Real-time Optimization of Personalized Assortments

Negin Golrezaei Hamid Nazerzadeh Paat Rusmevichientong

Marshall School of Business
University of Southern California
{golrezae,nazerzad,rusmevic}@usc.edu

February 1, 2014

Abstract

Motivated by the availability of real-time data on customer characteristics, we consider the problem of personalizing the assortment of products for each arriving customer. Using actual sales data from an online retailer, we demonstrate that personalization based on each customer's location can lead to over 10% improvements in revenue compared to a policy that treats all customers the same. We propose a family of index-based policies that effectively coordinate the real-time assortment decisions with the backend supply chain constraints. We allow the demand process to be arbitrary and prove that our algorithms achieve an optimal competitive ratio. In addition, we show that our algorithms perform even better if the demand is known to be stationary. Our approach is also flexible and can be combined with existing methods in the literature, resulting in a hybrid algorithm that brings out the advantages of other methods while maintaining the worst-case performance guarantees.

1 Introduction

The availability of real-time data on customer characteristics has encouraged companies to personalize operational decisions for each *arriving* customer. For instance, the product recommendations that Amazon.com makes to each customer dynamically change depending on recent reviews, ratings, purchases of the customer herself, purchases of other customers with similar interests to hers, and several other factors (Amazon's Recommendation Systems, 2012). Orbitz.com, as another example, has found that users of Apple Macintosh computers spend as much as 30% more per night on hotels; consequently, the company can show Mac users different and more expensive assortments of hotels and travel options, than Windows users (Mattioli, 2012). Location-based deals and coupons are offered by Groupon, Yelp, Foursquare, and other Internet companies (Wortham, 2012). In online advertising, the advertisements that are displayed to a user browsing a website are routinely personalized based on the user's browsing history, demographic information, and location (Helft and Vega, 2010). Even brick-and-mortar grocery stores are starting to offer personalized real-time coupons based on each customer's purchasing history and the available products on the shelf in the aisle where each customer is currently shopping (Clifford, 2012).

These examples raise a key question that motivates our work: Given the complexity of coordinating real-time, front-end, customer-facing decisions with the back-end supply chain constraints, what policies should companies use to take advantage of such data?

We answer these questions by formulating a real-time, personalized, choice-based assortment optimization problem involving multiple products with limited inventories, and arbitrary customer types. The type of each arriving customer can be arbitrary, and it is indexed by a (possibly infinite) set $\mathcal{Z}$. Examples of types include the customer's computer type (Mac vs PC), her current location, her purchasing history, the average household income in her neighborhood, competitors' current offerings and prices, time of day, or a combination of other observable characteristics.

For an arriving customer of type $z \in \mathcal{Z}$, the company must decide in real time, on the assortment of products to offer. Given an assortment S, the customers make choices on which products to buy, if any, according to a *general choice model* that is specific to each customer type. Our goal is to develop a revenue-maximizing policy that determines the assortment to offer to each arriving customer, taking into account the customer type and the current inventories.

The above formulation captures the essential features of the situation faced by companies that sell services, products, or advertising to heterogenous customer types that require real time decision-making, with inventory constraints.

We first observe that differentiating among customer types (even just their locations) can significantly increase revenues. We consider the top ten DVDs with the highest national sales volumes during the summer of 2005 and compare their sales in two locations: Urbana-Champaign, IL and Miami, FL. Table 1 shows the sales rate of each DVD in each location, which is defined as the proportion of the potential customers in each location[1] who purchased each DVD.

DVD	Title	Sales Rate		
		National	Miami	Urbana-Champaign
1	Lost–The Complete First Season	7.6%	8.9%	6.7%
2	Firefly–The Complete Series	6.9%	0.0%	12.0%
3	The Simpsons–The Complete Sixth Season	6.0%	5.0%	6.7%
4	Star Wars: Episode III–Revenge of the Sith	5.5%	7.4%	4.0%
5	Sin City	5.2%	9.5%	2.0%
6	Family Guy Presents Stewie Griffin–The Untold Story	4.8%	3.3%	6.0%
7	Batman Begins	4.3%	4.8%	4.0%
8	What the Bleep Do We Know!?	4.2%	9.8%	0.0%
9	Curb Your Enthusiasm: The Complete Fourth Season	3.8%	1.8%	5.3%
10	Seinfeld: Season Four	3.7%	3.3%	4.0%

Table 1: Top 10 DVDs nationally and the percentage of customers in each location who purchase each DVD.

We observe that these two locations exhibit very different purchasing behaviors, which are significantly different from the national sales pattern. Consider the sci-fi DVD *Firefly–The Complete Series*, which is the second most popular DVD nationally, with a sales rate of 6.9%. None of the customers in Miami purchased this DVD, while 12% of the customers in Urbana-Champaign, almost twice the national rate, bought the series. On the other hand, none of the customers from Urbana-Champaign bought *What the Bleep Do We Know!?*, while almost 10% of the customers in

[1] The potential customers in each location correspond to the people in the location who bought one of the top 200 DVDs with the highest national sales volumes during the summer of 2005. We choose 200 DVDs as the cutoff because they account for a large proportion of the sales volumes.

Miami bought it. In Section 7.3, we evaluate the performance of our algorithms and observe that customizing the assortment of DVDs for each customer's location leads to an increase in revenues of 10%. The improvement can be as high as 21% when the sizes of the assortments are constrained.

As our main contribution, **we propose a family of simple and effective algorithms, called Inventory-Balancing, for real-time personalized assortment optimization that do not require any forecasting:** an Inventory-Balancing algorithm maintains a "discounted-revenue index" for each product in which the (actual) revenue is multiplied by a (virtual) discount factor that depends on the fraction of the product's *remaining* inventory. Upon the arrival of each customer, based on the customer's type, the algorithm offers to her the assortment that maximizes the expected discounted revenue. Each Inventory-Balancing algorithm is characterized by a penalty function that discounts the marginal revenue of each product as the inventory level drops. By adjusting the revenue of each product according to its remaining inventory, the algorithms hedges against the uncertainty in the types of future customers by reducing the rate at which products with low inventory are offered. Thus, the discounted-revenue index serves as a simple mechanism that coordinates the front-end customer-facing decision with the back-end supply chain constraints. Our Inventory-Balancing algorithms offer the following benefits.

No forecasting: A traditional approach for dynamic assortment optimization, both in the literature and in practice, is to forecast demand over time by estimating the distribution of the number of customers of each type and then finding an optimal policy based on the forecast using re-optimization methods (Gallego and van Ryzin, 1994; Jasin and Kumar, 2012) or dynamic programming (Bernstein et al., 2011). In our sales data, we observe large variability in the number of customers across time and locations. Such a large variability in the demand process often makes forecasting difficult and, not surprisingly, could lead to poor performance for the policies established under this approach.

An alternative approach to making a real-time decision is to solve the "off-line" assortment optimization problem repeatedly using the most up-to-date inventory levels of each product and the latest demand forecast. This can be done by repeatedly solving a series of linear programs; see, for example, Jasin and Kumar (2012). When the number of customers is known in advance, the re-optimization methods work extremely well and yield nearly optimal revenue because they can effectively ration the inventory to all customers. However, when there is significant uncertainty in the market size, the problem becomes more challenging. In this setting, our Inventory-Balancing algorithms perform very well, yielding 5%-11% more revenues than re-optimization methods.

Strong performance under both non-stationary and stationary demand processes: as a performance benchmark, we compare the revenue of our algorithms to the revenue of a *clairvoyant optimal solution* that has complete knowledge of the sequence of the *types* of the customers that arrive in the future but does not know the (random) choice for each future customer.[2] We prove that Inventory-Balancing algorithms with a strictly concave penalty function *always* obtain more than 50% of the optimal revenue; see Theorem 1 and Corollary 1. We also provide an Inventory-Balancing algorithm that obtains at least $(1 - \frac{1}{e}) \approx 63\%$ of the benchmark revenue. This implies that, even when there are sudden shocks in the customers' arrival patterns, either from seasonality or other non-stationarity effects, the algorithm maintains a strong performance guarantee.

[2]For each future customer and an assortment, the clairvoyant algorithm knows the probability that the customer will purchase a product from that assortment but does not know the exact choice that the customer will make. In other words, the clairvoyant algorithm knows the choice model but does not know the realization of the random choices of a future customer.

The $(1 - \frac{1}{e})$ fraction of the benchmark revenue is optimal for non-stationary stochastic arrivals in the sense that in the worst-case, no deterministic or randomized policy can achieve a higher competitive ratio; see Theorem 2.

When customer arrivals are stationary, our algorithms perform even better. We show that, when the types of arriving customers are independently and identically distributed, our algorithm is *guaranteed* to obtain at least 75% of the benchmark; see Theorem 3. In our numerical experiments, presented in Section 7, our algorithms perform even better than what is predicted by the worst-case bound, obtaining revenues that are within 96%-99% of the benchmark.

Simplicity, robustness, and flexibility: in contrast to the existing methods, our Inventory-Balancing algorithms are extremely simple and fast. We do not need to solve any offline assortment optimization problem, so we can compute the decision for each customer quickly. Our formulation allows for infinite customer types, so our algorithms are robust to changing customer types over time. In addition, under mild assumptions, our analysis and performance guarantees continue to hold when the choice models of the customers are learned over time and the algorithm uses estimations of the parameters of the choice models; see Proposition 1.

Our proposed algorithms are also flexible, and they can be easily combined with existing re-optimization methods while maintaining worst-case performance guarantees; see Section 5.3 and Proposition 2. Our numerical experiments show that such a hybrid method brings out the advantages of all methods, especially when there is uncertainty in the number of future customers.

The key messages in this paper are that personalization can increase the revenue significantly and real-time optimization of personalized assortments can be done efficiently and robustly. Our proposed policies maintain a simple index for each product, which balances the nominal revenue with the value of each unit of remaining inventory. These indices are easy to implement, and they serve as a simple mechanism that coordinates between front-end real-time decisions and the back-end supply chain constraints. As the volumes of data on customer profiles and preferences continue to grow, we believe that companies will consider the personalization of other operational decisions, such as pricing or shipping options. The framework and analysis in this paper can serve as a starting point for more complex models.

1.1 Literature Review

Our work is related to the growing literature on assortment planning. We describe a brief overview of the area to provide a context for our work. Assortment planning problems focus on the relationships among assortment offerings, customer choices, and inventory constraints. van Ryzin and Mahajan (1999) introduced one of the first models that capture the tradeoffs between inventory costs and product variety. Mahajan and van Ryzin (2001) followed up on this work with a study on the optimal inventory levels in the presence of stockouts and substitution behavior. Since their seminal work was published, researchers have considered a variety of choice models and studied how such models affect the optimal assortment and the inventory level of products that should be carried. Examples include the demand substitution model (Smith and Agrawal, 2000), the Lancaster choice model (Gaur and Honhon, 2006), ranked-list preferences (Honhon et al., 2010; Goyal et al., 2011), and multinomial logit models (Talluri and van Ryzin, 2004; Gallego et al., 2004; Liu and van Ryzin, 2008; Topaloglu, 2013). Recently, Farias et al. (2013) introduced a very general class of choice models based on a distribution over permutations and developed efficient algorithms for determining the optimal assortment. For a survey of the assortment planning literature, the reader is referred to Kök et al. (2008).

Two of the most important decisions in modeling an assortment planning problem are determining the customers' choice models and capturing the arrival process of the customers. The family of choice models considered in our paper is quite general and includes most of the choice models used in practice or previously studied by researchers. What distinguishes this work from the existing literature is the fact that we do not impose any restrictions on the arrival process, and most of our results hold even if an adversary chooses the sequence of customers. In the following, we briefly discuss the prevalent approaches to model the arrival process.

A common approach to model customer arrivals is to assume that arrivals follow a stochastic process. In this model, the optimal sequence of assortments can be planned by solving a multi-dimensional dynamic program, see Appendix C. Not surprisingly, this approach suffers from the curse of dimensionality, even for stationary processes.

Recently, Bernstein et al. (2011) studied the aforementioned assortment planning model under the assumption that the type of customer (represented by multinomial logit choice models) is drawn identically and independently from a stationary distribution, i.e., I.I.D. arrivals. For two products with equal revenue, two customer types (with each type following a multinomial logit choice model), and Poisson arrivals, they provide structural properties of the optimal solution. Interestingly, they show that the optimal dynamic program may withhold products with low remaining inventory for future customers that are more interested in them. Based on this observation, they propose a heuristic that, roughly speaking, reduces the general problem with multiple products to a two-product problem by separating the products into two groups based on their inventory to demand ratio. They do not provide any performance guarantees for the heuristic.

In practice, we do not expect the distribution of customer types to remain constant over time because of seasonality effects or changing popular trends. In the context of airline revenue management, the fraction of business customers tends to increase as the departure date approaches. Prior to our work, to the extent of our knowledge, the best known performance guarantee for a heuristic with respect to a clairvoyant optimal solution in non-stationary stochastic environments was a ratio of $\frac{1}{2}$ that follows from Chan and Farias (2009). When the arrivals are stochastic, with some adjustments, the assortment planning studied in the paper would fit into the stochastic depletion framework proposed by Chan and Farias (2009). They show that, in their framework, the competitive ratio of a myopic policy is at least $\frac{1}{2}$.

Re-optimization policies (Jasin and Kumar, 2012) are applicable to non-stationary stochastic environments. However, we are not aware of any results that provide a performance guarantee.[3] The closest work to this line of research is by Ciocan and Farias (2013), who studied re-optimization policies for a network revenue management problem where the distribution of the valuation of the customer (i.e., the distribution of the types) is *constant* over time, but the size of the market changes over time according to a stochastic (e.g., multi-variate Gaussian) process. They showed that a re-optimization policy that adjusts prices of the products by solving a linear program obtains about one-third of the optimal revenue; see also Chen and Farias (2013). In contrast to dynamic pricing problems, where firms manage their profits and capacities by controlling prices, in assortment planning models, product prices are exogenously determined and remain constant over the horizon, and firms decide on the selection of the assortment to offer to each customer.

[3]The analysis of Liu and van Ryzin (2008) and Jasin and Kumar (2012) extends to the environment with time-varying demand if the demand varies slowly over time, but not when the demand is volatile.

We choose the competitive ratio as our performance benchmark because it allows for arbitrary non-stationary and even adversarial arrivals, and it does not require any prior knowledge about the arrival patterns. This notion has been previously applied by Ball and Queyranne (2009) to the problem of capacity allocation. Besbes and Zeevi (2011) and Besbes and Sauré (2012) also used similar notions of optimality when they studied revenue management problems where the demand may change dramatically because of shocks.

The problem we study here resembles some aspects of the Adwords problem (Mehta et al., 2007; Buchbinder et al., 2007; Goel et al., 2010; Azar et al., 2009), where the goal is to allocate a sequence of advertising spaces associated with search queries to budget-constrained advertisers. Both the Adwords and personalized assortment optimization problems contain the b-matching problem as a special case (Kalyanasundaram and Pruhs, 2000). Mehta et al. (2007) proposed an algorithm that achieves an optimal-competitive ratio for the Adwords problem by taking into account both the bid and the budget of the advertisers; see Buchbinder and Naor (2007) and Mehta (2012) for a survey on online algorithms and Acimovic and Graves (2011) for another application in the context of inventory management.

Organization: In Section 2, we formally define our problem. Our algorithm and the main results are presented in Section 3. We discuss the performance of our algorithm under stationary stochastic arrivals in Section 4, followed by discussions of extensions of our original model in Section 5. The proof of the competitive ratio and discussions of computational complexity are presented in Section 6. We present the numerical experiments in Section 7. The conclusion and direction for future work are given in Section 8.

2 Preliminaries and Problem Formulation

Consider a firm that sells n products, indexed by $1, 2, \ldots, n$, to customers that arrive sequentially over time. The firm obtains a revenue $r_i > 0$ for selling each unit of product i, which has an initial inventory of $c_i \in \mathbb{Z}_+$, with no replenishment. We denote the no-purchase option as product 0, with $r_0 = 0$. Let $\mathcal{Z}$ denote the set of possible customer types. Once a customer arrives, her type, denoted by $z \in \mathcal{Z}$, is revealed. For instance, the type of a customer can correspond to his or her computer type; i.e., $\mathcal{Z} = \{\text{Mac}, \text{PC}\}$.[4] As mentioned in the introduction about assortment personalization by Orbitz.com, the type $z = \text{Mac}$ may suggest that the user is more likely to choose expensive travel options. If we are interested in the location of each customer, then the type of the customer can correspond to his or her ZIP code.[5] In addition, the revelation of each customer's type can happen when the customer logs in to the website, e.g., Amazon or eBay.

Based on the customer's type and the remaining inventory, the firm offers an assortment $S \in \mathcal{S}$, where $\mathcal{S}$ denotes the set of all feasible assortments; we assume that $\{0\} \in \mathcal{S}$; i.e., the firm has the option to not offer any product. The set $\mathcal{S}$ allows us to incorporate a variety of constraints on the assortments, such as shelf-space or size constraints; see Section 6.4.

Associated with each customer type $z \in \mathcal{Z}$ is the probability of purchasing each product under each assortment. More specifically, each customer type $z \in \mathcal{Z}$ corresponds to a *general* choice

[4] This information is communicated to the website that the user is visiting.

[5] This information may be identified through each customer's IP address or (opt-in) cellphone's GPS signals; see Steel and Angwin (2010).

model that specifies the probability of purchasing each product under each assortment. We denote by $\phi_i^z(S)$ the probability that a customer of type z purchases product i, when assortment S is offered. In fact, all of our results continue to hold when each customer may purchase more than one product at a time: we define $\Phi^z : \mathcal{S} \times \mathcal{S} \to [0,1]$, where $\Phi^z(S', S)$ is the probability that a customer of type z will purchase exactly the products in set S' when assortment S is offered; in addition, $\Phi^z(S', S) = 0$ when $S' \not\subseteq S$ or $S \notin \mathcal{S}$ and $\sum_{S' \subseteq S} \Phi^z(S', S) = 1$. Hence, we have $\phi_i^z(S) = \sum_{S' : i \in S', S' \subseteq S} \Phi^z(S', S)$. In the remainder of the paper, we use only the notation $\phi_i^z(\cdot)$.

Our goal is to design an algorithm that offers an assortment to each arriving customer to maximize the total expected revenue. Let a vector $\{z_t\}_{t=1}^T = (z_1, z_2, \cdots, z_T)$ represent the sequence of the types of the arriving customers, where for each t, $z_t \in \mathcal{Z}$ denotes the type of the customer that arrives in period t.

Definition 1. *For any algorithm A and any sequence of customer types $\{z_t\}_{t=1}^T$, we denote by $\mathsf{Rev}_A\left(\{z_t\}_{t=1}^T\right)$ the expected revenue obtained by algorithm A from the customers $\{z_t\}_{t=1}^T$, where the expectation is taken with respect to the choices made by each customer and possibly random selections of the algorithm (if the algorithm is not deterministic).*

We do *not assume* any arrival patterns, and the algorithm does *not know* the sequence of the customers in advance. Therefore, we use the notion of the competitive ratio, defined below, to measure the performance of an algorithm. The following lemma establishes an upper bound on the expected revenue that can be obtained by any algorithm from a sequence of customers.

Lemma 1 (Revenue Upper Bound)**.** *For any sequence of customers $\left(\{z_t\}_{t=1}^T\right)$ and any algorithm A, $\mathsf{Rev}_A\left(\{z_t\}_{t=1}^T\right)$ is bounded by the optimal value of the linear program $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$ defined below:*

$$\begin{array}{llr}
\text{MAXIMIZE} & \sum_{t=1}^T \sum_{S\in\mathcal{S}} \sum_{i=1}^n r_i \phi_i^{z_t}(S) y^t(S) & (\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)) \\
\text{SUBJECT TO:} & \sum_{t=1}^T \sum_{S\in\mathcal{S}} \phi_i^{z_t}(S) y^t(S) \le c_i \quad 1 \le i \le n & \\
 & \sum_{S\in\mathcal{S}} y^t(S) = 1 \quad 1 \le t \le T & \\
 & y^t(S) \ge 0 \quad 1 \le t \le T,\ \forall S \in \mathcal{S} &
\end{array}$$

In the linear program above, $y^t(S)$ corresponds to the probability that the set S is offered to the customer of type z_t in period t. With a slight abuse of notation, we denote the optimal value of the linear program above by $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$ as well. The proof, given in Appendix A, follows from the fact that $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$ is an upper bound on the expected revenue of the optimal clairvoyant solution that knows the sequence of the customer types in advance. Namely, we construct a feasible solution for the linear program above based on the optimal clairvoyant solution, taking into account the realizations of the customers' choice models.

According to the above lemma, no algorithm without hindsight would obtain revenue equal to $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$. However, an algorithm with no knowledge of the future types might be able to obtain a fraction of the revenue of this clairvoyant optimal solution. Therefore, the competitive ratio of an algorithm is defined as follows:

Definition 2 (Competitive Ratio). *An algorithm A is α-competitive if:*

$$\inf_{T\geq 1} \quad \inf_{\{z_t\}_{t=1}^T : z_t \in \mathcal{Z}\ \forall t} \quad \frac{\mathsf{Rev}_A\left(\{z_t\}_{t=1}^T\right)}{\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)} \quad \geq \quad \alpha\ .$$

The infimum is taken over all possible sequences of customer arrivals of arbitrary lengths. In other words, the competitive ratio is defined as the worst-case ratio between the "expected revenues" of an algorithm and the optimal clairvoyant solution over a (possibly infinite) sequence of customer types, where the expectation is with respect to the customers' choice models.[6]

One potential criticism of the notion of the competitive ratio could be that it compares algorithms with a benchmark that is too strong. However, as we show in the following sections, in the context of assortment planning, it leads to simple algorithms that perform very well with respect to this benchmark. Moreover, our numerical simulations demonstrate the practical relevance of our method and show that our algorithms outperform existing methods in the literature.

3 Inventory-Balancing Algorithms

We present a family of algorithms called Inventory-Balancing (IB), which take into account both the revenue that would be obtained from the customer and the current inventory levels to decide which assortments to offer. Each Inventory-Balancing algorithm is defined with a *penalty function* $\Psi : [0, 1] \to [0, 1]$, which is an increasing function with $\Psi(0) = 0$ and $\Psi(1) = 1$.

Recall that c_i is the initial inventory of product i. Let I_i^t denote the remaining inventory of product i at the end of period t. Note that $I_i^0 = c_i$, and for $t \geq 1$, $I_i^t = \max\{I_i^{t-1} - Q_i^t, 0\}$, where Q_i^t is a binary random variable that is equal to 1 if the customer has chosen product i and 0 otherwise. We are now ready to describe the algorithm.

<u>INVENTORY-BALANCING WITH A PENALTY FUNCTION Ψ</u>

Upon the arrival of the customer in period $t \in \{1, \dots, T\}$ of type z_t, offer an assortment S^t:

$$S^t = \operatorname{argmax}_{S \in \mathcal{S}} \sum_{i \in S} \Psi\left(I_i^{t-1}/c_i\right) r_i \phi_i^{z_t}(S)$$

The assortment S^t can be found in polynomial time for a broad class of choice models; see Section 6.4. In the case of ties, we choose any of the sets with the smallest number of products. We can think of $r_i \Psi\left(I_i^{t-1}/c_i\right)$ as the discounted revenue associated with product i, where the discount factor $\Psi\left(I_i^{t-1}/c_i\right)$ is determined by the penalty function, and it depends on the fraction of the initial inventory that remains. As we discuss in Section 4.1, $\Psi\left(I_i^{t-1}/c_i\right)$ corresponds to a dual solution to $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$. Namely, for each t, using $\Psi\left(I_i^{t-1}/c_i\right)$, we can construct a feasible solution for the

[6] Note that we use the upper-bound linear program $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$ as a proxy for the revenue of the optimal clairvoyant algorithm. This can be interpreted as giving even more power to the clairvoyant algorithm since it can now respect inventory constraints only in expectation. However, such additional power would be negligible with large inventory levels, which we believe are more interesting and realistic instances of the problem.

dual of $\mathsf{Primal}\big(\{z_t\}_{t=1}^T\big)$, and the value of this feasible dual solution is within a "constant" factor of $\mathsf{Primal}\big(\{z_t\}_{t=1}^T\big)$.

The main idea behind the algorithm is simple. Sometimes it might be better to sell a product with a lower marginal revenue but a high inventory level than to sell a product with high marginal revenue but a small remaining inventory. This is because future customers might only be interested in products with low (or no) inventory, and if we have already sold those products, we would lose these profitable opportunities.[7] The penalty function thus protects against uncertainty in future customer types. The following example shows what will happen if we ignore the inventory level, and only offer assortments with the highest revenues; see Mehta et al. (2007) and Bernstein et al. (2011) for similar examples.

Example: Myopic Policy and Why Inventory Levels Matter Consider a myopic policy that does not take into account any inventory level. The policy corresponds to the following penalty function: $\Psi(x) = \mathbb{1}[x > 0]$; i.e., $\Psi(x)$ is equal to 1 if the remaining inventory of the product is positive and 0 otherwise. This algorithm, at any period, offers the assortment, among products with positive inventory, that maximizes the expected revenue. The following scenario shows that the competitive ratio of the myopic policy is at most $\frac{1}{2}$. Suppose that the length of the horizon is equal to T. There are two products with the following parameters: $r_1 = 1 + \epsilon$, $r_2 = 1$, and $c_1 = c_2 = \frac{T}{2}$. We have two customer types. The first type arrives during periods $1, \ldots, \frac{T}{2}$, and the second type arrives in periods $\frac{T}{2} + 1, \ldots, T$. For $t \in \{1, \ldots, \frac{T}{2}\}$, $\phi_1^{z_t}(\{1\}) = \phi_2^{z_t}(\{2\}) = 1$ and $\phi_i^{z_t}(S) = 0$ otherwise. For $t \in \{\frac{T}{2} + 1, \ldots, T\}$, $\phi_1^{z_t}(\{1\}) = 1$ and $\phi_i^{z_t}(S) = 0$ otherwise. In this setting, a customer of the first type is interested in both products 1 and 2, while the second type is interested only in product 1.

The myopic policy will allocate all the inventory of product 1 to the customers that arrive in periods $1, 2, \ldots, \frac{T}{2}$ and obtain a revenue of $\frac{T}{2}(1+\epsilon)$. However, the optimal solution of $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$ allocates all the inventory of product 2 to the customers that arrive in periods $1, 2, \ldots, \frac{T}{2}$, and then sells product 1 to customers that arrive afterwards in period $\frac{T}{2} + 1, \ldots, T$, yielding a revenue of $\frac{T}{2}(2 + \epsilon)$. Note that $\frac{T(1+\epsilon)}{T(2+\epsilon)} \leq \frac{1}{2} + \epsilon$. Since ϵ can be arbitrarily small, the competitive ratio of the myopic policy is at most $\frac{1}{2}$.[8]

The above example, though rather stylized, highlights the importance of inventory levels in assortment planning. We will show that, by discounting the revenue of each product based on its remaining inventory, the Inventory-Balancing algorithms obtain a better competitive ratio. Throughout the paper, we impose the following mild assumption on the choice models.

Assumption 1 (Substitutability)**.** *For all $z \in \mathcal{Z}$, $S \in \mathcal{S}$, and $i \neq j$, $\phi_i^z(S) \geq \phi_i^z(S \cup \{j\})$.*

The above assumption implies that adding another product to an assortment does not increase the probability of selling other products in the assortment. It is easy to verify that the above assumption encompasses all choice models that are consistent with random utility maximization[9]

[7]Bernstein et al. (2011) show that (under certain assumptions) the behavior of the optimal dynamic program is similar to this intuition.

[8]On the other hand, it is not difficult to show that the myopic policy obtains at least $\frac{1}{2}$ of the revenue of the benchmark revenue. Hence, the ratio $\frac{1}{2}$ is tight.

[9]This is because, under the random utility maximization (cf. Talluri and van Ryzin, 2004), $\phi_i^z(S) = \Pr\left\{U_i^z \geq \max_{\ell \in S \cup \{0\}} U_\ell^z\right\}$ where $(U_0^z, U_1^z, \ldots, U_n^z)$ is the random utility vector that a customer of type z assigns

including the multinomial logit choice model, the nested logit, and many others. The above assumption leads to the following desirable property. The proof is relegated to the appendix.

Lemma 2. *Under Assumption 1, the Inventory-Balancing algorithm never offers an assortment that includes a product with zero remaining inventory.*

In Section 5.4, we relax Assumption 1 and extend our results to a more general setting where stockouts are allowed. We now use Lemma 2 to establish the competitive ratio. The proof is given in Section 6.1.

Theorem 1 (Competitive Ratio)**.** *Let* $c_{\text{MIN}} = \min_{i=1,\ldots,n} c_i$*. Suppose that* Ψ *is an increasing, concave, and twice-differentiable penalty function. The competitive ratio of the Inventory-Balancing algorithm with a penalty function* Ψ *is at least equal to* $\alpha_{c_{\text{MIN}}}(\Psi)$*, where*

$$\alpha_{c_{\text{MIN}}}(\Psi) = \min_{x \in \left[0, 1 - \frac{1}{c_{\text{MIN}}}\right]} \left\{ \frac{1-x}{\frac{1}{c_{\text{MIN}}} + 1 - \Psi(x) + \int_{x+\frac{1}{c_{\text{MIN}}}}^{1} \Psi(y)dy} \right\}$$

We emphasize that the competitive ratio in the above theorem depends only on c_{MIN} and penalty function Ψ, and the ratio does **not** depend on the length of the horizon T. Hence, the above result holds when T increases to infinity. It also holds for any sequence of customer types of arbitrary length, even if the sequence is chosen by an *adversary.*

Many of the previous results in the literature have been established for an asymptotic regime where the size of the initial inventory c_{MIN} and the length of the horizon T tend to infinity. The justification for the asymptotic analysis is that the initial inventory of products and the number of customers are often large. In this asymptotic regime, we can simplify the expression for the competitive ratio. We define:

$$\alpha(\Psi) := \alpha_\infty(\Psi) = \min_{x \in [0,1]} \left\{ \frac{1-x}{1 - \Psi(x) + \int_x^1 \Psi(y)dy} \right\}.$$

We observe that the competitive ratio of the algorithm improves *slightly* as c_{MIN} becomes larger. For instance, for the polynomial penalty functions $\Psi(x) = \sqrt{x}$, the competitive ratio with $c_{\text{MIN}} = 2, 5$, and 10 is, respectively, 0.52, 0.55, and 0.57. This ratio approaches $\alpha(\sqrt{x}) = 0.60$ as c_{MIN} grows.

As a corollary of Theorem 1, we can show that the competitive ratio is at least $\frac{1}{2}$ for any increasing concave function. Therefore, by taking into account the remaining inventory levels, we obtain a better performance guarantee than a myopic policy that ignores inventory.

Corollary 1. *For the Inventory-Balancing algorithm with a linear penalty function* (LIB)*,* $\Psi(x) = x$*, the competitive ratio* $\alpha_{c_{\text{MIN}}}(x)$ *is equal to* $\frac{1}{2}$ *for any* $c_{\text{MIN}} \geq 1$*. For any increasing strictly concave and differentiable penalty function, the competitive ratio is strictly* greater *than* $\frac{1}{2}$*.*

---

to each product. The random variables $(U_0^z, U_1^z, \ldots, U_n^z)$ may be correlated and can have arbitrary distributions. If $j \neq i$, then

$$\phi_i^z(S \cup \{j\}) = \Pr\left\{U_i^z \geq \max_{\ell \in S \cup \{j\} \cup \{0\}} U_\ell^z\right\} \;\leq\; \Pr\left\{U_i^z \geq \max_{\ell \in S \cup \{0\}} U_\ell^z\right\} = \phi_i^z(S)\ .$$

The proof is given in Appendix A. Note that this is a worst-case performance guarantee and does not imply that the Inventory-Balancing algorithms outperform the myopic policy for every sequence of customers. In practice, as suggested by our numerical simulations, we expect that the IB algorithms and even the myopic policy often to perform better than their theoretical worst-case bounds; see Section 7.2.

The choice of the penalty function determines the trade-offs between the revenue from selling a product and the value of the remaining inventory. For a linear penalty function $\Psi(x) = x$, the derivative is always 1, and a reduction in a unit of inventory has the same penalty, regardless of the inventory level. On the other hand, the derivative of the exponential penalty function $\Psi(x) = \frac{e}{e-1}(1 - e^{-x})$ is given by $\frac{e}{e-1}e^{-x}$, which decreases from 1.58 at $x = 0$ to 0.58 at $x = 1$. Under the exponential penalty function, consuming one unit of inventory incurs a higher penalty when the inventory is scarce. In regimes with high demand and low inventory, we would expect that the Inventory-Balancing algorithm with an exponential penalty function (EIB) to be more conservative and hold back more products to hedge against future arrivals. As we show in the next section, the best competitive ratio can be obtained using an exponential penalty function $\Psi(x) = \frac{e}{e-1}(1 - e^{-x})$.

3.1 The Tight Upper Bound on the Competitive Ratio

We start this section by providing an upper bound on the competitive ratio. Then, in Theorem 2, we show that an IB algorithm with an exponential penalty function achieves this upper bound, showing that our proposed method achieves an optimal competitive ratio.

Lemma 3 (Upper Bound on the Competitive Ratio)**.** *For any number of products n, we can construct a non-stationary stochastic process for customer arrivals where, for every deterministic algorithm (including the optimal dynamic program), there exists a sequence of customer types $\{z_t\}_{t=1}^T$ such that the revenue of the algorithm is at most a fraction $\rho_n = \left(\frac{1}{n}\sum_{j=1}^n \min\left\{\sum_{t=1}^j \frac{1}{n-t+1}, 1\right\}\right)$ of $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$.*

For instance, for $n = 2, 5$, and 20, the upper bound ρ_n is respectively equal to 0.75, 0.69, and 0.64, and ρ_n approaches $\lim_{n\to\infty} \rho_n = 1 - \frac{1}{e} \approx 63\%$ as the number of products n increases.

In the proof of the above lemma, given in Section 6.2.1, we construct a stochastic process that consists of n products. The per-unit revenue from each product is equal to 1, and the initial inventories are equal to $\frac{T}{n}$. Think of T, the length of the horizon, as a very large number (that would tend to infinity) and a multiple of n. The number of types is equal to $2^n - 1$. Each type corresponds to a non-empty set Θ of products that a customer of that type equally likes; the "no-purchase" probability for all types is equal to zero. Note that this is a special case of the multinomial logit choice model where all the products have a weight of either 0 or 1. The arrival process is defined as follows: customers arrive in n phases of equal length; that is, the number of customers in each phase is $\frac{T}{n}$. All the customers in each phase have the same type. Customers in the first phase are interested in all the products. After that, in each phase, customers randomly lose interest in one of the products of interest in the previous phase; i.e., there are $n!$ sequences of customer arrivals, each with equal probability.

Now, we show that the Exponential Inventory-Balancing algorithm achieves the optimal competitive ratio.

Theorem 2 (Exponential IB Achieves the Optimal Competitive Ratio)**.** *The competitive ratio of the Inventory-Balancing algorithm with an exponential penalty function* (EIB)*,* $\Psi(x) = \frac{e}{e-1}(1-e^{-x})$*,* $x \in [0,1]$*, approaches* $\left(1 - \frac{1}{e}\right)$ *as* c_{MIN} *increases to infinity; i.e.,* $\alpha\left(\frac{e}{e-1}(1-e^{-x})\right) = 1-\frac{1}{e}$*. Moreover, no algorithm, deterministic or randomized, that does not know the sequence of customer types in advance can obtain a competitive ratio better than* $1 - \frac{1}{e}$*.*

The proof is given in Section 6.2. The first part of the proof is based on Theorem 1. The second part follows from applying Yao's Lemma (Yao, 1977) to Lemma 3. Yao's Lemma implies that the competitive ratio of any randomized algorithm that does not know the input sequence in advance is bounded by the competitive ratio of any deterministic algorithm that knows the distribution over the input sequence.

We note that the upper bound of $(1 - \frac{1}{e})$ applies to *all* deterministic algorithms, including the optimal dynamic programming (Bernstein et al., 2011) and re-optimization (Jasin and Kumar, 2012) policies. Thus, by the theorem above, in terms of the competitive ratio, the Inventory-Balancing algorithm with an exponential penalty function is optimal for this problem. We remark that this notion of optimality does not imply that the algorithm will yield the highest revenue from every sequence of customers.

Under Theorem 1, the competitive ratio of the Exponential Inventory-Balancing algorithm with limited inventory, such as for $c_{\text{MIN}} = 5$, 10, 20, and 30, is respectively equal to 0.57, 0.60, 0.61, and 0.62. The ratio approaches 0.63 rather rapidly as c_{MIN} grows. We emphasize that these ratios hold for **all** values of T, including the asymptotic regime where T increases to infinity (at a possibly faster rate than c_{MIN}).

Moreover, as shown in our numerical experiments, our algorithms often perform much better than the worst-case guarantee bounds.

4 Stochastic I.I.D. Arrivals

The competitive ratio of our IB algorithm in Theorems 1 and 2 hold for any arbitrary, possibly adversarially chosen sequence of customer types. It turns out that our IB algorithm performs even better if the customer arrivals follow a stochastic process, that is, when the sequence of customers $\{z_t\}_{t=1}^T$ is generated by a stochastic process that is known in advance. In this model, the optimal sequence of assortments can be planned by solving a multi-dimensional dynamic program; for more details, see Appendix C. Not surprisingly, this approach suffers from the curse of dimensionality, even for stationary processes; see Bernstein et al. (2011).

Under stochastic models, although a dynamic programming approach may be intractable, there is room for natural and powerful heuristics. First observe that, for any algorithm A, the *expected revenue* of the algorithm, denoted by $\mathbb{E}_{\{z_t\}_{t=1}^T}[\mathsf{Rev}_A\left(\{z_t\}_{t=1}^T\right)]$, is well-defined, where $\mathbb{E}_{\{z_t\}_{t=1}^T}$ is the expectation with respect to the sequence of customers. Recall that, by definition, the expectation of customers' choices is taken into account by $\mathsf{Rev}_A(\cdot)$. Furthermore, we can establish an upper bound on the revenue of the algorithm. The proof is omitted due to its similarity to Lemma 1.

Lemma 4 (Revenue Upper Bound for I.I.D. Arrivals)**.** *Suppose that the types of customers are drawn independently and identically from a known distribution. Let* η^z *be the* expected *number of customers of type* $z \in \mathcal{Z}$*. The expected revenue of any algorithm* A*,* $\mathbb{E}_{\{z_t\}_{t=1}^T}[\mathsf{Rev}_A\left(\{z_t\}_{t=1}^T\right)]$*, is* *bounded by the optimal value of the linear program* Primal-S*, defined below:*[10]

$$
\begin{array}{llllr}
\text{MAXIMIZE} & \sum_{z\in\mathcal{Z}}\sum_{S\in\mathcal{S}}\sum_{i\in S}\eta^z r_i \phi_i^z(S) y^z(S) & & & \\
\text{SUBJECT TO:} & \sum_{z\in\mathcal{Z}}\sum_{S\in\mathcal{S}}\eta^z\phi_i^z(S)y^z(S) \leq & c_i & 1\leq i\leq n, & (\textsf{Primal-S}) \\
 & \sum_{S\in\mathcal{S}} y^z(S) = & \eta_z & \forall z\in\mathcal{Z}, & \\
 & y^z(S) \geq & 0 & \forall z\in\mathcal{Z}, S\in\mathcal{S} &
\end{array}
$$

In the above linear program, $y^z(S)$ is the probability of offering the set S to a customer of type z. As before, we also denote the optimal solution of the above linear program by Primal-S. Note that Lemma 1 provides a stronger upper bound since it holds for every customer sequence, while the above upper bound holds only in expectation. Theorem 1 provides a bound on the performance of our algorithms with respect to the upper bound in Lemma 1. When we have I.I.D. arrivals and use Primal-S as the benchmark, as stated by the theorem below, we obtain an even stronger performance guarantee for our algorithms.

Theorem 3 (Improved Performance Guarantee in the I.I.D. Arrival Model)**.** *Suppose that, in every period t, the type of an arriving customer is drawn independently and identically from a common distribution over the set of types $\mathcal{Z}$. In the asymptotic regime where c_{MIN} and T increase to infinity with $\frac{T}{c_{\text{MIN}}} = k$ for some positive integer k, then with high probability, the Inventory-Balancing algorithms with linear (LIB) and exponential (EIB) penalty functions satisfy the following inequalities:*

$$\lim_{T,c_{\text{MIN}}\to\infty}\frac{\mathbb{E}_{\{z_t\}_{t=1}^T}\left[\textsf{Rev}_{LIB}\left(\{z_t\}_{t=1}^T\right)\right]}{\textsf{Primal-S}} \geq 0.72 \qquad \text{and} \qquad \lim_{T,c_{\text{MIN}}\to\infty}\frac{\mathbb{E}_{\{z_t\}_{t=1}^T}\left[\textsf{Rev}_{EIB}\left(\{z_t\}_{t=1}^T\right)\right]}{\textsf{Primal-S}} \geq 0.75,$$

where the expectations in $\mathbb{E}_{\{z_t\}_{t=1}^T}[\cdot]$ is taken with respect to the sequence of arriving customers.

The proof is given in Section 6.3. The basic idea is to construct a (factor-revealing) linear program, denoted by FRLP. With high probability, every solution obtained by the Inventory-Balancing algorithm corresponds to a feasible solution of FRLP, and the objective corresponds to the ratio of the expected revenue $\mathbb{E}_{\{z_t\}_{t=1}^T}\left[\textsf{Rev}_{IB}\left(\{z_t\}_{t=1}^T\right)\right]$ of the IB algorithm and Primal-S. FRLP is parameterized by a discretization parameter ϵ. For each ϵ, we can solve the linear program to determine the lower bound on the competitive ratio.

4.1 Motivating IB Algorithms via Dual-Based Heuristics

In this section, we provide the motivation and intuition behind our Inventory-Balancing algorithm. The discounted revenue index in our IB algorithm is a proxy (approximation) for the dual variables. To see this, consider the following policy for I.I.D. arrival model.

1. Observe the type of the first ϵT customers.[11]

[10] See Gallego et al. (2004); Liu and van Ryzin (2008) for a similar linear programming formulation in the context of choice-based network revenue management.

[11] For the purpose of analysis, assume that no product is shown to this customer that is $S^t = \{0\}$ for $t \leq \epsilon T$.

2. Solve the dual of $\mathsf{Primal}\big(\{z_t\}_{t=1}^T\big)$ for the first ϵT customer:

$$
\begin{array}{llll}
\text{MINIMIZE} & \sum_{t=1}^{\epsilon T} \lambda^t + \sum_{i=1}^{n} \theta_i \epsilon c_i & & \\
\text{SUBJECT TO:} & \lambda^t \ \geq \ \sum_{i\in S} (r_i - \theta_i)\, \phi_i^{z_t}(S) & 1 \leq t \leq \epsilon T, S \in \mathcal{S}, & \\
& \theta_i \ \geq \ 0 & 1 \leq i \leq n. &
\end{array} \tag{1}
$$

Let $\theta_i^*(\epsilon T), i = 1, 2, \ldots, n$, be the solution of the linear program above.

3. For each subsequent customer of type $z \in \mathcal{Z}$, we offer an assortment S^t:

$$S^t = \operatorname{argmax}_{S\in\mathcal{S}} \sum_{i\in S} (r_i - \theta_i^*(\epsilon T))\phi_i^z(S)$$

Note that the algorithm does not need to know the distribution in advance. Following from the results of Devenur and Hayes (2009); Agrawal et al. (2009); Feldman et al. (2010), and Jaillet and Lu (2012), we can show that this algorithm is asymptotically optimal for I.I.D. stochastic arrivals.[12] Since the proof is similar to the existing literature, we omit the details.

Note that the selection rule of the above heuristic is similar to our algorithms by replacing $r_i - \theta_i^*$ with $r_i \times \Psi\left(I_i^{t-1}/c_i\right)$. In our Inventory-Balancing algorithms, however, we do **not** assume any patterns for the arrival of future customers. Thus, we can think of the discounted revenue index $r_i \times \Psi\left(I_i^{t-1}/c_i\right)$ as the "estimate" of the dual parameters based on the current inventory levels. This index value does not require any forecasting, and by choosing an appropriate penalty function Ψ, we can obtain an optimal competitive ratio, as shown by Theorem 1. See Feldman et al. (2010), who apply similar ideas to the online allocation of display advertisement.

5 Extensions

In this section, we discuss how our policies can be extended to more general settings and incorporate additional information about the customers' choice models or arrival patterns.

5.1 Incorporating Partial Information and Learning Customer Types

So far, we have assumed that, in each period, the algorithm knows the customer choice models. Namely, the $\phi_i^z(S)$ of the customer of type z is the *exact* value of the selection probability. However, this may not always be the case. The firm may learn the choice model associated with each customer type over time. For instance, in our numerical simulations, we associate the type of each customer to his or her location. We consider the case where the firm estimates the parameters of an MNL model for each location by learning from the purchases of the previous customers from that location.

Suppose that an arriving customer in period t is of type z. We denote by $\bar{\phi}_i^z(S)$ the true selection probability of product i when assortment S is offered to a customer of type z. In this environment, $\phi_i^{z_t}(S)$ represents the current estimation of the selection probabilities.[13] These estimates can be

[12] The dual heuristic is $1-O(\epsilon)$-competitive for a real-time assortment optimization problem, with high probability if: i) $\max\left\{\frac{r_i}{\mathsf{Primal}(\{z_t\}_{t=1}^T)}\right\} \leq \frac{\epsilon}{(n+1)(\ln(T)+\ln(2^n))}$ ii) $\frac{1}{c_{\text{MIN}}} \leq \frac{\epsilon^3}{(n+1)(\ln(T)+\ln(2^n))}$.

[13] Note that $\bar{\phi}_i^z(S)$ does not depend on the mechanism; however, $\phi_i^{z_t}(S)$ is a function of the mechanism since the estimations of the mechanism for each customer type depend on the assortments offered in the past.

obtained using partial information or historical data. We do not require specifics about how these estimations are made. However, a good example would be when the customer types are drawn from a stationary distribution and the parameters of the choice model are learned from observing customers' choices. Under standard assumptions, we expect that the estimated selection probabilities would converge to the true selection probabilities; see Section 5.2 for an example.

The following proposition provides a lower bound on the competitive ratio when we have estimation errors.

Proposition 1 (Competitive Ratio with Estimation Errors)**.** *For each t, let $\epsilon^t = \max_{i,S} |\phi_i^{z_t}(S) - \bar{\phi}_i^{z_t}(S)|$ be the random variable corresponding to the maximum estimation error in period t. Suppose that the Inventory-Balancing algorithm sells at least one unit of each product. Then, the competitive ratio of the Inventory-Balancing algorithm is at least equal to*

$$\min_{x \in \left[0, 1-\frac{1}{c_{\text{MIN}}}\right]} \left\{ \frac{1-x}{\frac{1}{c_{\text{MIN}}} + 1 - \Psi(x) + \int_{x+\frac{1}{c_{\text{MIN}}}}^{1} \Psi(y)dy + \frac{2}{c_{\text{MIN}}}\mathbb{E}\left[\sum_{t=1}^{T} \epsilon^t\right]} \right\}.$$

The proof is given in Appendix A. Note that, when there is no estimation error, i.e., $\phi_i^{z_t}(S) = \bar{\phi}_i^{z_t}(S)$, then the above expression is the same as the competitive ratio in Theorem 1. Furthermore, the only assumption made on the estimation errors is that the algorithm should sell at least one unit of each product. This assumption is made mainly for technical reasons to rule out the situation that the estimations are so far off that the algorithm never sells the products sold by the optimal solution (which knows the true estimations). We expect this condition to be satisfied when the estimation errors are small or if they vanish over time as the mechanism gathers more data about each type.

In the next section, we present an example that demonstrates how learning customer types can be incorporated in our framework. Furthermore, in Appendix B.3, using numerical simulations, we evaluate the performance of the IB algorithms when the selection probability for each customer type is unknown and must be estimated from data collected in earlier periods.

5.2 Learning the Customer Types under the Multinomial Logit Model

Suppose that the choice model of each customer type is described by a multinomial logit. If we show all products to m independent customers of type z and compute the maximum likelihood estimates $(V_0^z, \ldots, V_n^z)$, then it is a standard result that $\Pr\left\{\max_{i,S} |\phi_i^z(S) - \bar{\phi}_i^z(S)| > \delta\right\} \leq d\, e^{-\delta^2 m}$. Here, $\phi_i^z(S) = V_i^z/(V_0^z + \sum_{\ell \in S} V_\ell^z)$ and d is a constant, see, for example, Rusmevichientong et al., 2010. Therefore, $\mathbb{E}\left[\max_{i,S} |\phi_i^z(S) - \bar{\phi}_i^z(S)|\right] \leq \delta + d\, e^{-\delta^2 m}$.

Now consider the following variation of the IB algorithm: upon the arrival of the customer in period $1 \leq t \leq T$, of type z_t, with a probability of $0 < \gamma < 1$, we do exploration, i.e., we show all the products to the customer and with a probability of $1-\gamma$, we offer an assortment $S^t = \text{argmax}_{S \in \mathcal{S}} \sum_{i \in S} \Psi\left(I_i^{t-1}/c_i\right) r_i \phi_i^{z_t}(S)$, where $\phi_i^{z_t}(S)$ is the estimated selection probability, as described above, using previous sales data.

Note that the number of observations up to period t with high probability, is approximately $\theta(\gamma t)$. Hence, by setting $\delta = \frac{1}{t^{(1-\delta_1)/2}}$ and $m = \gamma t$ where $0 < \delta_1 < 1$, we have $\mathbb{E}\big[\max_{i,S} |\phi_i^{z_t}(S) - \bar{\phi}_i^{z_t}(S)|\big] = O\big(\frac{1}{t^{(1-\delta_1)/2}} + e^{-\gamma t^{\delta_1}}\big)$, which implies that $\mathbb{E}\left[\sum_{t=1}^{T} \epsilon^t\right] = o(T)$, i.e., $\lim_{x \to \infty} \mathbb{E}\big[\sum_{t=1}^{T} \epsilon^t\big]/T =$

0. Observe that as c_{MIN} and T proportionally grow, $\mathbb{E}\left[\sum_{t=1}^{T}\epsilon^t\right]/c_{\text{MIN}}$ approaches 0. Note that we allocate γ fraction of the inventory of each product for "exploration". Using Proposition 1 and the fact that algorithm loses at most a γ fraction of its revenue during explorations, the competitive ratio of the modified algorithm, as c_{MIN} and T proportionally tend to infinity, would be equal to $(1-\gamma)\alpha(\Psi)$.

Note that the modified algorithm, because of the constant rate of sampling, will still perform well if the choice models change slowly over time.

5.3 Incorporating (Uncertain) Information about Arrivale Pattens

The Inventory-Balancing algorithms do not rely on any forecast of future customer arrivals; however, if such a forecast exists, it could potentially be used to improve the performance of the algorithms. Consider a heuristic $\mathcal{L}$ such as the linear program re-optimization, that relies on the distribution (e.g., the estimated number) of the customers of each type. This heuristic will perform well if the estimations are accurate, but it performs poorly when the estimate turns out to be inaccurate or there is a high degree of uncertainty; see Section 7. We propose a family of algorithms called the *Hybrid* algorithm that combines the solution of such heuristics and IB algorithms. These algorithms incorporate additional information about the arrival sequence while maintaining a reasonable competitive ratio in unpredictable scenarios; see Mahdian et al. (2007, 2012).

The Hybrid algorithm, given below, is parameterized by a number $\gamma \geq 1$. This parameter controls the extent to which one would rely on heuristic $\mathcal{L}$.

<u>THE HYBRID ALGORITHM WITH PARAMETER γ</u>

Upon the arrival of a customer in period $t \in \{1,\ldots,T\}$, of type z_t:

- Let $S^t_{\mathcal{L}}$ be the set that heuristic $\mathcal{L}$ recommends in period t.
- Offer the assortment $S^t_{\mathcal{L}}$ if:

$$\gamma\left(\sum_{i\in S^t_{\mathcal{L}}}\Psi\left(I_i^{t-1}/c_i\right)r_i\phi_i^{z_t}(S)\right) \geq \max_{S\in\mathcal{S}}\left\{\sum_{i\in S}\Psi\left(I_i^{t-1}/c_i\right)r_i\phi_i^{z_t}(S)\right\}$$

- Otherwise, offer an assortment $S^t \in \operatorname{argmax}_{S\in\mathcal{S}}\sum_{i\in S}\Psi\left(I_i^{t-1}/c_i\right)r_i\phi_i^{z_t}(S)$.

The next proposition provides a lower bound on the competitive ratio of the Hybrid algorithm.

Proposition 2 (Competitive Ratio of the Hybrid Algorithm). *Suppose that Ψ is an increasing, concave, and twice-differentiable penalty function. As $c_{\text{MIN}}\to\infty$, the competitive ratio of the Hybrid algorithm with a penalty function Ψ and parameter γ and for any heuristic $\mathcal{L}$ is at least equal to $\alpha_\infty^\gamma(\Psi)$, where*

$$\alpha_\infty^\gamma(\Psi) = \min_{x\in[0,1]}\left\{\frac{1-x}{\gamma(1-\Psi(x))+\int_x^1\Psi(y)dy}\right\}$$

For example, for the exponential penalty function, $\alpha_\infty^\gamma\left(\frac{e}{e-1}\left(1-e^{-x}\right)\right)$, for $\gamma = 1.5$ and $\gamma = 2$, is approximately equal to 0.48 and 0.39, respectively. The proof is very similar to the proof of Theorem 1 and is omitted. The main idea is to assign $\lambda^t = \gamma\left(\sum_{i\in S^t} r_i \Psi\left(I_i^{t-1}/c_i\right)\phi_i^{z_t}(S^t)\right)$. Intuitively, we are extending the feasible region of the dual problem, which allows the algorithm to follow the heuristics on the recommendations that are considered "safe."

In our simulation results in Section 7.2 and the appendix, we consider a Hybrid algorithm that combines the EIB algorithm (Theorem 2) and the linear program re-optimization heuristic. We show that the Hybrid algorithm outperforms the Inventory-Balancing algorithm, when the number of customers is known in advance by the re-optimization policies. On the other hand, when the number of customers is uncertain, the Hybrid algorithm outperforms the re-optimization methods.

5.4 Beyond Substitutability

In this section, we explain how we can relax Assumption 1. Recall that, according to Lemma 2, Assumption 1 implies that our algorithm does not benefit from showing a product with no remaining inventory. However, sometimes the assumption may not hold such as when the dissimilarity parameters in the nested logit model are larger than 1 (Davis et al., 2011; Bhat, 2002) or there are externalities among the products[14].

More specifically, we assume that the choice model satisfies the following property: suppose that a customer is offered a set S and then she chooses product $i \in S$. Then, the customer buys product i if it has positive inventory, or she leaves without making a purchase. Under this choice model, we can use the Inventory-Balancing algorithm exactly in the same way as before. The inventory level of the products that are out of stock remains at 0 even if the product is shown to the customer.

In this model, the optimal revenue can be upper-bounded by the linear program below.

$$
\begin{array}{lrcll}
\text{MAXIMIZE} & \multicolumn{4}{l}{\sum_{t=1}^T \sum_{S\in\mathcal{S}} \sum_{i=1}^n r_i \phi_i^{z_t}(S) y^t(S) - \sum_i r_i w_i} \\
\text{SUBJECT TO:} & \sum_{t=1}^T \sum_{S\in\mathcal{S}} \phi_i^{z_t}(S) y^t(S) - w_i & \leq & c_i & 1 \leq i \leq n, \\
& \sum_{S\in\mathcal{S}} y^t(S) & = & 1 & 1 \leq t \leq T, \\
& y^t(S) & \geq & 0 & 1 \leq t \leq T, \quad S \in \mathcal{S}, \\
& w_i & \geq & 0 & 1 \leq i \leq n.
\end{array}
$$

This linear program is the same as the linear program $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$ given in Section 2, with a new set of variables: w_i denotes the number of times that a customer selects product i after its inventory hits 0. In this case, the product will not be allocated to the customer.

We now argue that the algorithm obtains the same competitive ratio as before. The argument is based on the following observation. The dual of the linear program above is as follows:

$$
\begin{array}{lrcll}
\text{MINIMIZE} & \multicolumn{4}{l}{\sum_{t=1}^T \lambda^t + \sum_{i=1}^n \theta_i c_i} \\
\text{SUBJECT TO:} & \lambda^t & \geq & \sum_{i=1}^n \phi_i^{z_t}(S)\left(r_i - \theta_i\right) & 1 \leq t \leq T, S \in \mathcal{S}, \\
& \theta_i & \leq & r_i & 1 \leq i \leq n, \\
& \theta_i & \geq & 0 & 1 \leq i \leq n.
\end{array}
$$

[14] For example, William Poundstone, in his book *Priceless*, documented the following case: "Williams-Sonoma added a \$429 breadmaker next to their \$279 model: sales of the cheaper model doubled even though practically nobody bought the \$429 machine." Thompson (2012).

Note that, compared to the previous dual in Section 6, the above linear program has a new set of constraints: $\theta_i \leq r_i$. However, these constraints are satisfied by our construction of the feasible solution in the proof of Theorem 1. Therefore, the ratio of the primal and dual solutions and the competitive ratio of the algorithm are the same as those described in Theorem 1.

6 Analysis

In this section, we prove our main theorems.

6.1 Proof of Theorem 1

We start with the following lemma proved in Appendix A.

Lemma 5. *For any increasing, concave, twice-differentiable penalty function* $\Psi : [0,1] \to [0,1]$*, the function* $x \mapsto \frac{1-x}{2-x-\Psi(x)}$ *increases on* $[0,1]$*, and for any* $a \in [0,1]$*, the function* $C \mapsto \frac{1}{C} + \int_{a+\frac{1}{C}}^{1} \Psi(y)dy$ *decreases on* $[1/(1-a), \infty)$*.*

Let $\{z_t\}_{t=1}^T$ be an arbitrary sequence of customers. Note that, according to Lemma 2, the Inventory-Balancing algorithm respects the capacity constraints of the problem. However, its solution may not correspond to a feasible solution of $\mathsf{Primal}\big(\{z_t\}_{t=1}^T\big)$. To compare the expected revenue of our algorithm with the upper bound given by $\mathsf{Primal}\big(\{z_t\}_{t=1}^T\big)$, we construct a sequence of *feasible dual solutions*. The dual of $\mathsf{Primal}\big(\{z_t\}_{t=1}^T\big)$ is given below:

$$\begin{array}{lllll} \text{MINIMIZE} & \sum_{t=1}^T \lambda^t + \sum_{i=1}^n \theta_i c_i & & & \\ \text{SUBJECT TO:} & \lambda^t \geq & \sum_{i=1}^n \phi_i^{z_t}(S)\,(r_i - \theta_i) & 1 \leq t \leq T, S \in \mathcal{S}, & (\mathsf{Dual}\left(\{z_t\}_{t=1}^T\right)) \\ & \theta_i \geq & 0 & 1 \leq i \leq n. & \end{array}$$

Based on the realization of customers' choices, we construct a feasible solution for the linear program $\mathsf{Dual}\left(\{z_t\}_{t=1}^T\right)$ as follows:

$$\begin{aligned} \theta_i &= r_i(1 - \Psi\left(I_i^T/c_i\right)) & i &= 1, 2, \ldots, n, \\ \lambda^t &= \sum_{i \in S^t} r_i \Psi\left(I_i^{t-1}/c_i\right) \phi_i^{z_t}(S^t) & t &= 1, 2, \ldots, T. \end{aligned}$$

Note that θ_i and λ^t are random variables because they depend on the inventory levels, which are random. However, they form a feasible solution for the dual with a probability of one because

$$\lambda^t \;=\; \sum_{i \in S^t} r_i \Psi\left(I_i^{t-1}/c_i\right) \phi_i^{z_t}(S^t) \;\geq\; \sum_{i \in S^t} r_i \Psi\left(I_i^T/c_i\right) \phi_i^{z_t}(S^t) \;=\; \sum_{i \in S^t} (r_i - \theta_i) \phi_i^{z_t}(S^t)\ ,$$

where the inequality follows from the fact that Ψ is increasing and $I_i^{t-1} \geq I_i^T$ an the equality follows from the definition of θ_i; that is $r_i \Psi(I_i^T/c_i) = (r_i - \theta_i)$.

We now calculate the expected value of this dual solution, which will provide an upper bound on the value of $\mathsf{Primal}\big(\{z_t\}_{t=1}^T\big)$ by the weak duality theorem. Since the sequence of the customers $\big(\{z_t\}_{t=1}^T\big)$ is fixed, the expectation is with respect to the realization of each customer's choice. Recall that Q_i^t is a binary random variable that is equal to 1 if the customer chooses product i in period t and 0 otherwise. Thus,

$$\mathbb{E}\left[\sum_{t=1}^{T}\lambda^t\right] = \mathbb{E}\left[\sum_{t=1}^{T}\sum_{S'\subseteq S^t}\sum_{i\in S'} r_i\Psi(I_i^{t-1}/c_i)\phi^{z_t}(S',S^t)\right] \quad = \quad \mathbb{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} r_i\Psi(I_i^{t-1}/c_i)Q_i^t\right]$$

$$= \mathbb{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{n} r_i\Psi(I_i^{t-1}/c_i)\left(I_i^{t-1}-I_i^t\right)\right] \quad = \quad \mathbb{E}\left[\sum_{i=1}^{n} r_i \sum_{t=I_i^T+1}^{c_i}\Psi(t/c_i)\right],$$

where the second equality follows from the tower property of the conditional expectation and the fact that $\mathbb{E}\left[Q_i^t | I_1^{t-1},\ldots,I_n^{t-1}\right] = \phi_i^{z_t}(S^t)$ since S^t is a function of $I_1^{t-1},\ldots,I_n^{t-1}$. The third equality follows from the observation that $I_i^{t-1} - I_i^t = Q_i^t$. The final equality follows because the k^{th} sold unit of product i contributes an amount of $\Psi\big((c_i-k+1)/c_i\big)$ to the summation.

Since θ_i and λ^t are dual feasible, it follows from the weak duality theorem that

$$\mathbb{E}\left[\sum_{t=1}^{T}\lambda^t + \sum_{i=1}^{n} c_i\theta_i\right] = \mathbb{E}\left[\sum_{i=1}^{n} r_i\left(\sum_{t=I_i^T+1}^{c_i}\Psi(t/c_i) + c_i\left(1-\Psi\left(I_i^T/c_i\right)\right)\right)\right] \geq \mathsf{Primal}\left(\{z_t\}_{t=1}^T\right).$$

On the other hand, the expected revenue of the Inventory-Balancing algorithm is equal to $\mathbb{E}\big[\sum_{i=1}^n r_i(c_i - I_i^T)\big]$. Therefore, the competitive ratio is at least

$$\frac{\mathbb{E}\left[\sum_{i=1}^n r_i(c_i-I_i^T)\right]}{\mathsf{Primal}^m\left(\{z_t\}_{t=1}^T\right)} \;\geq\; \frac{\mathbb{E}\left[\sum_{i=1}^n r_i(c_i-I_i^T)\right]}{\mathbb{E}\left[\sum_{i=1}^n r_i\left(\sum_{t=I_i^T+1}^{c_i}\Psi(t/c_i) + c_i\left(1-\Psi\left(I_i^T/c_i\right)\right)\right)\right]}.$$

Note that, if $I_i^T = c_i$, then the contribution of product i to both the revenue of our algorithm and to the constructed dual solution is zero. Therefore, the competitive ratio of the algorithm is at least

$$\min_{(c_i,I_i^T):I_i^T\le c_i-1} \frac{c_i - I_i^T}{\sum_{t=I_i^T+1}^{c_i}\Psi(t/c_i) + c_i\left(1-\Psi\left(I_i^T/c_i\right)\right)} = \min_{(c_i,x):x\;\le\;1-\frac{1}{c_i}} \frac{1-x}{\frac{1}{c_i}\sum_{t=I_i^T+1}^{c_i}\Psi(t/c_i) + (1-\Psi(x))},$$

where the equality follows from the variable transformation $x = I_i^T/c_i$. Because $\Psi(1)=1$ and Ψ is increasing, we have

$$\frac{1}{c_i}\sum_{t=I_i^T+1}^{c_i}\Psi(t/c_i) \;=\; \frac{1}{c_i}\left(1+\sum_{t=I_i^T+1}^{c_i-1}\Psi(t/c_i)\right) \;\leq\; \frac{1}{c_i} + \int_{\frac{I_i^T+1}{c_i}}^{1}\Psi(y)dy.$$

Putting everything together, we have the following lower bound on the competitive ratio:

$$\min_{(c_i,x)\in\mathbb{R}_+\times\left[0,1-\frac{1}{c_i}\right]}\left\{\frac{1-x}{\frac{1}{c_i}+1-\Psi(x)+\int_{x+\frac{1}{c_i}}^{1}\Psi(y)dy}\right\}$$

To complete the proof, it suffices to show that the above ratio is lower bounded by $\alpha_{c_{\text{MIN}}}(\Psi) := \min_{x\in\left[0,1-\frac{1}{c_{\text{MIN}}}\right]} \frac{1-x}{\frac{1}{c_{\text{MIN}}}+1-\Psi(x)+\int_{x+\frac{1}{c_{\text{MIN}}}}^{1}\Psi(y)dy}$ defined in Theorem 1. Consider an arbitrary $(c_i,x)\in$

$\mathbb{R}_+ \times \left[0, 1 - \frac{1}{c_i}\right]$. There are two cases to consider: $x \le 1 - \frac{1}{c_{\text{MIN}}}$ and $x > 1 - \frac{1}{c_{\text{MIN}}}$. In the first case, since the function $C \mapsto \frac{1}{C} + \int_{x+\frac{1}{C}}^{1} \Psi(y)dy$ decreases by Lemma 3 and $c_i > c_{\text{MIN}}$, we have

$$\frac{1-x}{\frac{1}{c_i} + 1 - \Psi(x) + \int_{x+\frac{1}{c_i}}^{1} \Psi(y)dy} \quad \ge \quad \frac{1-x}{\frac{1}{c_{\text{MIN}}} + 1 - \Psi(x) + \int_{x+\frac{1}{c_{\text{MIN}}}}^{1} \Psi(y)dy} \ \ge \ \alpha_{c_{\text{MIN}}}(\Psi).$$

In the second case, we have $x > 1 - \frac{1}{c_{\text{MIN}}}$. Recall that, to compute the minimum competitive ratio, we need to consider $I_i^T < c_i$; thus, $(c_i, x) \in \mathbb{R}_+ \times \left[0, 1 - \frac{1}{c_i}\right]$ and as a result, we have $x \le 1 - \frac{1}{c_i}$ or equivalently, $c_i \ge 1/(1-x)$. Applying Lemma 3 once again with $\frac{1}{1-x}$ as a lower bound for c_i, we get

$$\begin{aligned}
\frac{1-x}{\frac{1}{c_i} + 1 - \Psi(x) + \int_{x+\frac{1}{c_i}}^{1} \Psi(y)dy} \quad &\ge \quad \frac{1-x}{1 - x + 1 - \Psi(x) + \int_{x+(1-x)}^{1} \Psi(y)dy} \ = \ \frac{1-x}{2 - x - \Psi(x)} \\
&\ge \quad \frac{1 - \left(1 - \frac{1}{c_{\text{MIN}}}\right)}{2 - \left(1 - \frac{1}{c_{\text{MIN}}}\right) - \Psi(1 - \frac{1}{c_{\text{MIN}}})} \quad \ge \quad \alpha_{c_{\text{MIN}}}(\Psi) \ ,
\end{aligned}$$

where the second inequality follows from the fact that $x > 1 - \frac{1}{c_{\text{MIN}}}$ and Lemma 5, which shows that $\frac{1-x}{2-x-\Psi(x)}$ is increasing in x. This completes the proof.

6.2 Proof of Theorem 2

From Theorem 1 we have:

$$\begin{aligned}
\alpha(\Psi) \quad &= \quad \min_{x\in[0,1]} \left\{ \frac{1-x}{1 - \frac{e}{e-1}\left(1 - e^{-x} - \int_x^1 (1-e^{-y})dy\right)} \right\} \\
&= \quad \min_{x\in[0,1]} \left\{ \frac{1-x}{1 - \frac{e}{e-1}\left(1 - e^{-x} - 1 + x - e^{-1} + e^{-x}\right)} \right\} \\
&= \quad \min_{x\in[0,1]} \left\{ \frac{1-x}{1 - \frac{e}{e-1}\left(x - e^{-1}\right)} \right\} \quad = \quad \frac{e-1}{e} = 1 - \frac{1}{e} \ .
\end{aligned}$$

The second part of the theorem is followed from Lemma 3.

6.2.1 Proof of Lemma 3

Consider a setting with n products, indexed by $1, \cdots, n$, all with revenue equal to 1 and initial inventory of $\frac{1}{n}T$.[15] Think of T, the length of the horizon, as a very large number (that would tend to infinity) and a multiple of n. The number of types is equal to $2^n - 1$. Each type corresponds to

[15]The proof is built upon ideas from Mehta et al. (2007). Our analysis is different, more rigorous, and applies to smaller number of products. For instance, theirs omits the corresponding proof of Lemma 6 which we establish via induction, using the dynamic programming formulation of the problem.

a set $\Theta \neq \emptyset$ of products that a customer of that type equally likes; the "no-purchase" probability for all types is equal to zero.

The arrival process is defined as follows: customer arrives in n phases of equal length, that is, the number of customers in each phase is $\frac{T}{n}$. All the customers in each phase have the same type. We denote the type of the customer in phase j by Θ_j. We have $\Theta_1 = \{1, 2, \ldots n\}$; for j, $2 \leq j \leq n$, $\Theta_j = \Theta_{j-1} \setminus \{\theta_{j-1}\}$ where θ_{j-1} is a randomly chosen element of Θ_{j-1}. In other words, the set of products of interest to customer during phase j is the set of products of interest to customers in phase $j-1$ minus one of those products and θ_n is the only product of interest to customers in phase n, i.e., customers in phase j randomly lose interest in one of the products of interest in phase $j-1$. An example of sequences of customer types in n phases is $\Big\{\{1, 2, \ldots, n\}, \{1, 2, \ldots, n-1\}, \ldots, \{1, 2\}, \{1\}\Big\}$. Therefore, there are $n!$ sequences of customer arrivals, each with equal probability.

In Lemma 6, stated below, we show that the following Inventory-Balancing policy is optimal among all deterministic policies: offer to each customer all the products with the highest (positive) remaining inventory that are of interest to her.[16] The proof is given in Appendix A.

Lemma 6. *For the arrival process described in the proof of Lemma 3, the following inventory balancing algorithm is optimal among all deterministic policies: offer to the customer all the products with the highest (positive) remaining inventory that are of interest to her.*

Each customer purchases one of the products (if any) offered to her because the no-purchase probability is zero. Hence, the policy described above, in each phase, sells equal portion of the remaining inventory of each product that is of interest to the customers in that phase (which are all of the same type). For instance, in the first phase $\frac{1}{n}$ fraction of the inventory of every product is sold. Note that the rounding error is negligible since T is large. Recall that θ_i denotes the product that will be of no more interest to the customers arriving after and including phase $i+1$. Let $q_{i,j}$ be the fraction of customers in phase j that bought product θ_i. We have

$$q_{i,j} = \begin{cases} \frac{1}{n-j+1} & j \leq i \\ 0 & j > i \end{cases} \tag{2}$$

where $n-j+1$ is the number of products of interest to customers in phase j. Therefore, the revenue obtained from product θ_i is $\frac{1}{n}T\left(\min\left\{\sum_{j=1}^{i}\frac{1}{n-j+1}, 1\right\}\right)$ and consequently the total revenue of the policy above is equal to $\frac{1}{n}T\left(\sum_{i=1}^{n}\min\left\{\sum_{j=1}^{i}\frac{1}{n-j+1}, 1\right\}\right)$. On the other hand, the optimal clairvoyant solution that knows the customers types in advance sells all units of product θ_i to customers in phase i and obtains in total a revenue of T. This completes the proof.

6.3 Proof of Theorem 3

In this section, we discuss the performance guarantee of the EIB and LIB algorithms in the random arrival model that encompasses I.I.D. arrivals. [17] In the random arrival model the total number of

[16] We do not investigate that a randomized algorithm would be able to outperform the aforementioned policy, but we are not studying that questions in the paper.

[17] Consider the sample space of sequences of customer types in I.I.D. model and divide it into groups such that in each group the number of customers of each type is the same for every sequence. Since each group includes all the equally likely permutations of some sequence, every sequence of the customers (and their types) in the I.I.D. model can be mapped to a sequence of the customers in the random arrival model.

the customers T and number of customers of each type are chosen by an adversary, but the order of the arrivals is chosen uniformly at random. For this arrival model, we obtain the following result.

Proposition 3 (Performance Guarantee in the Random Arrival Model)**.** *Suppose the penalty function is exponential (EIB), $\Psi(x) = \frac{e}{e-1}(1-e^{-x})$, $x \in [0,1]$. Suppose $c_{\text{MIN}} \to \infty$ and $T \to \infty$ and choose the discretization parameter ϵ* [18] *such that $\frac{1}{\epsilon \times c_{\text{MIN}}} = O(1)$,* [19] *Then, the ratio of the expected revenue of the EIB algorithm, $\mathbb{E}_{\{z_t\}_{t=1}^T}\left[Rev_{EIB}\left(\{z_t\}_{t=1}^T\right)\right]$, to the expected revenue of the optimal solution,* Primal-S*, is bounded below by the solution of the following linear program.*

$$\text{MINIMIZE}_{\rho,\gamma,\chi} \qquad \chi(\frac{1}{\epsilon}) - \frac{e}{e-1}\sum_{j=0}^{\frac{1}{\epsilon}-1}\rho^{j,\frac{1}{\epsilon}}\left(\frac{j\epsilon}{e} - e^{j\epsilon-1} + e^{-1}\right) \qquad \text{(FRLP)}$$

$$\begin{aligned}
\text{SUBJECT TO:} \qquad & \sum_{j=0}^{\frac{1}{\epsilon}-1}\gamma^{j,k} = 1 && 0 < k \le \frac{1}{\epsilon},\\
& \gamma^{j,k} \le \rho^{j,k} && 0 \le j \le \frac{1-\epsilon}{\epsilon},\quad 0<k \le \frac{1}{\epsilon},\\
& \sum_{l=j}^{\frac{1}{\epsilon}-1}\rho^{l,k+1} \ge \sum_{l=j}^{\frac{1}{\epsilon}-1}\rho^{l,k} && 0 \le j \le \frac{1-\epsilon}{\epsilon},\quad 0<k \le \frac{1-\epsilon}{\epsilon},\\
& \sum_{j=0}^{\frac{1}{\epsilon}-1}\rho^{j,k}\int_{y=1-j\epsilon}^{1}\Psi(y)dy \le \chi(k) && 0<k \le \frac{1}{\epsilon},\\
& \epsilon\sum_{j=0}^{\frac{1}{\epsilon}-1}\gamma^{j,k+1}\Psi\big(1-(j+1)\epsilon\big) \le \chi(k+1) - \chi(k) && 0 < k \le \frac{1}{\epsilon} - 1
\end{aligned}$$

For the linear penalty function (LIB), $\Psi(x) = x$, the ratio of the expected revenue of the Inventory-Balancing algorithm, $\mathbb{E}_{\{z_t\}_{t=1}^T}\left[Rev_{LIB}\left(\{z_t\}_{t=1}^T\right)\right]$ to Primal-S *is at least equal to the solution of the linear program FRLP with objective function $\chi(\frac{1}{\epsilon}) + \frac{1}{2}\sum_{j=0}^{\frac{1}{\epsilon}-1}\rho^{j,\frac{1}{\epsilon}}(j\epsilon)^2$.*

The main idea is that the objective function corresponds to the revenue of the IB algorithms when the value of Primal-S is normalized to 1 and any solution of the IB algorithms corresponds to a feasible solution of FRLP. The complete proof is given in Appendix A.1 and is built upon the factor-revealing linear program proposed by Mirrokni et al. (2012), Jain et al. (2003); Mehta et al. (2007); Mahdian et al. (2012), and Mahdian and Yan (2011). In the remaining of this section, we present a sketch of the proof of Proposition 3.

The revenue of the IB algorithms is evaluated by *potential* function χ, which is an integral of the penalty function. Roughly speaking, the IB algorithms perform well if the final value of potential function $\chi(\frac{1}{\epsilon})$ is high. Thus, we keep track of the increase in the potential function during the run of the algorithms; due to discretization, we only measure the amount of increase in χ at $\frac{1}{\epsilon}$ points. The last set of constraints implies that the potential function, with high probability, increases after every $\frac{T}{\epsilon}$ steps. This is established using the fact that in any period t the IB algorithm chooses an assortment S^t that maximizes $\sum_{i \in S} r_i \Psi\left(I_i^{t-1}/c_i\right)\phi_i^{z_t}(S)$. Hence, we can lower bound the final value $\chi(\frac{1}{\epsilon})$ or equivalently the overall performance of the IB algorithms, as a function of the optimal solution, measured by $\gamma_{j,k}$s.

More precisely, for any run of the IB algorithm, we construct a solution for FRLP which is with high probability *feasible*. We discretize the fraction of remaining inventory into $\frac{1}{\epsilon}$ segments, i.e., segment j corresponds to interval $((1-(j+1)\epsilon)\, c_i, (1-j\epsilon)\, c_i]$ of the inventory. In this solution,

[18] That is, the horizon T is divided into $\frac{1}{\epsilon}$ time slots.

[19] We use the standard Landau notation: $f(n) = O(g(n))$ denotes $|f(n)| \le d|g(n)|$ where d is some constant.

$\rho^{j,k}$ is defined as the sum of maximum achievable revenue (revenue times initial inventory) of products which at time slot k, their remaining inventories fall into segment j. Similarly, $\gamma^{j,k}$ is equal to the total revenue obtained from product i in the optimal solution (of Primal-S), where $\frac{I_i^{kT\epsilon}}{c_i} \in (1-(j+1)\epsilon, 1-j\epsilon]$. Note that the first set of constraints implies that the value (revenue) of Primal-S is normalized to 1. By this construction, we have $\chi(k) = \sum_{i=1}^{n} r_i \int_{y=I_i^{kT\epsilon}}^{c_i} \Psi(y/c_i)dy$.

The optimal value of the linear program FRLP for the LIB algorithm for $\epsilon = \frac{1}{20}, \frac{1}{30}$ and $\frac{1}{50}$ are, respectively, approximately equal to 0.69, 0.71, and 0.72. The corresponding values for the EIB algorithm are respectively 0.72, 0.74, and 0.75. We observe that as ϵ gets smaller, the optimal value of FRLP becomes more accurate and converges to 0.72 and 0.75 respectively for the LIB and EIB algorithms.

6.4 The Complexity of the Inventory-Balancing Algorithm

The complexity of the Inventory-Balancing algorithm is determined by the complexity of solving the optimization problem $\max_{S\in\mathcal{S}} \sum_{i\in S} w_i \phi_i^{z_t}(S)$ where $w_i = r_i\, \Psi(I^{t-1}/c_i)$. As shown in the following examples, for a broad class of choice models, that satisfy Assumption 1, this problem can be solved efficiently.

Multinomial and Nested Logit: Under the multinomial logit (MNL) choice model,

$$\phi_i^z(S) = \begin{cases} v_i^z/(v_0^z + \sum_{\ell\in S} v_\ell^z) & \text{if} \quad i \in S\cup\{0\}, \\ 0 & \text{otherwise} \end{cases},$$

where for each $i \in \{0,1,2,\ldots,n\}$, $v_i^z \in \mathbb{R}_+$ denote the preference weight parameter associated with product i for a customer of type z. As shown in Talluri and van Ryzin (2004), the assortment optimization problem $\max_S \sum_{i\in S} w_i\phi_i^z(S)$ can be solved efficiently by simply sorting the products in a descending order of the marginal revenue w_i, and the optimal assortment can be found among the revenue-ordered assortments $\{1\}, \{1,2\}, \ldots, \{1,2,\ldots,n\}$. Li et al. (2013) showed that the above assortment optimization problem can also be solved in $O(dn\log n)$ operations for a d-level nested logit choice model, which generalizes to MNL to allow for a product taxonomy that is described by a tree with d levels (the MNL corresponds to the special case where $d = 1$).

Choice Models with Constraints Our formulation automatically allows for constraints on the assortments, through the specification of the set $\mathcal{S}$ of feasible assortments. For example, if we have a budget or shelf-space constraint, we can define $\mathcal{S} = \left\{S : \sum_{i\in S} d_i \le B\right\}$, where d_i is the cost (or space) associated with showing product i and B is the budget. When the underlying choice model is an MNL or a nested logit model, the assortment optimization problem can still be solved efficiently; see, for example, Rusmevichientong et al. (2009, 2010); Gallego and Topaloglu (2012); Farias et al. (2013).

General Choice Models Farias et al. (2013) has pioneered a novel algorithm that learns a general nonparametric choice model from transaction data. Under their framework, for each customer of type z, the choice model corresponds to a probability distribution η^z over the set of all permutations of $\{1,\ldots,n\}$ where for each i, $\sigma(i) \in \{1,2,\ldots,n\}$ denote the rank of product i, with 1 be the highest rank (most preferred). They assume that customer chooses the product with the highest rank. In this case,

$$\phi_i^z(S) = \sum_\sigma \eta^z(\sigma)\mathbb{1}\left[\sigma(i) \le \min_{k\in S\cup\{0\}} \sigma(k)\right].$$

It is easy to verify that $\phi_i^z(S) \ge \phi_i^z(S\cup\{j\})$ for $j \ne i$. So, this choice model satisfies Assumption 1. Moreover, Farias et al. (2011) described an algorithm for solving the assortment optimization problem $\max_S \sum_{i\in S} w_i\phi_i^z(S)$ under their general choice model framework. This algorithm which can be used as a subroutine in our Inventory-Balancing algorithm is efficient for MNL choice model.

7 Numerical Experiments

In this section, we numerically evaluate our Inventory-Balancing (IB) and Hybrid algorithms and compare them with existing methods in the literature. The first IB algorithm is the Linear Inventory-Balancing (LIB) with a linear penalty function $\Psi(x) = x$. The second algorithm is the Exponential Inventory-Balancing (EIB) with penalty function $\Psi(x) = \frac{e}{e-1}(1 - e^{-x})$.

In the next section, we describe the dataset and the simulation setting. Then, we compare the revenues and running times of our IB and Hybrid algorithms with the myopic policy and existing heuristics based on resolving linear programs. Section 7.3 shows the benefits of differentiating customers by type. Finally, in Section 7.4, we present a synthetic simulation to compare our algorithm with the myopic policy.

7.1 Dataset and Simulation Setting

We use the DVD sales data from a large online retailer and consider DVDs sold during a four-month period from June 1, 2005 through September 30, 2005. During this period, the retailer sold over 5.7 million DVDs in the United States, spanning $55,875$ DVD titles.

We consider the location of each customer as her *type*, and for our analysis, we choose 10 different locations that would reflect the diverse purchasing patterns of the customers. Table 2 shows the list of locations and the top three DVDs with the highest sale volumes in each location.[20] In our experiments, we consider a total of 73 DVD titles ($n = 73$), obtained by taking the union of the top 20 DVDs in each location and removing duplicates. For each DVD $i = 1, 2, \ldots, 73$, we set the revenue r_i as the average selling price of the DVD during this period. The DVD prices range from \$9-\$81, with more than 50% of the DVDs priced less than \$20, and the average and standard deviation of the DVD selling prices are \$25.2 and \$12.9, respectively. We assume that all DVDs have the same initial inventory, with no replenishment. Finally, we assume a multinomial logit choice model for each customer type. For each type $z \in \mathcal{Z}$, we estimate the preference weight parameters $(v_0^z, v_1^z, ..., v_n^z) \in \mathbb{R}_n^+$ by computing the maximum likelihood estimate based on the sales data for each location.

In the simulations, each problem class is characterized by two parameters: the *loading factor* (LF) and the *coefficient of variation* (CV) associated with the proportions of customers of each type. Note that the coefficient of variation is the ratio of the standard deviation to the mean; that is, for a given fixed mean, a larger CV means that there is a larger uncertainty in the type of an arriving customer. The *loading factor (LF)* is the ratio between the (expected) total number of customers

[20] Two of the names of the DVDs in Table 2 are shortened; "Lost" and "The Simpsons," respectively, correspond to "Lost - The Complete First Season" and "The Simpsons - The Complete Sixth Season."

Location	Top Three DVDs
Jersey City, NJ	Lost; Sin City; The Simpsons
Manhattan NY	Lost; Sin City; Bob Dylan - No Direction Home
Orlando, FL	Lost; The Muppet Show: Season One; Firefly
Miami, FL	What the Bleep Do We Know!?; Sin City; Lost
San Jose, CA	Star Wars: Episode III; Lost; The Complete Thin Man Collection
Beverly Hills, CA	The Simpsons; L'Auberge Espagnole; Sin City
Inglewood, CA	Lost; Sin City; Final Fantasy VII - Advent Children
Dallas, TX	My Big Fat Greek Wedding; Lost; Shark Tale
Urbana Champaign, IL	Firefly; Lost; The Simpsons
Charlotte, NC	Lost; Curb Your Enthusiasm; Family Guy Presents Stewie Griffin

Table 2: The locations used in the simulations and the top three best-selling DVDs in each location.

and the total number of units. For a given loading factor, we can determine the (expected) total number of customers. The number of customers of *each type* is generated by multiplying the total number of customers by a 10-dimensional Dirichlet random variable $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_{10})$, where β_i represents the proportion of customers of type i.[21] Thus, on average, each type is equally likely.[22] Once the number of customers of each type is generated, the order of arrivals is random. We further assume that a single customer arrives in each period.

7.2 Performance Evaluation

In this section, we compare the performance of our two IB algorithms to the Hybrid algorithm, the myopic policy, and the LP-based heuristics.

The myopic policy offers the assortment with the maximum expected revenue among all assortments that include only products with positive remaining inventory.[23]

The first LP-based algorithm, called the *LP One-Shot* (LPO), is a heuristic based on the solution of the linear program $\mathsf{Primal\text{-}S}$ where the expected number of each customer type η^z is equal to $0.1 \times \mathbb{E}[T]$.[24] This follows because each of the 10 customer types is, on average, equally likely, and $\mathbb{E}[T]$ is the expected number of customers. Although the linear program (LP) $\mathsf{Primal\text{-}S}$ has exponentially many variables, we can solve it efficiently using techniques from Gallego et al. (2004); Liu and van Ryzin (2008), and Topaloglu (2013). Under the LPO, we solve the linear program $\mathsf{Primal\text{-}S}$ exactly once and denote the solution by $\{\bar{y}^z(S) : z \in \mathcal{Z}, S \in \mathcal{S}\}$. The LPO policy is constructed as follows: upon the arrival of a customer of type z, offer her assortment S with the probability $\bar{y}^z(S) / \sum_{S' \in \mathcal{S}} \bar{y}^z(S')$.

[21] For a k-dimensional Dirichlet distribution $\boldsymbol{\beta}$ with parameters $\alpha_1, \ldots, \alpha_k$, the mean and variance of β_i for $i = 1, 2, \ldots, k$ are respectively given by $\frac{\alpha_i}{\alpha_0}$ and $\frac{\alpha_i(\alpha_0 - \alpha_i)}{\alpha_0^2(\alpha_0+1)}$, where $\alpha_0 = \sum_{j=1}^k \alpha_j$. Given that $\alpha_1 = \alpha_2 = \ldots = \alpha_k = \alpha$, the coefficient of variation is $\sqrt{\frac{k-1}{k\alpha+1}}$. The parameter of the Dirichlet distribution, α, is chosen so that $\mathbb{E}[\beta_i] = 0.1$ for all i and β_i has the desired coefficient of variation.

[22] We get similar results when each type is not equally likely.

[23] The myopic policy corresponds to an Inventory-Balancing algorithm with a penalty function $\Psi(x) = \mathbb{1}[x > 0]$.

[24] When T is known, $\mathbb{E}[T]$ is replaced by T.

We also consider an *adaptive* variation of this policy, called ALPO, that excludes any product with zero remaining inventory from an offered set. Note that ALPO is an adaptive policy because the assortment that we offer to each customer may change over time depending on the product availability.

The second LP-based algorithm is called *LP Resolving* (LPR). Under this algorithm, the linear program Primal-S is resolved periodically every h periods with the up-to-date inventory levels and forecasts of the proportion of customers of each type. We denote this heuristic by LPR^h. In this heuristic, at the beginning of a resolving period t, η^z is replaced by $\eta^z(t)$. We define $\eta^z(t)$ as *the empirically estimated* expected number of customers of type z that will arrive between periods t and T, $\eta^z(t)$ is equal to the fraction of customers of type z that arrived between time 1 and $t-1$ multiplied by $\frac{\mathbb{E}[T-t+1|T\geq t]}{t-1}$, where $\mathbb{E}[T-t+1|T \geq t]$ is the expected number of future customers given that t customers have already arrived.[25] In addition, we set $\eta^z(1) = 0.1$. As the system evolves, the realized number of customers of each type will differ from the expected value, and $\eta^z(t)$ incorporates these differences into the estimates. The LPR algorithm also updates the inventory level to reflect the current remaining inventory of each product.

The Hybrid algorithm combines IB algorithms with LP based heuristics; see Section 5.3.[26] In particular, we assume that the Hybrid algorithm incorporates the solutions of the EIB algorithm and the LPR^{500} heuristic with parameter $\gamma = 1.5$ and 2. For $\gamma = 1.5$ and $\gamma = 2$, the competitive ratio of the Hybrid algorithm is at least equal to 0.48 and 0.39, respectively. Recall that parameter γ should be greater than 1 and that, the larger γ is, the more the Hybrid algorithm applies the LPR heuristic. We denote the Hybrid algorithm with parameter γ by $\mathcal{H}_\gamma$.

We evaluate the aforementioned algorithms by comparing the revenue with the upper bound given by a *clairvoyant optimal solution* that knows the arrival sequence of all the customers in advance; see Lemma 1 for more details.

Simulation Parameters: We consider nine problem classes, corresponding with loading factors 1.4, 1.6, and 1.8, and coefficients of variation of 2.0, 1.0, and 0.1. We choose the loading factor greater than 1 because we are interested in cases where demand exceeds supply. Each problem class consists of 250 problem instances. In each problem instance, we set the initial inventory levels to 100; i.e., $c_i = 100$, $i = 1, 2, \ldots, 73$, and the length of the horizon is randomly and uniformly chosen from an interval $\left[\underline{T}, \bar{T}\right]$.[27] In Table 3, for each problem class, we present the average revenue of each algorithm as a percentage of the upper bound, averaged over 250 instances in each problem class. We present the results for periodic LPR with $h = 50$ and $h = 500$.

As expected, the upper bound increases as the loading factor increases and the variability decreases. Both the LIB and EIB algorithms surpass all other policies. In addition, LPR policies cannot obtain more than 92% of the upper bound even if we increase the frequency of resolving.[28] In all cases, the $\mathcal{H}_{1.5}$ algorithm performs better than the $\mathcal{H}_2$ algorithm because the larger γ is, the more the Hybrid algorithm relies on the LPR heuristic. Note that the Hybrid algorithms get

[25] Note that, when the length of the horizon is known, $\mathbb{E}[T-t+1|T \geq t]$ is simply $T-t+1$.

[26] We do not present the results for the Hybrid algorithm that combines an LIB solution with the LP resolving heuristics since its performance is very close to the one we considered here.

[27] We choose the interval $\left[\underline{T}, \bar{T}\right]$ by first computing the expected number of customers $\mathbb{E}[T]$ from the loading factor and the total initial inventory of each problem class. Then, we set $\underline{T} = 0.5\mathbb{E}[T]$ and $\bar{T} = 1.5\mathbb{E}[T]$.

[28] When the degree of uncertainty in the number of customers drops the LPR heuristic performs better; e.g., when $\bar{T} = 1.3 \times E[T]$, $\underline{T} = 0.7 \times \mathbb{E}[T]$, LF=1.4 and CV=1, the LPR^{500} obtains 94% of the upper bound. Even in this case, IB algorithms beat the LPR heuristic by 3%.

Problem Class		Upper Bound,	Average Revenue under Different Policies (as % of the Upper Bound)								
			Inventory-Balancing		Myopic	One-shot LP		LP Resolving		Hybrid	
LF	CV	(in \$1000)	EIB	LIB	Policy	LPO	ALPO	LPR^{500}	LPR^{50}	$\mathcal{H}_{1.5}$	$\mathcal{H}_2$
1.4	2.0	165	97.0	96.9	96.4	73.6	81.4	91.3	91.3	96.5	95.8
	1.0	169	96.8	96.9	95.8	84.3	88.4	89.8	90.0	96.6	96.5
	0.1	172	97.5	97.5	96.4	92.8	93.5	90.9	90.8	96.8	96.7
1.6	2.0	170	97.3	97.3	96.8	66.7	73.6	88.5	88.6	96.8	95.9
	1.0	174	97.5	97.6	96.5	82.5	86.2	88.1	88.1	97.2	96.2
	0.1	178	98.4	98.5	97.3	91.8	92.4	89.5	89.7	98.3	97.7
1.8	2.0	174	98.0	97.9	97.2	62.9	68.8	86.9	87.0	97.9	97.2
	1.0	178	98.0	98.1	97.1	78.2	81.9	86.5	86.6	98.1	97.9
	0.1	178	97.8	97.9	96.7	85.1	85.2	86.1	86.3	97.8	97.5

Table 3: Revenue comparison when the length of the horizon is random. The standard errors of all numbers are less than 0.1%.

5%-12% more revenue than the LP resolving heuristics. This highlights the benefits of combining the solution of the IB policies and LP-based heuristics. As we will discuss later, we also consider the worst-case performance of all policies; see Figure 1 and Appendix B.1.

In most problem classes, the LPO algorithm has the lowest revenue and its performance decreases by increasing CV and loading factor. The main reason for the poor performance of the LPO algorithm is the lack of strategy adjustment. The LPO algorithm solves the linear program exactly once and does not update its decision based on the current inventory level (it just checks whether the product is available in the case of ALPO), nor does it consider the proportion of each customer type. When the uncertainty in the type of arriving customers and the number of customers is high, this lack of adjustment can result in a significant reduction in the performance of the LPO algorithm; e.g., for CV=0.1 and loading factor of 1.4, the ALPO algorithm obtains 93.5% of the optimal solution, while for $CV = 2$, it yields 81.4% of that. Note that the ALPO algorithm that adjusts its offered set based on the product availability (offering only those with positive inventories) yields a 1%-8% increase in revenue compared with the non-adaptive LPO algorithm. This also emphasizes the need for a real-time algorithm that adjusts its strategy based on the remaining product inventory and customers' choices.

Explanation of LPR performance: In Appendix B.2, we show that, when the horizon length T —corresponding to the total number of customers—is known in advance, the LPR heuristic performs very well and beats all other policies. We also observe that the LPR revenue increases linearly over time, which shows that the LPR method exploits the known length of the horizon to *ration* the inventory to *all* customers effectively. However, when T is random and the length of the horizon is less than the expectation, the LPR heuristic may end up with rather large left over inventories (of more profitable products). On the other hand, if T is larger than the expectation, then the LPR heuristic may run out of some the products, and customers will face stock-out.

We note that, in all problem classes, the myopic policy performs well because there is only a small chance that a customer will not purchase a product; that is, most of the preference weight parameters of the MNL model are positive for all customer types. Thus, if the myopic policy runs out of the most profitable products quickly by offering them to customers, then it can still collect high revenues because future customers continue to like the rest of the available products. To demonstrate this point, in Section 7.4, we conduct a simulation on synthetic data where each customer type is interested in only a subset of products, and the preference weights are small for most products. We observe that, in this setting, our IB algorithms can beat the myopic policy by up to 8%.

Running Time: Although the LPR^{50} and LPR^{500} heuristics do not solve the linear program very often, they are still 25 and 4 times slower than the Inventory-Balancing algorithms, respectively. We note that, in the special case of the multinomial logit choice model, the linear program Primal-S with $O(2^n)$ variables and $O(|\mathcal{Z}|)$ can be reduced to an equivalent LP with just $O(n|\mathcal{Z}|)$ variables and $O(n|\mathcal{Z}|)$ constraints Topaloglu (2013). However, for the general model where this reduction is not possible, the LPR heuristic will be even slower.

Distribution of Revenues across Problem Instances: To get more insight into the performance of different policies, in Figure 1, we depict the Complementary Cumulative Distribution Function (C.C.D.F.) of revenue as a percentage of the optimal clairvoyant revenues across 250 problems instances in the problem class with LF= 1.8 and CV= 2 for the myopic, LIB, EIB, LPR^{500}, LPO, ALPO, and $\mathcal{H}_{1.5}$ algorithms. In Figure 1, the curve for each algorithm shows, for $\alpha \in [90, 100]$, the fraction of problem instances whose revenues are at least α% of the upper bound. It can be seen that both IB polices almost dominate all other policies stochastically.[29] In addition, the myopic and Hybrid policies stochastically dominate the LPR^{500} heuristic, which demonstrates the weak performance of LPR heuristics in the worse-case scenario, when there is uncertainty in the number of customers. We observe that, for 80% of problem instances, the EIB and LIB algorithms obtain more than 95.8% of the optimal clairvoyant revenue. However, the LPR^{500} heuristic yields more than 95.8% of the upper bound only for about 33% of all instances.

7.3 Benefits of Personalization: The Importance of Knowing Customer Types

To quantify the benefits of personalization, we compare the revenue in two cases: *multiple-type* versus *single-type*. We consider the same 10 locations as in Section 7.1. The total number of DVDs is 73, corresponding to the union of the top 20 DVDs with the highest sales volumes in each location. The initial inventory of each DVD is set at 30. The setting for the multiple-type case is exactly the same as Section 7.1, where we estimate the parameters $(v_0^z, v_1^z, \ldots, v_n^z)$ for each type z and customize the decision to the type of each arriving customer. On the other hand, in the single-type case, we estimate a single parameter vector $(v_0, v_1, \ldots, v_n)$ and use it to make decisions for all customers. In our simulation, when we compute the revenue, we assume that each customer makes a decision based on her own type. Since the multiple-type model is more accurate, we expect that the multiple-type setting will yield higher revenue, but the key question is the magnitude of the improvements.

[29] Policy A stochastically dominates policy B if for any x, $\bar{F}_A(x) \geq \bar{F}_B(x)$ where $\bar{F}_A$ and $\bar{F}_B$ is the complementary cumulative distribution function of revenue (as a % of the upper bound) for policy A and B, respectively. Note that, when policy A dominates policy B, the worst-case performance of policy A is also better (higher) than that of policy B.

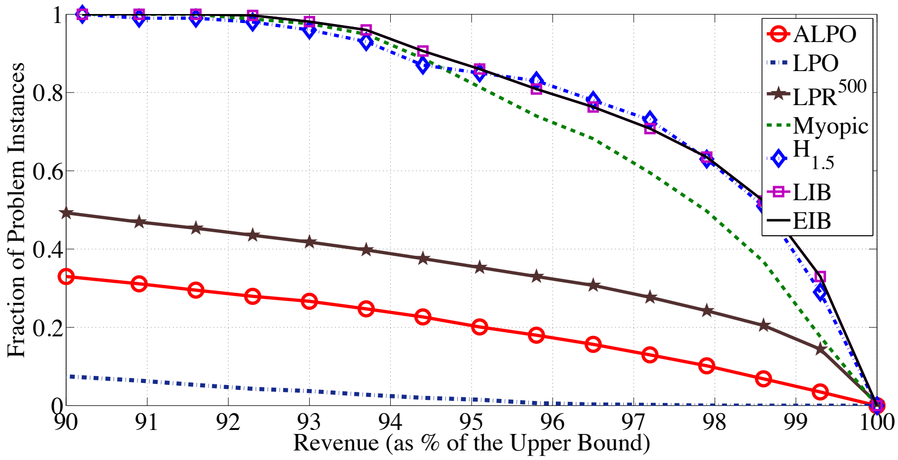

Figure 1: The C.C.D.F. of revenues with LF= 1.8 and CV= 2. For each algorithm, the curve shows the fractions of problem instances (out of 250) whose revenues exceed certain percentages of the upper bound.

We impose a constraint C on the size of the assortment that we can offer to each customer, and we consider $C \in \{10, 20\}$. Columns 4 and 5 in Table 4 show the revenues of the IB algorithms in the multiple-type case for different values of the size constraints. The corresponding difference between the revenues in the multiple-type and single-type cases is reported in columns 6 and 7. We consider the loading factors of 1.2, 1.4, and 1.6, but set the coefficient of variation at 0.2 because the results for the other coefficients of variations are similar.[30] As expected, the revenue from the multiple-type case exceeds that of the single-type, regardless of the algorithms we use. However, the benefits of customization are very significant when the assortment size is small. For example, for a loading factor of 1.6 and $C = 10$, using the LIB algorithm in the multiple-type case yields an improvement of over 21%. As the size of the assortment increases, however, the benefits of customization decreases.

7.4 Comparison to Myopic Policy

Theoretically, it is easy to show that our algorithm will perform significantly better than the myopic policy. Using the experiment on real data (for choice models) in Section 7.2, we also observe an improvement of 1%-2% over the myopic policies. In this section, we run a synthetic simulation in which our algorithm can get up to 7.4% more revenue compared to th myopic policy. In this simulation, there are 73 DVDs and 10 types. As in our simulations, for each DVD $i = 1, 2, \ldots, 73$, the revenue r_i is the average selling price of the DVD obtained from the dataset, and the initial inventory level c_i is set to 30. We also assume that customers purchase according to the MNL model. However, in this simulation, we do not obtain the preference weight parameters, $(v_0^z, v_1^z, ..., v_n^z)$ for each type $z \in \mathcal{Z}$, from the dataset. Rather, we order the products based on their prices so that product 1 has the highest price and product 73 has the lowest price. We assume that, for each of the

[30]Note that we did not use the optimal solution because it is rather challenging to solve the corresponding linear programs with a constrained assortment size. When there is no restriction on the size of the assortment, though the LP has exponentially many variables, we can solve it efficiently using the techniques from Topaloglu (2013).

Loading Factor (LF)	Assort-ment Size (C)	Upper Bound, (in $100)	Revenue in the Multiple-type Case (as % of the Upper Bound)		Improvements over the Single-type Case (as % of the Upper Bound)	
			LIB	EIB	LIB	EIB
1.2	10	526	82.3	82.5	17.4	15.4
	20		90.8	90.7	11.6	9.8
1.4	10	546	88.0	88.0	19.3	17.5
	20		95.9	95.6	11.1	10.0
1.6	10	549	95.3	94.9	21.3	19.2
	20		98.9	98.7	6.5	5.6

Table 4: The average revenue for the IB algorithms in the multiple-type model and improvement over the single-type model. All numbers are statistically significant, and the standard errors are less than 0.1%.

first nine customer types ($z \in \{1, 2, \ldots, 9\}$), the customer type z is only interested in the products indexed by $A^z = \{1, 2, \ldots, 7z\}$, with $v_i^z = 1$ for $i \in A^z$, and $v_i^z = 0.001$ for $i \notin A^z$. Customers of type $z = 10$ are likely interested in all products equally. We further assume that $v_0^z = 1$ for all 10 types. The way we construct the preference weight parameters reflects the fact that some customers are interested only in recently released DVDs that are of course more expensive, while some customers are less sensitive to this issue. The arrival process is exactly the same as indicated in previous sections; namely, the number of customers of each type is computed by multiplying the total number of customers by a 10-dimensional Dirichlet random variable. After generating the number of customers of each type, we uniformly permute them to determine the order of arrival.

Table 5 compares different policies in this setting when the length of the horizon is drawn from a uniform distribution with $\bar{T} - \underline{T} = \mathbb{E}[T]$ for coefficients of variations 0.5 and 1.0 and loading factors of 1.2, 1.4, and 1.6. We observe that the both the EIB and LIB algorithms beat the myopic policy by 4.5%-7.4%. In addition, the IB policies surpass all other policies as well.

Problem Class		Upper Bound,	Avg. Revenue under Different Policies (as % of the Upper Bound)					
			Inventory-Balancing		Myopic	One-shot LP		LP Resolving
LF	CV	(in $100)	EIB	LIB	Policy	LPO	ALPO	LPR^{500}
1.2	1.0	499	95.5	96.0	90.1	65.2	66.6	90.5
	0.5	516	94.9	95.5	88.1	80.8	82.2	90.8
1.4	1.0	514	96.1	96.6	90.8	63.9	65.3	89.8
	0.5	530	95.6	96.2	89.5	79.4	80.7	90.7
1.6	1.0	524	96.8	97.3	92.3	71.3	72.7	91.3
	0.5	538	96.5	97.0	90.9	88.7	89.8	92.7

Table 5: Revenue Comparison. The standard errors of all numbers are less than 0.1%.

8 Conclusion and Future Work

Motivated by the availability of instantaneous data on customer characteristics, we formulated a real-time, personalized, choice-based assortment optimization problem with arbitrary customer types. Our proposed Inventory-Balancing algorithms are simple, intuitive, and effective. We establish the competitive ratio for our algorithms and prove that it is the best possible for this problem.

The managerial insight from our work is that companies can increase their revenues by personalizing the operational decisions for each customer using real-time information about the customer's characteristics. This process requires coordination between front-end customer-facing decisions with back-end supply chain constraints. We demonstrate that such coordination can be achieved using simple index-based algorithms that can be easily implemented. As the volume and speed with which real-time data become available increase, the opportunity in this area will continue to grow, and we believe that this work can serve as a starting point for more complex models.

Our proposed Inventory-Balancing algorithms maintain an index for each product in which the marginal revenue is discounted by a penalty based on the product's remaining inventory. The index serves as a simple mechanism for coordinating fast-moving customer-facing decisions and the back-end operational constraints. In addition to inventory, it would be interesting to consider other supply chain constraints. For example, in network revenue management, each product corresponds to an itinerary, and it uses a common pool of resources, corresponding to seats on flights. When resources are shared among products, coordinating and valuing the benefit of each additional unit of a resource becomes challenging. One way to extend our index-based framework is to consider an index that depends on the inventory of all resources simultaneously.

Another exciting direction is to model explicitly the mechanism for learning the choice model of each customer. Recent advances by Farias et al. (2013) have allowed us to model and estimate a very rich class of choice models, and it would be interesting to understand how we can incorporate the learning mechanism with the real-time decision-making process considered in this paper. Finally, it would be interesting to consider the personalization of other operational decisions, such as pricing, warranty services, or shipping options. We believe that the framework in this paper can serve as a starting point for analyzing the personalization of these decisions.

Acknowledgements

We are grateful to Dimitris Bertsimas, the Associate Editor, and the three referees for their detailed and insightful comments, which greatly improved the results and presentation of the paper. We also would like to thank Vivek Farias, Stefanus Jasin, and Huseyin Topaloglu for their fruitful comments and discussions. The research has been supported in part by the National Science Foundation through grants CMMI-1158658, CMMI-1158659, and CMMI-1157569.

References

J. Acimovic and S. Graves. Making better fulfillment decisions on the fly in an online retail environment. Working Paper, MIT, 2011.

S. Agrawal, Z. Wang, and Y. Ye. A dynamic near-optimal algorithm for online linear programming. Working paper, Stanford University, 2009.

Amazon's Recommendation Systems. How does the amazon recommendations system work? `http://www.amazon.com/gp/help/customer/display.html/ref=pd_ys_help_iyr?ie=UTF8&nodeId=13316081`, 2012.

Y. Azar, B. E. Birnbaum, A. R. Karlin, and C. T. Nguyen. On revenue maximization in second-price ad auctions. In Amos Fiat and Peter Sanders, editors, *ESA*, volume 5757 of *Lecture Notes in Computer Science*, pages 155–166. Springer, 2009. ISBN 978-3-642-04127-3.

Michael O Ball and Maurice Queyranne. Toward robust revenue management: Competitive analysis of online booking. *Operations Research*, 57(4):950–963, 2009.

F. Bernstein, A. G. Kök, and L. Xie. Dynamic assortment customization with limited inventories. Working Paper, Duke University, 2011.

Omar Besbes and Denis Sauré. Dynamic pricing strategies in the presence of demand shocks. Working Paper, 2012.

Omar Besbes and Assaf Zeevi. On the minimax complexity of pricing in a changing environment. *Operations Research*, 59(1):66–79, February 2011.

Chandra R. Bhat. Recent methodological advances relevant to activity and travel behavior analysis. *In Perpetual Motion: Travel Behavior Research Opportunities and Application Challenges, edited by H.S. Mahmassani, Pergamon*, pages 381–414, 2002.

N. Buchbinder and J. Naor. The design of competitive online algorithms via a primal-dual approach. *Foundations and Trends in Theoretical Computer Science*, 3(2–3):93–263, 2007.

N. Buchbinder, K. Jain, and J. Naor. Online primal-dual algorithms for maximizing ad-auctions revenue. In Lars Arge, Michael Hoffmann, and Emo Welzl, editors, *ESA*, volume 4698 of *Lecture Notes in Computer Science*, pages 253–264. Springer, 2007. ISBN 978-3-540-75519-7.

C. W. Chan and V. F. Farias. Stochastic depletion problems: Effective myopic policies for a class of dynamic optimization problems. *Mathematics of Operations Research*, 34(2):333–350, 2009.

Y. Chen and V. F. Farias. Simple policies for dynamic pricing with imperfect forecasts. *Operations Research*, 61(3):612–624, 2013.

D. F. Ciocan and V. F. Farias. Dynamic allocation problems with volatile demand. To appear in *Mathematics of Operations Research*, 2013.

S. Clifford. Shopper alert: Price may drop for you alone. *The New York Times, August 9*, 2012.

J. Davis, G. Gallego, and H. Topaloglu. Assortment optimization under variants of the nested logit model. Working Paper, Cornell University, 2011.

Nikhil R. Devenur and Thomas P. Hayes. The adwords problem: online keyword matching with budgeted bidders under random permutations. In *Proceedings of the 10th ACM Conference on Electronic Commerce*, EC '09, pages 71–78, New York, NY, USA, 2009. ACM.

V. F. Farias, S. Jagabathula, and D. Shah. Assortment optimization under a general choice model. Working Paper, MIT, 2011.

Vivek F. Farias, Srikanth Jagabathula, and Devavrat Shah. A nonparametric approach to modeling choice with limited data. *Management Science*, 59(2):305–322, February 2013. ISSN 0025-1909.

J. Feldman, M. Henzinger, N. Korula, V. Mirrokni, and C. Stein. Online stochastic packing applied to display ad allocation. *Algorithms–ESA 2010, Springer*, pages 182–194, 2010.

G. Gallego and H. Topaloglu. Constrained assortment optimization for the nested logit model. Working Paper, Cornell University, 2012.

G. Gallego, G. Iyengar, R. Phillips, and A. Dubey. Managing flexible products on a network. Working Paper, Columbia University, 2004.

Guillermo Gallego and Garrett van Ryzin. On the relationship between inventory costs and variety benefits in retail assortments. *Management Science*, 40(8):999–1020, 1994.

V. Gaur and D. Honhon. Assortment planning and inventory decisions under a locational choice model. *Management Science*, 52(10):1528–1543, 2006.

Ashish Goel, Mohammad Mahdian, Hamid Nazerzadeh, and Amin Saberi. Advertisement allocation for generalized second-pricing schemes. *Operations Research Letters*, 38(6):571–576, 2010.

V. Goyal, R. Levi, and D. Segev. Near-optimal algorithms for the assortment planning problem under dynamic substitution and stochastic demand. Working Paper, Columbia University, 2011.

Miguel Helft and Tanzina Vega. Retargeting ads follow surfers to other sites. *http://www.nytimes.com/2010/08/30/technology/30adstalk.html*, August 30 2010.

D. Honhon, V. Gaur, and S. Seshadri. Assortment planning and inventory decisions under stock-out based substitution. *Operations Research*, 58(5):1364–1379, 2010.

P. Jaillet and X. Lu. Near-optimal online algorithms for dynamic resource allocations. Working Paper, MIT, 2012.

Kamal Jain, Mohammad Mahdian, Evangelos Markakis, Amin Saberi, and Vijay V. Vazirani. Greedy facility location algorithms analyzed using dual fitting with factor-revealing lp. *J. ACM*, 50(6):795–824, 2003.

S. Jasin and S. Kumar. A re-solving heuristic with bounded revenue loss for network revenue management with customer choice. *Mathematics of Operations Research*, 37(2):313–345, 2012.

Bala Kalyanasundaram and Kirk Pruhs. An optimal deterministic algorithm for online b-matching. *Theorictal Computer Science*, 233(1-2):319–325, 2000.

A. G. Kök, M. Fisher, and R. Vaidyanathan. Assortment planning: review of literature and industry practice. In *Retail Supply Chain Management*. Springer, 2008.

G. Li, P. Rusmevichientong, and H. Topaloglu. The d-level nested logit model: Assortment and price optimization problems. Working Paper, Marshall School of Business, 2013.

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

S. Mahajan and G. J. van Ryzin. Stocking retail assortments under dynamic consumer substitution. *Operations Research*, 49(3):334–351, 2001.

Mohammad Mahdian and Qiqi Yan. Online bipartite matching with random arrivals: an approach based on strongly factor-revealing lps. In Lance Fortnow and Salil P. Vadhan, editors, *STOC*, pages 597–606. ACM, 2011. ISBN 978-1-4503-0691-1.

Mohammad Mahdian, Hamid Nazerzadeh, and Amin Saberi. Allocating online advertisement space with unreliable estimates. In Jeffrey K. MacKie-Mason, David C. Parkes, and Paul Resnick, editors, *ACM Conference on Electronic Commerce*, pages 288–294. ACM, 2007. ISBN 978-1-59593-653-0.

Mohammad Mahdian, Hamid Nazerzadeh, and Amin Saberi. Online optimization with uncertain information. *ACM Transactions on Algorithms*, 8(1):2, 2012.

D. Mattioli. On orbitz, mac users steered to pricier hotels. *The Wall Street Journal, June 26*, June 26 2012.

A. Mehta, A. Saberi, U. V. Vazirani, and V. V. Vazirani. Adwords and generalized online matching. *Journal of the ACM*, 54(5):22, 2007.

Aranyak Mehta. Online matching and ad allocation. *Theoretical Computer Science*, 8(4):265–368, 2012.

V. S. Mirrokni, S. O. Gharan, and M. Zadimoghaddam. Simultaneous approximations for adversarial and stochastic online budgeted allocation. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1690–1701, 2012.

P. Rusmevichientong, Z.-J. Max Shen, and D. B. Shmoys. A PTAS for capacitated sum-of-ratios optimization. *Operations Research Letters*, 37(4):230–238, 2009.

P. Rusmevichientong, Z.-J. M. Shen, and D. B. Shmoys. Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations Research*, 58(6):1666–1680, 2010.

S. A. Smith and N. Agrawal. Management of multi-item retail inventories systems with demand substitution. *Operations Research*, 48:50–64, 2000.

E. Steel and J. Angwin. The web's cutting edge, anonymity in name only. *The Wall Street Journal, August 3*, 2010.

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

K. Talluri and G. J. van Ryzin. *The Theory and Practice of Revenue Management.* Springer, New York, 2004.

Derek Thompson. The 11 ways that consumers are hopeless at math. *The Atlantic*, Jul 6 2012.

Huseyin Topaloglu. Joint stocking and product offer decisions under the multinomial logit model. *Production and Operations Management*, 22(5), 2013. ISSN 1937-5956.

G. J. van Ryzin and S. Mahajan. On the relationship between inventory costs and variety benefits in retail assortments. *Management Science*, 45:1496–1509, 1999.

J. Wortham. Rather than share your location, Foursquare wants to suggest one. *The New York Times, June 7*, 2012.

A. Yao. Probabilistic computations: toward a unified measure of complexity. *Proceedings of the 18th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 222–227, 1977.

A Proofs

Proof of Lemma 1: Fix an arbitrary sequence $\left(\{z_t\}_{t=1}^T\right)$ of customers and an algorithm A. Let the random variables $S_1, \dots, S_T$ denote the sequence of assortments offered by the algorithm A, and let $\xi_1, \dots, \xi_T$ be the random variables corresponding to the customer's choices. Note that S_t may depend on $S_1, \dots, S_{t-1}$ and $\xi_1, \dots, \xi_{t-1}$. The expected revenue of the algorithm A is given by

$$\begin{aligned}\mathbb{E}\left[\sum_{t=1}^{T}\sum_{S\in\mathcal{S}}\sum_{i\in S} r_i \mathbb{1}\left[S_t = S, \xi_t = i\right]\right] &= \sum_{t=1}^{T}\sum_{S\in\mathcal{S}}\sum_{i\in S} r_i \mathbb{E}\left[\phi_i^{z_t}(S_t)\mathbb{1}\left[S_t = S\right]\right] \\ &= \sum_{t=1}^{T}\sum_{S\in\mathcal{S}}\sum_{i=1}^{n} r_i \phi_i^{z_t}(S) \Pr\left\{S_t = S\right\},\end{aligned}$$

where the first equality follows from the tower property of conditional expectation and the fact that $\mathbb{E}\left[\mathbb{1}\left[\xi_t = i\right] \;\middle|\; S_t\right] = \phi_i^{z_t}(S_t)$, and the last equality follows from the fact that $\mathbb{E}\left[\mathbb{1}\left[S_t = S\right]\right] = \Pr\left\{S_t = S\right\}$.

For $t = 1, \dots, T$ and $S \in \mathcal{S}$, let $\bar{y}^t(S) = \Pr\{S_t = S\}$. To complete the proof, it suffices to show that $\bar{y}^t(S)$ satisfies the constraints of the $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$. Clearly, $\bar{y}^t(S) \geq 0$. Furthermore, by definition, $\sum_{S\in\mathcal{S}} \bar{y}^t(S) = 1$. Finally, since the algorithm cannot sell more than the initial inventory of each product, for every product i, with a probability of one,

$$\sum_{t=1}^{T}\sum_{S\in\mathcal{S}} \mathbb{1}\left[S_t = S, \xi_t = i\right] \leq c_i,$$

and by taking expectation on both sides, we have $\sum_{t=1}^{T}\sum_{S\in\mathcal{S}} \phi_i^{z_t}(S)\bar{y}^t(S) \leq c_i$. This shows that $\left(\bar{y}^t(S) : S \in \mathcal{S}\right)$ is a feasible solution of $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$, which is the desired result. □

Proof of Lemma 2: When the customer arrives in period t, if product j has no remaining inventory, then $I_j^{t-1} = 0$, which implies that $r_j \Psi(I_j^{t-1}/c_j) = 0$. By Assumption 1, under our choice model, adding product j to an assortment does not increase the probability that a customer will select other products. Recall that in the case that both sets S and $S \cup \{j\}$ have the maximum discounted revenue, we choose the set with the smaller number of products. Therefore, product j will never be included as a part of the optimal assortment. □

Proof of Corollary 1: First, observe that $\Psi(x) \geq x$. This is because Ψ is increasing and concave, and we have $\Psi(0) = 0$ and $\Psi(1) = 1$. By this observation, we have

$$
\begin{aligned}
\alpha_\Psi(c_{\text{MIN}}) &\geq \min_{x \in \left[0, 1 - \frac{1}{c_{\text{MIN}}}\right]} \left\{ \frac{1 - x}{\frac{1}{c_{\text{MIN}}} + 1 - \Psi(x) + \int_{x + \frac{1}{c_{\text{MIN}}}}^{1} \Psi(y) dy} \right\} \\
&\geq \min_{x \in \left[0, 1 - \frac{1}{c_{\text{MIN}}}\right]} \left\{ \frac{1 - x}{\frac{1}{c_{\text{MIN}}} + 1 - x + \int_{x + \frac{1}{c_{\text{MIN}}}}^{1} dy} \right\} \\
&\geq \min_{x \in \left[0, 1 - \frac{1}{c_{\text{MIN}}}\right]} \left\{ \frac{1 - x}{\frac{1}{c_{\text{MIN}}} + 1 - x + 1 - x - \frac{1}{c_{\text{MIN}}}} \right\} \\
&\geq \frac{1}{2} \, .
\end{aligned}
$$

The second equality follows from the fact that for any $x \in \left[0, 1 - \frac{1}{c_{\text{MIN}}}\right]$, the lower limit of integral, $x + \frac{1}{c_{\text{MIN}}}$, is less than the upper limit of integral, 1, and $\Psi(x) \leq 1$.

In the following we show that the competitive ratio of the IB algorithm with an increasing, strictly concave, and differentiable penalty function is strictly greater than $\frac{1}{2}$. First note that for $x = 0$, the lower bound of the competitive ratio, $\left\{ \frac{1-x}{\frac{1}{c_{\text{MIN}}} + 1 - \Psi(x) + \int_{x + \frac{1}{c_{\text{MIN}}}}^{1} \Psi(y) dy} \right\}$ is greater than $\frac{1}{2}$. This is because for a differentiable penally function Ψ, $\int_{\frac{1}{c_{\text{MIN}}}}^{1} \Psi(y) dy$ is strictly less than $1 - \frac{1}{c_{\text{MIN}}}$ and $\Psi(0) = 0$. Thus, the result holds because

$$
\min_{x \in \left(0, 1 - \frac{1}{c_{\text{MIN}}}\right]} \left\{ \frac{1 - x}{\frac{1}{c_{\text{MIN}}} + 1 - \Psi(x) + \int_{x + \frac{1}{c_{\text{MIN}}}}^{1} \Psi(y) dy} \right\} > \min_{x \in \left(0, 1 - \frac{1}{c_{\text{MIN}}}\right]} \left\{ \frac{1 - x}{\frac{1}{c_{\text{MIN}}} + 1 - x + \int_{x + \frac{1}{c_{\text{MIN}}}}^{1} dy} \right\} = \frac{1}{2} \, .
$$

The inequality holds because for any differentiable and strictly concave penalty function $\Psi(x)$, we have $\Psi(x) > x$ for all $x \in (0, 1)$. □

Proof of Lemma 6. Since the revenue from all the products is the same and no-purchase probability is zero, by Eq. (8), to prove the lemma, it suffices to show the following Claim.

Claim 1: Consider any two products i and j and remaining inventory levels $(x_1, \ldots, x_n)$ such that $x_i > x_j$. If with t periods remaining, the type of the arriving customer is z such that $i \in z$ and $j \in z$, then we have

$$
V(t, x_1, \ldots, x_i, \ldots, x_j, \ldots, x_n \mid z) \leq V(t, x_1, \ldots, x_i - 1, \ldots, x_j + 1, \ldots, x_n \mid z)
$$

This claim implies that it would be better to equalize the inventory levels. Namely, if the inventory for product i is higher than product j, the value (i.e., expected revenue) of the DP policy would increase if instead we have one additional unit of product j and one less unit of product i.

We prove the claim using induction on the inventory levels, fixing product (any) two products i and j:

The induction basis is when $x_i = x_j + 1$ (and no restriction on the inventory of other products). In this case, because of the symmetry in the problem, the value function does not change if we replace one unit of product i with one unit of product j. The reason is that the current customer is interested in both i and j and because of the symmetry in the arrival process, the probability that a future customer is only interested in product i but not product j is the same as the probability that a future customer is only interested in product j but not product i.

Induction Step: Consider initial inventory levels $(y_1, \ldots, y_n)$ such that $y_i > y_j + 1$. Assume Claim 1 holds for any other initial inventory levels $(x_1, \ldots, x_n)$ such $x_k \leq y_k$, $1 \leq k \leq n$, and at least one of these inequalities is strict. To prove the induction step, suppose the optimal dynamic program starting with inventory levels $(y_1, y_2, \ldots, y_n)$ offers set S to the arriving customer. Hence, by conditioning on the type of the customer in the next period, denoted by z', we have:

$$\begin{aligned} & V(t, y_1, \ldots, y_n \mid z) \\ = & \sum_{z' \in \mathcal{Z}} \Pr[\text{next customer is of type } z'|z] \times \Big(1 + \frac{1}{|S|} \sum_{k \in S} V(t-1, y_k - 1, y_{-\{k\}} \mid z')\Big) \end{aligned}$$

where $(y_k - 1, y_{-\{k\}})$ represents the same inventory levels as before only with one less unit of product k. Now by applying the induction hypothesis, we get

$$\begin{aligned} & V(t, y_1, \ldots, y_n \mid z) \\ & \leq \sum_{z' \in \mathcal{Z}} \Pr[\text{next customer is of type } z'|z] \times \Big(1 + \frac{1}{|S|} \sum_{k \in S} V(t-1, y_k - 1, y_i - 1, y_j + 1, y_{-\{i,j,k\}} \mid z')\Big) \\ & \leq V(t-1, y_i - 1, y_j + 1, y_{-\{i,j\}} \mid z) \end{aligned}$$

The last inequality follows from the properties of the optimal dynamic program — note that the optimal policy starting with inventory levels $(y_i - 1, y_j + 1, y_{-\{i,j\}})$ may find a set more profitable than S to offer to the customer. Finally, we point out if $i \in S$ and $k = i$ is chosen by the customer, then $V(t-1, y_k - 1, y_i - 1, y_j + 1, y_{-\{i,j,k\}} \mid z')$ would be defined equivalent to $V(t-1, y_i - 2, y_j + 1, y_{-\{i,j\}} \mid z')$; note that in the induction step we assume $y_i \geq y_j + 2$. $\square$

Proof of Proposition 1: We prove the claim by revisiting the steps of the proof of Theorem 1. Let $\{z_t\}_{t=1}^T$ be the sequence of the customers. By Lemma 2, we never offer any product that has no inventory.

We now construct a solution for $\mathsf{Dual}\left(\{z_t\}_{t=1}^T\right)$, with the true selection probabilities, as follows:

$$\begin{aligned} \theta_i &= r_i(1 - \Psi\left(I_i^T/c_i\right)) \\ \lambda^t &= \textstyle\sum_{i=1}^n r_i \left[\Psi\left(I_i^{t-1}/c_i\right) \bar{\phi}_i^{z_t}(S^t) + 2\epsilon^t\right]. \end{aligned}$$

where $S^t = \text{argmax}_{S \in \mathcal{S}} \sum_{i \in S} r_i \Psi\left(I_i^{t-1}/c_i\right) \phi_i^{z_t}(S)$ is the assortment offered by the IB algorithm. Note that we add the error term ϵ^t to the value of λ^t because the assortment S^t is computed using the *estimated* selection probability ϕ^{z_t}. This construction gives us a feasible dual solution because

$$\lambda^t \;\geq\; \sum_{i=1}^n r_i \Psi\left(I_i^T/c_i\right)\left(\phi_i^{z_t}(S^t) - \epsilon^t\right) + 2\sum_{i=1}^n r_i \epsilon^t \geq \sum_{i=1}^n r_i \Psi\left(I_i^T/c_i\right)\phi_i^{z_t}(S^t) + \sum_{i=1}^n r_i \epsilon^t$$

where the first inequality follows from the fact that for all $i = 1, 2, \ldots, n$ and $S \in \mathcal{S}$, $|\bar{\phi}_i^{z_t}(S) - \phi_i^{z_t}(S)| \leq \epsilon^t$. The second inequality holds because Ψ is increasing and $I_i^{t-1} \geq I_i^T$. By definition of S^t,

$$\begin{aligned}
\lambda^t &\geq \max_S \left\{ \sum_{i \in S} r_i \Psi\left(I_i^T / c_i\right) \phi_i^{z_t}(S) \right\} + \sum_{i=1}^n r_i \epsilon^t \\
&\geq \max_S \left\{ \sum_{i \in S} r_i \Psi\left(I_i^T / c_i\right) \left(\bar{\phi}_i^{z_t}(S) - \epsilon^t\right) \right\} + \sum_{i=1}^n r_i \epsilon^t \\
&\geq \max_S \sum_{i \in S} (r_i - \theta_i) \bar{\phi}_i^{z_t}(S) \ ,
\end{aligned}$$

where the second inequality follows from the fact that for all i and $S \in \mathcal{S}$, $|\bar{\phi}_i^{z_t}(S) - \phi_i^{z_t}(S)| \leq \epsilon^t$. The third inequality follows from the definition of θ_i and the fact that $\Psi\left(I_i^T / c_i\right)$ is less than or equal to 1.

It follows from the Weak Duality Theorem that

$$\begin{aligned}
\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right) &\leq \mathbb{E}\left[\sum_{t=1}^T \lambda^t + \sum_{i=1}^n c_i \theta_i\right] \\
&\leq \mathbb{E}\left[\sum_{i=1}^n r_i \left(c_i \left(1 - \Psi\left(I_i^T / c_i\right)\right) + \sum_{t=I_i^T+1}^{c_i} \Psi(t/c_i) + 2 \sum_{t=1}^T \epsilon^t \right)\right]
\end{aligned}$$

Hence,

$$\frac{\mathbb{E}\left[\sum_{i=1}^n r_i (c_i - I_i^T)\right]}{\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)} \geq \frac{\mathbb{E}\left[\sum_{i=1}^n r_i (c_i - I_i^T)\right]}{\mathbb{E}\left[\sum_{i=1}^n r_i \left(c_i \left(1 - \Psi\left(I_i^T / c_i\right)\right) + \sum_{t=I_i^T+1}^{c_i} \Psi(t/c_i) + 2 \sum_{t=1}^T \epsilon^t\right)\right]} \ .$$

where $\mathbb{E}\left[\sum_{i=1}^n r_i (c_i - I_i^T)\right]$ is the revenue of the Inventory-Balancing algorithm. Note that the contribution of any product i that is not sold by the optimal solution to $\mathsf{Primal}\left(\{z_t\}_{t=1}^T\right)$ is zero. Thus, to find the competitive ratio, we only consider products that are sold by the optimal solution. Since it is assumed that the IB algorithm sells at least one unit of any product that is sold by the optimal solution, the competitive ratio of the algorithm is at least

$$\min_{(c_i, x): x \,\leq\, 1 - \frac{1}{c_i}} \frac{1 - x}{\frac{1}{c_i} \sum_{t=I_i^T+1}^{c_i} \Psi(t/c_i) + (1 - \Psi(x)) + \frac{2}{c_i} \mathbb{E}\left[\sum_{t=1}^T \epsilon^t\right]},$$

where i is a product sold by the optimal solution. Therefore, by the same argument as in the proof of Theorem 1, the competitive ratio is at least

$$\min_{x \in \left[0, 1 - \frac{1}{c_{\text{MIN}}}\right]} \left\{ \frac{1 - x}{\frac{1}{c_{\text{MIN}}} + 1 - \Psi(x) + \int_{x + \frac{1}{c_{\text{MIN}}}}^1 \Psi(y) dy + \frac{2}{c_{\text{MIN}}} \mathbb{E}\left[\sum_{t=1}^T \epsilon^t\right]} \right\}.$$

$\square$

Proof of Lemma 5: The derivative of $x \mapsto \frac{1-x}{2-x-\Psi(x)}$ is equal to $\frac{-1+\Psi(x)+(1-x)\Psi'(x)}{(2-x-\Psi(x))^2}$. The numerator $-1+\Psi(x)+(1-x)\Psi'(x)$ is equal to 0 at $x=1$, and its derivative is equal to $(1-x)\Psi''(x) \leq 0$ because Ψ is concave. Thus, $-1+\Psi(x)+(1-x)\Psi'(x) \geq 0$ for all $x \in [0,1]$. To complete the proof, note that the derivative of $C \mapsto \frac{1}{C} + \int_{a+\frac{1}{C}}^{1} \Psi(y)dy$ is equal to $-(1/C)^2(1-\Psi(a+\frac{1}{C})) \leq 0$. $\square$

A.1 Proof of Proposition 3

In the following, we prove the claim for the EIB algorithm; a similar argument can be applied to the LIB algorithm. First, note that we divide the horizon T into $\frac{1}{\epsilon}$ time slots such that $T\epsilon$ is an integer. We only observe the remaining inventory of each product i in periods $kT\epsilon$, $0 < k \leq \frac{1}{\epsilon}$. Consider the solution (allocation) of the EIB algorithm for a sequence of customers $\{z_t\}_{t=1}^{T}$. For this solution, let $\rho^{j,k}$ be the sum of *revenue times capacity* of any product i whose fraction of the remaining inventory $\frac{I_i^{kT\epsilon}}{c_i}$ in the k^{th} time slot (period $kT\epsilon$) is between $1-(j+1)\epsilon$ and $1-j\epsilon$ (inclusive); that is, $\frac{I_i^{kT\epsilon}}{c_i} \in (1-(j+1)\epsilon, 1-j\epsilon]$ where $0 \leq j \leq \frac{1}{\epsilon}-1$ and $0 < k \leq \frac{1}{\epsilon}$. Similarly, let $\gamma^{j,k}$ be the total revenue obtained from product i in the optimal solution (of $\mathsf{Primal\text{-}S}$) where $\frac{I_i^{kT\epsilon}}{c_i} \in (1-(j+1)\epsilon, 1-j\epsilon]$, i.e.,

$$\begin{aligned}\rho^{j,k} &= \sum_{i:\frac{I_i^{kT\epsilon}}{c_i} \in (1-(j+1)\epsilon, 1-j\epsilon]} r_i c_i, \\ \gamma^{j,k} &= \sum_{i:\frac{I_i^{kT\epsilon}}{c_i} \in (1-(j+1)\epsilon, 1-j\epsilon]} o_i \ ,\end{aligned}$$

where o_i is the revenue obtained from selling product i in the optimal solution. Note that $o_i \leq r_i c_i$. For any time slot k, we define

$$\chi(k) = \sum_{i=1}^{n} c_i r_i \int_{y=\frac{I_i^{kT\epsilon}}{c_i}}^{1} \Psi(y)dy = \sum_{i=1}^{n} r_i \int_{y=I_i^{kT\epsilon}}^{c_i} \Psi(y/c_i)dy.$$

Notice that $\chi(k)$ is an increasing function of k. Using the fact that in any period t the IB algorithm chooses an assortment S^t that maximizes $\sum_{i\in S} r_i \Psi\left(I_i^{t-1}/c_i\right) \phi_i^{z_t}(S)$, we will bound the change in χ function at two consecutive time slots, see the fifth sets of constraints in linear program FRLP.

Next, we will show that the objective function of FRLP is less than the total revenue of the EIB algorithm. Since the penalty function is exponential, $\Psi(x) = \frac{e}{e-1}(1-e^{-x})$, $x \in [0,1]$, the first term in the objective function is given by

$$\chi(\frac{1}{\epsilon}) = \sum_{i=1}^{n} c_i r_i \int_{y=\frac{I_i^T}{c_i}}^{1} \Psi(y)dy = \frac{e}{e-1} \sum_{i=1}^{n} c_i r_i \left(1 - \frac{I_i^T}{c_i} + e^{-1} - e^{-\frac{I_i^T}{c_i}}\right).$$

By definition of $\rho^{j,\frac{1}{\epsilon}}$, the second term of the objective function is lower bounded as follows

$$\frac{e}{e-1} \sum_{j=0}^{1-\frac{1}{\epsilon}} \rho^{j,\frac{1}{\epsilon}} \left(\frac{j\epsilon}{e} - e^{j\epsilon-1} + e^{-1}\right) \geq \frac{e}{e-1} \sum_{i=1}^{n} c_i r_i \left(\frac{1-\frac{I_i^T}{c_i}}{e} - e^{-\frac{I_i^T}{c_i}} + e^{-1}\right).$$

The inequality holds because $\frac{x}{e} - e^{x-1}$ is a decreasing function of x. Therefore,

$$\chi(\frac{1}{\epsilon}) - \frac{e}{e-1} \sum_{j=0}^{\frac{1}{\epsilon}-1} \rho^{j,\frac{1}{\epsilon}} \left(\frac{j\epsilon}{e} - e^{j\epsilon - 1} + e^{-1} \right) \leq \sum_{i=1}^{n} r_i (c_i - I_i^T),$$

where the left hand side is the objective function of linear program FRLP and the right hand side is the revenue of the EIB algorithm.

The next step is to show that any solution of the EIB algorithm corresponds to a feasible solution of linear program FRLP. Without loss of generality, we can normalize the revenue of the optimal solution, Primal-S, to 1 which implies the first set of constraints. The second set of constraints holds because $o_i \leq r_i c_i$. The third set of constraints follows from the definition of $\rho^{j,k}$ and the fact that $I_i^{kT\epsilon}$ is a decreasing function of k. The forth set of constraints holds because of the definition of χ and the fact that $\int_{y=x}^{1} \Psi(y)dy$ is a decreasing function of x. Lemma 7, stated below, leads us to the last set of constraints. This set of constraints gives a lower bound for difference of $\chi(k+1)$ and $\chi(k)$ as a function of the optimal solution $\gamma^{j,k}$. To prove the lower bound, we show that under uniform permutation, the revenue obtained from product i in the optimal solution during periods in $[kT\epsilon, (k+1)T\epsilon)$ denoted by $o_{i,k}$ is concentrated around its average ϵo_i. Using this concentration and the fact that the IB algorithm chooses a set that maximizes discounted revenue, we will get the desired bound for the change in function χ. Hence, any solution of the EIB algorithm, corresponds to a feasible solution of the linear program FRLP where the objective is less than the revenue of the algorithm. Considering the fact that the optimal solution, Primal-S, is normalized to one, the solution of the linear program FRLP is a number in $[0, 1]$ and by minimizing the objective function, we obtain a lower bound for the performance of the EIB algorithm, namely the ratio of $\mathbb{E}_{\{z_t\}_{t=1}^T} \left[Rev_{EIB} \left(\{z_t\}_{t=1}^T \right) \right]$ to Primal-S.

Lemma 7. *Suppose Ψ is increasing, concave, and twice differentiable with a bounded derivative, in $[0, 1]$ and $\frac{1}{\epsilon \times c_{\text{MIN}}} = O(1)$, then with high probability, for any $0 < k < \frac{1}{\epsilon}$,*

$$\epsilon \sum_{j=0}^{\frac{1}{\epsilon}-1} \gamma^{j,k+1} \Psi\big(1 - (j+1)\epsilon\big) \leq \chi(k+1) - \chi(k) + O\left(\frac{1}{c_{\text{MIN}}} \right).$$

Proof of Lemma 7: Note that the contribution of product i to $\chi(k+1) - \chi(k)$ is equal to $r_i c_i \int_{I_i^{t_1}/c_i}^{I_i^{t_0}/c_i} \Psi(y)dy$ where $t_0 = kT\epsilon$ and $t_1 = (k+1)T\epsilon$. By the assumption that the derivative of Ψ is bounded, we can substitute the integral with the sum and get

$$r_i c_i \int_{I_i^{t_1}/c_i}^{I_i^{t_0}/c_i} \Psi(y)dy = r_i \int_{I_i^{t_1}}^{I_i^{t_0}} \Psi(y/c_i)dy \geq r_i \sum_{z=I_i^{t_1}}^{I_i^{t_0}} \Psi(z/c_i) - O\left(\frac{1}{c_{\text{MIN}}} \right). \tag{3}$$

Now let us consider the optimal solution. Let S_{opt}^t denote the set that is shown to the customer in period t by the optimal solution. Since the Inventory-Balancing algorithm shows to each customer an assortment that maximizes $\sum_{i \in S} r_i \Psi\big(\frac{I_i^t}{c_i}\big) \phi_i^t(S)$, we have

$$\sum_{i=1}^{n} r_i \sum_{z=I_i^{t_1}}^{I_i^{t_0}} \Psi(z/c_i) \geq \sum_{t=t_0}^{t_1-1} \sum_{i \in S_{opt}^t} r_i \Psi\big(I_i^t/c_i\big) \phi_i^t(S_{opt}^t) \geq \sum_{t=t_0}^{t_1-1} \sum_{i \in S_{opt}^t} r_i \Psi(I_i^{t_1}/c_i) \phi_i^t(S_{opt}^t). \tag{4}$$

The last inequality holds because Ψ is an increasing function. Recall that $o_{i,k}$ is the revenue obtained by the optimal solution from product i during periods in $[kT\epsilon, (k+1)T\epsilon)$. By this definition and the inequality above, we have

$$\sum_{i=1}^{n} r_i \sum_{z=I_i^{t_1}}^{I_i^{t_0}} \Psi(z/c_i) \geq \sum_{i=1}^{n} \Psi(I_i^{t_1}/c_i) o_{i,k} \ . \tag{5}$$

In Lemma 8 which is borrowed from Mirrokni et al. (2012) (with some modifications), we show that under uniform permutation, with high probability, $o_{i,k}$ is concentrated around its average ϵo_i where $\frac{1}{\epsilon}$ is the number of time slots. Note that this concentration holds under the uniform permutation and as we will discuss below it is useful when $\frac{1}{\epsilon \times c_{\text{MIN}}} = O(1)$. By this lemma and the above equation,

$$\sum_{i=1}^{n} r_i \sum_{z=I_i^{t_1}}^{I_i^{t_0}} \Psi(z/c_i) \geq \sum_{i=1}^{n} \Psi(I_i^{t_1}/c_i) \epsilon o_i \ , \tag{6}$$

with high probability. Therefore, by Equation (6) and the definition of γ, we have

$$\sum_{i=1}^{n} r_i \sum_{z=I_i^{t_1}}^{I_i^{t_0}} \Psi(z/c_i) - O\left(\frac{1}{c_{\text{MIN}}}\right) \geq \epsilon \sum_{j=0}^{\frac{1}{\epsilon}-2} \Psi\big(1-(j+1)\epsilon\big) \gamma^{j,k+1} - O\left(\frac{1}{c_{\text{MIN}}}\right). \tag{7}$$

The proof is completed since the left hand side is less than $\chi(k+1) - \chi(k)$. $\square$

Lemma 8. *If the customers arrive according to a random order (i.e., a permutation chosen uniformly at random), $\sum_{i=1}^{n} o_i = 1$, then for any $\delta > 0$ and $\frac{1}{T\epsilon} \leq k \leq \frac{1}{\epsilon}$,*

$$\Pr\left[\sum_{i=1}^{n} |o_{i,k} - \epsilon o_i| > \frac{5}{\epsilon \times c_{\text{MIN}} \times \delta}\right] < 1 - \delta.$$

The assumption that $\sum_{i=1}^{n} o_i = 1$ implies that we have normalized Primal-S to 1. In this lemma, we need $\frac{5}{\epsilon \times c_{\text{MIN}}}$ to be either constant or go to 0 which justifies the assumption $\frac{1}{\epsilon \times c_{\text{MIN}}} = O(1)$ in Lemma 7.

B Numerical Experiments: Appendix to Section 7

B.1 Worst-Case Performance

In Section 7.2, we have compared different polices in term of their average performance. Here, we investigate the worst-case performance of different policies. To this aim, we consider 250 random arrival sequences. For each of them we compute the ratio of revenue collected by each policy and the corresponding optimal clairvoyant solution. Then, the worst-case performance of any policy is defined as the minimum of these ratios. Table 6 presents the worst-case performance of all policies for LF$= 1.4, 1.6$, and 1.8 and CV$= 0.1$, 1, and 2 when the length of the horizon is drawn from the uniform distribution with $\bar{T} - \underline{T} = \mathbb{E}[T]$. Our IB polices outperform other policies in term of the worst-case performance, that is they can obtain at least 91% of the optimal clairvoyant solution, which is much higher than the theoretical bounds, i.e., 63% for the EIB policy and 50% for the LIB policy. We observe that the LP resolving heuristics perform poorly compare to IB and Hybrid algorithms. Furthermore, One-shot LP heuristics are very sensitive to uncertainty in arrival sequence (large CV). For instance, when LF=1.8 and CV=2 there is an arrival sequence in which they only get 3.8% of the optimal clairvoyant solution.

Problem Class		Worst Case Revenue under Different Policies (as % of the Upper Bound)								
		Inventory-Balancing		Myopic	One-shot LP		LP Resolving		hybrid	
LF	CV	EIB	LIB	Policy	LPO	ALPO	LPR^{500}	LPR^{50}	$\mathcal{H}_{1.5}$	$\mathcal{H}_2$
1.4	2.0	91.8	91.4	91.0	13.2	14.4	75.8	76.7	88.4	83.2
	1.0	92.2	92.0	91.9	16.8	17.8	76.7	77.4	88.8	84.7
	0.1	92.2	91.8	91.4	72.6	73.3	78.9	79.1	89.5	85.4
1.6	2.0	92.5	92.0	92.0	8.6	8.8	70.9	73.3	90.0	86.9
	1.0	93.2	91.7	91.2	41.8	41.9	72.7	73.1	89.8	87.3
	0.1	92.7	92.8	91.3	66.5	67.4	73.4	74.5	91.1	87.3
1.8	2.0	92.4	92.3	91.2	3.8	3.8	66.2	67.4	90.5	87.1
	1.0	92.8	92.5	92.0	20.9	20.3	68.4	69.0	91.8	88.9
	0.1	93.1	93.2	91.6	60.3	60.5	67.8	68.2	92.6	90.3

Table 6: Worst-Case Performance Comparison when the length of horizon is unknown.

B.2 Known Length of the Horizon

In this Section we compare the performance of the EIB, LIB and Hybrid algorithms to the myopic policy and the LP-based heuristics when the length of the horizon is known in advance. We set the initial inventory levels to 100, i.e., $c_i = 100$, $i = 1, 2, \ldots, 73$.

Performance Evaluation: In Table 7, we present the average revenue of each algorithm as a percentage of the upper bound, which is averaged over all 250 problem instances, for loading factors 1.4, 1.6 and 1.8 and for coefficients of variation of 0.1, 1, and 2. As the table shows, when the number of customers is known in advance, LPR^{500} algorithm can obtain more that 99% of the optimal solution for all the considered problem classes which implies that having more resolving periods is not necessary.

We note that both the LIB and EIB algorithms outperform the myopic and LPO algorithms. Moreover, the revenue of the EIB and LIB algorithms is within $\pm 2\%$ of that of the resolving heuristics. Comparing the performance of LPR^{500} in Tables 7 and 3 implies that that LPR heuristics are sensitive to uncertainty in number of customers. Precisely, the performance of LPR heuristic decreases significantly when it does not know the exact length of the horizon. In all problem classes, the Hybrid algorithms yield more revenue than the IB polices since they incorporates additional information about arrival sequence by using the LP resolving heuristic.

Again, in all cases, the LPO algorithm has the lowest revenue and its performance decreases by increasing CV and loading factor. Observe that when CV= 0.1, One-shot LP heuristics obtain more than 95% of the optimal clairvoyant solution. For small value of CV, the number of customers

Problem Class		Upper Bound	Avg. Revenue under Different Policies (as % of the Upper Bound)							
			Inventory-Balancing		Myopic	One-shot LP		LP Resolving	Hybrid	
LF	CV	(in \$1000)	EIB	LIB	Policy	LPO	ALPO	LPR^{500}	$\mathcal{H}_{1.5}$	$\mathcal{H}_2$
1.4	2.0	173	97.3	97.2	96.2	69.3	77.9	99.3	98.6	98.9
	1.0	179	97.6	97.8	95.4	83.5	89.2	99.3	98.5	98.8
	0.1	182	98.1	98.5	95.8	95.8	98.0	99.5	98.9	99.1
1.6	2.0	175	98.2	98.3	97.2	68.4	75.4	99.4	98.9	99.0
	1.0	181	98.7	98.9	97.4	83.6	88.3	99.6	99.0	99.0
	0.1	183	99.3	99.3	97.8	95.8	97.8	99.7	99.4	99.5
1.8	2.0	177	98.8	98.9	98.0	65.5	72.1	99.4	99.1	99.1
	1.0	182	99.2	99.3	98.4	79.8	84.6	99.5	99.3	99.4
	0.1	183	99.7	99.8	99.4	95.5	97.6	99.8	99.8	99.8

Table 7: Revenue comparison when the length of horizon is known in advance. The standard errors of all numbers are less than 0.1%.

of each type is very concentrated around its average. Therefore, these heuristics do not suffer from fixing their strategies at the beginning of the horizon. Note that even for small value of CV, our IB algorithms perform better than One-shot LP heuristics.

Transient Behavior: Figures 2 shows the cumulative revenue over time for the myopic, LIB, LPR^{500}, LPO and ALPO algorithms with $\text{LF} = 1.8$ and $\text{CV} = 2$. We observe that the myopic policy and the LIB algorithm are very aggressive during the initial periods, resulting in higher cumulative revenues than One-shot LP and resolving heuristics. Since resolving heuristics know exactly the number of customers in advance, they manage to earn revenue linearly over time. This implies that knowing the true estimate of the length of the horizon (number of customers) is essential for the resolving heuristics, that is, *if the number of customers is less than its estimated value, these heuristics will suffer from significant revenue loss*, see Section 7.2.

B.3 Learning the Customer Types

Here we investigate the performance of the IB algorithms when we do not know the exact value of the selection probability $\bar{\phi}_i^{z_t}(S)$. Rather, we only have an estimate $\phi_i^z(S)$ based on data collected in the previous periods. Since we assume the multinomial logit choice model for each customer type, we maintain an estimate $\mathbf{V^z}(t) = (V_0^z(t), V_1^z(t), \ldots, V_n^z(t))$ of the preference weight parameters, where for each product i, we set $V_i^z(t)$ to be proportional to the number of times that a customer of type z purchases DVD during the previous $t-1$ periods, and we normalize $\mathbf{V^z}(t)$ so that $V_0^z(t) = 1$. Similar to the previous section, we have 10 customer types and 73 products with initial inventory of $c_i = 30$.

Table 8 shows the revenue of the IB algorithms when these algorithms only have estimates of the preference weight parameters. In absolute terms, the IB algorithms perform well despite not knowing the true parameter values; they obtain 83% − 98% of the upper bound, depending of the coefficient of variations and the loading factor. We observe better performance for loading factor of 1.6 in compare to smaller loading factors. The reason is that larger loading factor or longer the

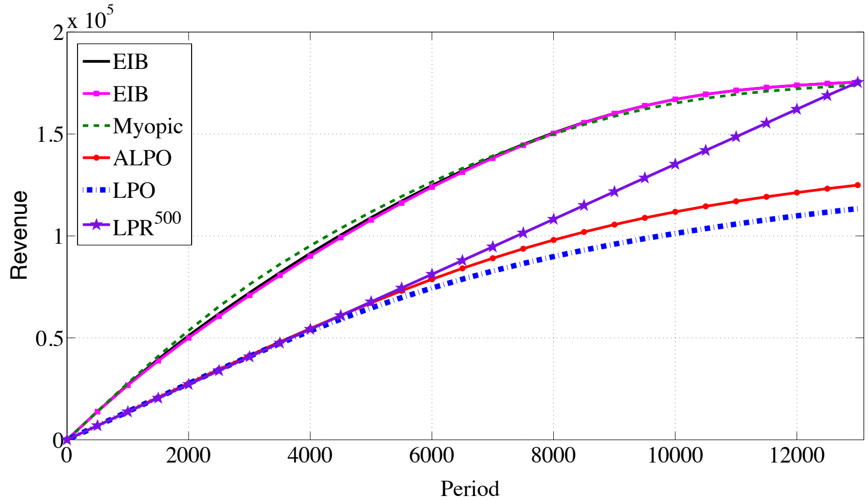

Figure 2: The cumulative revenue over time for $LF = 1.8$ and $CV = 2$ when the length of the horizon is known in advance.

horizon allows the algorithms to obtain better estimates of the unknown parameters. Note that the IB algorithms perform well even with few observations. One of the reasons is that in the setting above, we do not impose any constraint on the size of the assortment that policies can offer to each customer. This could compensate for the inaccuracy in estimation of choice model since the algorithm can offer large assortments. Furthermore, by Proposition 1, we expect the IB algorithms to be robust with respect to the preference weight parameters.

C Asymptotic Optimality of the Dynamic Programming Policy

In this section, we show asymptotic optimality of the dynamic programming (DP) policy when the type of customers is drawn independently from a known distribution. Namely, we show that the value obtained by the DP policy approaches Primal-S asymptotically when both the capacities and the horizon scale proportionally.

Let $\eta^z > 0$, $z \in \mathcal{Z}$, be the probability that in each period a customer of type z arrives.[31] Let $V(t, x_1, \ldots, x_n \mid z)$ denote the maximum expected revenue with t periods *remaining*, given that a customer of type $z \in \mathcal{Z}$ arrives, and the remaining inventories are $(x_1, \ldots, x_n)$. Then, the dynamic programming formulation of this problem is given by

$$\begin{aligned} & V(t, x_1, \ldots, x_n \mid z) & (8) \\ = & \max_{S \in \mathcal{S} \,:\, x_i \geq 1 \,\forall\, i \in S} \left\{ \sum_{i \in S} \phi_i^z(S) \left[r_i + V(t-1, x_1, \ldots, x_i - 1, \ldots, x_n) \right] + \phi_0^z(S) V(t-1, x_1, \ldots, x_n) \right\} \end{aligned}$$

where $V(t, x_1, \ldots, x_n) = \sum_{z \in \mathcal{Z}} \eta^z V(t, x_1, \ldots, x_n \mid z)$. Also, the terminal condition is given by $V(0, \cdot) = 0$. We denote the optimal revenue under the dynamic programming formulation by

[31]This is with abuse of notation and done for the sake of economy of notation, we previously used η^z as the expected number of customers of type z, not the probability.

Problem Class		Upper	Revenue	
Loading	Coefficient of	Bound	(as % of the Upper Bound)	
Factor (LF)	Variation (CV)	(in \$100)	LIB	EIB
1.2	0.2	526	91.3	91.4
	0.5	524	89.6	89.6
	0.8	522	86.2	86.4
1.4	0.2	546	95.4	95.7
	0.5	544	92.9	93.2
	0.8	541	89.9	90.6
1.6	0.2	549	98.6	98.5
	0.5	548	96.7	96.9
	0.8	546	93.8	93.9

Table 8: The average revenue for the LIB and EIB algorithms when the underlying parameters are unknown, and each algorithm uses the estimated parameters based on data collected in the previous periods.

$V(T, \boldsymbol{c})$ where $\boldsymbol{c}$ is the vector of initial inventories. We note that in computing $V(T, \boldsymbol{c})$, we take expectation with respect to sequence of customers and the customers' choices. For simplicity, we assume that the policy can always offer an "empty" assortment with $S = \varnothing$. Thus, the maximum in the dynamic programming equation is always well-defined.

The asymptotic optimality result is stated in the following Proposition.

Proposition 4 (Asymptotic Optimality of DP)**.** *Given that the type of customers is drawn independently from a known distribution such that the probability of arriving a customer of type $z \in \mathcal{Z}$ in any period t is η^z, then*

$$\lim_{\beta \to \infty} \frac{V(\beta T, \beta \boldsymbol{c})}{\textsf{Primal-S}(\beta T, \beta \boldsymbol{c})} \;=\; 1,$$

where $\textsf{Primal-S}(\beta T, \beta \boldsymbol{c})$ *is the linear programming* $\textsf{Primal-S}$ *with initial inventories $\beta \boldsymbol{c}$ and the length of the horizon βT.*

In the above proposition, we scale both the horizon and initial inventory with a scalar β. The corresponding problem is called *β-scaled stochastic problem.* Then, to see the asymptotic behavior of dynamic programming, we let β go to infinity. We note that Proposition 4 does not imply that the dynamic programming policy is asymptotically optimal for every sequence of customer types. Instead it shows that it is asymptotically optimal only when take average over all sequences.

Proof of Proposition 4: By Lemma 4, $V(\beta T, \beta \boldsymbol{c}) \leq \textsf{Primal-S}(\beta T, \beta \boldsymbol{c})$ for all T, $\boldsymbol{c}$ and $\beta > 0$. Now, let $\{\bar{y}^z(S) : S \in \mathcal{S}\ ,\ z \in \mathcal{Z}\}$ denote an optimal solution for the (unscaled) Primal-S$(T, \boldsymbol{c})$ for all T, $\boldsymbol{c}$ and $\beta > 0$. Then, it is easy to verify that $\{\beta \bar{y}^z(S) : S \in \mathcal{S}\ ,\ z \in \mathcal{Z}\}$ is an optimal solution to $\textsf{Primal-S}(\beta T, \beta \boldsymbol{c})$.

To show that $\lim_{\beta \to \infty} \frac{V(\beta T, \beta \boldsymbol{c})}{\textsf{Primal-S}(\beta T, \beta \boldsymbol{c})} = 1$, we construct a deterministic policy μ for the β-scaled stochastic problem whose expected revenue approaches $\textsf{Primal-S}(\beta T, \beta \boldsymbol{c})$ as β increases toward infinity. We show that this policy is admissible, that is the total sales of product i is less than its initial inventory. Therefore, $V(\beta T, \beta \boldsymbol{c})$ also approaches $\text{Primal-S}(\beta T, \beta \boldsymbol{c})$ as $\beta \to \infty$.

The policy μ operates as follows: Offer a set $S \in \mathcal{S}$ to customers of type z for up to $\beta \eta^z \bar{y}^z(S)$ times. The order in which the sets are offered is arbitrary. Under this policy, we will *NOT* accept all of the demands generated by offering S. *Rather, we will limit the sales of product i from offering S to customers of type z to at most $\beta \eta^z \phi_i^z(S) \bar{y}^z(S)$.*

Let $\boldsymbol{N}(\beta T) = (N^z(\beta T) : z \in \mathcal{Z})$ be a multinomial random vector, where $N^z(\beta T)$ denotes the total number of customers of type z over βT periods. Note that $N^z(\beta T)$ has a binomial distribution with parameter βT and η^z. We define the random variable $D_i^z(S, q)$ as the total number of customers of type z who select product i when S is offered under the policy μ, *given that there are q customers of type z*. Since under the policy μ we do not accept all the demands, the total sales of product i from customers of type z generated from offering S under the policy μ is given by

$$\mathsf{Sale}_i^{\mu,z}(S) = \min \left\{ D_i^z(S, N^z(\beta T)) \ , \ \beta \eta^z \phi_i^z(S) \bar{y}^z(S) \right\} \ .$$

We point out that $\mathsf{Sale}_i^{\mu,z}(S)$ is a random variable because $D_i^z(S, N^z(\beta T))$ is a random variable. Since $\beta \bar{y}^z(S)$ is a feasible solution of linear program Primal-S$(\beta T, \beta \boldsymbol{c})$, we have

$$\beta \sum_{z \in \mathcal{Z}} \sum_{S \in \mathcal{S}} \eta^z \phi_i^z(S) \bar{y}^z(S) \leq \beta c_i \quad , \quad i = 1, \ldots, n \ ,$$

which implies that, with probability one,

$$\sum_{z \in \mathcal{Z}} \sum_{S \in \mathcal{S}} \mathsf{Sale}_i^{\mu,z}(S) \leq \beta \sum_{z \in \mathcal{Z}} \sum_{S \in \mathcal{S}} \eta^z \phi_i^z(S) \bar{y}^z(S) \ \leq \ \beta c_i \quad , \quad i = 1, \ldots, n \ .$$

Therefore, the policy μ is admissible because the total sales of product i does not exceed its initial inventory. The total revenue over βT periods under the policy μ is given by a random variable $\sum_{i=1}^n r_i \sum_{S \in \mathcal{S}} \sum_{z \in \mathcal{Z}} \mathsf{Sale}_i^{\mu,z}(S)$. Then,

$$\begin{aligned}
\lim_{\beta \to \infty} \frac{1}{\beta} \sum_{i=1}^n r_i \sum_{S \in \mathcal{S}} \sum_{z \in \mathcal{Z}} \mathsf{Sale}_i^{\mu,z}(S) &= \lim_{\beta \to \infty} \frac{1}{\beta} \sum_{i=1}^n r_i \sum_{S \in \mathcal{S}} \sum_{z \in \mathcal{Z}} \min \left\{ D_i^z(S, N^z(\beta T)) \ , \ \beta \eta^z \bar{y}^z(S) \phi_i^z(S) \right\} \\
&= \sum_{i=1}^n r_i \sum_{S \in \mathcal{S}} \sum_{z \in \mathcal{Z}} \min \left\{ \lim_{\beta \to \infty} \frac{1}{\beta} D_i^z(S, N^z(\beta T)) \ , \ \eta^z \bar{y}^z(S) \phi_i^z(S) \right\} \\
&= \sum_{i=1}^n r_i \sum_{S \in \mathcal{S}} \sum_{z \in \mathcal{Z}} \eta^z \bar{y}^z(S) \phi_i^z(S) = \mathsf{Primal\text{-}S}(T, \boldsymbol{c}) \ .
\end{aligned}$$

To establish the third equality above, note that

$$\frac{1}{\beta} D_i^z(S, N^z(\beta T)) \ = \ \frac{1}{\beta} \sum_{t=1}^{M^z} \mathbb{1}_{\{B_{t,z}^i(S)=1\}} = \frac{M^z}{\beta} \times \frac{1}{M^z} \sum_{t=1}^{M^z} \mathbb{1}_{\{B_{t,z}^i(S)=1\}} \ ,$$

where $M^z := \min\{N^z(\beta T) \ , \ \beta \eta^z \bar{y}^z(S)\}$ and $B_{t,z}^i(S) = 1$ denotes the event that the t^{th} customer of type z selects product i when S is offered, with $\mathbb{E}[\mathbb{1}_{\{B_{t,z}^i(S)=1\}}] = \phi_i^z(S)$. By SLLN, we know that $\lim_{\beta \to \infty} N^z(\beta T)/\beta = \eta^z T$ almost surely (a.s.). Since under the policy μ, we only offer S up to $\beta \eta^z \bar{y}^z(S)$ customers of type z,

$$\lim_{\beta \to \infty} \frac{M^z}{\beta} = \lim_{\beta \to \infty} \frac{\min\{N^z(\beta T) \ , \ \beta \eta^z \bar{y}^z(S)\}}{\beta} = \eta^z \bar{y}^z(S) \quad a.s.$$

By a similar argument, $\frac{1}{M^z}\sum_{t=1}^{M^z} \mathbb{1}_{\{B^i_{t,z}(S)=1\}} = \phi^z_i(S)$. Thus, with probability one,

$$\lim_{\beta\to\infty} \frac{1}{\beta} D^z_i(S, N^z(\beta T)) \quad = \quad \eta^z \bar{y}^z(S)\phi^z_i(S),$$

which gives us the desired result. Then, by the Dominated Convergence Theorem, it follows that $\lim_{\beta\to\infty} \frac{1}{\beta}\mathbb{E}\left[\sum_{i=1}^n r_i \sum_{S\in\mathcal{S}} \sum_{z\in\mathcal{Z}} \mathsf{Sale}^{\mu,z}_i(S)\right] = \mathsf{Primal\text{-}S}(T, \boldsymbol{c})$. Since the policy μ is admissible,

$$1 \geq \lim_{\beta\to\infty} \frac{V(\beta T, \beta \boldsymbol{c})}{\mathsf{Primal\text{-}S}(\beta T, \beta \boldsymbol{c})} \geq \lim_{\beta\to\infty} \frac{\frac{1}{\beta}\mathbb{E}\left[\sum_{i=1}^n r_i \sum_{S\in\mathcal{S}} \sum_{z\in\mathcal{Z}} \mathsf{Sale}^{\mu,z}_i(S)\right]}{\frac{1}{\beta}\mathsf{Primal\text{-}S}(\beta T, \beta \boldsymbol{c})} = 1,$$

which completes the proof. $\square$