Online Resource Allocation under Arbitrary Arrivals: Optimal Algorithms and Tight Competitive Ratios

Will Ma
Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA 02139, willma@mit.edu

David Simchi-Levi
Institute for Data, Systems, and Society, Department of Civil and Environmental Engineering, and Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA 02139, dslevi@mit.edu

We consider the problem of allocating fixed resources to heterogeneous customers arriving sequentially. We study this problem under the framework of competitive analysis, which does not assume any predictability in the sequence of customer arrivals. Previous work has culminated in optimal algorithms under two scenarios: (i) there are multiple resources, each of which yields reward at a constant rate when allocated; or (ii) there is a single resource, which yields reward at different rates when allocated to different customers.

In this paper, we derive optimal allocation algorithms when there are multiple resources, each with multiple reward rates. Our algorithms are simple, intuitive, and robust against forecast error. Their tight competitive ratio cannot be achieved by combining existing algorithms, which consider the tradeoffs between multiple resources and multiple reward rates separately.

By showing how to integrate these tradeoffs while making allocation decisions, we expand the applicability of competitive analysis in many areas, such as online advertising, matching markets, and personalized e-commerce. We test our methodological contribution on the hotel data set of Bodea et al. (2009), where there are multiple resources (hotel rooms), each with multiple reward rates (fares at which the room could be sold). We find that applying our algorithms, in conjunction with algorithms which attempt to forecast and learn the future transactions, results in the best performance.

1. Introduction

In this paper we study a general online resource allocation problem, stated in revenue management terminology. A firm has multiple items, each with an unreplenishable starting inventory, and a set of feasible prices at which its units of inventory could be sold. Heterogeneous customers arrive sequentially over time. Upon a customer's arrival, the probability that she would buy each item at each price is revealed; these probabilities can be 0 for items she is not interested in, or prices that are too high. The firm then chooses an available item and feasible price to offer her, after which her purchase decision is immediately realized according to the probability given. The firm's goal is to maximize its expected revenue before the inventories run out, or there are no more customers.

A special case of our problem is the *deterministic case*, where all purchase probabilities are 0 or 1. In this case, the firm knows the maximum a customer is willing to pay for each item, possibly 0. Therefore, the firm's decision can be reduced to choosing an item to *assign* to the customer (charging her maximum willingness-to-pay for that item), or *rejecting* the customer if her willingness-to-pay is low for every item.

We study these problems under the framework of competitive analysis. In competitive analysis, no information is given about the sequence of customers, nor are they assumed to follow any observable pattern. The algorithm's performance is expressed as a fraction of an optimum which knows the complete customer sequence in advance. For $c \leq 1$, if an algorithm can guarantee that this fraction is at least c for every problem instance (and customer sequence), then it is said to *achieve a competitive ratio of c*. The goal is to develop robust algorithms which achieve the *optimal* competitive ratio, i.e. a ratio c^* such that no algorithm, without knowing the customer sequence in advance, can do better.

1.1. Previous Work in Competitive Analysis

Our model involves multiple items, as well as multiple feasible prices for each item. This combines two challenges in competitive analysis, which have previously been studied separately.

1. **Multiple Items**: The challenge of how to prioritize between multiple items, when a customer can only be offered (or assigned) one of them, has been considered in the online b-matching problem (Kalyanasundaram and Pruhs 2000), Adwords problem (Mehta et al. 2007, Buchbinder et al. 2007), and online assortment problem (Golrezaei et al. 2014). The optimal

algorithms for these problems all perform some kind of *inventory balancing*, placing lower priority on selling items with lower remaining inventory. Inventory balancing algorithms are also related to the *randomized ranking* algorithms used in the online bipartite matching problem (Karp et al. 1990, Aggarwal et al. 2011).

2. **Multiple Prices**: The challenge of when to reject a customer only willing to pay a low price, to preserve inventory for customers willing to pay higher prices, has been considered in the single-item, deterministic case of our problem (Ball and Queyranne 2009, Lan et al. 2008). The optimal algorithm employs *booking limits*, rejecting customers with low willingness-to-pay once a threshold amount of the item has been sold.

Our model studies the challenges introduced when multiple prices are incorporated into the aforementioned problems with multiple items. In Section 6, we explain how our techniques can be extended to allow for fractional inventory consumption, like in the Adwords problem; or multiple items to be offered to each customer, like in the online assortment problem. We now discuss two additional ways to view our model, which emphasize the increase in modeling power from allowing for multiple prices:

- First, one can think of each of our (item, price)-combinations as an independent *product*. By allowing for multiple prices, we have allowed the multiple products corresponding to each item to draw from the same inventory, or *resource*. The different products can also consume different amounts of that resource, under the extension with fractional inventory consumption.
- Second, in some applications, the customers are classified under a finite number of *types*, and instead of a pricing decision, there is a different reward (corresponding to "match quality") for allocating each item to each customer type. This can be reduced to our problem, with the feasible prices for an item being that item's "match qualities" over all types. By allowing for multiple prices, we have allowed each item to yield different rewards when allocated to different types, as opposed to yielding the same reward for all types.

1.2. Integrating the Challenges

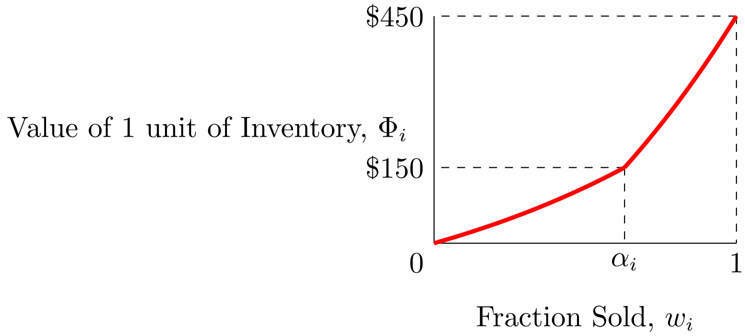
We introduce a *bid-price control* policy which achieves the optimal competitive ratio under both multiple items and multiple prices. Our algorithm maintains for each item a bid price, which is the value placed on one unit of its inventory. The *pseudorevenue* associated with an (item, price)-combination is then the price minus the value of the item. The algorithm offers to each customer the (item, price)-combination with the highest expected pseudorevenue, never offering combinations with non-positive pseudorevenue.

Bid-price control is a classical idea in revenue management (see Talluri and Van Ryzin (2006), Liu and Van Ryzin (2008)), where the bid prices are computed using an LP, based on the remaining inventory and forecasted distribution of remaining customers. However, since we make no assumptions about future customers, our bid prices are based on only the remaining inventory. Our bid prices are very simple—they are computed separately for each item i, like the multiplicative *penalties* in Golrezaei et al. (2014). Let w_i be the fraction of the starting inventory of i which has already been sold. At each point in time, the bid price of item i is set to $\Phi_i(w_i)$, where Φ_i is a *value function* dependent on the set of feasible prices for item i.

To illustrate our algorithm, we display the form of Φ_i for an example item i which could be sold at fares \$150 or \$450, in Figure 1. Note the following:

- As the fraction of item i sold increases over time, the value of one unit of inventory increases, hence the pseudorevenues associated with the feasible prices of item i decrease, and the bid-price algorithm places lower priority on offering/assigning item i. This captures the "inventory balancing" used to address the challenge of multiple items.
- Let α_i be the value at which $\Phi_i(\alpha_i) = 150$. The algorithm stops selling item i at the lower price of 150 once its fraction sold reaches α_i, because the pseudorevenue associated with the lower price is $150 - \Phi_i(w_i)$, which is non-positive for $w_i \geq \alpha_i$. Therefore, our algorithm captures the "booking limits" used to address the challenge of multiple prices. The specific value of $\Phi_i(w_i)$ also tells the algorithm how to choose between a lower price which may have higher expected revenue, versus a higher price which has lower expected inventory consumption.

Figure 1 The value function for an item with feasible price set $\{150, 450\}$.



- Φ_i increases from 0 to the maximum price of 450 over [0,1], and is piecewise-convex.

In general, each value function Φ_i is designed to maximize the competitive ratio CR_i associated with it. As we will explain, the exact function Φ_i is defined as the solution to a differential equation arising from a primal-dual analysis.

The booking limits implied by such a Φ_i are different than the booking limits derived by Ball and Queyranne (2009) which are optimal when i is the single item being sold. For example, if item i has two prices, with r_i being the ratio of high to low price, then the value of α_i is $\alpha(r_i)$, where

$$\alpha(r) = \ln \frac{2(r-1)}{\sqrt{1+4r(r-1)/e}-1}. \tag{1}$$

Meanwhile, the optimal booking limit from the single-item case is $\frac{r_i}{2r_i-1}$. α_i is greater than $\frac{r_i}{2r_i-1}$, with the intuition being that with multiple items, there is less upside to reserving inventory for higher prices, because the reserved units may have to compete with other items to be sold. Indeed, when there are both multiple items and multiple prices, the optimal algorithm must integrate inventory balancing when setting booking limits, instead of using the single-item booking limits.

1.3. Competitive Ratio Results

The overall competitive ratio associated with our algorithm is $\min_i \mathsf{CR}_i$, being limited by the item i with the smallest value of CR_i. While this competitive ratio is not achieved by the exact bid-price algorithm specified in the previous subsection, we prove the following results in this paper:

1. A variant of the bid-price algorithm, which we call Multi-price Balance, achieves a competitive ratio of $\min_i \mathsf{CR}_i$ in the *asymptotic regime*, where all starting inventories go to ∞.

2. A different variant of the bid-price algorithm, which we call Multi-price Ranking, achieves a competitive ratio of $\min_i \mathsf{CR}_i$ in the *deterministic* case of our problem.
3. A counterexample, which can be made to fall under both the asymptotic regime and the deterministic case, shows that the competitive ratio of any algorithm cannot exceed $\min \mathsf{CR}_i$.

When there is a single feasible price for an item i, $\mathsf{CR}_i = 1 - \frac{1}{e}$. Our statements 1–3 are generalizations of results that exist when every item has only one price. Statement 1 corresponds to the inventory balancing algorithm of Golrezaei et al. (2014) achieving a competitive ratio of $1 - \frac{1}{e}$. Statement 2 corresponds to the ranking-based algorithm of Aggarwal et al. (2011) achieving the same competitive ratio. Statement 3 shows that both of these results are tight.

These results may not be tight in the non-asymptotic, non-deterministic setting, which is an important open problem (Devanur et al. 2013) in the single-price case as well. Nonetheless, we establish lower bounds on the competitive ratio achieved which hold in the non-deterministic setting, and are parametrized by k, the minimum starting inventory of an item. As k increases, these bounds sharply approach the tight guarantee of $\min_i \mathsf{CR}_i$ from the asymptotic regime. In the single-price case, our bounds show that the multiplicative gap from $1 - \frac{1}{e}$ is at most $(1+k)(1-e^{-1/k})$, which improves the previously-best-known gap from Golrezaei et al. (2014).

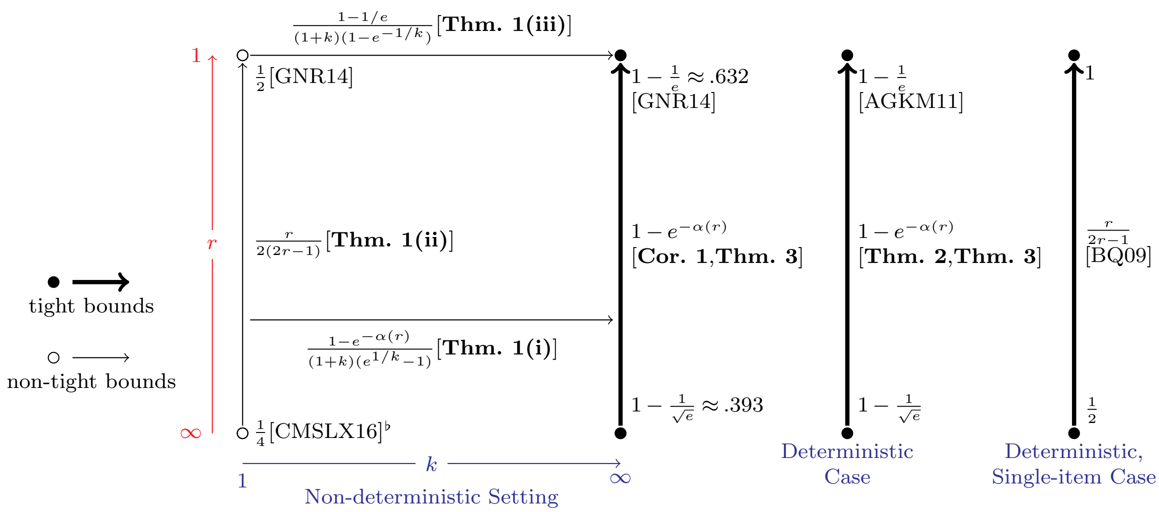
We illustrate our bounds on the case where every item has two feasible prices, in Figure 2. The competitive ratio CR_i associated with an item i is $1 - e^{-\alpha(r_i)}$, where r_i is its ratio of high to low price, and α is defined in (1). Thus the overall competitive ratio $\min_i \mathsf{CR}_i$ can be written as

$$1 - e^{-\alpha(r)}, \tag{2}$$

where $r = \max_i r_i$. (2) is decreasing in r. As $r \to 1$, $\alpha(r) \to 1$ and (2) approaches the known value of $1 - \frac{1}{e} \approx .632$. The smallest competitive ratio occurs as $r \to \infty$, with (2) approaching $1 - \frac{1}{\sqrt{e}} \approx .393$.

The formal statements of our theorems, which allow each item to have an arbitrary set of feasible prices, are deferred to Section 2. We analyze Multi-price Balance in Section 3 and Multi-price Ranking in Section 4. Descriptions of our techniques are also deferred to these sections.

Figure 2 Competitive ratios achieved in the two-price case, where r denotes the maximum ratio of an item's high to low price, and k denotes the minimum starting inventory of an item. The guarantees improve from bottom to top (as r decreases), and from left to right (as k increases).



$\flat$ The smallest guarantee of $\frac{1}{4}$ in this diagram is also implied by the results of Chen et al. (2016).

In general, the tight competitive ratio of CR_i can approach 0 is the feasible price set for item i contains both a large number of prices and a large ratio from highest to lowest price, which is a known negative result (Aggarwal et al. 2011). Nonetheless, in many applications, one can enumerate the price points (e.g., an item which could only be sold at \$19.99 or \$24.99), or bound the ratio between the highest and lowest prices (e.g., an advertiser who bids between .1 and .2).

1.4. Application on Hotel Data Set of Bodea et al. (2009)

We first summarize the general benefits of applying competitive analysis, and the competitive algorithms derived from this research. In contrast to traditional algorithms, which optimize based on a forecast of future demand, or attempt to learn the demand, competitive algorithms hedge against some worst case, and operate without any demand information. Most immediately, they are useful for products with highly unpredictable demand (Ball and Queyranne 2009, Lan et al. 2008), or for initializing new products with no historical sales data (Van Ryzin and McGill 2000). Second, by eschewing stochastic processes for generating demand, competitive algorithms are usually simple and flexible, leading to clean insights about the problem (Borodin and El-Yaniv 2005). Third, past research has reported on cases where competitive algorithms perform well in practice (Feldman et al. 2010), or on average in numerical experiments (Golrezaei et al. 2014, Chen et al. 2016).

In Section 7, we run simulations on the publicly-accessible hotel data set of Bodea et al. (2009). We use the product availability information to estimate customer choice models, and the transactional data as the sequence of arrivals. This leads to an online assortment problem like in Golrezaei et al. (2014), with multiple prices (advance-purchase rate, rack rate, etc.) for each item (King room, Two-double room, etc.). We compare the performance of our MULTI-PRICE BALANCE algorithm, using the extension discussed in Section 6 which can offer assortments, to various benchmarks and forecasting algorithms.

The main conclusion from our simulations is that the best performance is achieved by *hybrid* algorithms (see Golrezaei et al. (2014)). These are forecasting-based algorithms which continuously reference our forecast-independent value functions $\Phi_1, \ldots, \Phi_n$, and adjust their decisions accordingly. Although this only changes a small fraction ($\approx 5\%$) of decisions, these tend to be the decisions where the forecast is being most overconfident. Therefore, not only does this boost average performance, it drastically reduces the variance in performance caused when the forecast is wrong.

1.5. Other Related Work

We briefly discuss some related papers which has not been mentioned until now.

Alternate Approaches to Online Matching. Our problem captures the *online edge-weighted bipartite matching* problem, which has been studied under various settings designed to get around a basic impossibility result (see Aggarwal et al. (2011)). One such setting is *free disposal* (Feldman et al. 2009). Alternatively, one could assume that the arrivals appear in a *random order*, which allows for some form of learning (Kesselheim et al. 2013); this approach is very general and has been extended to online linear programming (Agrawal et al. 2014). However, to our knowledge, we are the first to focus on the *weight-dependent* competitive ratio for the online edge-weighted bipartite matching problem, instead of making assumptions such as free disposal or randomly-ordered arrivals. For a survey of online matching, we refer to Mehta (2013).

Known Stochastic Processes. When the stochastic process generating the arrivals in our problem is given, the resulting optimization problem is still computationally intractable. Nonetheless, many effective heuristics have been proposed, under different variations of the model (Zhang and Cooper 2005, Jasin and Kumar 2012, Ciocan and Farias 2012, Chen and Farias 2013). These heuristics can earn $\frac{1}{2}$ of the LP optimum in general settings (Chan and Farias 2009, Wang et al. 2015, Gallego et al. 2015). Manshadi et al. (2012) derive an improved performance ratio when the given stochastic process is IID. From a modeling perspective, our problem with multiple items and multiple prices is similar to the multi-fare, parallel flights problem of Zhang and Cooper (2005), and the appointment scheduling with customer preferences problem of Wang et al. (2015).

Alternate Metrics. Competitive/approximation ratio both consider the algorithm's expected reward as a fraction of an LP optimum. Our problem has been analyzed under other metrics as well. When the arrival process is unknown but assumed to be IID, one popular metric is regret, which measures the *additive* loss from optimum (see Ferreira et al. (2016)). When the arrival process is known, the *fluid* and *diffusion* analysis approaches have also been used (see Reiman and Wang (2008)). However, unlike competitive ratio, these metrics all tend to focus on asymptotic performance as the number of customers grows to infinity. Finally, a recent metric which has been studied is *regret ratio* (Zhang et al. 2016). For a comprehensive review of different metrics to use under different models of demand (for a single item), we refer the reader to Araman and Caldentey (2011).

2. Problem Definition, Algorithm Sketch, and Theorem Statements

A firm is selling $n \in \mathbb{N}$ different items. Each item $i \in [n]$[1] starts with a fixed inventory of $k_i \in \mathbb{N}$ units, and could be offered at one of $m_i \in \mathbb{N}$ feasible prices, with corresponding fares $r_i^{(1)}, \dots, r_i^{(m_i)} \in \mathbb{R}$

[1] For a general positive integer b, let $[b]$ denote the set $\{1, \dots, b\}$.

satisfying $0 < r_i^{(1)} < \ldots < r_i^{(m_i)}$. For convenience, we let $r_i^{(0)} = 0$ for each i. In Appendix E.1, we allow for a continuum of feasible prices in some range $[r^{\min}, r^{\max}]$.

There are $T \in \mathbb{N}$ customers arriving sequentially. Upon the arrival of customer $t \in [T]$, the firm observes $p_{t,i}^{(j)}$, the probability that customer t would buy item i at price j, for all $i \in [n]$ and $j \in [m_i]$.[2] The firm chooses up to one of the items i with inventory remaining, and offers it to customer t, at any price j. The customer accepts the offer with probability $p_{t,i}^{(j)}$, in which case the firm earns revenue $r_i^{(j)}$, and the inventory of item i is decremented by 1. In Section 6, we discuss models where multiple items can be offered or multiple units of inventory can be consumed at a time.

We define an *instance* $\mathcal{I}$ of the problem to consist of all of the following:

1. Initial information—$n, \{k_i, m_i, r_i^{(1)}, \ldots, r_i^{(m_i)} : i \in [n]\}$;
2. Arrival information—$T, \{p_{t,i}^{(j)} : t \in [T], i \in [n], j \in [m_i]\}$.

An *online algorithm* prescribes, based on the initial information, how to make the offering decision at each time t, without knowing $\{p_{t',i}^{(j)} : i \in [n], j \in [m_i]\}$ for future customers $t' > t$ nor the length of the time horizon T. For an online algorithm, let $\mathsf{ALG}(\mathcal{I})$ denote the revenue earned on a run on instance $\mathcal{I}$, which is a random variable with respect to the customers' purchase decisions as well as any coin flips in the algorithm.

Meanwhile, we can write the following LP based on instance $\mathcal{I}$:

$$\max \sum_{t=1}^{T} \sum_{i=1}^{n} \sum_{j=1}^{m_i} p_{t,i}^{(j)} r_i^{(j)} x_{t,i}^{(j)} \tag{3a}$$

$$\sum_{t=1}^{T} \sum_{j=1}^{m_i} p_{t,i}^{(j)} x_{t,i}^{(j)} \le k_i \qquad i \in [n] \tag{3b}$$

$$\sum_{i=1}^{n} \sum_{j=1}^{m_i} x_{t,i}^{(j)} \le 1 \qquad t \in [T] \tag{3c}$$

$$x_{t,i}^{(j)} \ge 0 \qquad t \in [T], i \in [n], j \in [m_i] \tag{3d}$$

LP (3) encapsulates the execution of any algorithm, which could make full use of the arrival information at the start, on instance $\mathcal{I}$. $x_{t,i}^{(j)}$ represents the unconditional probability of the algorithm

[2] These probabilities can be 0 for items the customer is not interested in, or prices that are too high. A rational customer would have $p_{t,i}^{(1)} \ge \ldots \ge p_{t,i}^{(m_i)}$, although we do not need this assumption.

offering item i at price j to customer t. (3b) enforces that starting inventories are respected, while (3c) enforces that at most one combination of item and price is offered to each customer. Objective function (3a) represents the expected revenue earned by the algorithm. Let $\mathsf{OPT}(\mathcal{I})$ denote the optimal objective value.

The *competitive ratio* of the online algorithm is then defined to be

$$\inf_{\mathcal{I}} \frac{\mathbb{E}[\mathsf{ALG}(\mathcal{I})]}{\mathsf{OPT}(\mathcal{I})}. \tag{4}$$

We say that an algorithm *achieves a competitive ratio of* c if (4) is lower-bounded by c.

Given any fixed online algorithm, (4) considers the worst-case instance, including the worst-case arrival sequence. The goal for the algorithm is to hedge against the worst-case arrival sequence, possibly by using randomness. Definition (4) provides a guarantee on $\mathbb{E}[\mathsf{ALG}(\mathcal{I})]$ relative to any algorithm which could have been possible, due to the following result.

LEMMA 1. $\mathsf{OPT}(\mathcal{I})$ *is an upper bound on the expected revenue of any algorithm, which could make full use of the arrival information at the start, on instance* $\mathcal{I}$.

The proof of Lemma 1 is deferred to Appendix A. The definition of OPT based on the LP is standard in problems with stochastic purchase realizations and arbitrary customer arrivals—we leave its justification to Mehta and Panigrahi (2012), Golrezaei et al. (2014).

In the *deterministic* case of our problem, every $p_{t,i}^{(j)}$ is 0 or 1. The problem can be simplified by letting $j_{t,i} = \max\{j \in [m_i] : p_{t,i}^{(j)} = 1\}$, with $j_{t,i} = 0$ if the set is empty, for all $t \in [T]$ and $i \in [n]$. We say that item i is *assigned* to customer t to indicate that i is offered to customer t at price $j_{t,i}$, which results in a sale; there is no reason to offer any other price. Customer t can also be *rejected*, e.g. if $j_{t,i}$ is low for every i. In the deterministic case, the LP (3) is integral, so $\mathsf{OPT}(\mathcal{I})$ is *equal* to the revenue of the best algorithm knowing the arrival sequence at the start.

2.1. The Multi-price Value Function Φ_i

For an arbitrary item i, we specify its value function Φ_i, which is dependent on its feasible prices $r_i^{(1)}, \ldots, r_i^{(m_i)}$. Recall that Φ_i is a function of w_i, the fraction of item i sold. For $w_i \in [0,1]$, $\Phi_i(w_i)$ is the value the algorithm currently places on one unit of inventory of i.

First we define booking limits $\alpha_i^{(1)},\dots,\alpha_i^{(m_i)}$, which are the fractions of starting inventory "reserved" for the respective fares $r_i^{(1)},\dots,r_i^{(m_i)}$, via the following proposition.

PROPOSITION 1. *Consider any item i, with an arbitrary number of discrete prices satisfying $0 < r_i^{(1)} < \dots < r_i^{(m_i)}$. There are unique positive values $\alpha_i^{(1)},\dots,\alpha_i^{(m_i)}$ which sum to 1 and satisfy*

$$1-e^{-\alpha_i^{(1)}} = \frac{1}{1-r_i^{(1)}/r_i^{(2)}}\cdot(1-e^{-\alpha_i^{(2)}}) = \dots = \frac{1}{1-r_i^{(m_i-1)}/r_i^{(m_i)}}\cdot(1-e^{-\alpha_i^{(m_i)}}). \tag{5}$$

There are also unique positive values $\sigma_i^{(1)},\dots,\sigma_i^{(m_i)}$ which sum to 1 and satisfy

$$\sigma_i^{(1)} = \frac{1}{1-r_i^{(1)}/r_i^{(2)}}\cdot\sigma_i^{(2)} = \dots = \frac{1}{1-r_i^{(m_i-1)}/r_i^{(m_i)}}\cdot\sigma_i^{(m_i)}. \tag{6}$$

Furthermore,

$$\alpha_i^{(1)} \geq \frac{1}{m_i}. \tag{7}$$

The proof of Proposition 1 is deferred to Appendix A. While finding the exact solution to (5) requires finding the roots of a degree-m polynomial, a numerical solution can easily be found via bisection search.

Proposition 1 contrasts $\alpha_i^{(1)},\dots,\alpha_i^{(m_i)}$ in (5) with the booking limits $\sigma_i^{(1)},\dots,\sigma_i^{(m_i)}$ in (6) originally derived by Ball and Queyranne (2009), which are optimal when i is the single item being sold. Inequality (7) says that the fraction of starting inventory reserved for the lowest fare is at least its "fair share", i.e. $1/m_i$ where m_i is the number of fares.

Booking limits $\alpha_i^{(1)},\dots,\alpha_i^{(m_i)}$ and value function Φ_i are a by-product of our analysis, and maximize the competitive ratio. Our method for deriving them is deferred to Appendix E. For now, we complete the definition of Φ_i:

DEFINITION 1. For each item i, define the following based on $\alpha_i^{(1)},\dots,\alpha_i^{(m_i)}$:

- $L_i^{(j)}$: the sum $\sum_{j'=1}^{j}\alpha_i^{(j')}$, defined for all $j=0,\dots,m_i$ (note that $L_i^{(0)}=0$ and $L_i^{(m_i)}=1$);
- $\ell_i(\cdot)$: a function on $[0,1]$, where $\ell_i(w)$ is the unique $j\in[m_i]$ for which $w\in[L_i^{(j-1)},L_i^{(j)})$ (note that $\ell_i(L_i^{(j)})=j+1$ for $j=0,\dots,m_i-1$; we define $\ell_i(L_i^{(m_i)})$ to be m_i).

The value function Φ_i is then defined over $w_i \in [0,1]$ by

$$\Phi_i(w_i) = r_i^{(\ell_i(w_i)-1)} + \big(r_i^{(\ell_i(w_i))} - r_i^{(\ell_i(w_i)-1)}\big)\frac{\exp(w_i - L_i^{(\ell_i(w_i)-1)}) - 1}{\exp(\alpha_i^{(\ell_i(w))}) - 1}. \tag{8}$$

An example of a value function Φ_i with 2 prices was plotted in Figure 1. In general, Φ_i is continuously increasing and piecewise-convex over m_i *segments* of lengths $\alpha_i^{(1)}, \ldots, \alpha_i^{(m_i)}$, separated by *segment borders* $L_i^{(0)}, \ldots, L_i^{(m_i)}$. For each j, Φ_i reaches the value of $r_i^{(j)}$ at $L_i^{(j)}$, hence price j stops being offered once the fraction sold w_i reaches $L_i^{(j)}$.

We will see that the competitive ratio CR_i associated with Φ_i is $1 - e^{-\alpha_i^{(1)}}$, which is optimal. When $m_i = 1$, it can be seen that $\alpha_i^{(1)} = 1$, $\Phi_i(w_i) = r_i^{(1)} \cdot \frac{e^{w_i} - 1}{e - 1}$, and $\mathsf{CR}_i = 1 - \frac{1}{e}$, which correspond to known results. The functions $\Phi_1, \ldots, \Phi_n$ facilitate the tradeoff between immediate reward and future inventory. We develop two algorithms, which use them in different ways.

2.2. Sketch of our MULTI-PRICE BALANCE and MULTI-PRICE RANKING Algorithms

We first sketch Multi-price Ranking, which is simpler. It assumes that $k_i = 1$ for all i, which does not lose generality since an item which starts with multiple units of inventory can be transformed into multiple disparate items. At the start, the algorithm fixes for each item i a random seed W_i, drawn independently and uniformly from $[0,1]$. It then treats $\Phi_i(W_i)$ as the bid price for the single unit of item i: it offers to each customer t the available item i and price j maximizing the expected pseudorevenue, $p_{t,i}^{(j)}\big(r_i^{(j)} - \Phi_i(W_i)\big)$.

Multi-price Ranking hedges against the ambiguity in customer arrivals by using randomness, which is standard in competitive analysis. The random seed W_i determines the random minimum price at which the algorithm is willing to sell item i, as well as a random priority for selling i when the algorithm is choosing between multiple items.

We now sketch Multi-price Balance, which updates the bid price of each item i based on the fraction w_i of its k_i units which has been sold. However, the algorithm does not directly use $\Phi_i(w_i)$ as the bid price of item i, because w_i would always be a multiple of $\frac{1}{k_i}$, while the booking limits and segment borders which Φ_i is based on may not be multiples of $\frac{1}{k_i}$. Instead, the algorithm first uses a randomized scheme for rounding the booking limits to multiples of $\frac{1}{k_i}$.

Specifically, at the start, the algorithm fixes for each item i *random* segment borders $\tilde{L}_i^{(0)}, \ldots, \tilde{L}_i^{(m_i)}$, which are multiples of $\frac{1}{k_i}$ satisfying $0 = \tilde{L}_i^{(0)} \le \ldots \le \tilde{L}_i^{(m_i)} = 1$. These realizations imply a *random*, *perturbed* value function $\tilde{\Phi}_i$. Function $\tilde{\Phi}_i$ is defined on $\{0, \frac{1}{k_i}, \ldots, 1\}$, since the fraction sold w_i is always a multiple of $\frac{1}{k_i}$. Function $\tilde{\Phi}_i$ still satisfies $0 = \tilde{\Phi}_i(0) \le \tilde{\Phi}_i(\frac{1}{k_i}) \le \ldots \le \tilde{\Phi}_i(1)$. The algorithm treats $\tilde{\Phi}_i(w_i)$ as the bid price for item i: it offers to each customer t the item i and price j maximizing the expected pseudorevenue

$$p_{t,i}^{(j)}\big(\tilde{\Phi}_i(\tilde{L}_i^{(j)}) - \tilde{\Phi}_i(w_i)\big). \tag{9}$$

In expression (9), the definition of pseudorevenue at price j is $\tilde{\Phi}_i(\tilde{L}_i^{(j)}) - \tilde{\Phi}_i(w_i)$, i.e. we have used $\tilde{\Phi}_i(\tilde{L}_i^{(j)})$ in place of $r_i^{(j)}$, so that the pseudorevenue is exactly 0 when $w_i = \tilde{L}_i^{(j)}$. However, the realized $\tilde{\Phi}_i$ will generally be close to Φ_i, so that $\tilde{\Phi}_i(\tilde{L}_i^{(j)}) \approx r_i^{(j)}$. In the asymptotic regime with $k_i \to \infty$, $\tilde{\Phi}_i = \Phi_i$ deterministically. However, for small k_i, optimizing a *randomized procedure* for initializing $\tilde{\Phi}_i$ (based on $r_i^{(1)}, \ldots, r_i^{(m_i)}$ as well as k_i) instead of having a deterministic Φ_i (based on only $r_i^{(1)}, \ldots, r_i^{(m_i)}$) allows us to achieve a greater competitive ratio.

2.3. Statements of Our Results

THEOREM 1. MULTI-PRICE BALANCE *achieves a competitive ratio of* $\min_i \widetilde{\mathsf{CR}}_i$*, where for all* i*,* $\widetilde{\mathsf{CR}}_i$ *is lower-bounded by all of: (i)* $\frac{1-e^{-\alpha_i^{(1)}}}{(1+k_i)(e^{1/k_i}-1)}$*; (ii)* $\frac{\sigma_i^{(1)}}{2}$*; and (iii)* $\frac{1-e^{-1}}{(1+k_i)(1-e^{-1/k_i})}$*, if* $m_i = 1$*.*

COROLLARY 1. MULTI-PRICE BALANCE *achieves a competitive ratio approaching* $1 - \exp(-\min_i \alpha_i^{(1)})$ *as each starting inventory* k_i *approaches* ∞*.*

COROLLARY 2. *Suppose that each item has at most* m *discrete prices and at least* k *units of starting inventory. Then the competitive ratio achieved by* MULTI-PRICE BALANCE *is lower-bounded by* $\frac{1-e^{-1/m}}{(1+k)(e^{1/k}-1)}$*, which approaches* $1 - e^{-1/m}$ *as* k *approaches* ∞*.*

Theorem 1 is our general result, where for each i, $\widetilde{\mathsf{CR}}_i$ is the competitive ratio associated with the optimal randomized procedure for initializing $\tilde{\Phi}_i$, based on $r_i^{(1)},\ldots,r_i^{(m_i)}$ and k_i.

Lower bound (i) on $\widetilde{\mathsf{CR}}_i$ is attained by a randomized procedure which defines $\tilde{\Phi}_i$ based on the fixed function Φ_i. The numerator in (i) is a function of the feasible prices $r_i^{(1)},\ldots,r_i^{(m_i)}$, while the denominator is a function of the starting inventory k_i. Note that the denominator $(1+k_i)(e^{1/k_i}-1)$ decreases toward 1 as $k_i\to\infty$, resulting in Corollary 1. Corollary 2 is a further simplification of the bound presented, using inequality (7). Lower bound (ii) is attained from solving an optimization problem for the best randomized procedure, which is tractable when $k_i=1$. Interestingly, the bound turns out to be based on the single-item booking limits $\sigma_i^{(1)},\ldots,\sigma_i^{(m_i)}$, instead of $\alpha_i^{(1)},\ldots,\alpha_i^{(m_i)}$. Lower bound (iii) is the improvement of (i) in the single-price case, where we have gained a factor of e^{1/k_i}. It simplifies and improves the bound from Golrezaei et al. (2014).

Multi-price Balance is formalized and Theorem 1 is proven in Section 3. We explain the ideas behind our primal-dual analysis, why we need *random* value functions, and how to overcome the resulting analytical challenges.

Theorem 2. *Multi-price Ranking achieves a competitive ratio of* $1-\exp(-\min_i\alpha_i^{(1)})$ *in the deterministic case of our problem.*

Multi-price Ranking is formalized and Theorem 2 is proven in Section 4. While we described Multi-price Ranking as an algorithm for our general problem in Subsection 2.2, it is most amenable to analysis in the deterministic case. This is also true in the single-price case, as our analysis uses the framework of Devanur et al. (2013) and extends it to handle to multiple prices.

Theorem 3. *Consider a set of m prices satisfying $0<r^{(1)}<\ldots<r^{(m)}$, from which $\alpha^{(1)}$ and $\sigma^{(1)}$ are defined according to Proposition 1. Then there exists a distribution over instances $\mathcal{I}$ (a "randomized instance") with $m_i=m$ and $r_i^{(1)}=r^{(1)},\ldots,r_i^{(m)}=r^{(m)}$ for each item i, on which no online algorithm can have expected revenue greater than $(1-e^{-\alpha^{(1)}})\mathbb{E}[\mathsf{OPT}(\mathcal{I})]$. Furthermore, for every instance $\mathcal{I}$ in the support of the distribution:*

1. *the starting inventories k_i can be made arbitrarily large;*

2. *$\mathcal{I}$ falls under the deterministic case of our problem.*

Theorem 3 is proven in Section 5. It implies that no online algorithm can achieve a competitive ratio greater than $1-e^{-\alpha^{(1)}}$, via Yao's minimax principle (Yao 1977). The counterexample can be made to satisfy the conditions of both Corollary 1 and Theorem 2, showing that these results are tight.

In our counterexample, a large number of customers arrive according to a random permutation, like in Karp et al. (1990), Mehta et al. (2007), Golrezaei et al. (2014). In our case, the customers are further split into m "phases", where the customers in phase j are willing to pay $r^{(j)}$ for any of the items they are interested in. The lengths of the phases are optimized by an adversary to minimize the competitive ratio.

Interestingly, on the related counterexamples from the literature (Karp et al. 1990, Mehta et al. 2007, Ball and Queyranne 2009, Golrezaei et al. 2014), all (reasonable) algorithms have the same performance. On our counterexample, with the adversarially-optimized phase lengths, the unique optimal algorithm turns out to be our two algorithms. We say *unique* because Multi-price Balance and Multi-price Ranking converge toward the same algorithm as the starting inventories go to ∞; this phenomenon has also been noted in the single-price case by Aggarwal et al. (2011).

Proposition 2. *For $m \geq 2$ prices satisfying $0 < r^{(1)} < \ldots < r^{(m)}$, from which $\alpha^{(1)}$ and $\sigma^{(1)}$ are defined according to Proposition 1, the following inequalities hold:*

$$(1-\frac{1}{e})\cdot\sigma^{(1)} < 1-e^{-\sigma^{(1)}} < 1-e^{-\alpha^{(1)}}; \tag{10}$$

$$\frac{1}{1+\ln\frac{r^{(m)}}{r^{(1)}}} < \sigma^{(1)}; \tag{11}$$

$$1-e^{-\alpha} < 1-e^{-\alpha^{(1)}},\ \textit{where } \alpha \textit{ is the unique solution to } 1-e^{-\alpha} = \frac{1-\alpha}{\ln\frac{r^{(m)}}{r^{(1)}}}. \tag{12}$$

Finally, Proposition 2, which is proven in Appendix A, puts our tight competitive ratio of $1-e^{-\alpha^{(1)}}$ into perspective. $\sigma^{(1)}$ is the existing tight competitive ratio for a single item, while $1-\frac{1}{e}$ is the existing tight competitive ratio for multiple items with one price each. (10) shows that our competitive ratio for multiple items with multiple prices is not a naive combination of the existing competitive ratios, and hence our algorithms cannot be obtained by combining existing algorithms.

With a single item whose price can take any value in the continuum $[r^{(1)}, r^{(m)}]$, the tight competitive ratio is $\frac{1}{1+\ln(r^{(m)}/r^{(1)})}$ (Ball and Queyranne 2009). $1-e^{-\alpha}$, with α as defined in (12), is our corresponding competitive ratio when there are multiple items (α can be solved to equal $1-W(Re^{R-1})/R$, where W is the inverse of the function $f(x)=xe^x$, and $R=\ln(r^{\max}/r^{\min})$—see Appendix E.1). (11) and (12) say that when the prices vary within a discrete subset of $[r^{(1)}, r^{(m)}]$, the competitive ratios can only be greater. (11) combined with (10) shows that our competitive ratio of $1-e^{-\alpha^{(1)}}$ is $\Omega(\frac{1}{\log(r^{(m)}/r^{(1)})})$.

3. MULTI-PRICE BALANCE and the Proof of Theorem 1

Multi-price Balance, as sketched in Subsection 2.2, is formalized in Algorithm 1. For now, we consider a generic randomized procedure for initializing $\tilde{L}_i^{(0)}, \ldots, \tilde{L}_i^{(m_i)}$ and $\tilde{\Phi}_i$ in Step 1, which deterministically satisfies the following monotonicity conditions:

$$\tilde{L}_i^{(0)}, \ldots, \tilde{L}_i^{(m_i)} \in \{0, \frac{1}{k_i}, \ldots, 1\},\ 0 = \tilde{L}_i^{(0)} \le \ldots \le \tilde{L}_i^{(m_i)} = 1; \tag{13}$$

$$\tilde{\Phi}_i(0), \tilde{\Phi}_i(\frac{1}{k_i}), \ldots, \tilde{\Phi}_i(1) \in \mathbb{R},\ 0 = \tilde{\Phi}_i(0) \le \tilde{\Phi}_i(\frac{1}{k_i}) \le \ldots \le \tilde{\Phi}_i(1). \tag{14}$$

Since $\tilde{\Phi}_i$ is non-decreasing, the expression $\tilde{\Phi}_i(\tilde{L}_i^{(j)}) - \tilde{\Phi}_i(\frac{N_i}{k_i})$ in (15) is non-positive once the number sold N_i reaches k_i. Therefore, Algorithm 1 never tries to offer an item i which has stocked out.

Theorem 4. *Suppose in Line 1 of Algorithm 1, for each $i \in [n]$, the segment borders $\tilde{L}_i^{(1)}, \ldots, \tilde{L}_i^{(m_i)}$ and value function $\tilde{\Phi}_i$ are randomly initialized in a way such that*

$$k_i(\tilde{\Phi}_i(\frac{N+1}{k_i}) - \tilde{\Phi}_i(\frac{N}{k_i})) + \tilde{\Phi}_i(\tilde{L}_i^{(j)}) - \tilde{\Phi}_i(\frac{N}{k_i}) \le \frac{r_i^{(j)}}{F}, \qquad j \in [m_i], N \in \{0, \ldots, \tilde{L}_i^{(j)} k_i - 1\}; \tag{16}$$

$$\mathbb{E}[\tilde{\Phi}_i(\tilde{L}_i^{(j)})] \ge r_i^{(j)}, \qquad j \in [m_i]. \tag{17}$$

Then Algorithm 1 achieves a competitive ratio of F.

Theorem 4 identifies conditions which, when satisfied by the randomized procedure for each i, yields a competitive ratio of F. Note that (16) needs to hold for every potential instantiation of

Algorithm 1 MULTI-PRICE BALANCE

1: Initialize $\tilde{L}_i^{(0)},\dots,\tilde{L}_i^{(m_i)},\tilde{\Phi}_i$ randomly and independently for each $i\in[n]$
2: $N_i\leftarrow 0$ for all $i\in[n]$ (N_i tracks the total number of copies of item i sold, at any price)
3: **for** $t=1,2,\dots$ **do**
4: Compute
$$\max_{i\in[n],j\in[m_i]} p_{t,i}^{(j)}(\tilde{\Phi}_i(\tilde{L}_i^{(j)})-\tilde{\Phi}_i(\frac{N_i}{k_i})) \tag{15}$$
5: **if** the value of (15) is strictly positive **then**
6: Offer any item i_t^* and price j_t^* maximizing (15) to customer t
7: **if** customer t accepts (occurring with probability $p_{t,i_t^*}^{(j_t^*)}$) **then**
8: $Z_t\leftarrow\tilde{\Phi}_{i_t^*}(\tilde{L}_{i_t^*}^{(j_t^*)})-\tilde{\Phi}_{i_t^*}(N_{i_t^*}/k_{i_t^*})$ (this is the pseudorevenue earned)
9: $N_{i_t^*}\leftarrow N_{i_t^*}+1$
10: **end if**
11: **end if**
12: **end for**

$\tilde{\Phi}_i$, while (17) only needs to hold in expectation over the instantiations. We prove Theorem 4 in Appendix B, but outline its proof here and provide some intuition.

First, we take the dual of the LP (3):

$$\min\sum_{i=1}^{n}k_iy_i+\sum_{t=1}^{T}z_t \tag{18a}$$

$$p_{t,i}^{(j)}y_i+z_t\ge p_{t,i}^{(j)}r_i^{(j)} \qquad t\in[T],i\in[n],j\in[m_i] \tag{18b}$$

$$y_i,z_t\ge 0 \qquad i\in[n],t\in[T] \tag{18c}$$

By weak duality, $\mathsf{OPT}(\mathcal{I})$ is bounded from above by the objective value of any feasible dual solution.

During the (random) execution of Algorithm 1, it maintains a dual variable $y_i=\tilde{\Phi}_i(\frac{N_i}{k_i})$ for each i. At each time t, only if a sale is realized, does the algorithm set z_t to a non-zero value Z_t (Line 8) and increment the y_i-variables by incrementing $N_{i_t^*}$ (Line 9). We prove three claims:

1. During each time $t\in[T]$, the gain in the dual objective is at most some multiple $\frac{1}{F}$ of the revenue earned by the algorithm;
2. During each time $t\in[T]$, the conditional expectation of Z_t over the random purchase decision of customer t, combined with the current value of y_i, make the LHS of (18b) at least $p_{t,i}^{(j)}\cdot\tilde{\Phi}_i(\tilde{L}_i^{(j)})$, for all $i\in[n]$ and $j\in[m_i]$;
3. The expectation of $\tilde{\Phi}_i(\tilde{L}_i^{(j)})$, over the random segment borders and value function initially chosen by the algorithm, is at least $r_i^{(j)}$, for all $i\in[n]$ and $j\in[m_i]$.

Claim 1 follows from condition (16), while Claim 3 follows from condition (17). Claims 2 and 3 can be combined to show that the dual variables y_i and z_t maintained by the algorithm are feasible, after taking an expectation over all sample paths.

We explain the intuition behind our idea of a *random* value function, and the resulting analysis. Even for a single item, with a small starting inventory and a large ratio r from its highest to lowest price, in order to achieve a constant competitive ratio which does not scale with r, one must use *random* booking limits (Ball and Queyranne 2009). With multiple items, our equivalent is to have the configuration of segment borders $\tilde{L}_i^{(0)}, \dots, \tilde{L}_i^{(m_i)}$ be random, and define an arbitrary value function $\tilde{\Phi}_i$ corresponding to each one. In order to "average" over these configurations in the analysis, we relax dual feasibility to only hold in expectation. The idea of feasibility in expectation has been previously seen, but in different contexts: in Devanur et al. (2013), over a random seed, and in Golrezaei et al. (2014), over a random purchase decision (similar to our Claim 2).

3.1. Optimizing the Randomized Procedures

Theorem 4 reduces the problem of deriving a competitive algorithm to that of finding a randomized procedure for initializing $\tilde{\Phi}_1, \dots, \tilde{\Phi}_n$ satisfying (16)–(17). We can consider this problem separately for each i, based on $r_i^{(1)}, \dots, r_i^{(m_i)}$ and k_i, and omit the subscript i.

A randomized procedure consists of a distribution over the all of the configurations satisfying (13), and for each configuration, values for $\tilde{\Phi}(\frac{1}{k}), \tilde{\Phi}(\frac{2}{k}), \dots, \tilde{\Phi}(1)$ satisfying (14). We would like to find a randomized procedure which satisfies (16)–(17) with a maximal value of F. While this optimization problem is intractable in general, we can use the intuition behind the definitions of $L^{(0)}, \dots, L^{(m)}$ and Φ from Subsection 2.1 to specify a near-optimal randomized procedure.

Definition 2. Define the following randomized procedure for initializing $\tilde{\Phi}$:

1. Draw a random seed W uniformly from $[0,1]$;
2. For each j, set $\tilde{L}^{(j)} = \frac{\lfloor L^{(j)}k \rfloor + 1}{k}$ if $W < L^{(j)}k - \lfloor L^{(j)}k \rfloor$, and $\tilde{L}^{(j)} = \frac{\lfloor L^{(j)}k \rfloor}{k}$ otherwise;
3. For $q \in \{0, \frac{1}{k}, \dots, 1\}$, let $\tilde{\ell}(q)$ be the unique $j \in [m]$ such that $\tilde{L}^{(j-1)} \le q < \tilde{L}^{(j)}$ (note that $\tilde{\ell}(\tilde{L}^{(j)}) = j+1$ for $j = 0, \dots, m-1$; we define $\tilde{\ell}(\tilde{L}^{(m)})$ to be m).

The value function $\tilde{\Phi}$ is then defined over $q \in \{0, \frac{1}{k}, \ldots, 1\}$ by

$$\tilde{\Phi}(q) = \sum_{j=1}^{\tilde{\ell}(q)-1} (r^{(j)} - r^{(j-1)}) \frac{\exp(\tilde{L}^{(j)} - \tilde{L}^{(j-1)}) - 1}{\exp(\alpha^{(j)}) - 1} + (r^{(\tilde{\ell}(q))} - r^{(\tilde{\ell}(q)-1)}) \frac{\exp(q - \tilde{L}^{(\tilde{\ell}(q)-1)}) - 1}{\exp(\alpha^{(\tilde{\ell}(q))}) - 1}. \quad (19)$$

It is important that the random segment borders $\tilde{L}^{(0)}, \ldots, \tilde{L}^{(m)}$ are rounded *comonotonically* (in a perfectly positively correlated fashion) using a single seed, both to ensure that they satisfy the monotonicity condition in (13), and to reduce the number of potential configurations on which (16) needs to hold. $\tilde{\Phi}$ increases over the m (possibly empty) "segments" of its domain $\{0, \frac{1}{k}, \ldots, 1\}$, which are "bordered" by $\tilde{L}^{(0)}, \ldots, \tilde{L}^{(m)}$. (19) is similar to definition (8) for Φ, except the sum in (19) does not telescope, since $\tilde{L}^{(j)} - \tilde{L}^{(j-1)}$ equals $\alpha^{(j)}$ only in expectation.

Theorem 5. *The randomized procedure for initializing $\tilde{\Phi}$ from Definition 2 satisfies (16)–(17) with $F = \frac{1-e^{-\alpha^{(1)}}}{(1+k)(e^{1/k}-1)}$. Furthermore, if $m = 1$, then the value of F can be improved to $\frac{1-e^{-\alpha^{(1)}}}{(1+k)(1-e^{-1/k})}$.*

Theorem 5 is proven in Appendix B. It, in conjunction with Theorem 4, establishes bounds (i) and (iii) from our main result for Multi-price Balance, Theorem 1. In Appendix B, we state the complete proof of Theorem 1, including bound (ii), which involves explicitly formulating the optimization problem over randomized procedures and solving it when $k = 1$.

4. MULTI-PRICE RANKING and the Proof of Theorem 2

In Subsection 2.2, we sketched Multi-price Ranking, for our general problem. In Algorithm 2, we formalize it in the deterministic case, which is the case analyzed in Theorem 2. Recall that we have assumed, without loss of generality, that $k_i = 1$ for each item i.

Our analysis extends the framework of Devanur et al. (2013) to incorporate multiple prices. It uses the dual LP defined in (18), where every $p_{t,i}^{(j)}$ is 0 or 1.

If Algorithm 2 assigns item i to customer t (charging price $j_{t,i}$), then we set dual variables $Z_t = r_i^{(j_{t,i})} - \Phi_i(W_i)$ and $Y_i = \Phi_i'(W_i)$, where Φ_i is the fixed function defined in Subsection 2.1 (we ignore the measure-zero set where Φ_i' is undefined). All dual variables not set during a time period are defined to be zero. The following lemmas are proven in Appendix C:

Lemma 2. *If Algorithm 2 assigns item i to customer t, then $(1 - e^{-\alpha_i^{(1)}})(Y_i + Z_t) \le r_i^{(j_{t,i})}$ w.p.1.*

Algorithm 2 Multi-price Ranking in the Deterministic Case

Initialize W_i uniformly at random from $[0,1]$, independently for each $i \in [n]$
$\mathsf{available}_i \leftarrow \mathbf{true}$ for all $i \in [n]$
for $t = 1, 2, \ldots$ **do**
  Compute

$$\max_{i \in [n], j \in [m_i]: \mathsf{available}_i = \mathbf{true}} (r_i^{(j_{t,i})} - \Phi_i(W_i)) \tag{20}$$

  if the value of (20) is strictly positive **then**
    Offer any item i_t^* maximizing (20) to customer t, at price j_{t,i_t^*}
    $\mathsf{available}_{i_t^*} \leftarrow \mathbf{false}$
  end if
end for

Lemma 3. *Setting $y_i = \mathbb{E}[Y_i], z_t = \mathbb{E}[Z_t]$ for all i, t forms a feasible solution to the dual LP (18).*

The proof of Theorem 2 is straight-forward given these lemmas:

Proof of Theorem 2. Lemma 3 implies $\mathsf{OPT}(\mathcal{I}) \leq \sum_{i=1}^n \mathbb{E}[Y_i] + \sum_{t=1}^T \mathbb{E}[Z_t]$, via weak duality. However, by Lemma 2, the revenue earned by Algorithm 2, or ALG, is at least $\min_{i \in [n]}\{1 - e^{-\alpha_i^{(1)}}\} \cdot \left(\sum_{i=1}^n Y_i + \sum_{t=1}^T Z_t\right)$, with probability 1. Thus, $\mathbb{E}[\mathsf{ALG}] \geq (1 - \exp(-\min_{i \in [n]} \alpha_i^{(1)})) \cdot \mathsf{OPT}(\mathcal{I})$. □

5. Randomized Instance and the Proof of Theorem 3

We formalize the randomized instance described in Subsection 2.3 and use it to prove Theorem 3.

The $n \in \mathbb{N}$ items i all have $m_i = m$, $r_i^{(j)} = r^{(j)}$ for all j, and $k_i = k$ for some $k \in \mathbb{N}$. We think of n as going to ∞, while k is arbitrary. Throughout this example, we often express quantities as portions τ of n. We abuse notation and write τn to refer to an integer, even if τ is irrational, since the error from rounding τn to the nearest integer is negligible as $n \to \infty$.

The arrival sequence is randomized following the classical construction of Karp et al. (1990). There are $T = nk$ customers, split into n "groups" of k identical customers each. Uniformly draw a random permutation $\pi = (\pi_1, \ldots, \pi_n)$ of $(1, \ldots, n)$ from the $n!$ possibilities. For $i \in [n]$, all k customers in group i would deterministically buy any item in $\{\pi_i, \ldots, \pi_n\}$. Our construction differs from existing ones in that the n groups of customers are further split into m "phases". Let $\beta_1, \ldots, \beta_m$ be positive numbers summing to 1, corresponding to the fraction of groups in each phase, whose values we specify later. For all $j \in [m]$, the customers in groups $(\beta_1 + \ldots + \beta_{j-1})n + 1, \ldots, (\beta_1 + \ldots + \beta_j)n$ are willing to pay $r^{(j)}$ for any of the items in their interest set.

Definition 3. Define the following shorthand notation for all $j=1,\ldots,m+1$:

- $A_j := \sum_{\ell=j}^{m} \alpha^{(\ell)}$ (note that $A_1=1$ and $A_{m+1}=0$);
- $B_j := \sum_{\ell=j}^{m} \beta_\ell$ (note that $B_1=1$ and $B_{m+1}=0$).

Proposition 3. *Given $m\in\mathbb{N}$, $0<r^{(1)}<\ldots<r^{(m)}$, and $\alpha^{(1)},\ldots,\alpha^{(m)}$ as defined in Proposition 1, there exists a unique solution to the following system of equations in variables $B_2,\ldots,B_m$:*

$$B_m r^{(m)} e^{-\alpha^{(m)}} = \ldots = B_2 r^{(2)} e^{-\alpha^{(2)}} = r^{(1)} e^{-\alpha^{(1)}}, \tag{21}$$

with $0<B_m<\ldots<B_2<B_1=1$.

We define $B_2,\ldots,B_m$ according to Proposition 3. This implies definitions for $\beta_1,\ldots,\beta_m$, which are strictly positive and sum to 1.

Now, regardless of the permutation π, the optimal algorithm allocates the k copies of item π_i to the customers in group i, for each $i\in[n]$, successfully serving all $T=nk$ customers and earning revenue $\sum_{j=1}^{m} r^{(j)}(\beta_j n)k$. This is also the optimal objective value of the LP (3). Therefore, $\mathsf{OPT}(\mathcal{I})=\sum_{j=1}^{m} r^{(j)}(\beta_j n)k$ deterministically, which we can rewrite as

$$\sum_{j=1}^{m}(r^{(j)}-r^{(j-1)})B_j nk. \tag{22}$$

5.1. Upper Bound on Performance of Online Algorithms

Lemma 4. *The expected revenue of an online algorithm on this randomized instance is upper-bounded by the maximum value of*

$$\sum_{j=1}^{m} r^{(j)} B_j n(1-e^{-\lambda_j})k \tag{23}$$

subject to $0\le\lambda_j\le\ln\frac{B_j}{B_{j+1}}$ for $j\in[m-1]$, $0\le\lambda_m$, and $\sum_{j=1}^{m}\lambda_j\le 1$.

Lemma 4 drastically simplifies the analysis of the online algorithm, because it restricts to algorithms which are *indifferent* to the realized permutation π, allowing for a deterministic analysis. However, our analysis differs from existing ones (e.g. (Golrezaei et al. 2014, Lem. 6)) in that despite the item symmetry, the online algorithm has a decision—how many customers in each phase to serve, as opposed to reserving inventory for customers in future phases.

This is controlled by the λ-variables, where λ_j denotes the expected fraction of item π_n's inventory sold to phase-j customers. The expected number of groups served during phase j is then at most $B_j n(1-e^{-\lambda_j})$, resulting in the upper bound (23). Constraint $\lambda_j \le \ln\frac{B_j}{B_{j+1}}$ comes from the fact that $B_j n(1-e^{-\lambda_j})$ must not exceed the total number of groups in phase j, $\beta_j n$.

LEMMA 5. *Let $j \in [m]$ and $\tau \in [0,1]$. The maximum value of*

$$\sum_{\ell=j}^{m} r^{(\ell)} B_\ell n(1-e^{-\lambda_\ell})k \tag{24}$$

subject to $\lambda_\ell \ge 0$ for all $\ell = j,\dots,m$ as well as $\sum_{\ell=j}^{m}\lambda_\ell \le \tau$ is

$$nk\sum_{\ell=j}^{m} r^{(\ell)} B_\ell\Big(1-\exp\big(-\alpha^{(\ell)}+\frac{A_j-\tau}{m-j+1}\big)\Big). \tag{25}$$

Lemma 5 establishes the optimal objective value of the optimization problem from Lemma 4. The upper bound of $\ln\frac{B_j}{B_{j+1}}$ on λ_j for $j\in[m-1]$ turns out to not be binding. With both lemmas, the proof of Theorem 3 is easy.

Proof of Theorem 3. The value of (25) with $j=1$ and $\tau=1$ is

$$nk\sum_{\ell=1}^{m} r^{(\ell)} B_\ell(1-e^{-\alpha^{(\ell)}}) = (1-e^{-\alpha^{(1)}})\sum_{\ell=1}^{m}(r^{(\ell)}-r^{(\ell-1)})B_\ell nk, \tag{26}$$

where we have used (5) to derive the equality. Combining Lemmas 4–5, we get that the RHS of (26) is an upper bound on $\mathbb{E}[\mathsf{ALG}(\mathcal{I})]$, for any online algorithm. Meanwhile, $\mathsf{OPT}(\mathcal{I})$ on this randomized instance is always equal to (22), which is exactly the RHS of (26) divided by $(1-e^{-\alpha^{(1)}})$. We have established that $\mathbb{E}[\mathsf{ALG}(\mathcal{I})] \le (1-e^{-\alpha^{(1)}})\mathbb{E}[\mathsf{OPT}(\mathcal{I})]$, which is the desired result. Finally, the second condition of Theorem 3 is clearly satisfied; the first condition is also satisfied because our analysis holds for any value of $k\in\mathbb{N}$, hence k can be made arbitrarily large. $\square$

REMARK 1. Suppose $k\to\infty$. It can be seen that our algorithm (either MULTI-PRICE BALANCE or MULTI-PRICE RANKING, which behave identically on this instance—see Aggarwal et al. (2011)), with booking limits $\alpha^{(1)},\dots,\alpha^{(m)}$, is the unique optimal algorithm on this instance. The proof of Lemma 4 shows that given $\lambda_1,\dots,\lambda_m$, the dominant strategy for the online algorithm is to deplete the inventories of items evenly (which is possible since $k\to\infty$), in which case upper bound (23) is attained. The proof of Lemma 5 shows that the unique optimal values for $\lambda_1,\dots,\lambda_m$ are $\alpha^{(1)},\dots,\alpha^{(m)}$.

It only remains to show that $\lambda_j=\alpha^{(j)}$ is feasible, namely $\alpha^{(j)}\le\ln\frac{B_j}{B_{j+1}}$ for $j<m$. Applying (21), this is equivalent to showing $e^{-\alpha^{(j)}}\ge\frac{r^{(j)}e^{-\alpha^{(j)}}}{r^{(j+1)}e^{-\alpha^{(j+1)}}}$, or $e^{-\alpha^{(j+1)}}\ge\frac{r^{(j)}}{r^{(j+1)}}$, which follows from (5) since $1-e^{-\alpha^{(1)}}\le 1$.

6. Extending our Techniques

We explain how our techniques can be extended to allow for fractional inventory consumption like in the Adwords problem (Mehta et al. 2007), or offering multiple items like in the online assortment problem (Golrezaei et al. 2014). The extension to continuous price sets in deferred to Appendix E.1.

Consider the following modification of our problem from Section 2: when customer t is offered item i at price j, she *deterministically* pays $p_{t,i}^{(j)}r_i^{(j)}$ and consumes a fractional amount $p_{t,i}^{(j)}\le 1$ of item i's inventory, instead of paying $r_i^{(j)}$ and consuming 1 unit with probability $p_{t,i}^{(j)}$. We assume that $\min_i k_i\to\infty$. This generalizes the Adwords problem under the small bids assumption, by allowing each budget i to be depleted at m_i different rates $r_i^{(1)},\dots,r_i^{(m_i)}$.

For this problem, we use Multi-price Balance, except since we are taking $\min_i k_i\to\infty$, we can deterministically set each $\tilde{\Phi}_i=\Phi_i$. The three claims used to establish Theorem 4 are simpler: Claim 2 now holds deterministically instead of requiring a conditional expectation over Z_t, while Claim 3 also holds deterministically since $\tilde{\Phi}_i$ is always Φ_i. In Theorem 5, condition (16) is now only satisfied under an additional error term ε, since N is no longer a discrete integer. Nonetheless, the rounding error ε approaches 0 as $k_i\to\infty$, so the optimal competitive ratio is still achieved.

For online assortment, we use the term *product* to refer to an (item, price)-combination (i,j). Consider the following modification of our problem from Section 2: upon the arrival of customer t, for any subset (assortment) S of products and $(i,j)\in S$, we are given $p_{t,i}^{(j)}(S)$, the probability that customer t would pick product (i,j) when offered the choice from S. After being given these probabilities, we must offer an assortment S to customer t. This generalizes the original online assortment problem, by allowing each item to have multiple feasible prices. The execution of an algorithm can be encapsulated by the following modification of the LP (3):

$$\max \sum_{t=1}^{T} \sum_{S} x_t(S) \sum_{(i,j)\in S} r_i^{(j)} p_{t,i}^{(j)}(S) \tag{27a}$$

$$\sum_{t=1}^{T} \sum_{S} x_t(S) \sum_{j:(i,j)\in S} p_{t,i}^{(j)}(S) \le k_i \qquad i \in [n] \tag{27b}$$

$$\sum_{S} x_t(S) = 1 \qquad t \in [T] \tag{27c}$$

$$x_t(S) \ge 0 \qquad t \in [T], S \subseteq \{(i,j) : i \in [n], j \in [m_i]\} \tag{27d}$$

Multi-price Balance can be directly applied to this problem, with the change that it offers the *assortment* S maximizing expected pseudorevenue, $\sum_{(i,j)\in S} p_{t,i}^{(j)}(S)\big(\tilde{\Phi}_i(\tilde{L}_i^{(j)}) - \tilde{\Phi}_i(w_i)\big)$, to each customer t. In the analysis, dual constraints (18b) now require $z_t \ge \sum_{(i,j)\in S} p_{t,i}^{(j)}(S)(r_i^{(j)} - y_i)$ for all t and S, which is still implied by the conditions of Theorem 4 so long as the choice probabilities for customers satisfy a mild *substitutability* assumption (see Golrezaei et al. (2014) for details).

7. Simulations on Hotel Data Set of Bodea et al. (2009)

We test our algorithms on the publicly-accessible hotel data set collected by Bodea et al. (2009). Based on the data, we consider a multi-price online assortment problem, as defined in Section 6. In general, we aim to follow the experimental setup of Golrezaei et al. (2014).

7.1. Experimental Setup

We consider Hotel 1 from the data set, which has more transactions than the other four hotels. For each transaction, we use *booking* to refer to the date the transaction occurred, and *occupancy* to refer to the dates the customer will stay in the hotel. We consider occupancies spanning the 5-week period from Sunday, March 11th, 2007 to Sunday, April 15th, 2007. Although the data contains occupancies for a couple of weeks outside this range, such transactions are sparse.

We merge the different rooms into 4 categories: King rooms, Queen rooms, Suites, and Two-double rooms. Rooms under the same category draw from the same inventory. We merge the different fare classes into two: discounted advance-purchase fares and regular rack rates. We use *product* to refer to any of the 8 combinations formed by the 4 room categories and 2 fares.

We estimate a Multinomial Logit (MNL) choice model on these 8 products, for each of 8 customer types. The customer types are based on the booking channel, party size, and VIP status (if any) associated with a transaction. These types capture preference heterogeneity (for example, party sizes greater than 1 tend to prefer Suites and Two-double rooms). The details of our choice estimation are deferred to Appendix F.

We should point out that more sophisticated segmentation and estimation techniques have been employed on this data set (van Ryzin and Vulcano 2014, Newman et al. 2014). Nonetheless, MNL has been reported to perform relatively well (van Ryzin and Vulcano 2014, sec. 5.2). The MNL choice model is convenient for our purposes because under it, both the assortment optimization problem, as well as the choice-based LP (27) with exponentially many variables, can be solved efficiently (Talluri and Van Ryzin 2004, Liu and Van Ryzin 2008, Cheung and Simchi-Levi 2016).

We treat each occupancy date as a separate instance of the problem, for which we define a sequence of arrivals, with one arrival for each transaction which occupies that date. The choice probabilities for each arrival are determined by the customer type associated with the corresponding transaction.[3] The number of days in advance of occupancy that each arrival occurred is also recorded, but this information is only relevant for algorithms which attempt to forecast the remaining number of arrivals based on the remaining length of time.

Before we proceed, we discuss the limitations of our analysis and the data set:

1. In the data set, 55% of the transactions occupy multiple, *consecutive* days. However, we treat such a transaction as a separate arrival in the instances for each of those occupancy dates. While this is a simplifying assumption, the focus of our paper is on the basic allocation problem

[3] The choice realized in that transaction was used for choice model estimation, but is not used in defining the arrival.

without complementarity effects across consecutive days, and our goal in using the data set is to extract an arrival pattern over time.

2. It is not possible to deduce from the data the fixed capacity for each category of room. Nonetheless, we consider a wide range of starting capacities in our tests.

3. Estimating the number of customers who do not make a purchase is a standard challenge in choice modeling, which is exacerbated in this data set by the fact that the arrivals are rather non-stationary. We test various assumptions on the weight of the no-purchase option in the MNL model for each customer type. In general, we assume that this weight is large, which causes the revenue-maximizing assortments to be large, allowing for tension between offering large assortments which maximize immediate revenue, and offering small assortments which regulate inventory consumption (details in Appendix F).

7.2. Instance Definition

An instance consists of a fixed capacity for each room category, corresponding to a specific occupancy date. Each customer interested in that occupancy date arrives in sequence, after which her characteristics (channel, party size, VIP status) are revealed. The problem is to show a *personalized* assortment of (room, fare)-options to each customer. The instances we test are defined below.

- Arrival sequence: 35 possibilities, one for each day in the 5-week occupancy period. We multiply the arrivals by 10 (i.e. instead of a type-1 customer followed by a type-2 customer, we have 10 type-1 customers followed by 10 type-2 customers), being interested in the high-inventory regime. After multiplication, the average number of arrivals per day is 1340, peaking on Sundays and Mondays, although the number and breakdown of customers varies every day.
- Number of products: 8 (room, fare)-combinations, identical for all instances.
- Prices of products: displayed in Table 1, identical for all instances. These prices were determined by taking the average price of that (room, fare)-combination over all transactions.
- Starting inventories: 3 possibilities, defined by the *loading factor*, which is the average number of customers per unit of starting inventory. We use the same loading factors (1.4, 1.6, 1.8) as

Table 1 Details on Room Categories and Fares

Room Category	Low Fare	High Fare	Fraction of Rooms
King	\$307	\$361	52%
Queen	\$304	\$361	15%
Suite	\$384	\$496	13%
Two Double	\$306	\$342	20%

Golrezaei et al. (2014). The breakdown of starting inventory is fixed, based on the relative frequency with which each room type is booked over all transactions (see Table 1).

We test additional synthetic instances, with greater differentiation between high and low fares and a greater range of loading factors, in Subsection 7.5.

7.3. Algorithms Compared

We compare the performances of 9 algorithms on each instance.

First we describe the forecast-independent algorithms we test.

1. **Myopic**: offer each customer the assortment maximizing immediate expected revenue, from the items that have not stocked out.
2. **Conservative**: only offer items at their maximum prices, using the optimal algorithm of Golrezaei et al. (2014) to choose assortments.
3. **Multi-price Balance**: offer each customer t the assortment S maximizing

$$\sum_{(i,j)\in S} p_{t,i}^{(j)}(S)(r_i^{(j)} - \Phi_i(w_i)), \tag{28}$$

where w_i is the fraction of item i sold. Expression (28) is the expected *pseudorevenue* of assortment S. Since we are in the high-inventory regime, for simplicity we have used the fixed value function Φ_i, instead of the random $\tilde{\Phi}_i$, to define the bid price of each item i.

The Myopic and Conservative algorithms represent two extremes, where the former extracts the maximum in expectation from every customer and is optimal as the loading factor approaches 0, while the latter extracts the maximum from every unit of inventory and is optimal as the loading factor approaches ∞. In-between these extremes, our algorithm attempts to balance revenue-per-customer and revenue-per-item, as it chooses items and prices to put in the assortment.

Next we describe the forecasting-based algorithms. These algorithms all estimate the number of each type of customer yet to arrive, and then incorporate this information into the LP (27) to set bid prices. They differ in how they perform the forecasting, and how frequently they update the bid prices by re-solving the LP. Further details about these algorithms, as well as discussion of alternative algorithms, are deferred to Appendix F.1.

4. **One-shot LP**: solve the LP only once, at the start, using the average number of customers of each type to appear on a given day.
5. **LP Resolving**: re-solve the LP every 100 arrivals, using updated forecasts and inventory counts. During each re-solve, the estimated number of remaining customers is updated, taking into account the length of time remaining until occupancy, and the number of customers that have arrived. The estimated type breakdown is fixed, based on the aggregate distribution.
6. **LP Learning**: same as LP Resolving, except the estimated type breakdown is also updated, based on the empirical distribution observed thus far.
7. **LP Clairvoyant**: same as LP Resolving, but given the true number of customers of each type remaining.

Finally, we describe the hybrid algorithms we test. These algorithms combine a forecasting algorithm with Multi-price Balance, based on a parameter $\gamma > 1$. For each customer t, the hybrid algorithm considers the expected pseudorevenue (as defined in (28)) of the assortment S^{fcst} suggested by the forecasting algorithm. If this is at least $\frac{1}{\gamma}$ of the maximum value of (28) over all assortments S, then the hybrid algorithm offers S^{fcst}. Otherwise, the hybrid algorithm offers the assortment suggested by Multi-price Balance, which maximizes (28).

8. **Resolve-1.5**: hybrid algorithm based on LP Resolving and parameter $\gamma = 1.5$.
9. **Learn-1.5**: hybrid algorithm based on LP Learning and parameter $\gamma = 1.5$.

7.4. Results

On every instance, we express the performance of each algorithm as a percentage of the LP upper bound. That is, we take the expected revenue of the algorithm (approximated over 10 runs), and

Table 2 The percentages of optimum achieved by different algorithms. The 3 highest percentages in each row are **bolded**. The 3 lowest standard deviations in each row are *italicized*.

Loading Factor		Forecast-independent			Forecast-dependent				Hybrid	
		Myopic	Conservative	Balance	One-shot	Resolve	Learn	Clairvoyant	Resolve-1.5	Learn-1.5
1.4	Mean	0.974	0.940	**0.976**	0.973	0.962	0.958	0.991	**0.977**	**0.977**
	Stdev	0.023	0.034	*0.013*	*0.016*	0.039	0.041	0.008	*0.018*	0.020
1.6	Mean	0.965	0.960	**0.971**	0.964	0.961	0.963	0.990	**0.977**	**0.978**
	Stdev	0.025	0.036	*0.014*	0.021	0.031	0.030	0.008	*0.008*	*0.010*
1.8	Mean	0.957	**0.972**	0.968	0.808	0.962	0.968	0.990	**0.977**	**0.977**
	Stdev	0.020	0.036	*0.012*	0.100	0.029	0.023	0.009	*0.008*	*0.007*

divide it by the optimal objective value of the LP (27) with the true arrival sequence. In Table 2, we report the mean and standard deviation of each algorithm's percentages over the 35 arrival sequences, for each loading factor.

In general, MULTI-PRICE BALANCE is the most profitable and robust among the forecast-independent algorithms. The forecast-dependent algorithms have much greater fluctuation in their performance for different occupancy days, dependent on how accurate their forecasts were for that day. LP Learning is slightly better than the others, but is most prone to overfitting in its forecasts.

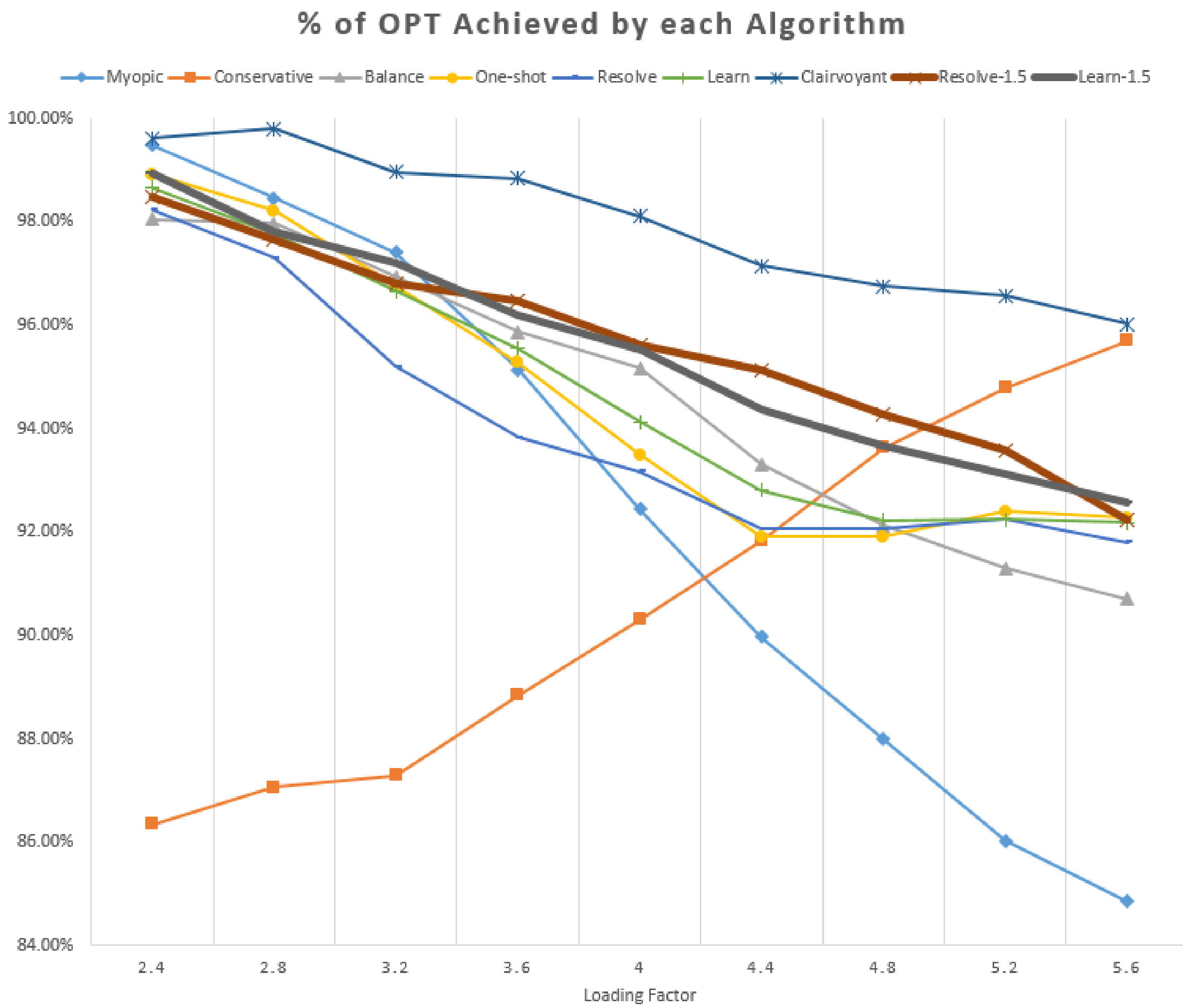
Nonetheless, by combining these algorithms with MULTI-PRICE BALANCE, the hybrid algorithms are able to correct for forecast overconfidence and achieve the best performance overall (aside from the Clairvoyant algorithm, which has a perfect forecast of the future). We find that although the hybrid algorithm only changes a small fraction ($\approx 5\%$) of the forecasting algorithm's decisions, this drastically improves the profitability and robustness.

7.5. Results under Greater Fare Differentiation

The instances tested in Subsection 7.4 were "easy" in that there was not so much difference between selling rooms at their low or high fares. In this subsection, we synthetically modify the higher fare for each room category to be *twice* its lower fare. We also increase the utility of the no-purchase option in the MNL model for each customer type (see Appendix F), to maintain the tension between lower fares which maximize expected revenue, and higher fares which limit inventory consumption.

Furthermore, we test the complete range of loading factors, including both the extreme where the Myopic algorithm is optimal, and the extreme where the Conservative algorithm is optimal.

Figure 3 Algorithm performances in the setting with greater fare differentiation. The lines corresponding to the two hybrid algorithms, which perform the best overall, have been **bolded**.



In Figure 3, we plot the average percentages of optimum attained by each algorithm over the 35 arrival sequences, for each loading factor.

The conclusion again is that the two hybrid algorithms, which use forecasts but continuously reference our forecast-independent value functions, are the most profitable and robust, with Multi-price Balance coming third. However, it is important to note that our methodology is only relevant in-between the extremes, where there is a non-trivial tradeoff between immediate revenue and future inventory. If a firm knew that its inventory constraints tend to not be binding, then it would be better off using the Myopic algorithm. Similarly, if a firm knew that it has too much demand for its inventory, then it would be better off always offering the maximum prices, using the Conservative algorithm.

8. Conclusion

Competitive analysis is a well-established methodology in sequential decision-making problems, providing a baseline decision in the absence of a reliable forecast of the future. Previously, optimal algorithms have been derived for allocating a single unreplenishable resource to customers from different fare classes, or allocating multiple resources which each have a fixed price. In this paper, we derive optimal allocation algorithms which jointly consider the tradeoffs between different fares and different resources. This broadly expands the applicability of competitive analysis, in areas such as online advertising, matching markets, personalized e-commerce, and appointment scheduling.

Acknowledgments

The authors would like to thank Rong Jin of Alibaba for pointing out a technical error in an earlier version of the appendix. The authors would also like to thank Ozan Candogan for asking a question which led to the simpler bound presented in Corollary 2.

References

Aggarwal G, Goel G, Karande C, Mehta A (2011) Online vertex-weighted bipartite matching and single-bid budgeted allocations. *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms*, 1253–1264 (Society for Industrial and Applied Mathematics).

Agrawal S, Wang Z, Ye Y (2014) A dynamic near-optimal algorithm for online linear programming. *Operations Research* 62(4):876–890.

Araman VF, Caldentey R (2011) Revenue management with incomplete demand information. *Wiley Encyclopedia of Operations Research and Management Science* .

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

Bodea T, Ferguson M, Garrow L (2009) Data setchoice-based revenue management: Data from a major hotel chain. *Manufacturing & Service Operations Management* 11(2):356–361.

Borodin A, El-Yaniv R (2005) *Online computation and competitive analysis* (cambridge university press).

Buchbinder N, Jain K, Naor JS (2007) Online primal-dual algorithms for maximizing ad-auctions revenue. *European Symposium on Algorithms*, 253–264 (Springer).

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

Chen X, Ma W, Simchi-Levi D, Xin L (2016) Dynamic recommendation at checkout under inventory constraint. *manuscript on SSRN* .

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

Cheung WC, Simchi-Levi D (2016) Efficiency and performance guarantees for choice-based network revenue management problems with flexible products .

Ciocan DF, Farias V (2012) Model predictive control for dynamic resource allocation. *Mathematics of Operations Research* 37(3):501–525.

Devanur NR, Jain K, Kleinberg RD (2013) Randomized primal-dual analysis of ranking for online bipartite matching. *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, 101–107 (SIAM).

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

Feldman J, Korula N, Mirrokni V, Muthukrishnan S, Pál M (2009) Online ad assignment with free disposal. *International Workshop on Internet and Network Economics*, 374–385 (Springer).

Ferreira KJ, Simchi-Levi D, Wang H (2016) Online network revenue management using thompson sampling. *manuscript on SSRN* .

Gallego G, Li A, Truong VA, Wang X (2015) Online resource allocation with customer choice. *arXiv preprint arXiv:1511.01837* .

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

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

Kalyanasundaram B, Pruhs KR (2000) An optimal deterministic algorithm for online b-matching. *Theoretical Computer Science* 233(1):319–325.

Karp RM, Vazirani UV, Vazirani VV (1990) An optimal algorithm for on-line bipartite matching. *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, 352–358 (ACM).

Kesselheim T, Radke K, Tönnis A, Vöcking B (2013) An optimal online algorithm for weighted bipartite matching and extensions to combinatorial auctions. *European Symposium on Algorithms*, 589–600 (Springer).

Lan Y, Gao H, Ball MO, Karaesmen I (2008) Revenue management with limited demand information. *Management Science* 54(9):1594–1609.

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

Manshadi VH, Gharan SO, Saberi A (2012) Online stochastic matching: Online actions based on offline statistics. *Mathematics of Operations Research* 37(4):559–573.

Mehta A (2013) Online matching and ad allocation. *Foundations and Trends® in Theoretical Computer Science* 8(4):265–368.

Mehta A, Panigrahi D (2012) Online matching with stochastic rewards. *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, 728–737 (IEEE).

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

Newman JP, Ferguson ME, Garrow LA, Jacobs TL (2014) Estimation of choice-based models using sales data from a single firm. *Manufacturing & Service Operations Management* 16(2):184–197.

Reiman MI, Wang Q (2008) An asymptotically optimal policy for a quantity-based network revenue management problem. *Mathematics of Operations Research* 33(2):257–282.

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

Talluri KT, Van Ryzin GJ (2006) *The theory and practice of revenue management*, volume 68 (Springer Science & Business Media).

Van Ryzin G, McGill J (2000) Revenue management without forecasting or optimization: An adaptive algorithm for determining airline seat protection levels. *Management Science* 46(6):760–775.

van Ryzin G, Vulcano G (2014) A market discovery algorithm to estimate a general class of nonparametric choice models. *Management Science* 61(2):281–300.

Wang X, Truong V, Bank D (2015) Online advance admission scheduling for services, with customer preferences. *Working paper.*

Yao ACC (1977) Probabilistic computations: Toward a unified measure of complexity. *Foundations of Computer Science, 1977., 18th Annual Symposium on*, 222–227 (IEEE).

Zhang D, Cooper WL (2005) Revenue management for parallel flights with customer-choice behavior. *Operations Research* 53(3):415–431.

Zhang H, Shi C, Qin C, Hua C (2016) Stochastic regret minimization for revenue management problems with nonstationary demands. *Naval Research Logistics (NRL)* 63(6):433–448.

Appendix A: Deferred Proofs from Section 2

Proof of Lemma 1. Fix any adaptive algorithm (which knows the arrival information, but not the realizations of the customers' purchase decisions, at the start) and consider its execution on instance $\mathcal{I}$. Let $X_{t,i}^{(j)}$ be the indicator random variable (0 or 1) for the algorithm offering item i at price j to customer t, and $P_{t,i}^{(j)}$ be the indicator random variable for customer t accepting when item i is offered to her at price j. On a given run, the constraints $\sum_{t=1}^{T}\sum_{j=1}^{m_i} P_{t,i}^{(j)} X_{t,i}^{(j)} \leq k_i$ and $\sum_{i=1}^{n}\sum_{j=1}^{m_i} X_{t,i}^{(j)} \leq 1$ are satisfied. Therefore, they are still satisfied after taking an expectation over all runs, and furthermore we can use independence to show that $\mathbb{E}[P_{t,i}^{(j)} X_{t,i}^{(j)}] = \mathbb{E}[P_{t,i}^{(j)}] \cdot \mathbb{E}[X_{t,i}^{(j)}] = p_{t,i}^{(j)} x_{t,i}^{(j)}$. Therefore, the algorithm must satisfy constraints (3b) and (3c) of the LP. Since its revenue on a given run is $\sum_{t=1}^{T}\sum_{i=1}^{n}\sum_{j=1}^{m_i} P_{t,i}^{(j)} r^{(j)} X_{t,i}^{(j)}$, taking an expectation over it yields (3a), completing the proof. $\square$

Proof of Proposition 1. The statement for $\sigma^{(j)}, \ldots, \sigma^{(j)}$ is immediate from the fact that the explicit value of $\sigma^{(j)}$ is $(1 - \frac{r^{(j-1)}}{r^{(j)}})(1 + \sum_{j'=2}^{m}(1 - \frac{r^{(j'-1)}}{r^{(j')}}))^{-1}$, for all $j \in [m]$. To prove the statement for $\alpha^{(1)}, \ldots, \alpha^{(m)}$, we show that the solution to the system of n equations formed by (5) and $\alpha^{(1)} + \ldots \alpha^{(m)} = 1$ is unique and strictly positive.

Let $\gamma^{(j)} = e^{-\alpha^{(j)}}$ for all j. Then the constraint $\alpha^{(1)} + \ldots \alpha^{(m)} = 1$ can be rewritten as $\prod_{j=1}^{m} \gamma^{(j)} = \frac{1}{e}$. Furthermore, we derive from (5) that for all $j > 1$, $\gamma^{(j)} = (1 - \frac{r^{(j-1)}}{r^{(j)}})\gamma^{(1)} + \frac{r^{(j-1)}}{r^{(j)}}$. Therefore,

$$\gamma^{(1)} \cdot \prod_{j=2}^{m} \Big((1 - \frac{r^{(j-1)}}{r^{(j)}})\gamma^{(1)} + \frac{r^{(j-1)}}{r^{(j)}} \Big) = \frac{1}{e}. \tag{29}$$

Consider the LHS of (29) as a function of $\gamma^{(1)}$ on $[\frac{1}{e}, 1]$. This is a continuous, strictly increasing function which is at most $\frac{1}{e}$ when $\gamma^{(1)} = \frac{1}{e}$ and 1 when $\gamma^{(1)} = 1$. Therefore, there is a unique solution with $\gamma^{(1)} \in [\frac{1}{e}, 1)$, and the resulting value of $\alpha^{(1)}$ is positive. For $j > 1$, since $\gamma^{(j)}$ can also be written as $\gamma^{(1)} + \frac{r^{(j-1)}}{r^{(j)}}(1 - \gamma^{(1)})$, it can be seen that $\gamma^{(j)} \in [\frac{1}{e}, 1)$, hence the unique value for $\alpha^{(j)}$ is positive as well.

Finally, to see that $\alpha^{(1)} \geq \frac{1}{m}$, observe that when $\gamma^{(1)} = e^{-1/m}$, the LHS of (29) is at least $(e^{-1/m})^m = \frac{1}{e}$. Therefore, $\gamma^{(1)}$ is at most $e^{-1/m}$ which implies that $\alpha^{(1)} \geq \frac{1}{m}$. $\square$

Proof of Proposition 2. For the first inequality in (10), observe that $f(x) = \frac{x}{1-e^{-x}}$ is a strictly increasing function on $[0, 1]$. Since $\sigma^{(1)} \in (0, 1)$, $\frac{\sigma^{(1)}}{1-e^{-\sigma^{(1)}}} < \frac{1}{1-\frac{1}{e}}$, which is the desired result.

For the second inequality in (10), we show $\alpha^{(1)} > \sigma^{(1)}$, by showing that for all $j = 2, \ldots, m$, $\alpha^{(j)}$ is a smaller multiple of $\alpha^{(1)}$ than $\sigma^{(j)}$ is of $\sigma^{(1)}$. This suffices because both the fractions $\alpha^{(1)}, \ldots, \alpha^{(m)}$ and $\sigma^{(1)}, \ldots, \sigma^{(m)}$ must sum to 1. For a given j, we must establish that $\frac{\alpha^{(j)}}{\alpha^{(1)}} < \frac{\sigma^{(j)}}{\sigma^{(1)}}$. By definition, $\frac{\sigma^{(j)}}{\sigma^{(1)}} = 1 - \frac{r^{(j-1)}}{r^{(j)}} = \frac{1-e^{-\alpha^{(j)}}}{1-e^{-\alpha^{(1)}}}$. Therefore, is suffices to show that $\frac{\alpha^{(j)}}{\alpha^{(1)}} < \frac{1-e^{-\alpha^{(j)}}}{1-e^{-\alpha^{(1)}}}$, or $\frac{\alpha^{(j)}}{1-e^{-\alpha^{(j)}}} < \frac{\alpha^{(1)}}{1-e^{-\alpha^{(1)}}}$. This follows from the fact that the function $f(x) = \frac{x}{1-e^{-x}}$ is strictly increasing.

To prove (11), note that $\sigma^{(1)} = (1+\sum_{j=2}^{m}(1-\frac{r^{(j-1)}}{r^{(j)}}))^{-1}$, while $1+\ln\frac{r^{(m)}}{r^{(1)}} = 1+\sum_{j=2}^{m}\ln\frac{r^{(j)}}{r^{(j-1)}}$. Therefore, it suffices to show that for any $j=2,\ldots,m$, $\ln\frac{r^{(j)}}{r^{(j-1)}} > 1-\frac{r^{(j-1)}}{r^{(j)}}$. Letting $x=\ln\frac{r^{(j-1)}}{r^{(j)}} < 0$, the desired inequality becomes $-x > 1-e^{x}$, which is immediate.

For (12), we would like to prove that $\alpha < \alpha^{(1)}$. Note that $\alpha^{(1)}$ is the unique solution to

$$\alpha^{(1)} + \sum_{j=2}^{m}\Big[-\ln\big(1-(1-e^{-\alpha^{(1)}})(1-\frac{r^{(j-1)}}{r^{(j)}})\big)\Big] = 1, \tag{30}$$

while α is the unique solution to

$$\alpha + \sum_{j=2}^{m}(1-e^{-\alpha})\ln\frac{r^{(j)}}{r^{(j-1)}} = 1. \tag{31}$$

The LHS of (30), as a function of $\alpha^{(1)}$, is increasing over $(0,1)$; the same can be said about the LHS of (31) as a function of α. Therefore, it suffices to show that if $\alpha^{(1)}=\alpha=x$, then the LHS of (30) is strictly less than the LHS of (31), for all $x\in(0,1)$.

Let $F=1-e^{-x}$ and consider any $j>1$. Let $s=\frac{r^{(j-1)}}{r^{(j)}}\in(0,1)$. It suffices to show that $-\ln(1-F(1-s)) < F\cdot\ln\frac{1}{s}$, which can be rearranged as $\frac{1-s^F}{1-s} > F$. For the final inequality, note that $f(s)=s^F$ is a strictly concave function on $(0,1)$, since $F\in(0,1)$. Therefore, $\frac{1-s^F}{1-s} > F$, because the LHS is the slope of the secant line through (s,s^F) and $(1,1)$, while the RHS is the slope of the tangent line through $(1,1)$. $\square$

Appendix B: Supplement to Section 3

The first subsection contains the deferred proofs from Section 3. In the second subsection, we explain how to optimize the randomized procedure for generating a single value function. In the third subsection, we put together the proof of Theorem 1.

The following inequality will be useful throughout the paper. For all $j=2,\ldots,m$, (5) says that $1-e^{-\alpha^{(j)}} \le 1-\frac{r^{(j-1)}}{r^{(j)}}$, where we have used the fact that $1-e^{-\alpha^{(1)}}\le 1$. Therefore, for all $j=2,\ldots,m$, we can derive that

$$\frac{r^{(j-1)}}{r^{(j)}} \le e^{-\alpha^{(j)}}. \tag{32}$$

B.1. Deferred Proofs

Proof of Theorem 4. Define $N_{t,i}$ to be the algorithm's value for N_i at the *end* of time t ($N_{0,i}$ is understood to be 0), for all $t\in[T]$ and $i\in[n]$. For all $t\in[T]$, define $R_t = r_{i_t^*}^{(j_t^*)}$ and $Z_t = \tilde{\Phi}_{i_t^*}(\tilde{L}_{i_t^*}^{(j_t^*)}) - \tilde{\Phi}_{i_t^*}(N_{i_t^*}/k_{i_t^*})$ if a sale was made during time t; define $R_t=Z_t=0$ otherwise.

Consider the solution to the dual LP (18) formed by setting $y_i=\mathbb{E}[\tilde{\Phi}_i(\frac{N_{T,i}}{k_i})]$ for all $i\in[n]$, and $z_t=\mathbb{E}[Z_t]$ for all $t\in[T]$. We claim that this solution is feasible. The non-negativity constraint (18c) can be verified directly from the definitions.

Now, consider constraint (18b) for a fixed $t \in [T], i \in [n], j \in [m_i]$. Given the initializations of $\tilde{L}_i^{(1)}, \ldots, \tilde{L}_i^{(m_i)}, \tilde{\Phi}_i$ and the value of $N_{t-1,i}$, the algorithm will always make a decision during time t which earns pseudorevenue whose conditional expectation is at least $p_{t,i}^{(j)}(\tilde{\Phi}_i(\tilde{L}_i^{(j)}) - \tilde{\Phi}_i(\frac{N_{t-1,i}}{k_i}))$, by definition (15). Formally,

$$\mathbb{E}[Z_t | \tilde{L}_i^{(1)}, \ldots, \tilde{L}_i^{(m_i)}, \tilde{\Phi}_i, N_{t-1,i}] \geq p_{t,i}^{(j)}(\tilde{\Phi}_i(\tilde{L}_i^{(j)}) - \tilde{\Phi}_i(\frac{N_{t-1,i}}{k_i})),$$

for all values of $\tilde{L}_i^{(1)}, \ldots, \tilde{L}_i^{(m_i)}, \tilde{\Phi}_i, N_{t-1,i}$. By the tower property of conditional expectation, $z_t = \mathbb{E}[Z_t] \geq \mathbb{E}[p_{t,i}^{(j)}(\tilde{\Phi}_i(\tilde{L}_i^{(j)}) - \tilde{\Phi}_i(\frac{N_{t-1,i}}{k_i}))]$. Meanwhile, y_i has been set to $\mathbb{E}[\tilde{\Phi}_i(\frac{N_{T,i}}{k_i})]$. Since $N_{T,i} \geq N_{t-1,i}$ and $\tilde{\Phi}_i$ is increasing, $y_i \geq \mathbb{E}[\tilde{\Phi}_i(\frac{N_{t-1,i}}{k_i})]$. Therefore, the LHS of (18b), $p_{t,i}^{(j)} y_i + z_t$, is at least $\mathbb{E}[p_{t,i}^{(j)}(\tilde{\Phi}_i(\tilde{L}_i^{(j)}))]$. By (17), this is at least $r_i^{(j)}$, completing the proof of feasibility.

Applying weak duality, we obtain

$$\begin{aligned}
\mathsf{OPT}(\mathcal{I}) &\leq \sum_{i=1}^{n} k_i \mathbb{E}[\tilde{\Phi}_i(\frac{N_{T,i}}{k_i})] + \sum_{t=1}^{T} \mathbb{E}[Z_t] \\
&= \sum_{i=1}^{n} k_i \mathbb{E}\Big[\sum_{t=1}^{T}(\tilde{\Phi}_i(\frac{N_{t,i}}{k_i}) - \tilde{\Phi}_i(\frac{N_{t-1,i}}{k_i}))\Big] + \sum_{t=1}^{T} \mathbb{E}[Z_t] \\
&= \sum_{t=1}^{T} \mathbb{E}\Big[\sum_{i=1}^{n} k_i(\tilde{\Phi}_i(\frac{N_{t,i}}{k_i}) - \tilde{\Phi}_i(\frac{N_{t-1,i}}{k_i})) + Z_t\Big]. \qquad (33)
\end{aligned}$$

We now analyze the term inside the expectation,

$$\sum_{i=1}^{n} k_i(\tilde{\Phi}_i(\frac{N_{t,i}}{k_i}) - \tilde{\Phi}_i(\frac{N_{t-1,i}}{k_i})) + Z_t, \qquad (34)$$

for every $t \in [T]$. We would like to argue that it is at most $\frac{R_t}{F}$, on every sample path.

There are two cases. If an item $i = i_t^*$ was sold at price $j = j_t^*$ during time t, then (34) equals

$$k_i(\tilde{\Phi}_i(\frac{N_{t-1,i}+1}{k_i}) - \tilde{\Phi}_i(\frac{N_{t-1,i}}{k_i})) + \tilde{\Phi}_i(\tilde{L}_i^{(j)}) - \tilde{\Phi}_i(\frac{N_{t-1,i}}{k_i}). \qquad (35)$$

Indeed, $N_{t,i} = N_{t-1,i} + 1$, $N_{t,i} = N_{t-1,i}$ for all $i \neq i$, and $Z_t = \tilde{\Phi}_i(\tilde{L}_i^{(j)}) - \tilde{\Phi}_i(\frac{N_{t-1,i}}{k_i})$ by definition. Furthermore, since Z_t is positive, $N_{t-1,i}$ must by less than $\tilde{L}_i^{(j)}k$. Therefore, we can invoke (16) to get that (35) is at most $r_i^{(j)}/F$, which is equal to $\frac{R_t}{F}$ by definition. In the other case, if no item was sold during time t, then (35) is 0, while $R_t = 0$ too, so (35) is still at most $\frac{R_t}{F}$.

Substituting back into (33), we conclude that $\mathsf{OPT}(\mathcal{I}) \leq \sum_{t=1}^{T} \mathbb{E}[\frac{R_t}{F}]$, which is equal to $\frac{1}{F}\mathbb{E}[\mathsf{ALG}(\mathcal{I})]$ by definition. This completes the proof of Algorithm 1 having a competitive ratio at least F. $\square$

Proof of Theorem 5. First we prove the following two properties implied by the comonotonic randomized rounding procedure for $\tilde{L}^{(0)}, \ldots, \tilde{L}^{(m)}$ from Definition 2:

$$\mathbb{E}[\tilde{L}^{(j)}] = L^{(j)}, \qquad j = 0, \ldots, m; \qquad (36)$$

$$|(\tilde{L}^{(j)} - \tilde{L}^{(j')}) - (L^{(j)} - L^{(j')})| \leq \frac{1}{k}, \qquad 1 \leq j' < j \leq m. \qquad (37)$$

For (36), note that $\mathbb{E}[\tilde{L}^{(j)}] = \frac{\lfloor L^{(j)}k \rfloor + 1}{k}(L^{(j)}k - \lfloor L^{(j)}k \rfloor) + \frac{\lfloor L^{(j)}k \rfloor}{k}(1 - (L^{(j)}k - \lfloor L^{(j)}k \rfloor)) = \frac{1}{k}(L^{(j)}k - \lfloor L^{(j)}k \rfloor) + \frac{\lfloor L^{(j)}k \rfloor}{k} = L^{(j)}$.

For (37), note that $|(\tilde{L}^{(j)} - \tilde{L}^{(j')}) - (L^{(j)} - L^{(j')})| = |(\tilde{L}^{(j)} - L^{(j)}) - (\tilde{L}^{(j')} - L^{(j')})|$. We will prove that $(\tilde{L}^{(j)} - L^{(j)}) - (\tilde{L}^{(j')} - L^{(j')}) \le \frac{1}{k}$; the inequality that $(\tilde{L}^{(j)} - L^{(j)}) - (\tilde{L}^{(j')} - L^{(j')}) \ge -\frac{1}{k}$ follows by symmetry. The maximum value of $k\tilde{L}^{(j)}$ is $\lfloor kL^{(j)} \rfloor + 1$ while the minimum value of $k\tilde{L}^{(j')}$ is $\lfloor kL^{(j')} \rfloor$, hence the result is immediate unless $(\lfloor kL^{(j)} \rfloor + 1) - kL^{(j)} + kL^{(j')} - \lfloor kL^{(j')} \rfloor > 1$, i.e. $kL^{(j')} - \lfloor kL^{(j')} \rfloor > kL^{(j)} - \lfloor kL^{(j)} \rfloor$. However, in this case, if $k\tilde{L}^{(j)} = \lfloor kL^{(j)} \rfloor + 1$, then $W < kL^{(j)} - \lfloor kL^{(j)} \rfloor < kL^{(j')} - \lfloor kL^{(j')} \rfloor$ and hence $\tilde{L}^{(j')}$ is rounded up as well. Similarly, if $\tilde{L}^{(j')}$ is rounded down, then $\tilde{L}^{(j)}$ must be rounded down as well. If $\tilde{L}^{(j)}$ and $\tilde{L}^{(j')}$ are rounded in the same direction, then (iii) holds.

Having established (36) and (37), we now show that (16)–(17) are satisfied.

First we prove (17), the claim that $\mathbb{E}[\tilde{\Phi}(\tilde{L}^{(j)})] \ge r^{(j)}$, inductively. Clearly $\mathbb{E}[\Phi(\tilde{L}^{(0)})] \ge r^{(0)} = 0$. Now consider $j \in [m]$ and suppose we have established (17) for the $j-1$ case. We can compare expression (19) with $q = \tilde{L}^{(j)}$ and $q = \tilde{L}^{(j-1)}$ to obtain $\tilde{\Phi}(\tilde{L}^{(j)}) = \tilde{\Phi}(\tilde{L}^{(j)}) + (r^{(j)} - r^{(j-1)})\frac{\exp(\tilde{L}^{(j)} - \tilde{L}^{(j-1)}) - 1}{\exp(\alpha^{(j)}) - 1}$. Therefore,

$$\begin{aligned}
\mathbb{E}[\tilde{\Phi}(\tilde{L}^{(j)})] &\ge r^{(j-1)} + (r^{(j)} - r^{(j-1)})\frac{\mathbb{E}[\exp(\tilde{L}^{(j)} - \tilde{L}^{(j-1)})] - 1}{\exp(\alpha^{(j)}) - 1} \\
&\ge r^{(j-1)} + (r^{(j)} - r^{(j-1)})\frac{\exp(\mathbb{E}[\tilde{L}^{(j)} - \tilde{L}^{(j-1)}]) - 1}{\exp(\alpha^{(j)}) - 1} \\
&= r^{(j-1)} + (r^{(j)} - r^{(j-1)})\frac{\exp(\alpha^{(j)}) - 1}{\exp(\alpha^{(j)}) - 1}
\end{aligned}$$

where the first inequality uses the induction hypothesis, and the second inequality uses Jensen's inequality (the exponential function exp is convex). The equality follows from (36) and the definition that $\alpha^{(j)} = L^{(j)} - L^{(j-1)}$, completing the induction.

Now we prove (16) for an arbitrary $j \in [m]$ and $N \in \{0, \dots, \tilde{L}^{(j)}k - 1\}$. Let $q = \frac{N}{k}$ and $\ell = \tilde{\ell}(q)$. Note that $1 \le \ell \le j$, and $\tilde{L}^{(\ell-1)} \le q < \tilde{L}^{(\ell)}$. Substituting $q = \frac{N}{k}$ into the LHS of (16), we get $k\big(\tilde{\Phi}(q + \frac{1}{k}) - \tilde{\Phi}(q)\big) + \tilde{\Phi}(\tilde{L}^{(j)}) - \tilde{\Phi}(q)$. Adding and subtracting $\tilde{\Phi}(\tilde{L}^{(\ell)})$ and rearranging, we get

$$k\big(\tilde{\Phi}(q + \frac{1}{k}) - \tilde{\Phi}(q)\big) + \tilde{\Phi}(\tilde{L}^{(\ell)}) - \tilde{\Phi}(q) + \tilde{\Phi}(\tilde{L}^{(j)}) - \tilde{\Phi}(\tilde{L}^{(\ell)}). \tag{38}$$

The following upper bound can be derived for expression (38):

$$\begin{aligned}
&k\big(\tilde{\Phi}(q + \frac{1}{k}) - \tilde{\Phi}(q)\big) + \tilde{\Phi}(\tilde{L}^{(\ell)}) - \tilde{\Phi}(q) + \tilde{\Phi}(\tilde{L}^{(j)}) - \tilde{\Phi}(\tilde{L}^{(\ell)}) \\
=&(r^{(\ell)} - r^{(\ell-1)})\frac{e^{q+1/k-\tilde{L}^{(\ell-1)}}(k - (k+1)e^{-1/k}) + e^{\tilde{L}^{(\ell)} - \tilde{L}^{(\ell-1)}}}{e^{\alpha^{(\ell)}} - 1} + \sum_{\ell'=\ell+1}^{j}(r^{(\ell')} - r^{(\ell'-1)})\frac{e^{\tilde{L}^{(\ell')} - \tilde{L}^{(\ell'-1)}} - 1}{e^{\alpha^{(\ell')}} - 1} \\
\le&(r^{(\ell)} - r^{(\ell-1)})\frac{e^{\tilde{L}^{(\ell)} - \tilde{L}^{(\ell-1)}}(k - (k+1)e^{-1/k}) + e^{\tilde{L}^{(\ell)} - \tilde{L}^{(\ell-1)}}}{e^{\alpha^{(\ell)}} - 1} + \sum_{\ell'=\ell+1}^{j}(r^{(\ell')} - r^{(\ell'-1)})\frac{e^{\tilde{L}^{(\ell')} - \tilde{L}^{(\ell'-1)}} - 1}{e^{\alpha^{(\ell')}} - 1} \\
=&(r^{(\ell)} - r^{(\ell-1)})\frac{e^{\tilde{L}^{(\ell)} - \tilde{L}^{(\ell-1)} - \alpha^{(\ell)}}(1+k)(1 - e^{-1/k})}{1 - e^{-\alpha^{(\ell)}}} + \sum_{\ell'=\ell+1}^{j}(r^{(\ell')} - r^{(\ell'-1)})\frac{e^{\tilde{L}^{(\ell')} - \tilde{L}^{(\ell'-1)} - \alpha^{(\ell')}} - e^{-\alpha^{(\ell')}}}{1 - e^{-\alpha^{(\ell')}}}.
\end{aligned} \tag{39}$$

The inequality holds because $k-(1+k)e^{-1/k}>0$ for all $k\in\mathbb{N}$, and q is at most $\tilde{L}^{(\ell)}-1/k$.

It suffices to show that expression (39) is bounded from above by

$$r^{(j)}\frac{(1+k)(e^{1/k}-1)}{1-e^{-\alpha^{(1)}}}. \tag{40}$$

To assist in this task, we would like to establish the following for all $\ell'=\ell+1,\dots,j$ and $\ell''\in\{\ell,\dots,\ell'-1\}$:

$$\begin{aligned}&(r^{(\ell'-1)}-r^{(\ell'-2)})\frac{e^{\tilde{L}^{(\ell'-1)}-\tilde{L}^{(\ell''-1)}-L^{(\ell'-1)}+L^{(\ell''-1)}}(1+k)(1-e^{-1/k})}{1-e^{-\alpha^{(\ell'-1)}}}+(r^{(\ell')}-r^{(\ell'-1)})\frac{e^{\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}-\alpha^{(\ell')}}-e^{-\alpha^{(\ell')}}}{1-e^{-\alpha^{(\ell')}}}\\ \le&(r^{(\ell')}-r^{(\ell'-1)})\frac{e^{\max\{\tilde{L}^{(\ell')}-\tilde{L}^{(\ell''-1)}-L^{(\ell')}+L^{(\ell''-1)},\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}-L^{(\ell')}+L^{(\ell'-1)}\}}(1+k)(1-e^{-1/k})}{1-e^{-\alpha^{(\ell')}}}.\end{aligned} \tag{41}$$

But $\frac{r^{(\ell'-1)}-r^{(\ell'-2)}}{1-e^{-\alpha^{(\ell'-1)}}}=\frac{r^{(\ell')}-r^{(\ell'-1)}}{1-e^{-\alpha^{(\ell')}}}\cdot\frac{r^{(\ell'-1)}}{r^{(\ell')}}$ due to the definition of α in (5), and $\frac{r^{(\ell'-1)}}{r^{(\ell')}}\le e^{-\alpha^{(\ell')}}$ due to (32). Substituting back into inequality (41), it suffices to prove

$$\begin{aligned}&e^{\tilde{L}^{(\ell'-1)}-\tilde{L}^{(\ell''-1)}-L^{(\ell')}+L^{(\ell''-1)}}(1+k)(1-e^{-1/k})+e^{\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}-\alpha^{(\ell')}}-e^{-\alpha^{(\ell')}}\\ \le&e^{\max\{\tilde{L}^{(\ell')}-\tilde{L}^{(\ell''-1)}-L^{(\ell')}+L^{(\ell''-1)},\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}-L^{(\ell')}+L^{(\ell'-1)}\}}(1+k)(1-e^{-1/k})\end{aligned}$$

where we have used Definition 1 to rewrite the first exponent. Now,

$$e^{\tilde{L}^{(\ell'-1)}-\tilde{L}^{(\ell''-1)}-L^{(\ell')}+L^{(\ell''-1)}}(k-(1+k)e^{-1/k})\le e^{\tilde{L}^{(\ell')}-\tilde{L}^{(\ell''-1)}-L^{(\ell')}+L^{(\ell''-1)}}(k-(1+k)1-e^{-1/k}),$$

since $k-(1+k)e^{-1/k}>0$ and $\tilde{L}^{(\ell'-1)}\le\tilde{L}^{(\ell')}$. Thus it remains to prove that

$$e^{\tilde{L}^{(\ell'-1)}-\tilde{L}^{(\ell''-1)}-L^{(\ell')}+L^{(\ell''-1)}}+e^{\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}-\alpha^{(\ell')}}-e^{-\alpha^{(\ell')}}\le e^{\max\{\tilde{L}^{(\ell')}-\tilde{L}^{(\ell''-1)}-L^{(\ell')}+L^{(\ell''-1)},\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}-L^{(\ell')}+L^{(\ell'-1)}\}}. \tag{42}$$

We consider two cases. First suppose $\tilde{L}^{(\ell')}-\tilde{L}^{(\ell''-1)}-L^{(\ell')}+L^{(\ell''-1)}\le\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}-L^{(\ell')}+L^{(\ell'-1)}$, i.e. $\tilde{L}^{(\ell'-1)}-\tilde{L}^{(\ell''-1)}\le L^{(\ell'-1)}-L^{(\ell''-1)}$. Then the LHS of (42) equals $e^{-\alpha^{(\ell')}}+e^{\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}-\alpha^{(\ell')}}-e^{-\alpha^{(\ell')}}=e^{\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}-\alpha^{(\ell')}}$, which equals the RHS of (42) by the assumption that $\tilde{L}^{(\ell'-1)}-\tilde{L}^{(\ell''-1)}\le L^{(\ell'-1)}-L^{(\ell''-1)}$. In the second case, suppose $\tilde{L}^{(\ell')}-\tilde{L}^{(\ell''-1)}-L^{(\ell')}+L^{(\ell''-1)}>\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}-L^{(\ell')}+L^{(\ell'-1)}$, i.e. $\tilde{L}^{(\ell'-1)}-\tilde{L}^{(\ell''-1)}>L^{(\ell'-1)}-L^{(\ell''-1)}$. Then inequality (42) can be rearranged as

$$e^{-\alpha^{(\ell')}}(e^{\tilde{L}^{(\ell'-1)}-\tilde{L}^{(\ell''-1)}-L^{(\ell'-1)}+L^{(\ell''-1)}}-1)(e^{\tilde{L}^{(\ell')}-\tilde{L}^{(\ell'-1)}}-1)\ge 0.$$

The first bracket is positive by the assumption that $\tilde{L}^{(\ell'-1)}-\tilde{L}^{(\ell''-1)}>L^{(\ell'-1)}-L^{(\ell''-1)}$ and the second bracket is non-negative since $\tilde{L}^{(\ell'-1)}\le\tilde{L}^{(\ell')}$. This finishes the proof of (42), and hence (41).

Equipped with (41), we return the task of proving that expression (39) is at most expression (40). If we inductively apply inequality (41) to expression (39) for $\ell' = \ell+1, \dots, j$ (when $\ell' = \ell+1$, $\ell'' = \ell$; when $\ell' = \ell+2$, $\ell'' = \ell$ if we arrived at case two during iteration $\ell+1$ and $\ell'' = \ell+1$ otherwise,...), we conclude that expression (39) is bounded from above by

$$(r^{(j)} - r^{(j-1)}) \frac{e^{\tilde{L}^{(j)} - \tilde{L}^{(\ell''-1)} - L^{(j)} + L^{(\ell''-1)}}(1+k)(1-e^{-1/k})}{1 - e^{-\alpha^{(j)}}}$$

for some $\ell'' \in \{\ell, \dots, j\}$. The fact that $1 - e^{-\alpha^{(1)}} = \frac{r^{(j)}}{r^{(j)} - r^{(j-1)}}(1 - e^{-\alpha^{(j)}})$, due to (5), and the fact that $(\tilde{L}^{(j)} - \tilde{L}^{(\ell''-1)}) - (L^{(j)} - L^{(\ell''-1)}) \le 1/k$, due to (37), complete the proof of expression (39) being at most expression (40), and thus the proof of Theorem 5 for general m.

Finally, when $m = 1$, $\alpha^{(1)} = 1$. In the above proof, since j and ℓ are always 1, (39) can be replaced by $r^{(1)} \cdot \frac{(1)(1+k)(1-e^{-1/k})}{1-e^{-1}}$, where we have used the fact that $L^{(1)} = k$ always. This is immediately at most $\frac{r^{(j)}}{F}$, for the improved value of $F = \frac{1-e^{-\alpha^{(1)}}}{(1+k)(1-e^{-1/k})}$, completing the proof of Theorem 5 in its entirety. $\square$

B.2. Optimizing the Randomized Procedure

We can explicitly formulate the optimization problem over randomized procedures for a single item with starting inventory k and m prices $r^{(1)}, \dots, r^{(m)}$. Using the "balls in bins" counting argument, the number of configurations satisfying (13) is $D := \binom{k+m-1}{m-1}$.

We refer to these configurations in an arbitrary order using the index $d \in [D]$, where we let ρ_d denote the probability of choosing configuration d, $f_d(\cdot)$ denote the value function for d, and $L_d^{(j)}$ denote the value of $\tilde{L}^{(j)}$ under configuration d for all $j = 0, \dots, m$. The optimization problem of satisfying (16)–(17) with a maximal value of F can be formulated as follows:

$$\widetilde{\mathsf{CR}} := \sup F \tag{43a}$$

$$k(f_d(\frac{N+1}{k}) - f_d(\frac{N}{k})) + f_d(L_d^{(j)}) - f_d(\frac{N}{k}) \le \frac{r^{(j)}}{F} \qquad d \in [D], j \in [m], 0 \le N \le kL_d^{(j)} - 1 \tag{43b}$$

$$f_d(1) \ge \dots \ge f_d(\frac{1}{k}) \ge f_d(0) = 0 \qquad d \in [D] \tag{43c}$$

$$\sum_{d=1}^{D} \rho_d f_d(L_d^{(j)}) \ge r^{(j)} \qquad j \in [m] \tag{43d}$$

$$\sum_{d=1}^{D}\rho_d = 1 \tag{43e}$$

$$f_d(0), f_d(\frac{1}{k}), \ldots, f_d(1) \in \mathbb{R}; \rho_d \geq 0 \qquad d \in [D] \tag{43f}$$

Constraint (43b) corresponds to (16), constraint (43d) corresponds to (17), while constraint (43c) enforces the definition of a value function in (14). We let $\widetilde{\mathsf{CR}}$ denote the optimal objective value of (43). Unfortunately, it is difficult to solve (43) exactly, since the number of configurations D is exponential in the number of prices m, and constraint (43d) is non-linear.

Nonetheless, (43) is useful at determining the best competitive ratio which could be established *using our analysis.* We know that the randomized procedure from Definition 2 (based on Φ) is an optimal solution to (43) as $k \to \infty$, since it achieves the optimal competitive ratio possible.

We can also solve (43) exactly when $k = 1$, in which case $D = m$, where we will let $d \in [D]$ denote the configuration with $\tilde{L}^{(0)} = \ldots = \tilde{L}^{(d-1)} = 0$ and $\tilde{L}^{(d)} = \ldots = \tilde{L}^{(m)} = 1$. (43b) reduces to $2f_d(1) \leq \frac{r^{(j)}}{F}$, and needs to hold for $d \in [D]$, $j \geq d$ (for $j < d$, $kL_d^{(j)} - 1 = -1$). However, clearly only the constraint with $j = d$ is binding. As a result, (43b) corresponds to m constraints. (43d) corresponds to m constraints of the form $\sum_{d=1}^{j}\rho_d f_d(1) \geq r^{(j)}$, for $j \in [m]$.

Not counting $f_d(0)$, which must be set to 0, there are $2m+1$ variables: $\{f_d(1), \rho_d : d \in [D]\}$ and F. Consider the system of equations obtained in these $2m+1$ variables by setting (43b), (43d), and (43e) to equality. It can be checked that the unique solution is

$$f_d(1) = \frac{r^{(d)}}{\sigma^{(1)}}, \forall d \in [D]; \rho_d = \sigma^{(d)}, \forall d \in [D]; F = \frac{\sigma^{(1)}}{2} \tag{44}$$

with $\sigma^{(1)}, \ldots, \sigma^{(m)}$ defined from $r^{(1)}, \ldots, r^{(m)}$ according to (6). Furthermore, this solution is both feasible, satisfying the non-negativity constraints in (43c) and (43f), and optimal. Therefore, the value of $\widetilde{\mathsf{CR}}$ is $\frac{\sigma^{(1)}}{2}$.

B.3. Proof of Theorem 1

Now we put together the proof of Theorem 1. For all items $i \in [n]$, $\widetilde{\mathsf{CR}}_i$ is defined to be the optimal objective value of (43), with $k = k_i$, $m = m_i$, and $r^{(1)} = r_i^{(1)}, \ldots, r^{(m)} = r_i^{(m_i)}$. Consider Algorithm 1, where for all i, the randomized procedure used to initialize $\tilde{\Phi}_i$ is an optimal solution to (43) achieving the objective value of $\widetilde{\mathsf{CR}}_i$. For all i, (16)–(17) is satisfied as long as $F \leq \widetilde{\mathsf{CR}}_i$. Therefore, the maximum value of F satisfying the conditions of Theorem 4 is $\min_i \widetilde{\mathsf{CR}}_i$. By Theorem 4, this algorithm achieves a competitive ratio of $\min_i \widetilde{\mathsf{CR}}_i$.

To establish bounds (i)–(iii) from Theorem 1, for all i, we need to find a feasible randomized procedure with an objective value in (43) equal to the bound. For bounds (i) and (iii), this is established directly by the randomized procedure from Definition 2 and Theorem 5. For bound (ii), we need to split the k_i units of item i into k_i disparate items. For each single-unit item, its value function in Algorithm 1 is initialized according to the randomized procedure described by (44). This yields a value of $\frac{\sigma_i^{(1)}}{2}$, completing the proof of Theorem 1.

Appendix C: Deferred Proofs from Section 4

Proof of Lemma 2. Since the algorithm was willing to sell item i at price j, it must be the case that $W_i < L_i^{(j)}$. Let ℓ denote $\ell_i(W_i)$, which is at most j. Since we can ignore measure-zero events, we assume that $W_i \neq L^{(\ell-1)}$. We can rearrange Z_t as

$$
\begin{aligned}
&r_i^{(j)} - r_i^{(\ell)} + r_i^{(\ell)} - \left(r_i^{(\ell-1)} + (r_i^{(\ell)} - r_i^{(\ell-1)}) \frac{\exp(W_i - L_i^{(\ell-1)}) - 1}{\exp(\alpha_i^{(\ell)}) - 1} \right) \\
&= r_i^{(j)} - r_i^{(\ell)} + (r_i^{(\ell)} - r_i^{(\ell-1)}) \frac{\exp(\alpha_i^{(\ell)}) - \exp(W_i - L_i^{(\ell-1)})}{\exp(\alpha_i^{(\ell)}) - 1}.
\end{aligned}
$$

Adding $Y_i = \Phi_i'(W_i) = (r_i^{(\ell)} - r_i^{(\ell-1)}) \frac{\exp(W_i - L_i^{(\ell-1)})}{\exp(\alpha_i^{(\ell)})-1}$ to this expression, we get $r_i^{(j)} - r_i^{(\ell)} + \frac{r_i^{(\ell)} - r_i^{(\ell-1)}}{1-\exp(-\alpha_i^{(\ell)})}$, which can be re-written as $r_i^{(j)} - r_i^{(\ell)} + \frac{r_i^{(\ell)}}{1-\exp(-\alpha_i^{(1)})}$ due to (5). The result follows immediately. $\square$

Proof of Lemma 3. It suffices to show that constraint (18b) holds for all $t \in [T]$ and $i \in [n]$. Since $p_{t,i}^{(j)} \in \{0,1\}$ and the constraint clearly holds when $p_{t,i}^{(j)} = 0$, it suffices to show that $\mathbb{E}[Y_i + Z_t] \geq r_i^{(j_{t,i})}$, where $j_{t,i} \neq 0$. We will let $j = j_{t,i}$ for brevity.

Fix the realization of $W_{i'}$ for all $i' \neq i$, and consider the run of the algorithm on a modified instance with item i removed. Having fixed the values of $W_{i'}$, such a run is deterministic. Let Z^{crit} denote the pseudorevenue earned on this run during time t, possibly 0. Φ_i maps $[0, L_i^{(j)}]$ to $[0, r_i^{(j)}]$ bijectively, so we can set W^{crit} to be the value in $[0, L_i^{(j)}]$ for which $\Phi_i(W^{\mathsf{crit}}) = \max\{r_i^{(j)} - Z^{\mathsf{crit}}, 0\}$.

We now consider the run of the algorithm on the full instance with item i, which is dependent on the realization of W_i. The following two claims from Devanur et al. (2013) generalize to our multi-price setting.

1. Dominance: if $W_i \in [0, W^{\mathsf{crit}})$, then in the run with item i, item i gets matched.

Proof: Since $W^{\mathsf{crit}} > W_i$ and $W_i \geq 0$, $W^{\mathsf{crit}} > 0$. Therefore, $\Phi_i(W^{\mathsf{crit}}) > 0$. Thus $\Phi_i(W^{\mathsf{crit}}) = r_i^{(j)} - Z^{\mathsf{crit}}$ (as opposed to $\Phi_i(W^{\mathsf{crit}}) = 0$), and moreover since $W_i < W^{\mathsf{crit}}$ and Φ_i is strictly increasing, $\Phi_i(W_i) < r_i^{(j)} - Z^{\mathsf{crit}}$. This implies $r_i^{(j)} - \Phi_i(W_i) > \max\{Z^{\mathsf{crit}}, 0\}$, since $Z^{\mathsf{crit}} \geq 0$. Thus on the run with item i, either i is already matched before time t, or it is matched to customer t.

2. Monotonicity: $Z_t \geq Z^{\mathsf{crit}}$ (regardless of the realization of W_i).

Proof: fix the realization of W_i. We compare two deterministic runs of the algorithm: one with item i, and one without. We can inductively establish over $t = 0, \ldots, T$ that at the end of time t, the set of unmatched items in the run with i is a superset of that in the run without i. Therefore, in the run with i, since the algorithm is maximizing pseudorevenue over a superset of items, its pseudorevenue Z_t can be no less than Z^{crit}.

Now, conditioned on the realizations of $W_{i'}$ for $i' \neq i$, which determines the values of Z^{crit} and W^{crit}, we have $Z_t \geq Z^{\mathsf{crit}}$ (by Monotonicity) and in turn $Z^{\mathsf{crit}} \geq r_i^{(j)} - \Phi_i(W^{\mathsf{crit}})$ (by the definition of W^{crit}). Meanwhile, as long as i gets matched, Y_i gets set to $\Phi_i'(W_i)$, so by Dominance, $\mathbb{E}[Y_i | \{W_{i'} : i' \neq i\}] \geq \int_0^{W^{\mathsf{crit}}} \Phi_i'(w)dw = \Phi_i(W^{\mathsf{crit}}) - \Phi_i(0) = \Phi_i(W^{\mathsf{crit}})$. Therefore, $\mathbb{E}[Y_i + Z_t | \{W_{i'} : i' \neq i\}] \geq r_i^{(j)}$. The proof follows from the tower property of conditional expectation. $\square$

Appendix D: Deferred Proofs from Section 5

Proof of Proposition 3. The unique solution to the system (21) is obtained inductively over $j = 2, \ldots, m$ by setting $B_j = \frac{r^{(j-1)} e^{-\alpha^{(j-1)}}}{r^{(j)} e^{-\alpha^{(j)}}} B_{j-1}$. By (32), $\frac{r^{(j-1)}}{r^{(j)}} \leq e^{-\alpha^{(j)}}$, hence $B_j \leq e^{-\alpha^{(j-1)}} B_{j-1}$. But $\alpha^{(j-1)} > 0$ by Proposition 1, completing the proof that $B_j < B_{j-1}$ for $j = 2, \ldots, m$. The fact that $0 < B_m$ is immediate. $\square$

Proof of Lemma 4. Consider the execution of an online algorithm on this randomized instance. For all $i \in [n]$ and group of customers $t \in [n]$, let $Q_{t,i}$ denote the number of group-t customers to which item π_i is sold, which is a random variable with respect to the random permutation π as well as any randomness in the algorithm. Let $q_{t,i} = \mathbb{E}[Q_{t,i}]$.

Clearly if $i < t$, then $Q_{t,i} = 0$, because group-t customers have no interest in item π_i. Otherwise, for any $i, i' \geq t$, we argue that $q_{t,i} = q_{t,i'}$. This is because while group t is arriving, the online algorithm cannot distinguish between items π_i and $\pi_{i'}$, hence any items it allocates are equally likely to be item π_i and item $\pi_{i'}$. Therefore, we let q_t denote the value of $q_{t,i}$ for $i \geq t$.

Now, consider item π_n. Since it only has k units of inventory, we know that $\sum_{t=1}^{n} Q_{t,n} \leq k$ on every sample path. Using the linearity of expectation, we get that

$$\sum_{t=1}^{n} q_t \leq k. \tag{45}$$

Furthermore, for a $t \in [n]$, on every sample path, $\sum_{i=t}^{n} Q_{t,i} \leq k$, since there are only k customers in group t. Therefore, $(n+1-t)q_t \leq k$, or

$$q_t \leq \frac{k}{n+1-t}. \tag{46}$$

For this proof, let $M_j = \sum_{j'=1}^{j} \beta_{j'}$, for all $j = 0, \ldots, m$. For all $j \in [m]$, let $\lambda_j = \frac{1}{k} \sum_{t=M_{j-1}n+1}^{M_j n} q_t$. Substituting into (45), we get the constraint that $\sum_{j=1}^{m} \lambda_j \leq 1$. For any $j \in [m-1]$, summing inequality (46) for $t = M_{j-1}n+1, \ldots, M_j n$ yields $\lambda_j \leq \ln \frac{B_j}{B_{j+1}}$, since $n \to \infty$, and $B_j = 1 - M_{j-1}$, $B_{j+1} = 1 - M_j$ by definition. It is also clear from definition that $\lambda_j \geq 0$ for all $j \in [m]$.

Finally, the total expected revenue is

$$\sum_{j=1}^{m} r^{(j)} \sum_{t=M_{j-1}n+1}^{M_j n} q_t(n+1-t), \tag{47}$$

since for each group t there are $n+1-t$ items for each of which q_t copies are sold in expectation. Consider any $j \in [m]$. Since $\sum_{t=M_{j-1}n+1}^{M_j n} q_t = \lambda_j k$ by definition, $\sum_{t=M_{j-1}n+1}^{M_j n} q_t(n+1-t)$ is maximized by setting q_t to its upper bound in (46) for $t = M_{j-1}n+1, M_{j-1}n+2, \ldots$ until the capacity of $\lambda_j k$ is reached. Since $n \to \infty$, we can simply compute the value of t for which

$$\frac{k}{n - M_{j-1}n} + \ldots + \frac{k}{n-t} = \lambda_j k, \tag{48}$$

with $t \in [M_{j-1}n, M_j n]$. Letting $t = (M_{j-1} + y\beta_j)n$ with $y \in [0,1]$, and using the definition of B_j, (48) becomes $\ln \frac{B_j}{B_j - y\beta_j} = \lambda_j$, or $y\beta_j = B_j(1 - e^{-\lambda_j})$. Therefore,

$$\sum_{t=M_{j-1}n+1}^{M_j n} q_t(n+1-t) \le \sum_{t=M_{j-1}n+1}^{(M_{j-1}+B_j(1-e^{-\lambda_j}))n} \frac{k}{n+1-t} \cdot (n+1-t)$$
$$= B_j(1-e^{-\lambda_j})nk$$

Substituting into (47), we get that the expected revenue of the online algorithm is at most (23), where $\sum_{j=1}^{m} \lambda_j \le 1$, $\lambda_j \le \ln \frac{B_j}{B_{j+1}}$ for $j \in [m-1]$, and $\lambda_j \ge 0$ for $j \in [m]$, completing the proof. □

Proof of Lemma 5. We use backward induction over $j = m, \ldots, 1$. When $j = m$, (25) becomes $nkr^{(m)}B_m(1 - \exp(-\tau))$, since $A_m = \alpha^{(m)}$ by definition. Meanwhile, (24) is maximized by setting $\lambda_m = \tau$, resulting in the same expression and establishing the base case.

Now suppose $j < m$ and that we have already established the lemma in the $j+1$ case. If we set $\lambda_j = \lambda$, for some $\lambda \in [0,\tau]$, then the maximum value of (24) subject to $\lambda_{j+1}, \ldots, \lambda_m \ge 0$ and $\lambda_{j+1} + \ldots + \lambda_m \le \tau - \lambda$ is, by the inductive hypothesis,

$$r^{(j)}B_j(1-\exp(-\lambda))nk + nk\sum_{\ell=j+1}^{m} r^{(\ell)}B_\ell\Big(1 - \exp\big(-\alpha^{(\ell)} + \frac{A_{j+1} - (\tau-\lambda)}{m-(j+1)+1}\big)\Big). \tag{49}$$

Consider this expression as a function of λ. The derivative is

$$r^{(j)}B_j\exp(-\lambda)nk + nk\sum_{\ell=j+1}^{m} r^{(\ell)}B_\ell \cdot \frac{-1}{m-j} \cdot \exp\big(-\alpha^{(\ell)} + \frac{A_{j+1} - (\tau-\lambda)}{m-j}\big) \tag{50}$$

and the second derivative is clearly negative, so the function is concave. Therefore, it is maximized by setting the derivative to 0. By definition (21), $r^{(\ell)}B_\ell e^{-\alpha^{(\ell)}}$ is identical for all $\ell = j+1, \ldots, m$, and equal to $r^{(j)}B_j e^{-\alpha^{(j)}}$. Thus setting (50) to 0 implies:

$$\exp(\alpha^{(j)} - \lambda) = \frac{1}{m-j}\sum_{\ell=j+1}^{m} \exp\big(\frac{A_{j+1} - (\tau-\lambda)}{m-j}\big)$$
$$\alpha^{(j)} - \lambda = \frac{A_{j+1} - (\tau-\lambda)}{m-j}.$$

Rearranging and using the definition that $A_{j+1} = A_j - \alpha^{(j)}$, we get $\lambda = \alpha^{(j)} - \frac{A_j - \tau}{m-j+1}$. Substituting this value of λ into (49), the expression $\frac{A_{j+1}-(\tau-\lambda)}{m-(j+1)+1}$ is equal to $\frac{A_j - \tau}{m-j+1}$, hence (49) is equal to (25), completing the induction and the proof of the lemma. □

Appendix E: Deriving the Multi-price Value Function Φ_i

Throughout this paper, we have proven results critically dependent on the exact definitions of $\alpha_i^{(1)},\ldots,\alpha_i^{(m_i)}$ in (5), and Φ_i in (8). In this section we explain how to derive the system of equations in (5), and the functional form in (8). In Subsection E.1, we use the same method to derive the optimal value function when the price of an item i can take any value in the continuum $[r^{\min}, r^{\max}]$. We omit the subscript i throughout this section.

Consider constraints (16)–(17) in Theorem 5 for a single item with $k\to\infty$. Let $w=\frac{N}{k}$, and we deterministically set $\tilde{\Phi}$ to some Φ. The goal is to solve for the Φ which maximizes the value of F.

Observe that

$$\lim_{k\to\infty} k(\Phi(\frac{N+1}{k})-\Phi(\frac{N}{k}))=\lim_{k\to\infty}\frac{\Phi(w+1/k)-\Phi(w)}{1/k},$$

which is equal to the derivative of Φ as w, by definition (Φ will end up not being differentiable on a discrete set of measure 0, which can be ignored). Therefore, (16) is equivalent to

$$\Phi'(w)-\Phi(w)\le r^{(j)}(\frac{1}{F}-1), \tag{51}$$

and needs to hold for all $j\in[m], w\in[0,L^{(j)}]$. For a fixed $w\in(L^{(j-1)},L^{(j)})$, (51) needs to hold for all $j'=j,\ldots,m$, but is clearly binding when $j'=j$. Therefore, it suffices to fix a $j\in[m]$ and consider (51) when $w\in(L^{(j-1)},L^{(j)})$.

We should point out that this simplification via the "binding" argument is not possible for a finite k and random $\tilde{\Phi}$, because then (51) becomes $\tilde{\Phi}'(w)-\tilde{\Phi}(w)\le\frac{r^{(j)}}{F}-\tilde{\Phi}(L^{(j)})$, and the RHS in fact may not be increasing in j.

If we set (51) to equality for some $j\in[m]$ and all $w\in(L^{(j-1)},L^{(j)})$, and solve the differential equation, we get that $\Phi(w)$ must be of the form $Ce^w-r^{(j)}(\frac{1}{F}-1)$ on $(L^{(j-1)},L^{(j)})$. Setting $\Phi(L^{(j-1)})=r^{(j-1)}$ and $\Phi(L^{(j)})=r^{(j)}$, we obtain

$$C=\frac{r^{(j)}-r^{(j-1)}}{e^{L^{(j)}}-e^{L^{(j-1)}}};$$

$$F=\frac{1}{1-\frac{r^{(j-1)}}{r^{(j)}}}\cdot(1-e^{-\alpha^{(j)}}). \tag{52}$$

The RHS of (52) is the largest value of F which allows (51) to hold on segment j. It is dependent on $\alpha^{(j)}$, which is equal to $L^{(j)} - L^{(j-1)}$, the length of segment j. For (51) to hold on all segments $j \in [m]$, F must be set to $\min_j \frac{1}{1-r^{(j-1)}/r^{(j)}} \cdot (1 - e^{-\alpha^{(j)}})$.

Therefore, we would like to choose segment lengths $\alpha^{(1)}, \ldots, \alpha^{(m)}$ summing to 1 to maximize the minimum $\frac{1}{1-r^{(j-1)}/r^{(j)}} \cdot (1 - e^{-\alpha^{(j)}})$, which is accomplished by setting $\frac{1}{1-r^{(j-1)}/r^{(j)}} \cdot (1 - e^{-\alpha^{(j)}})$ equal for all $j \in [m]$. This yields the system of equations (5), and Proposition 1. The resulting value of F is equal to $1 - e^{-\alpha^{(1)}}$, since $r^{(0)} = 0$. The resulting value of C, when substituted into the equation for $\Phi(w)$ on each segment $(L^{(j-1)}, L^{(j)})$, yields (8).

The derivation of Φ we just completed, starting from condition (51), comes from our analysis of Multi-price Balance. We note that the exact same inequality (51) can also be derived from our analysis of Multi-price Ranking.

E.1. Continuum of Feasible Prices

Let the feasible price set for the item be $[r^{\min}, r^{\max}]$, where $0 < r^{\min} < r^{\max}$. Using the same "binding" argument, it suffices to maximize the value of F for which the following can hold:

$$\Phi'(w) - \Phi(w) \le r^{\min}(\frac{1}{F} - 1), \qquad w \in (0, \alpha); \tag{53}$$

$$\Phi'(w) - \frac{\Phi(w)}{F} \le 0, \qquad w \in (\alpha, 1). \tag{54}$$

Φ must also satisfy $\Phi(0) = 0, \Phi(\alpha) = r^{\min}, \Phi(1) = r^{\max}$, while $\alpha \in (0, 1)$ is an arbitrary "booking limit" for the lowest price of $r^{\min}$.

We know from before that under the optimal solution to (53), the value of F can be at most $1 - e^{-\alpha}$. Solving the differential equation where (54) is set to equality, $\Phi(w)$ must take the form $Ce^{w/F}$ on $(\alpha, 1)$. Substituting $\Phi(\alpha) = r^{\min}$ and $\Phi(1) = r^{\max}$ yields

$$C = (r^{\min})^{\frac{1}{1-\alpha}} (r^{\max})^{-\frac{\alpha}{1-\alpha}};$$

$$F = \frac{1-\alpha}{\ln \frac{r^{\max}}{r^{\min}}}.$$

Therefore, the value of F is also bounded from above by $\frac{1-\alpha}{\ln(r^{\max}/r^{\min})}$. F is maximized by setting $\frac{1-\alpha}{\ln(r^{\max}/r^{\min})}$ equal to the other upper bound of $1 - e^{-\alpha}$; the value at which equality is achieved is then the competitive ratio.

Letting $R = \ln(r^{\max}/r^{\min})$, the solution to $\frac{1-\alpha}{R} = 1 - e^{-\alpha}$ can be written as $W(Re^{R-1}) - R + 1$, where W is the Lambert-W function, the inverse function to $f(x) = xe^x$ for $x \in \mathbb{R}_{\geq 0}$. Indeed, when $\alpha = W(Re^{R-1})$, the following can be derived:

$$\begin{aligned}
\frac{1-\alpha}{R} &= 1 - e^{-\alpha} \\
Re^{-\alpha} &= \alpha + R - 1 \\
Re^{R-1} &= (\alpha + R - 1)e^{\alpha + R - 1} \\
W(Re^{R-1}) &= \alpha + R - 1
\end{aligned}$$

Substituting $\alpha = W(\ln(r^{\max}/r^{\min})e^{\ln(r^{\max}/r^{\min})-1}) - \ln(r^{\max}/r^{\min}) + 1$ into the formula for C, and using the fact that $\Phi(w) = Ce^{w/F}$, we get

$$\Phi(w) = (r^{\min})^{\frac{1-w}{1-\alpha}}(r^{\max})^{\frac{w-\alpha}{1-\alpha}}, \qquad w \in [\alpha, 1].$$

Meanwhile, the derivation preceding Subsection E.1 implies that

$$\Phi(w) = r^{\min} \cdot \frac{e^w - 1}{e^\alpha - 1}, \qquad w \in [0, \alpha].$$

It can be checked that indeed $\Phi(0) = 0$, $\Phi(\alpha) = r^{\min}$ (Φ is continuous at $w = \alpha$), and $\Phi(1) = r^{\max}$. Furthermore, unlike the case of discrete prices, it can be checked that Φ is also differentiable at $w = \alpha$ (on $[\alpha, 1]$, use the form that $\Phi(w) = Ce^{w/F}$, hence $\Phi'(\alpha) = \frac{\Phi(\alpha)}{F}$).

Appendix F: Supplement to Numerical Experiments

We provide additional details about our choice estimation. We define 8 customer types, one for each combination of the 3 following binary features.

1. Group: whether the customer indicated a party size greater than 1.
2. CRO: whether the customer booked using the Central Reservation Office, as opposed to the hotel's website or a Global Distribution System (for details on these terms, see Bodea et al. (2009)).
3. VIP: whether the customer had any kind of VIP status.

Table 3 MNL choice models for the 8 customer types. The suffix "L" on a room type means lower fare, while the suffix "H" on a room type means higher fare.

Customer Type				MNL Mean Utilities								
Group?	CRO?	VIP?	Share	KingL	QueenL	SuiteL	2DoubleL	KingH	QueenH	SuiteH	2DoubleH	NoBuy
			0.16	-0.36	-1.22	-2.56	-1.04	0	-0.23	-2.25	-1.8	0
		✓	0.03	-0.82	-1.98	-2.16	-2.09	0	-1.02	-1.45	-1.82	0
	✓		0.28	-1.67	$-\infty$	-3.78	-2.71	0	-1.33	-1.8	-1.58	0
	✓	✓	0.09	-2.13	$-\infty$	-3.38	-3.76	0	-2.12	-1	-1.59	0
✓			0.19	-0.54	-0.97	-2.26	0	-0.91	-1.47	-2.78	-1.41	0
✓		✓	0.04	-0.09	-0.82	-0.95	-0.14	0	-1.35	-1.07	-0.51	0
✓	✓		0.18	-0.93	$-\infty$	-2.56	-0.76	0	-1.66	-1.41	-0.27	0
✓	✓	✓	0.03	-1.39	$-\infty$	-2.16	-1.8	0	-2.45	-0.61	-0.28	0

We did not use features such as: whether the booking date is a weekend, whether the check-in date is a weekend, the length of stay, or the number of days in advance booked. Such features did not result in a more predictive model.

We estimate the mean MNL utilities for each of the 8 products separately for each customer type. The results are displayed in Table 3. The total share of each customer type (out of all the transactions) is also displayed. We should point out that it is possible for a customer to choose the higher fare for a room, even if the lower fare was also offered. This is because the higher fares are often packaged with additional offers, such as airline services, city attractions, in-room services, etc.

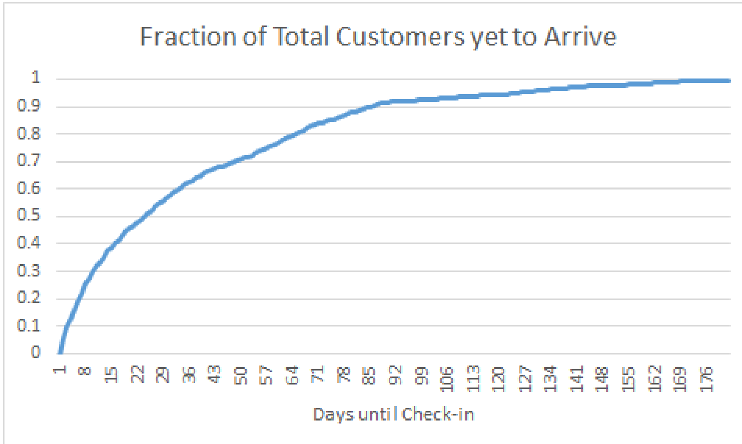
We have shifted the mean utilities so that for each customer type, the weights of both the no-purchase option, and the most-preferred purchase option, is equal to 0. The large weights on the no-purchase options ensure that the revenue-maximizing assortments tend to include both the low and high fares.

In the setting with greater fare differentiation (Subsection 7.5), the high prices of the King, Queen, Suite, and Two-double rooms are adjusted to \$614, \$608, \$768, \$612, respectively (twice the lower fares). The mean utility of the no-purchase option is increased by 2 for every customer type, to ensure that the revenue-maximizing assortments still include both the low and high fares.

F.1. Details on the Forecasting Bid-price Algorithms

To forecast the remaining number of customers, we assume that we know the average number of customers interested in each occupancy date (1340), as well as the overall trend for how far

Figure 4 Distribution of arrivals over the days before check-in, formed by aggregating all transactions.



in advance customers book, which is plotted in Figure 4. As an example of how to use these numbers, consider the occupancy date March 31st. At the start, we forecast there to be 1340 arrivals. However, suppose by March 6th, 500 customers have arrived. Since we know from Figure 4 that roughly 50% of the total population interested in March 31st will have already booked by March 6th (25 days in advance), we expect there to only be 500 customers remaining.

To forecast the breakdown of remaining customers by type, we assume that we know the aggregate distribution of customer type over all occupancy dates. For example, from Appendix F, we know that 28% of all customers are of Type 3. Then we would estimate $28\% \times 500 = 140$ of the 500 remaining customers to be of Type 3. Alternatively, one can try to learn the specific distribution of customers interested in March 31st. Suppose that only 100, or 20%, of the 500 bookings made before March 6th came from customers of Type 3. Then we would instead estimate $20\% \times 500 = 100$ of the 500 remaining customers to be of Type 3.

To use the forecasted information, algorithms incorporate it into the LP (27), and set the *bid price* of each item i equal to the shadow price of constraint i in (27b). These algorithms then offer each customer t the assortment S (from the available items) maximizing $\sum_{(i,j)\in S} p_{t,i}^{(j)}(S)\big(r_i^{(j)} - \lambda_i\big)$.

We clarify the exact way in which the forecasted information is incorporated into the LP. Let there be A customer types, indexed by $a = 1, \ldots, A$. We use $p_{a,i}^{(j)}(S)$ to denote the probability of a customer of type a choosing product (i,j) from assortment S. Suppose that when we want to re-solve the LP (27), the forecasted number of remaining customers of type a is N_a, for all $a \in [A]$, and the remaining inventory of item i is K_i, for all $i \in [n]$. We can formulate the following LP, which is a modification of (27):

$$
\begin{aligned}
\max \sum_{a=1}^{A} \sum_{S} x_a(S) \sum_{(i,j)\in S} r_i^{(j)} p_{a,i}^{(j)}(S) & & \\
\sum_{a=1}^{A} \sum_{S} x_a(S) \sum_{j:(i,j)\in S} p_{a,i}^{(j)}(S) \le k_i & & i \in [n] \\
\sum_{S} x_a(S) = N_a & & a \in [A] \\
x_a(S) \ge 0 & & a \in [A], S \subseteq \{(i,j) : i \in [n], j \in [m_i]\}
\end{aligned}
$$

We have set $T = \sum_{a=1}^{A} N_a$ and $|\{t : \text{type of customer } t \text{ is } a\}| = N_a$; note that the ordering of remaining customers is inconsequential for the LP.

Although this LP has an exponential number of variables, it can easily be solved using column generation (e.g., see Liu and Van Ryzin (2008)). Fix an optimal primal solution $(x_a^*(S) : a \in [A], S \subseteq \{(i,j) : i \in [n], j \in [m_i]\})$ and an optimal dual solution $(y_i^* : i \in [n]), (z_a^* : a \in [A])$. The bid-price algorithm sets the bid price of each item i equal to y_i^*.

We should point out that for every bid-price algorithm based on dual variables, there is a corresponding *random assignment* algorithm based on primal variables. Such an algorithm would, for each customer type a, offer each assortment S with probability proportional to $x_a^*(S)$. We have confirmed that these algorithms perform similarly in the simulations. We compare with the bid-price algorithms instead of the random assignment algorithms because they follow a form more similar to our Multi-price Balance algorithm.