τ, y_τ with respect to θ^* by comparing (an upper bound of) $\bar{\rho}^y$ with θ.

9.2.3. Regret Analysis of Algorithm 3.

The following theorem is the main regret upper bound on Algorithm 3.

Theorem 5. *Suppose Algorithm 3 is run with $\gamma_1 = 800$, $\gamma_2 = 4\sqrt{6}$, and $\gamma_3 = 2$. Suppose also that $N \leqslant T$. Then its cumulative regret can be upper bounded by*

$$\mathbb{E}\sum_{t=1}^{T} R(S^*) - R(S_t) \leqslant \widetilde{O}(\sqrt{NT}),$$

where in $\widetilde{O}(\cdot)$, we omit universal constants and poly-logarithmic terms in N and T.

We note that the choices of constants $\gamma_1, \gamma_2, \gamma_3$ in Theorem 5 are solely for the convenience of our technical proofs. In practice, they can be set at much smaller, reasonable levels, or be selected using historical data and cross-validation.

Remark 1. The $\widetilde{O}(\sqrt{NT})$ upper bound in Theorem 5 matches the results in existing works (Agarwal et al. [1, 2]) up to logarithmic terms in N and T. An improvement of Theorem 5 over Agarwal et al. [1, 2] is that no assumptions on $\{v_i\}_{i\in\mathcal{N}}$ are imposed. In contrast, previous works assume $v_i \leqslant 1$, essentially meaning that the probability of no purchase is always the largest regardless of the products provided in an assortment. We note that Agarwal et al. [2] is able to remove the this assumption but incurs a larger regret bound. Our algorithm and analysis therefore are more general and practical, being suitable to scenarios in which very popular products/items exist whose v_i far exceeds the no-purchase utility one.

We also note that compared with the UCB (Agarwal et al. [2]) and Thompson sampling (Agarwal et al. [1]) algorithms, our capacitated trisection algorithm uses a more computational efficient optimization step. Indeed, the optimization step in our algorithm is described at Line 3 of Algorithm 4, in which one only needs to sort the items according to the values $(r_i - \theta)\bar{v}_i$ and pick the largest K nonnegative ones. In comparison, both the UCB and Thompson sampling algorithms have to solve the static assortment optimization problem.

9.3. Numerical Study

For any given N (number of items), we generate the revenue and preference parameters of the N items from the same probability distribution described in Section 8. We then test our capacitated trisection algorithm on the synthetic data with different settings of N, K (the capacity limit of an assortment), and T (the time horizon). In Table 3, we report the mean and maximum regret from 20 independent runs of each parameter setting and compare the performance with that of the UCB and Thompson sampling algorithms. From the numerical results, we see that our capacitated trisection algorithm performs comparably with the UCB algorithm when T is relatively large. This is expected because our method also utilizes the idea of upper confidence bounds in the construction. Also, from the literature, it is known that Thompson sampling usually achieves better empirical performance even when the regret bounds are of the same order. Moreover, as we have explained in Remark 1, our algorithm is computationally more efficient, and runs about 10 times faster than the UCB and Thompson sampling algorithms with the parameter settings reported in Table 3.

Table 3. Average (mean) and worst-case (max) regret of the UCB and Thompson sampling algorithms our capacitated trisection algorithm (Cap Trisec) on synthetic data, where N is the number of items, K is the capacity limit of an assortment, and T is the time horizon.

	Ucb		Thompson		Cap Trisec	
(N, K, T)	Mean	Max	Mean	Max	Mean	Max
(20, 4, 100,000)	1,997	4,828	74	107	12,325	14,564
(20, 4, 1,000,000)	19,783	44,504	129	228	27,095	40,601
(30, 5, 100,000)	1,429	3,573	116	177	17,252	18,888
(30, 5, 1,000,000)	17,107	46,599	196	309	36,766	50,284
(40, 6, 100,000)	2,008	3,666	159	235	22,201	23,797
(40, 6, 1,000,000)	28,262	56,468	231	314	58,318	77,689

10. Conclusion and Future Directions

In this paper, we consider the dynamic assortment planning problem under uncapacitated MNL models and derive an optimal regret bound, which is independent of N.

There are a few interesting future work. In this paper, we assume that the time horizon length T is known. It is interesting to design "horizon-free" algorithms that adapt to the time horizon T. Moreover, the uncapacitated MNL can be viewed as a capacitated MNL with the capacity upper bound $K = N$. It is known from Agarwal et al. [2] and Chen and Wang [8] that the optimal regret is $\Theta(\sqrt{NT})$ when $K \leqslant N/4$, and from this paper that the optimal regret is $\Theta(\sqrt{T})$ when $K = N$. It is interesting to investigate the phase transition from $\Theta(\sqrt{NT})$ to $\Theta(\sqrt{T})$. Finally, another direction is to investigate "instance-optimal" regret bounds whose regret depends explicitly on the problem parameters $\{r_i\}_{i=1}^n, \{v_i\}_{i=1}^n$ and matching corresponding (instance-dependent) minimax lower bounds in which $\{v_i\}_{i=1}^n$ are known up to permutations. Such instance-optimal regret might potentially depend on "revenue gaps" $\Delta_i = R(S^*) - R(\mathcal{L}_{r_i})$, where S^* is the optimal assortment, and r_i is the revenue parameter of the item with the i th largest revenue.

Acknowledgments

The authors are grateful to area editor Prof. Benjamin Van Roy, the associate editor, and referees for their constructive comments and suggestions that have helped to significantly improve this paper.

Endnotes

[1] From random utility theory, we have $v_i = \exp(u_i)$, where u_i is the underlying mean utility. For the ease of presentation, we will call v_i the "utility parameter" because we only use v_i throughout this paper.

[2] By Lemma 3, we have $y_\tau - F(y_\tau) \geqslant y_\tau - F(x_\tau) \geqslant y_\tau - x_\tau$.

References

[1] Agrawal S, Avandhanula V, Goyal V, Zeevi A (2017) Thompson sampling for the MNL-bandit. Kale S, Shamir O, eds. Proc. *30th Annual Conf. Learn. Theory* (ML Research Press), 76–78.

[2] Agrawal S, Avandhanula V, Goyal V, Zeevi A (2019) MNL-bandit: A dynamic learning approach to assortment selection. *Oper. Res.* 67(5): 1453–1485.

[3] Agarwal A, Foster DP, Hsu D, Kakade SM, Rakhlin A (2013) Stochastic convex optimization with bandit feedback. *SIAM J. Optim.* 23(1): 213–240.

[4] Audibert JY, Bubeck S (2009) Minimax policies for adversarial and stochastic bandits. Dasgupta S, Klivans, A, eds. *Proc. 22nd Annual Conf. Learn. Theory* (ML Research Press), 217–226.

[5] Bubeck S, Cesa-Bianchi N (2012) Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations Trends Machine Learn.* 5(1):1–122.

[6] Bubeck S, Munos R, Stoltz G (2009) Pure exploration in multi-armed bandits problems. Gavaldà R, Lugosi G, Zeugmann T, Zilles S, eds. *Proc. Internat. Conf. Algorithmic Learn. Theory*, (Springer, Berlin), 23–37.

[7] Caro F, Gallien J (2007) Dynamic assortment with demand learning for seasonal consumer goods. *Management Sci.* 53(2):276–292.

[8] Chen X, Wang Y (2018) A note on tight lower bound for MNL-bandit assortment selection models. *Oper. Res. Lett.* 46(5):534–537.

[9] Chen X, Krishnamurthy A, Wang Y (2019) Robust dynamic assortment optimization in the presence of outlier customers. Preprint, submitted October 9, https://arxiv.org/abs/1910.04183.

[10] Chen X, Wang Y, Zhou Y (2020) Dynamic assortment optimization with changing contextual information. *J. Machine. Learn. Res.* 21(216): 1−44.

[11] Chen X, Wang Y, Zhou Y (2021) Dynamic assortment selection under nested logit models. *Production Oper. Management* 30(1):85−102.

[12] Chen X, Ma W, Simchi-Levi D, Xin L (2016) Dynamic recommendation at checkout under inventory constraint. Preprint, submitted October 17, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2853093.

[13] Cheung WC, Simchi-Levi D (2017) Thompson sampling for online personalized assortment optimization problems with multinomial logit choice models. Technical report, Massachusetts Institute of Technology, Cambridge, MA.

[14] Cheung WC, Ma W, Simchi-Levi D, Wang X (2018) Inventory balancing with online learning. Preprint, submitted October 11, https://arxiv.org/abs/1810.05640.

[15] Cohen M, Lobel I, Paes Leme R (2020) Feature-based dynamic pricing. *Management Sci.* 66(11):4921–4943.

[16] Cohen-Addad V, Kanade V (2017) Online optimization of smoothed piecewise constant functions. Singh A, Zhu J, eds. *Proc.* 20*th Internat. Conf. Artificial Intelligence Statist.* (ML Research Press), 412–420.

[17] Combes R, Proutiere A (2014) Unimodal bandits: Regret lower bounds and optimal algorithms. Xing EP, Jebara T, eds. *Proc.* 31*st Internat. Conf. Machine Learn.* (ML Research Press), 521–529.

[18] Cope EW (2009) Regret and convergence bounds for a class of continuum-armed bandit problems. *IEEE Trans. Automatic Control* 54(6): 1243–1253.

[19] Gallego G, Iyengar G, Phillips R, Dubey A (2004) Managing flexible products on a network. Technical Report CORC TR-2004-01, Department of Industrial Engineering and Operations Research, Columbia University, New York.

[20] Golrezaei N, Nazerzadeh H, Rusmevichientong P (2014) Real-time optimization of personalized assortments. *Management Sci.* 60(6):1532–1551.

[21] Hoeffding W (1963) Probability inequalities for sums of bounded random variables. *J. Amer. Statist. Assoc.* 58(301):13–30.

[22] Leme RP, Schneider J (2018) Contextual search via intrinsic volumes. *IEEE Annual Sympos. Found. Comput. Sci.* (IEEE Computer Society, Piscataway, NJ), 268–282.

[23] Liu Q, van Ryzin G (2008) On the choice-based linear programming model for network revenue management. *Manufacturing Service Oper. Management* 10(2):288–310.

[24] Lobel I, Leme RP, Vladu A (2018) Multidimensional binary search for contextual decision-making. *Oper. Res.* 66(5):1346–1361.

[25] Mahajan S, van Ryzin G (2001) Stocking retail assortments under dynamic consumer substitution. *Oper. Res.* 49(3):334–351.

[26] McFadden D (1974) Conditional logit analysis of qualitative choice behavior. Zarembka P, ed. *Frontiers in Econometrics* (Academic Press, New York), 105–142.

[27] Rusmevichientong P, Topaloglu H (2012) Robust assortment optimization in revenue management under the multinomial logit choice model. *Oper. Res.* 60(4):865–882.

[28] Rusmevichientong P, Shen ZJ, Shmoys D (2010) Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Oper. Res.* 58(6):1666–1680.

[29] Saure D, Zeevi A (2013) Optimal dynamic assortment planning with demand learning. *Manufacturing Service Oper. Management* 15(3):387–404.

[30] Talluri K, van Ryzin G (2004) Revenue management under a general discrete choice model of consumer behavior. *Management Sci.* 50(1): 15–33.

[31] van Ryzin G, Mahajan S (1999) On the relationships between inventory costs and variety benefits in retail assortments. *Management Sci.* 45(11):1496–1509.

[32] Wang Y, Chen X, Zhou Y (2018) Near-optimal policies for dynamic multinomial logit assortment selection models. Bengio S, Wallach H, Larochelle H, Grauman K, Cesa-Bianchi N, Garnett R, eds. *Proc. Adv. Neural Inform. Processing Systems* (Curran Associates, Red Hook, NY), 3105–3114.

[33] Yu JY, Mannor S (2011) Unimodal bandits. Getoor L, Scheffer T, eds. *Proc.* 28*th Internat. Conf. Machine Learn* (Omnipress, Madison, WI), 41–48.