informs
http://pubsonline.informs.org/journal/opre

OPERATIONS RESEARCH
Vol. 69, No. 3, May–June 2021, pp. 895–915
ISSN 0030-364X (print), ISSN 1526-5463 (online)

Methods

Online Resource Allocation Under Partially Predictable Demand

Dawsen Hwang,[a] Patrick Jaillet,[b] Vahideh Manshadi[c,*]

[a] Google, Chicago, Illinois 60607; [b] Department of Electrical Engineering and Computer Science, Laboratory for Information and Decision Systems, Operations Research Center, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139; [c] Yale School of Management, New Haven, Connecticut 06511

*Corresponding author

Contact: dawsen@gmail.com (DH); jaillet@mit.edu (PJ); vahideh.manshadi@yale.edu, https://orcid.org/0000-0001-9103-7797 (VM)

Received: September 20, 2018
Accepted: April 1, 2020
Published Online in Articles in Advance: January 18, 2021

Area of Review: Optimization

https://doi.org/10.1287/opre.2020.2017

Abstract. For online resource allocation problems, we propose a new demand arrival model where the sequence of arrivals contains both an adversarial component and a stochastic one. Our model requires no demand forecasting; however, because of the presence of the stochastic component, we can partially predict future demand as the sequence of arrivals unfolds. Under the proposed model, we study the problem of the online allocation of a single resource to two types of customers and design online algorithms that outperform existing ones. Our algorithms are adjustable to the relative size of the stochastic component; our analysis reveals that as the portion of the stochastic component grows, the loss due to making online decisions decreases. This highlights the value of (even partial) predictability in online resource allocation. We impose no conditions on how the resource capacity scales with the maximum number of customers. However, we show that using an adaptive algorithm—which makes online decisions based on observed data—is particularly beneficial when capacity scales linearly with the number of customers. Our work serves as a first step in bridging the long-standing gap between the two well-studied approaches to the design and analysis of online algorithms based on (1) adversarial models and (2) stochastic ones. Using novel algorithm design, we demonstrate that even if the arrival sequence contains an adversarial component, we can take advantage of the limited information that the data reveal to improve allocation decisions. We also study the classical secretary problem under our proposed arrival model, and we show that randomizing over multiple stopping rules may increase the probability of success.

Funding: This work was supported by the Office of Naval Research [Grants N00014-12-1-0999 and N00014-16-1-2786].

Supplemental Material: The online appendices are available at https://doi.org/10.1287/opre.2020.2017.

Keywords: online resource allocation • competitive analysis • analysis of algorithms

1. Introduction

E-commerce platforms host markets for perishable resources in various industry sectors ranging from airlines to hotels to internet advertising. In these markets, demand realizes sequentially and the firms need to make online (irrevocable) decisions regarding how (and at what price) to allocate resources to arriving demand without precise knowledge of future demand. The success of any online allocation algorithm depends crucially on a firm's ability to predict future demand. If demand can be predicted, then under some conditions on the amount of available resources, making online decisions incurs little loss (as shown in Agrawal et al. (2014), among others). However, in many markets, demand cannot be perfectly predicted because of unpredictable components, such as traffic spikes and strategy changes by competitors. The emergence of sharing-economy platforms, such as Airbnb, which can scale supply at negligible cost and on short notice (Zervas et al. 2017), has significantly added to unpredictable variability in demand even for products that are not new (e.g., existing hotels).

In such cases, firms can take a worst-case approach and assume that demand is controlled by an imaginary adversary and thus is unpredictable. Such an approach, however, usually results in online policies that are too conservative (as studied in Ball and Queyranne (2009) and others). Instead, firms may wish to employ online policies based on models that assume the future demand can partially be predicted, avoiding being too conservative while not being reliant on fully accurate predictions. This paper aims to investigate to what extent the above goal is achievable. We propose a new demand model, called *partially predictable*, that contains both adversarial (thus unpredictable) and stochastic (predictable) components. We design novel algorithms to demonstrate that even though demand is assumed to include an unpredictable component, firms can make use of the limited information that the data reveal and improve upon the completely conservative approach.

We study a basic online allocation problem of a single resource with an arbitrary capacity to a sequence of customers, each of which belongs to one of two types. Each customer demands one unit of the resource. If the resource is allocated, the firm earns a type-dependent revenue. Type-1 and -2 customers generate revenue of 1 and $a < 1$, respectively. Our demand model takes a parameter $0 < p < 1$ and works as follows. An unknown number of customers of each type will be revealed to the firm in an unknown order. Both the number and the order of customers are assumed to be controlled by an imaginary adversary. However, a fraction p of randomly chosen customers does *not follow* this prescribed order and instead arrives at uniformly random times. This group of customers represents the stochastic component of the demand that is mixed with the adversarial element. Although we cannot identify which customers belong to the stochastic group, we can still *partially* predict future demand, because this group is almost uniformly spread over the time horizon. Therefore, parameter p determines the level of predictability of demand.

From a practical point of view, our demand model requires *no forecast* for the number of customers of each type prior to arrival; instead, it assumes a rather mild "regularity" in the arrival pattern: a fraction p of customers of each type is spread throughout the time horizon. We motivate this through a simple example. Suppose an airline launches a new flight route for which it has no demand forecast. However, using historical data on customer booking behavior, the airline knows that there is heterogeneity in booking behavior of customers, namely, the time they request a booking varies across customers of each type. Such heterogeneity results in the gradual arrival of a portion of customers of each type. For example, CWT (2016) illustrates a significant disparity in the advanced booking behavior of business travelers based on their age, gender, and travel frequency. Therefore, the airline can reasonably assume that demand from business travelers (who correspond to type-1 in our model) is, to some degree, spread over the sale horizon.

From a theoretical point of view, our demand model aims to address the limitations of the main two approaches that have been taken so far in the literature: (1) adversarial models and (2) stochastic ones.[1] Under the adversarial modeling approach, the sequence of arrivals is assumed to be completely unpredictable. The online algorithms developed for these models aim to perform well in the worst-case scenario, often resulting in very conservative bounds (see Ball and Queyranne (2009) for the single-resource revenue management [RM] problem and Mehta et al. (2007) and Buchbinder and Naor (2009) for online allocation problems in internet advertising). On the other hand, the stochastic modeling approach assumes that demand follows an unknown distribution (Kleinberg 2005, Devanur and Hayes 2009, Agrawal et al. 2014).[2] In this case, we can predict future demand after observing a small fraction of it. For example, after observing the first 10% of the demand, if we observe that 15% of customers are of type-1, we can predict that roughly 15% of the remaining customers are also of type-1. The limitation of such an approach is that it cannot model variability across time. In some cases, real data do not confirm the stochastic structure presumed in these models, as shown in Wang et al. (2006) and Shamsi et al. (2014). In fact, as discussed in Mirrokni et al. (2012) and Esfandiari et al. (2015), large online markets (such as internet advertising systems) often use modified versions of these algorithms to make them less reliant on accurate demand prediction. Our model provides a middle ground between the aforementioned approaches by assuming that the arrival sequence contains both an adversarial component and a stochastic one.

For the above problem, we design two online algorithms (a nonadaptive and an adaptive one[3]) that perform well in the partially predictable model. We use the metric of competitive ratio, which is commonly used to evaluate the performance of online algorithms. Competitive ratio is the worst-case ratio between the revenue of the online scheme to that of a clairvoyant solution (see Definition 1). The competitive ratio of our algorithms is parameterized by p, and for both algorithms the ratio increases with p: *as the relative size of the stochastic component grows, the loss due to making online decisions decreases.* We further show that using an adaptive algorithm is particularly beneficial when the capacity scales linearly with the maximum number of customers. Our algorithms are easily adjustable with respect to parameter p. Therefore, if a firm wishes to use different levels of predictability for different products, then it can use the same algorithm with different parameters p.

In designing our algorithms, we keep track of the number of accepted customers of each type and we decide whether to accept an arriving type-2 customer by comparing the number of already accepted type-2 customers with optimally designed *dynamic* thresholds.[4] Our nonadaptive algorithm strikes a balance between "smoothly" allocating the inventory over time (by not accepting many type-2 customers toward the beginning) and not protecting too much inventory for potential late-arriving type-1 customers (see Algorithm 1 and Theorem 1). Our adaptive algorithm frequently recomputes upper bounds on the number of future customers of each type based on observed data and uses these upper bounds to ensure that we protect enough inventory for future type-1 customers.

We show that such an adaptive policy significantly improves the performance guarantee when the initial inventory is large relative to the maximum number of customers (see Algorithm 2 and Theorem 2). Both algorithms could reject a type-2 customer early on but accept another type-2 customer later. This is consistent with practice. For example, in online airline booking systems, lower fare classes can open up after being closed out previously (Cheapair 2016).

From a methodological standpoint, an analysis of the competitive ratio of our algorithms presents many new technical challenges arising from the fact that our arrival model contains both an adversarial and a stochastic component. Our analysis crucially relies on a concentration result that we establish for our arrival model (see Lemma 1) as well as fairly intricate case analyses for both algorithms. Further, to prove the lower bound on the competitive ratio of our adaptive algorithm, we construct a novel *factor-revealing* nonlinear mathematical program (see (MP1) and Section 5.2).

The two extreme cases of our model where all or none of the customers belong to the adversarial group (i.e., $p = 0$ and $p = 1$) reduce to the adversarial and stochastic modeling approaches that have been mainly studied in the literature thus far (for instance, Ball and Queyranne (2009) study the former model and Agrawal et al. (2014) study the latter). Our algorithms recover the known performance guarantees for these two extreme cases. For the regime in between (i.e., when $0 < p < 1$), we show that our algorithms achieve competitive ratios better than what can be achieved by any of the algorithms designed for these extreme cases (or even any combination of them). This highlights the need to design new algorithms when departing from traditional arrival models.

We also study the classic secretary problem under our partially predictable arrival model. The secretary problem, a stopping time problem, corresponds to the setting in which we initially have one unit of inventory; each customer is of a different type, and we wish to maximize the probability of allocating the inventory to the type generating the highest revenue. We show that, unlike the classic setting (which corresponds to $p = 1$ in our model), the celebrated deterministic stopping rule policy based on a deterministic observation period is no longer optimal. Because of the presence of the adversarial component, randomizing over the length of the observation period may result in improvement (see Algorithm 3, Theorem 3, and the ensuing proposition).

We conclude this section by highlighting our motivations and contributions. For many applications, demand arrival processes are inherently prone to contain unpredictable components that even advanced information technologies cannot mitigate. An allocation policy whose design is based on stochastic modeling cannot incorporate the presence of such unpredictable components. At the same time, taking a worst-case adversarial approach usually leads to allocation polices that are too conservative. We introduce the *first arrival model* that contains *both adversarial (thus unpredictable) and stochastic* components. Through novel algorithm design, we show that (1) we can take advantage of even limited available information (because of the presence of the stochastic component) to improve a firm's revenue when compared with algorithms that take a worst-case approach and that (2) there is an unavoidable loss due to the presence of an adversarial component, which emphasizes *the value of stochastic information and predictability* in online resource allocation.

The rest of the paper is organized as follows. In Section 2, we review the related literature and highlight the differences between the current paper and previous work. In Section 3, we formally introduce our demand arrival model and our performance metric and prove a consequential concentration result for the arrival process. Sections 4 and 5 are dedicated to description and analysis of our two algorithms. In Section 6, we present upper bounds on the performance of any online algorithm and compare the performance of our algorithms with that of existing ones. Section 7 studies the secretary problem under our new arrival model. In Section 8, we conclude by outlining several directions for future research. For the sake of brevity, we include proofs of only selected results in the main text. Detailed proofs of the remaining statements are deferred to clearly marked online appendices.

2. Literature Review

Online allocation problems have broad applications in revenue management: internet advertising and scheduling appointments (through web applications) in healthcare, just to name a few. Thus, it has been studied in various forms in operations research and management as well as computer science. As discussed in the introduction, the approach taken in modeling the arrival process is the first consequential step in studying these problems. Therefore, in this literature review, we categorize related streams of research by modeling approach rather than by the particular problem formulation and application.

First, we note that the single-leg revenue management problem and its generalizations have been extensively studied using frameworks other than online resource allocation problems and competitive analysis. Earlier papers assumed *low-before-high* models (where all low-fare demand realizes before high-fare demand) with known demand distributions (Belobaba 1987, 1989; Brumelle and McGill 1993; Littlewood 2005) or assumed the arrival process is known, and formulated the problem as a Markov decision problem (Lee and Hersh 1993, Lautenbacher and Stidham Jr. 1999). We refer the reader to Talluri and Van Ryzin (2006) for a comprehensive review of RM literature. Further, many recent papers in revenue management study dynamic pricing when the underlying price-sensitive demand process is unknown. See, for example, seminal work by Besbes and Zeevi (2009) and Araman and Caldentey (2009). For the sake of brevity, we will not review these streams of work.

2.1. Adversarial Models

Ball and Queyranne (2009) studied the single-leg revenue management problem under an adversarial model and showed that, in the two-fare case, the optimal competitive ratio is $\frac{1}{2-a}$ where $a < 1$ is the ratio of two fares. As discussed in the introduction, our model reduces to that of Ball and Queyranne (2009) for $p = 0$. In this special case, our nonadaptive algorithm reduces to the threshold policy of Ball and Queyranne (2009) and recovers the same performance guarantee. However, when $0 < p < 1$, we show that for a certain class of instances, our algorithms perform better than that of Ball and Queyranne (2009) (see Subsection 6.2), indicating the need for new algorithms for our new arrival model. Further, we point out that Ball and Queyranne (2009) extended their analysis to a setting with more than two types. In particular, they show that if the value of types is in interval $[v_H, v_L]$ where $0 < v_L \le v_H$, the tight competitive ratio is $\frac{1}{1+\log(v_H/v_L)}$. More recently, Ma and Simchi-Levi (2019) studied a generalized online assortment problem with multiple types as well as multiple resources to offer, again, in the adversarial setting.

Several papers studied the adwords problem under the adversarial model (Mehta et al. 2007, Buchbinder and Naor 2009). This problem concerns allocating ad impressions to budget-constrained advertisers. As mentioned in Mehta et al. (2007), even though the optimal competitive ratio under an adversarial model is $1 - 1/e$, one would expect to do better when statistical information is available. Later, Mirrokni et al. (2012) showed that it is impossible to design an algorithm with a near-optimal competitive ratio under both adversarial and random arrival models. Such an impossibility result affirms the need for new modeling approaches to serve as a middle ground between these two models. Our paper takes a step in this direction. (In Online Appendix EC.1, we show that under our setting, it is not possible to design an online algorithm that simultaneously achieves the best possible competitive ratio in the adversarial model as well as the random order model.)

2.2. Stationary Stochastic Models

A general form of these models is the *random order model*, which assumes that the sequence of arrivals is a random permutation of an arbitrary sequence (Kleinberg 2005, Devanur and Hayes 2009, Agrawal et al. 2014). In such a model, after observing a small fraction of the input, one can predict pattern of future demand. This intuition is used to develop primal- and dual-based online algorithms that achieve near-optimal revenue, under appropriate conditions on the relative amount of available resources to allocate. These algorithms rely heavily on learning from observed data, either once (Devanur and Hayes 2009) or repeatedly (Kleinberg 2005, Agrawal et al. 2014, Kesselheim et al. 2014). As discussed in the introduction, arrival patterns could experience high variability across time, limiting the performance of these algorithms in practice (Mirrokni et al. 2012, Esfandiari et al. 2015). We note that assuming independent and identically distributed (i.i.d.) arrivals with known or unknown distributions also falls into this category of modeling approaches. Several revenue management papers provided asymptotic analysis of linear programming – based approaches for such settings; see Talluri and Ryzin (1998), Cooper (2002), and Jasin (2015).

Our model reduces to a special case of the model studied by Agrawal et al. (2014) only for $p = 1$; and like their algorithm, ours also achieves near-optimal revenue when $p = 1$. However, when $0 < p < 1$, we show, in Subsection 6.2, that for a certain class of instances our algorithms perform better than that of Agrawal et al. (2014). We point out that Devanur and Hayes (2009), Kleinberg (2005), and Kesselheim et al. (2014) also studied similar settings that can be viewed as a generalized version of our online resource allocation problem but only when $p = 1$.

2.3. Nonstationary Stochastic Models

Motivated by advanced service reservation and scheduling, Wang and Truong (2015) and Stein et al. (2020) studied online allocation problems where demand arrival follows a *known* nonhomogeneous Poisson process. For such settings, they developed online algorithms with constant competitive ratios. Further, Ciocan and Farias (2012) considered another interesting setting where the (unknown) arrival process belongs to a broad class of stochastic processes. They proved a constant factor guarantee for the case where arrival rates are uniform. Our modeling strategy differs from both approaches by assuming that $(1 - p)$ fraction of the input is adversarial. Even for the stochastic component, we assume no prior knowledge of the distribution. However, we limit the adversary's power by assuming that these two components are *mixed*. Also, we note that the aforementioned papers studied more general allocation problems in settings like network revenue management.

2.4. Other Models

Several earlier papers also acknowledged and addressed the limitation of both the adversarial and random order (or stochastic) models using various approaches. Mahdian et al. (2007) and Mirrokni et al. (2012) considered allocation problems where the demand can *either* be perfectly estimated or adversarial. They designed and analyzed algorithms that have good performance guarantees in both worst-case and average-case scenarios. Unlike these works, our demand model contains *both* stochastic and adversarial components at the same time; and we design algorithms that take advantage of partial predictability.

Another approach to address unpredictable patterns in demand is to use robust stochastic optimization (Ben-Tal and Nemirovski 2002, Bertsimas et al. 2004). These papers aim to optimize allocations when the demand belongs to a class of distributions (or uncertainty set). This approach limits the adversary's power by restricting the class of demand distributions. Here, we take a different approach. We do not limit the class of distribution that the adversary can choose from; instead, we assume that a fraction p of the demand will not follow the adversary.

Lan et al. (2008) also took a robust approach, studying the single-leg multifare class revenue management problem in a very interesting setting, where the only prior knowledge about demand is the lower and upper bounds on the number of customers from each fare class. Lan et al. (2008) used fixed upper and lower bounds to develop optimal static policies in the form of nested booking limits and also showed that dynamically adjusting these policies can improve the competitive ratio. Unlike their work, we do not assume prior knowledge of lower and upper bounds on the number of customers from each class. Instead, in our model, we learn the bounds as the sequence unfolds.

Shamsi et al. (2014) used a real data set from display advertising at AOL/Advertising.com to show that arrival patterns do not satisfy the crucial property implied by assuming a random order model for demand. In particular, they showed that the dual prices of the offline allocation problem at different times can vary significantly. They used a risk minimization framework to devise allocation rules that outperform existing algorithms when applied to AOL data. Even though the results are practically promising, the paper provides no performance guarantee nor does it offer insights on how to model traffic in practice.

Further, Esfandiari et al. (2015) also considered a hybrid arrival model where the input comprises known stochastic i.i.d. demand and an unknown number of arrivals that are chosen by an adversary (which is motivated by traffic spikes). They do not assume any knowledge of the traffic spikes; but the performance guarantee of their algorithm is parameterized by λ, roughly the fraction of the revenue in the optimal solution that is obtained from the stochastic (predictable) part of the demand. Parameter λ plays a similar role as parameter p in our model, in that it controls the adversary's power. However, the underlying arrival processes in these two models differ considerably and cannot be directly compared. In particular, we do not assume any prior knowledge of the stochastic component; instead we partially predict it. However, we do assume that the adversary determines only the initial order of arrivals (i.e., before knowing which customer will eventually follow its order).

Our work is also closely related to the literature on the secretary problem. In the original formulation of the problem, n secretaries with unique values arrive in uniformly random order; the goal is to maximize the probability of hiring the most valuable secretary. The optimal solution to this problem is an observation-selection policy: observe the first n/e secretaries, then select the first one whose value exceeds that of the best of the previously observed secretaries (Lindley 1961, Dynkin 1963, Freeman 1983, Ferguson 1989). Recently, Kesselheim et al. (2015) relaxed the assumption of uniformly random order and analyzed the performance of the above policy under certain classes of nonuniform distribution over permutations. Here, we study the secretary problem in our new arrival model (i.e., only a p fraction of secretaries arrive in uniformly random order) and show that a deterministic observation period is not optimal.

3. Model and Preliminaries

A firm is endowed with b (identical) units of a product to sell over $n > 24$ periods, where $n \geq b$.[5] In each period, at most one customer arrives demanding one unit of the product; customers belong to two types depending on the revenue they generate. Type-1 and type-2 customers generate revenue of 1 and $0 < a < 1$, respectively. Upon the arrival of a customer, the firm observes the type of the customer and must make an irrevocable decision to accept this customer and allocate one unit or to reject this customer. If a firm accepts a type-1 (type-2) customer, it will earn \$1 (\$$a$). Our goal is to devise online allocation algorithms that maximize the firm's revenue. We evaluate the performance of an algorithm by comparing it to the optimum offline solution (i.e., the clairvoyant solution).

Before proceeding with the model, we introduce a few notations and briefly discuss the structure of the problem. We represent each customer by the value of revenue the customer generates if accepted and the sequence of arrival by $\vec{v} = (v_1, v_2, \ldots, v_n)$, where $v_i \in \{0, a, 1\}$; $v_i = 0$ implies that no customer arrives at period i. We denote the number of type-1 (type-2) customers in the entire sequence as n_1 (n_2). Note that the optimum offline solution that we denote by $OPT(\vec{v})$ has the following simple structure: accept all the type-1 customers and if $n_1 < b$, then accept $\min\{n_2, b - n_1\}$ type-2 customers. Therefore,

$$OPT(\vec{v}) = \min\{b, n_1\} + a\min\{n_2, (b - n_1)^+\}, \quad (1)$$

where $(x)^+ \triangleq \max\{x, 0\}$, and we use the symbol "$\triangleq$" for definitions. At each period, a reasonable online algorithm will accept an arriving type-1 customer if there is inventory left. Thus, the main challenge for an online algorithm is to decide whether to accept/reject an arriving type-2 customer facing the following natural trade-off: accepting a type-2 customer may result in rejecting a potential future type-1 customer because of limited inventory; on the other hand, rejecting a type-2 customer may lead to having unused inventory at the end. Therefore, any good online algorithm needs to strike a balance between accepting *too few* and *too many* type-2 customers. We denote by $ALG(\vec{v})$ the revenue obtained by an online algorithm.

Next we introduce our *partially predictable* demand arrival model that works as follows. The adversary determines an initial sequence, which we denote by $\vec{v}_I = (v_{I,1}, v_{I,2}, \ldots, v_{I,n})$, where $v_{I,j} \in \{0, a, 1\}$, for $1 \le j \le n$. However, a subset of customers will not follow this order. We call this subset the *stochastic group*, which we denote by $\mathcal{S}$. Each customer joins the stochastic group independently and with the same probability p. Other customers are in the *adversarial* group denoted by $\mathcal{A}$. Customers in the stochastic group are permuted uniformly at random among themselves. Formally, a permutation $\sigma_{\mathcal{S}} : \mathcal{S} \to \mathcal{S}$ is chosen uniformly at random and determines the order of arrivals among the stochastic group. In the resulting overall arriving sequence, the adversarial group follows the adversarial sequence according to $\vec{v}_I$; and those in the stochastic group follow the random order given by $\sigma_{\mathcal{S}}$. Given $\vec{v}_I$, we denote the *random* customer arrival sequence by $\vec{V} = (V_1, V_2, \ldots, V_n)$ and the *realization* of it by $\vec{v} = (v_1, v_2, \ldots, v_n)$. We highlight that we assume the initial sequence is determined without knowing which customer will belong to the stochastic group. Said differently, first an adversary determines the initial sequence $\vec{v}_I$. Then "nature" decides randomly which customers belong to the stochastic and adversarial groups.

The example presented in Figure 1 illustrates the arrival process. The top row (gray nodes) shows the initial sequence ($\vec{v}_I$). The middle row shows which customers belong to the stochastic group (the black nodes) and which belong to the adversarial group (the white ones). The bottom row shows both the permutation $\sigma_{\mathcal{S}}$ and the actual arrival sequence. In this example, $\mathcal{S} = \{2, 5, 6, 8\}$, and $\sigma_{\mathcal{S}}(2) = 6, \sigma_{\mathcal{S}}(5) = 2$, $\sigma_{\mathcal{S}}(6) = 5$, and $\sigma_{\mathcal{S}}(8) = 8$.

Note that the extreme cases $p = 0$ and $p = 1$ correspond to the adversarial and random order models that have been studied before (e.g., Ball and Queyranne 2009 and Agrawal et al. 2014, respectively). Hereafter, we assume that $0 < p < 1$. For a given $p \in (0, 1)$, at any time over the horizon, we can use the number of past observed type-1 (type-2) customers to obtain bounds on the number of customers of each type to be expected over the rest of the horizon. This idea is formalized later in Subsection 3.2 along with further analysis of our model.

Having described the arrival process, we now define the competitive ratio of an online algorithm under the proposed partially predictable model as follows:

Definition 1. An online algorithm is c-competitive in the proposed partially predictable model if for any adversarial instance $\vec{v}_I$,

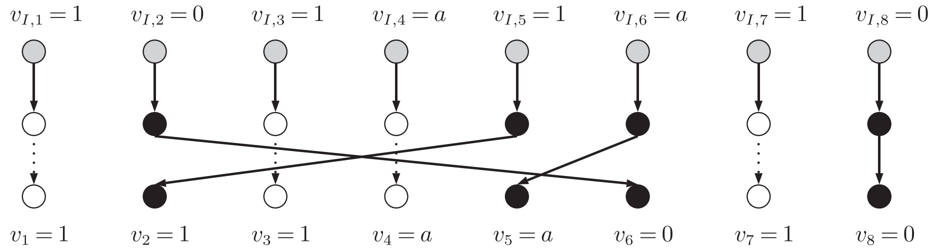
$$\mathbb{E}\Big[ALG\Big(\vec{V}\Big)\Big] \ge cOPT(\vec{v}_I),$$

where the expectation is taken over which customers belong to the stochastic group (i.e., subset $\mathcal{S}$), the choice of the random permutation $\sigma_{\mathcal{S}}$, and any possibly randomized decisions of the online algorithm.

Note that $OPT(\vec{V}) = OPT(\vec{v}_I)$ and, thus, in Definition 1, $\mathbb{E}[ALG(\vec{V})] \ge cOPT(\vec{v}_I)$ is equivalent to $\mathbb{E}[ALG(\vec{V})] \ge c\mathbb{E}[OPT(\vec{V})]$.

In Sections 4 and 5, we present two online algorithms that perform well in the proposed partially predictable model for various ranges of b and n. Before introducing our online algorithms, in the following subsections, we introduce a series of notations used throughout the paper and state a consequential concentration result that will allow us to partially predict future demand using past observed data.

Figure 1. Illustration of the Customer Arrival Model



3.1. Notational Conventions

Throughout the paper, we use uppercase letters for random variables and lowercase ones for realizations. We have already used this convention in defining $\vec{V}$ versus $\vec{v}$. We normalize the time horizon to 1, and represent time steps by $\lambda = 1/n, 2/n, \ldots, 1$. First, we introduce notations related to the random customer arrival sequence $\vec{V}$. At any time step λ, for $j = 1, 2$, the number of type-j customers *to be observed* by the online algorithms up to time λ is denoted by $O_j(\lambda)$. Further, we denote by $O_j^{\mathcal{S}}(\lambda)$ the number of type-j customers in the stochastic group that arrive up to time λ in $\vec{V}$. Note that the online algorithm cannot distinguish between customers in the stochastic group and customers in the adversarial group. Therefore, the online algorithm does *not* observe the realizations of $O_j^{\mathcal{S}}(\lambda)$. We denote realizations of $O_j(\lambda)$ and $O_j^{\mathcal{S}}(\lambda)$ by $o_j(\lambda)$ and $o_j^{\mathcal{S}}(\lambda)$, respectively.

Next, we introduce notations related to the initial adversarial sequence $\vec{v}_I$. As discussed earlier, we denote the total number of type-j customers in $\vec{v}_I$ by n_j. In addition, given the sequence $\vec{v}_I$, we denote the total number of type-j customers among the first λn customers by $\eta_j(\lambda)$. Note that both n_j and $\eta_j(\lambda)$ are deterministic. Also, we define $\tilde{o}_j(\lambda) \triangleq (1-p)\eta_j(\lambda) + p\lambda n_j$ and $\tilde{o}_j^{\mathcal{S}}(\lambda) \triangleq p\lambda n_j$, which will serve as deterministic approximations for $O_j(\lambda)$ and $O_j^{\mathcal{S}}(\lambda)$, respectively (see Lemma 1 and the subsequent discussion for motivation of this definition).

Here we return to the example in Figure 1 and review the notations. Suppose $\lambda = 5/8$ and $p = 0.5$; in this example, looking at the bottom row that shows the sequence $\vec{v}$, we have $o_1(5/8) = 3$, $o_1^{\mathcal{S}}(5/8) = 1$, which are realizations of random variables $O_1(5/8)$ and $O_1^{\mathcal{S}}(5/8)$, respectively. Looking at the top row that shows sequence $\vec{v}_I$, we have $n_1 = 4$, $\eta_1(5/8) = 3$, $\tilde{o}_1(5/8) = 0.5 \times 3 + 0.5 \times 4 \times (5/8) = 2.75$, and $\tilde{o}_1^{\mathcal{S}}(5/8) = 0.5 \times 4 \times (5/8) = 1.25$ that are all deterministic quantities. Similarly, for type-2 customers, $o_2(5/8) = 2$, $o_2^{\mathcal{S}}(5/8) = 1$, $n_2 = 2$, $\eta_2(5/8) = 1$, $\tilde{o}_2(5/8) = 0.5 \times 1 + 0.5 \times 2 \times (5/8) = 1.125$, and $\tilde{o}_2^{\mathcal{S}}(5/8) = 0.5 \times 2 \times (5/8) = 0.625$.

For convenience of reference, in Table 1, we present a summary of the defined notations.

Finally, to avoid carrying cumbersome expressions in the statement of our results for second-order quantities (e.g., bounds on approximation errors), we use the following approximation notations.

Definition 2. Suppose $f, g : \mathcal{X} \to \mathbb{R}$ are two functions defined on set $\mathcal{X}$. We use the notation $f = O(g)$ if there exists a constant k such that $f(x) < kg(x)$ for all $x \in \mathcal{X}$.

Definition 3. Suppose $f, g : \mathbb{N} \to \mathbb{R}$ are two functions defined on natural numbers. We use the notation $f = o(g)$ if $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$ and the notation $f = \omega(g)$ if $\lim_{n\to\infty} \left|\frac{f(n)}{g(n)}\right| = \infty$.

Table 1. Notations

$\vec{v}_I$	$\vec{v}_I = (v_{I,1}, v_{I,2}, \ldots, v_{I,n})$, initial customer sequence
$\mathcal{S}$	Subset of customers in the stochastic group
$\mathcal{A}$	Subset of customers in the adversarial group
$\vec{V}$	$\vec{V} = (V_1, V_2, \ldots, V_n)$, random customer arrival sequence
$\vec{v}$	$\vec{v} = (v_1, v_2, \ldots, v_n)$, a realization of $\vec{V}$ (what online algorithm actually observes)
n_j	Number of type-j, $j = 1, 2$, customers in $\vec{v}_I$ (which is the same as in $\vec{V}$)
λ	Normalized time: $\lambda = 1/n, \ldots, 1$
$O_j(\lambda)$	Random number of type-j customers arriving up to time λ
$o_j(\lambda)$	A realization of $O_j(\lambda)$
$O_j^{\mathcal{S}}(\lambda)$	Random number of type-j customers in $\mathcal{S}$ arriving up to time λ
$o_j^{\mathcal{S}}(\lambda)$	A realization of $O_j^{\mathcal{S}}(\lambda)$
$\eta_j(\lambda)$	Number of type-j, $j = 1, 2$, customers among the first λn ones in $\vec{v}_I$
$\tilde{o}_j(\lambda)$	$(1-p)\eta_j(\lambda) + p\lambda n_j$ (a deterministic approximation of $O_j(\lambda)$)
$\tilde{o}_j^{\mathcal{S}}(\lambda)$	$p\lambda n_j$ (a deterministic approximation of $O_j^{\mathcal{S}}(\lambda)$)

3.2. Estimating Future Demand

At time $\lambda < 1$, upon observing $o_j(\lambda)$, $j = 1, 2$ (but not n_j and $\eta_j(\lambda)$), we wish to estimate future demand, or equivalently the total demand n_j. To make such an estimation, we establish the following concentration result:

Lemma 1. *Define constants* $\alpha \triangleq 10 + 2\sqrt{6}$, $\bar{\epsilon} \triangleq 1/24$, *and* $k \triangleq 16$. *For any* $\epsilon \in [\frac{1}{n}, \bar{\epsilon}]$, *with probability at least* $1 - \epsilon$, *all the following statements hold:*

- *If* $n_1 \geq \frac{k}{p^2} \log n$, *then for all* $\lambda \in \{0, 1/n, 2/n, \ldots, 1\}$,

$$|O_1(\lambda) - \tilde{o}_1(\lambda)| < \alpha\sqrt{n_1 \log n}, \text{ and} \tag{2a}$$

$$|O_1(\lambda) + O_2(\lambda) - (\tilde{o}_1(\lambda) + \tilde{o}_2(\lambda))| < \alpha\sqrt{(n_1 + n_2)\log n} \tag{2b}$$

- *If* $n_2 \geq \frac{k}{p^2} \log n$, *then for all* $\lambda \in \{0, 1/n, 2/n, \ldots, 1\}$,

$$|O_2(\lambda) - \tilde{o}_2(\lambda)| < \alpha\sqrt{n_2 \log n}, \text{ and} \tag{3a}$$

$$\left|O_2^{\mathcal{S}}(\lambda) - \tilde{o}_2^{\mathcal{S}}(\lambda)\right| < \alpha\sqrt{n_2 \log n}. \tag{3b}$$

The lemma is proven in Online Appendix EC.2. Given that there are two layers of randomization (selection of subset $\mathcal{S}$ and the random permutation), proving the above concentration results requires a fairly delicate analysis that builds upon several existing concentration bounds. Because proving concentration results is not the main focus of our work, we will not outline the proof in the main text and refer the interested reader to Online Appendix EC.2.[6] Here we focus on the following two questions: (i) What is our motivation for using deterministic approximations $\tilde{o}_j(\lambda)$ and $\tilde{o}_j^{\mathcal{S}}(\lambda)$? (ii) How do such approximations help us to estimate n_j?

To answer the first question, let us count the number of type-j customers in $O_j(\lambda)$ that belong to the stochastic and adversarial groups separately. We start with the stochastic group. Roughly, a total of pn_j type-j customers belong to the stochastic group, and a λ fraction of them arrive by time λ because these customers are spread almost uniformly over the entire time horizon. As a result, there are approximately $pn_j\lambda$ type-j customers from $\mathcal{S}$ arriving up to time λ. Now we move on to the adversarial group: there are a total of $\eta_j(\lambda)$ of type-j customers in the first λn customers in $\vec{v}_I$. Because with probability $1-p$ each of them will be in the adversarial group, the total number of type-j customers from the adversarial group arriving up to time λ is approximately $(1-p)\eta_j(\lambda)$. Combining these two approximate counting arguments gives us

$$O_j(\lambda) \approx \big(1-p\big)\eta_j(\lambda) + p\lambda n_j = \tilde{o}_j(\lambda). \tag{4}$$

A similar argument shows that $O_j^{\mathcal{S}}(\lambda) \approx p\lambda n_j = \tilde{o}_j^{\mathcal{S}}(\lambda)$. Lemma 1 confirms that these approximations hold with high probability. Lemma 1 also provides upper bounds on the corresponding approximation errors. Further, we note that $\tilde{o}_j(\lambda) \neq \mathbb{E}[O_j(\lambda)]$, as shown in Online Appendix EC.2.1. However, the difference between the two is very small and vanishing in n. Given that $\tilde{o}_j(\lambda)$ provides a very intuitive deterministic approximation for random variable $O_j(\lambda)$ and admits a simple closed-form expression, we use it instead of the $\mathbb{E}[O_j(\lambda)]$.

Now, let us answer the second posed question. There are simple relations between n_j and $\eta_j(\lambda)$, such as $n_j \geq \eta_j(\lambda)$ and $\eta_j(\lambda) + (1-\lambda)n \geq n_j$.[7] Combining these with our deterministic approximations leads us to compute upper bounds on the total number of customers as established in a lemma in Section 5.1.

Finally, based on Lemma 1, we partition the sample space of arriving sequences into two subsets, $\mathcal{E}$ and its complement $\bar{\mathcal{E}}$, and define event $\mathcal{E}$ as follows:

Definition 4. Given the initial sequence $\vec{v}_I$, event $\mathcal{E}$ occurs if the realized arrival sequence $\vec{v}$ satisfies all the conditions of Lemma 1, that is,

- If $n_1 \geq \frac{k}{p^2}\log n$, then for all $\lambda \in \{0, 1/n, 2/n, \ldots, 1\}$,

$$|o_1(\lambda) - \tilde{o}_1(\lambda)| < \alpha\sqrt{n_1 \log n} \quad \text{and}$$
$$|o_1(\lambda) + o_2(\lambda) - (\tilde{o}_1(\lambda) + \tilde{o}_2(\lambda))| < \alpha\sqrt{(n_1+n_2)\log n},$$

- If $n_2 \geq \frac{k}{p^2}\log n$, then for all $\lambda \in \{0, 1/n, 2/n, \ldots, 1\}$,

$$|o_2(\lambda) - \tilde{o}_2(\lambda)| < \alpha\sqrt{n_2 \log n} \quad \text{and}$$
$$\big|o_2^{\mathcal{S}}(\lambda) - \tilde{o}_2^{\mathcal{S}}(\lambda)\big| < \alpha\sqrt{n_2 \log n}.$$

If $n_1 < \frac{k}{p^2}\log n$ and $n_2 < \frac{k}{p^2}\log n$, then event $\mathcal{E}$ occurs as well.

Lemma 1 confirms that event $\mathcal{E}$ occurs *with high probability*. In all our analyses, we use the above definition to focus on the event that the deterministic approximations (i.e., $\tilde{o}_j(\lambda)$) are in fact "very close" to the observed sequence. This greatly helps us simplify the analysis and its presentation. We conclude this section by remarking that in Online Appendix EC.3, we discuss how to use Lemma 1 to estimate parameter p when n_1 and n_2 are large enough and the firm has access to multiple past runs of the arrival process.

4. A Nonadaptive Algorithm

In this section, we present and analyze our first online algorithm for the resource allocation problem and the demand model described in Section 3. First, in Section 4.1, we describe the algorithm. Then, in Section 4.2, we present the analysis of its competitive ratio.

4.1. The Algorithm

Our first algorithm is a nonadaptive online algorithm that uses predetermined dynamic thresholds to accept or reject customers. This algorithm combines some ideas from the primal algorithm of Kesselheim et al. (2014) and the threshold algorithm of Ball and Queyranne (2009) to generate maximal revenue from both the stochastic and adversarial components of the demand.

In particular, our nonadaptive algorithm makes use of the fact that customers from the stochastic group are uniformly spread over the entire horizon. Therefore, at least a fraction p of the inventory should be allocated at a roughly constant rate. To this end, we define an *evolving threshold* that works as follows: at any time λ, accept a type-2 customer if the total number of accepted customers by this rule does not exceed $\lfloor \lambda pb \rfloor$.

However, the arrival pattern of the other $1-p$ fraction can take any arbitrary form. In particular, if the adversary puts many type-2 customers at the very beginning of the time horizon but none toward the end, then we may reject too many type-2 customers early on. To prevent this loss, we keep another quota for a type-2 customer rejected by the evolving threshold. We only reject that customer if the number of such type-2 customers accepted so far exceeds the *fixed* threshold of $\theta \triangleq \frac{1-p}{2-a}$. When $p = 0$, this is the same threshold as in Ball and Queyranne (2009).

The formal definition of our algorithm is presented in Algorithm 1. Note that q_1, $q_{2,e}$, and $q_{2,f}$, respectively, represent counters for the number of accepted type-1 customers, the number of type-2 customers accepted by the evolving threshold, and the number of type-2 customers accepted by the fixed threshold.

Algorithm 1 (Online Nonadaptive Algorithm (ALG_1))

1. Initialize $q_1, q_{2,e}, q_{2,f} \leftarrow 0$, and define $\theta \triangleq \frac{1-p}{2-a}$.
2. Repeat for time $\lambda = 1/n, 2/n, \ldots, 1$, accept customer $i = \lambda n$ arriving at time λ if there is remaining inventory and one of the following conditions holds:
   a. $v_i = 1$; update $q_1 \leftarrow q_1 + 1$.
   b. *Evolving threshold rule:* $v_i = a$ and $q_1 + q_{2,e} < \lfloor \lambda pb \rfloor$; update $q_{2,e} \leftarrow q_{2,e} + 1$.
   c. *Fixed threshold rule:* $v_i = a$ and $q_{2,f} < \lfloor \theta b \rfloor$; update $q_{2,f} \leftarrow q_{2,f} + 1$.

We prioritize the evolving threshold rule if both of the last two conditions are satisfied.

4.2. Competitive Analysis

In this subsection, we analyze the competitive ratio of Algorithm 1. Our main result is the following theorem:

Theorem 1. *For $p \in (0,1)$, the competitive ratio of Algorithm 1 is at least $p + \frac{1-p}{2-a} - O(\frac{1}{a(1-p)p}\sqrt{\frac{\log n}{b}})$ in the partially predictable model.*

Before proceeding to the proof of Theorem 1, we make the following remarks:

Remark 1. When $p = 0$, our model reduces to an adversarial model. This setting was studied in Ball and Queyranne (2009), which developed a nonadaptive algorithm with competitive ratio of $\frac{1}{2-a}$. Further they show this is the best possible competitive ratio for this setting. On the other hand, when $p = 1$, our model reduces to the so-called random ordered model studied in several papers, including Agrawal et al. (2014) among others. The online algorithm of Agrawal et al. (2014) is learning based and, thus, it is adaptive. The competitive ratio of their algorithms—when applied to our setting—is $1 - O(\sqrt{\log n/b})$.[8]

Remark 2. Our competitive analysis of Algorithm 1 is tight (up to an $O(\sqrt{\frac{\log n}{b}})$ term). In particular, for the following instance, Algorithm 1 can attain only a $p + \frac{1-p}{2-a}$ fraction of the optimum offline solution: Suppose $b = n$ and all customers are of type-2. The revenue of the optimum offline algorithm is ab. On the other hand, if we employ Algorithm 1, at the end we will have $q_1 = 0$, $q_{2,e} \le pb$, and $q_{2,f} \le \theta b$. This results in a competitive ratio of at most $p + \theta = p + \frac{1-p}{2-a}$.

Remark 3. In Subsection 6.1, we prove that no online algorithm can have a competitive ratio larger than $p + \frac{1-p}{2-a} + o(1)$ when $b = o(\sqrt{n})$. On the other hand, Theorem 1 indicates that Algorithm 1 achieves a competitive ratio of $p + \frac{1-p}{2-a} - o(1)$ when $b = \omega(\log n)$. Combining the two results implies that for fixed a and p, Algorithm 1 achieves the best possible competitive ratio (up to an $o(1)$ term) in the regime where conditions $b = \omega(\log n)$ and $b = o(\sqrt{n})$ hold simultaneously.

Remark 4. Note that even though $p + \frac{1-p}{2-a}$ is the convex combination of the competitive ratios of Ball and Queyranne (2009) and of Agrawal et al. (2014), it cannot be achieved by simply randomizing between these two algorithms. Suppose we flip a biased coin; with probability p, we follow the algorithm of Agrawal et al. (2014) (or any other algorithms designed for a random order model, such as Kesselheim et al. (2014)); and with probability $(1-p)$, we follow the fixed threshold algorithm of Ball and Queyranne (2009). In Subsection 6.2, we show that for a certain class of instances, such a randomized algorithm does not generate $p + \frac{1-p}{2-a}$ fraction of the optimum offline solution.

Proof of Theorem 1. We start the proof by making the following observation: Theorem 1 is nontrivial only if $\sqrt{\frac{\log n}{b}}$ is small enough, such that the approximation term $O(\cdot)$ is negligible. Therefore, without loss of generality, we can restrict attention to the case where $\sqrt{\frac{\log n}{b}}$ is small. In particular, recalling that we defined constant $\bar{\epsilon} = 1/24$ in Lemma 1, if $\frac{1}{a(1-p)p}\sqrt{\frac{\log n}{b}} \ge \bar{\epsilon}$, then $O(\frac{1}{a(1-p)p}\sqrt{\frac{\log n}{b}})$ becomes $O(1)$ and Theorem 1 becomes trivial. Therefore, without loss of generality, we assume $\frac{1}{a(1-p)p}\sqrt{\frac{\log n}{b}} < \bar{\epsilon}$, or equivalently,

$$b > \frac{1}{\bar{\epsilon}^2}\frac{\log n}{a^2(1-p)^2p^2}. \tag{5}$$

We denote the random revenue generated by Algorithm 1 by $ALG_1(\vec{V})$. To analyze $\mathbb{E}[ALG_1(\vec{V})]$, we condition it on the event $\mathcal{E}$. Thus, we have

$$\frac{\mathbb{E}\left[ALG_1\left(\vec{V}\right)\right]}{OPT(\vec{v}_I)} \ge \frac{\mathbb{E}\left[ALG_1\left(\vec{V}\right)|\mathcal{E}\right]\mathbb{P}\,\mathcal{E}}{OPT(\vec{v}_I)}.$$

Define $\epsilon \triangleq \frac{1}{a(1-p)p}\sqrt{\frac{\log n}{b}}$. For b that satisfies condition (5), and assuming that $n > 24$, we have $\frac{1}{n} \le \epsilon \le \bar{\epsilon}$. Therefore, we can apply Lemma 1 to get

$$\frac{\mathbb{E}\left[ALG_1\left(\vec{V}\right)\right]}{OPT(\vec{v}_I)} \ge \frac{\mathbb{E}\left[ALG_1\left(\vec{V}\right)|\mathcal{E}\right]\mathbb{P}\,\mathcal{E}}{OPT(\vec{v}_I)} \ge \frac{\mathbb{E}\left[ALG_1\left(\vec{V}\right)|\mathcal{E}\right]}{OPT(\vec{v}_I)}(1-\epsilon).$$

This will allow us to focus on the realizations that belong to event $\mathcal{E}$. In the main part of the proof, we show that, for any realization $\vec{v}$ belonging to event $\mathcal{E}$,

$$\frac{ALG_1(\vec{v})}{OPT(\vec{v}_I)} \geq p + \frac{1-p}{2-a} - O(\epsilon).$$

Fixing a realization $\vec{v}$ that belongs to event $\mathcal{E}$, we define $q_1(\lambda)$, $q_{2,e}(\lambda)$, and $q_{2,f}(\lambda)$ to be the values of counters q_1, $q_{2,e}$, and $q_{2,f}$ right after the algorithm determines whether to accept the customer arriving at time λ. Further, we define $\Delta \triangleq \alpha\sqrt{b\log n}$ (constant α is defined in Lemma 1). To analyze the competitive ratio, we analyze three cases separately.

Case (i). $n_1 \geq \frac{k}{p^2}\log n$, and Algorithm 1 exhausts the inventory.

Note that when $n_1 \geq \frac{k}{p^2}\log n$, we can apply the concentration result (2a) from Lemma 1. When Algorithm 1 exhausts the inventory, it is possible that the algorithm accepts *too many* type-2 customers, which results in rejecting type-1 customers and losing revenue. We control for this loss by establishing the following upper bound on the number of type-2 customers accepted by the evolving threshold.[9] In particular, we have the following lemma:

Lemma 2. *Under event $\mathcal{E}$, if $n_1 \geq \frac{k}{p^2}\log n$, then*

$$q_{2,e}(1) \leq p(b-n_1)^+ + \Delta.$$

Proof. We assume, without loss of generality, that $n_1 \leq b$. Otherwise, we construct a modified adversarial instance, denoted by $\vec{v}_{I,M}$, as follows: keep an arbitrary subset of type-1 customers with size b in $\vec{v}_I$ (before the random permutation), and remove the remaining type-1 customers (e.g., set their revenue to be zero). For the same realization of the stochastic group and random permutation, we claim that at any time $\lambda \in \{1/n, \dots, 1\}$, the number of type-2 customers accepted through the evolving threshold rule in the original instance is not larger than that in the modified one. This holds because $o_1(\lambda, \vec{v}) \geq o_1(\lambda, \vec{v}_M)$, where the second argument is added to $o_1(\cdot,\cdot)$ to indicate the corresponding instance. Note that because the algorithm accepts all type-1 customers, this implies $q_1(\lambda, \vec{v}) \geq q_1(\lambda, \vec{v}_M)$, which proves our claim (i.e., $q_{2,e}(\lambda, \vec{v}) \leq q_{2,e}(\lambda, \vec{v}_M)$). Thus, without loss of generality, we assume $n_1 \leq b$. Further, note that because of condition (5), we have $n_1(\vec{v}_M) = b \geq \frac{k}{p^2}\log n$.[10] Thus, we are still in Case (i) for the modified instance.

If no type-2 customer is accepted by the evolving threshold, then $q_{2,e}(1) = 0$ and the proof is complete. Otherwise, let $\bar{\lambda} \leq 1$ be the last time that a type-2 customer is accepted by the evolving rule. Then we have

$$\begin{aligned} q_{2,e}(1) = q_{2,e}(\bar{\lambda}) &\leq \bar{\lambda}pb - o_1(\bar{\lambda}) && \text{(Evolving threshold rule)} \\ &\leq \bar{\lambda}pb - (\bar{\lambda}pn_1 + (1-p)\eta_1(\bar{\lambda}) - \Delta) && ((2a)) \\ &\leq p(b-n_1) + \Delta. && (\eta_1(\bar{\lambda}) \geq 0, n_1 \leq b, \text{ and } \bar{\lambda} \leq 1) \end{aligned}$$

The reason for each inequality appears in the same line. We remark that in the second inequality, we crucially use the concentration result of Lemma 1. This completes the proof.

Using Lemma 2, we prove, in Online Appendix EC.4, the following lemma that gives a lower bound on the competitive ratio for Case (i):

Lemma 3. *Under event $\mathcal{E}$, if $n_1 \geq \frac{k}{p^2}\log n$ and $q_1(1) + q_{2,e}(1) + q_{2,f}(1) = b$, then $\frac{ALG_1(\vec{v})}{OPT(\vec{v})} \geq p + \frac{1-p}{2-a} - \frac{(1-a)\Delta}{ab}$.*

Case (ii). *$n_1 \geq \frac{k}{p^2}\log n$, and Algorithm 1 does not exhaust the inventory.*

In this case, all type-1 customers are accepted. Therefore, the ratio between $ALG_1(\vec{v})$ and $OPT(\vec{v})$ can be expressed as

$$\frac{ALG_1(\vec{v})}{OPT(\vec{v})} = \frac{n_1 + a[q_{2,e}(1) + q_{2,f}(1)]}{n_1 + a\min\{n_2, (b-n_1)\}}.$$

The only "mistake" that the algorithm may make is to reject too many type-2 customers. The following lemma establishes a lower bound on the number of accepted type-2 customers:

Lemma 4. *Under event $\mathcal{E}$, if $n_1 \geq \frac{k}{p^2}\log n$ and $q_1(1) + q_{2,e}(1) + q_{2,f}(1) < b$, then one of the following conditions holds:*

a. $q_{2,e}(1) + q_{2,f}(1) = n_2$,
b. $q_{2,f}(1) = \lfloor\theta b\rfloor$ *and* $n_1 > bp - 3\Delta$, *or*
c. $q_{2,f}(1) = \lfloor\theta b\rfloor$, $n_1 \leq bp - 3\Delta$, *and* $q_{2,e}(1) \geq (p(n_1 + n_2) - n_1 - 5\Delta)^+$.

Proof. First note that $q_{2,f}(1) < \lfloor\theta b\rfloor$ means that Algorithm 1 never rejects a type-2 customer. This implies that $q_{2,e}(1) + q_{2,f}(1) = n_2$, that is, condition (a) holds. Now suppose $q_{2,f}(1) = \lfloor\theta b\rfloor$. If $n_1 > bp - 3\Delta$, then condition (b) holds. The most interesting case is when $q_{2,f}(1) = \lfloor\theta b\rfloor$ and $n_1 \leq bp - 3\Delta$. In the following, we show that in this case, condition (c) will hold.

In this case, without loss of generality, we can assume $n_1 + n_2 \leq b$. Otherwise, we construct an alternative adversarial instance, denoted by $\vec{v}_{I,A}$, as follows: keep an arbitrary subset of type-2 customers with size $b - n_1$ in $\vec{v}_I$ (before the random permutation) and remove the remaining type-2 customers (e.g., set their revenue to be zero). With the same realization of the stochastic group and random permutation, we claim that

$$q_{2,e}(\lambda,\vec{v}) \geq q_{2,e}(\lambda,\vec{v}_A), \quad \lambda \in \{0, 1/n, \ldots, 1\}. \tag{6}$$

To show (6), we use induction. The base case, corresponding to taking $\lambda = 0$, is trivial. Suppose (6) holds for $\lambda - 1/n$. We show it will hold for λ as well. At time λ, if $q_{2,e}(\lambda,\vec{v}_A) = q_{2,e}(\lambda - 1/n,\vec{v}_A)$, then (6) holds because $q_{2,e}(\lambda,\vec{v}) \geq q_{2,e}(\lambda - 1/n,\vec{v})$. Otherwise, $q_{2,e}(\lambda,\vec{v}_A) = q_{2,e}(\lambda - 1/n,\vec{v}_A) + 1$. This implies that a type-2 customer arrives at time λ in $\vec{v}_A$, and thus also in $\vec{v}$. If $q_{2,e}(\lambda,\vec{v}) = q_{2,e}(\lambda - 1/n,\vec{v}) + 1$, then (6) again holds. Otherwise, under customer arrival sequence $\vec{v}$, we do not accept the type-2 customer at time λ by the evolving threshold rule, which means that $o_1(\lambda,\vec{v}) + q_{2,e}(\lambda,\vec{v}) = \lfloor \lambda pb \rfloor$. Because $o_1(\lambda,\vec{v}_A) + q_{2,e}(\lambda,\vec{v}_A) \leq \lfloor \lambda pb \rfloor$, and $o_1(\lambda,\vec{v}) = o_1(\lambda,\vec{v}_A)$, we can conclude that (6) holds in the last case as well. This concludes the induction. Thus, without loss of generality, we assume $n_1 + n_2 \leq b$.

To prove that condition (c) holds when $q_{2,f}(1) = \lfloor \theta b \rfloor$ and $n_1 \leq bp - 3\Delta$, we make two important observations: (i) In this case, the number of type-2 customers is large enough to apply the concentration results of (3b). In particular, we have

$$n_2 \geq \theta b \geq \frac{k \log n}{p^2}, \tag{7}$$

where the last inequality holds because of (5), and definitions of $\theta = \frac{1-p}{2-a}$ and k (defined in Lemma 1). (ii) The number of type-1 customers is so small that after a certain time the evolving threshold accepts a sufficient number of type-2 customers that ensures condition (c) holds. In particular, define

$$\bar{\lambda} \triangleq \frac{1}{n}\left\lceil \frac{n(n_1(1-p) + 3\Delta)}{p(b - n_1)} \right\rceil.$$

Note that $\bar{\lambda} \leq 1$ when $n_1 \leq bp - 3\Delta$. For any $\lambda \geq \bar{\lambda}$, we have

$$\begin{aligned}
&o_1(\lambda) + o_2^S(\lambda) - o_2^S(\bar{\lambda}) \\
&\quad \leq \lambda p n_1 + (1-p)\eta_1(\lambda) + \Delta + \lambda p n_2 + \Delta - (\bar{\lambda} p n_2 - \Delta) \\
&\quad\quad ((2a), (3b)) \\
&\quad \leq \lambda p n_1 + (1-p) n_1 + (\lambda - \bar{\lambda}) p n_2 + 3\Delta \\
&\quad\quad (\eta_1(\lambda) \leq n_1) \\
&\quad = \bar{\lambda} p n_1 + (1-p) n_1 + (\lambda - \bar{\lambda}) p (n_1 + n_2) + 3\Delta \\
&\quad \leq \bar{\lambda} p n_1 + (1-p) n_1 + (\lambda - \bar{\lambda}) p b + 3\Delta \\
&\quad\quad (n_1 + n_2 \leq b) \\
&\quad \leq \lambda p b. \\
&\quad\quad (\text{definition of } \bar{\lambda}).
\end{aligned}$$

Note that because $o_1(\lambda) + o_2^S(\lambda) - o_2^S(\bar{\lambda})$ is an integer, the above inequality also implies

$$o_1(\lambda) + o_2^S(\lambda) - o_2^S(\bar{\lambda}) \leq \lfloor \lambda pb \rfloor \qquad \text{for all } \lambda \geq \bar{\lambda}. \tag{8}$$

Further, the above inequality implies that for $\lambda \geq \bar{\lambda}$, there is a gap between $o_1(\lambda)$ and the evolving threshold $\lfloor \lambda pb \rfloor$, which in turn implies that the evolving threshold will accept type-2 customers. Next, for $\lambda \geq \bar{\lambda}$, we establish a lower bound on the number of type-2 customers that the evolving threshold accepts. In particular, we show that

$$q_{2,e}(\lambda) \geq o_2^S(\lambda) - o_2^S(\bar{\lambda}) \qquad \text{for all } \lambda \geq \bar{\lambda}. \tag{9}$$

We show (9) by induction. The base case $\lambda = \bar{\lambda}$ is trivial. Suppose (9) holds for $\lambda - 1/n \geq \bar{\lambda}$. We show it will also hold for λ: If the arriving customer is not a type-2 customer belonging to the stochastic group, then $o_2^S(\lambda) = o_2^S(\lambda - 1/n)$; but $q_{2,e}(\lambda) \geq q_{2,e}(\lambda - 1/n)$, and thus (9) holds. Otherwise, we have $o_2^S(\lambda) = o_2^S(\lambda - 1/n) + 1$. Now if this customer is accepted by the evolving threshold rule, then both sides of (9) are increased by one, and thus inequality (9) still holds. Otherwise, if the customer is not accepted, it implies we have reached the threshold. Therefore,

$$q_{2,e}(\lambda) = \lfloor \lambda pb \rfloor - o_1(\lambda). \tag{10}$$

Now we utilize the gap between $\lfloor \lambda pb \rfloor$ and $o_1(\lambda)$ that we established above in (8). Combining (10) and (8) proves that (9) holds in this case as well. This completes the induction and thus the proof of (9).

We complete the proof of the lemma by using (9) with $\lambda = 1$, to have the following lower bound:

$$\begin{aligned}
q_{2,e}(1) &\geq o_2^S(1) - o_2^S(\bar{\lambda}) && ((9)) \\
&\geq pn_2 - \Delta - (\bar{\lambda} p n_2 + \Delta) && ((3b)) \\
&\geq pn_2 - (n_1(1-p) + 3\Delta) - 2\Delta && ((b - n_1 \geq n_2)) \\
&= p(n_1 + n_2) - n_1 - 5\Delta.
\end{aligned}$$

This completes the proof.

Using Lemma 4, we prove, in Online Appendix EC.4, the following lemma that gives a lower bound on the competitive ratio for Case (ii):

Lemma 5. *Under event $\mathcal{E}$, if $n_1 \geq \frac{k}{p^2} \log n$ and $q_1(1) + q_{2,e}(1) + q_{2,f}(1) < b$, then*

$$\frac{ALG_1(\vec{v})}{OPT(\vec{v})} \geq p + \frac{1-p}{2-a} - \frac{5\Delta}{\theta b}.$$

Case (iii). $n_1 < \frac{k}{p^2} \log n$.

The competitive ratio analysis for Case (iii) is fairly similar to that for Case (ii). It follows from the next two lemmas. The proofs are deferred to Online Appendix EC.4.

Lemma 6. *Under event $\mathcal{E}$, if $n_1 < \frac{k}{p^2}\log n$, then one of the following conditions holds:*

a. $q_1(1) + q_{2,e}(1) + q_{2,f}(1) = b$,

b. $q_1(1) = n_1$ *and* $q_{2,e}(1) + q_{2,f}(1) = n_2$, *or*

c. $q_1(1) = n_1, q_{2,f}(1) = \lfloor \theta b \rfloor$ *and* $q_{2,e}(1) \geq pn_2 - \frac{k}{p^2}\log n - 4\Delta$.

Using Lemma 6, the following lemma (proven in Online Appendix EC.4) gives a lower bound on the competitive ratio for Case (iii):

Lemma 7. *Under event $\mathcal{E}$, if $n_1 < \frac{k}{p^2}\log n$, then*

$$\frac{ALG_1(\vec{v})}{OPT(\vec{v})} = \frac{n_1 + a[q_{2,e}(1) + q_{2,f}(1)]}{n_1 + a\min\{n_2, (b - n_1)\}} \geq \min\left\{p + \frac{1-p}{2-a} - \frac{\frac{k}{p^2}\log n}{ab}, p + \frac{1-p}{2-a} - \frac{\frac{k}{p^2}\log n + 4\Delta}{\theta b}\right\}.$$

Using Lemmas 3, 5, and 7, we have lower bounds on the competitive ratio of Algorithm 1 for all possible cases. We complete the proof of the theorem by the following lemma (proven in Online Appendix EC.4) that ensures that the error terms in Lemmas 3, 5, and 7 are $O(\epsilon)$.

Lemma 8. *The error terms in Lemmas 3, 5, and 7 are $O(\epsilon)$, that is, we have (a) $\frac{(1-a)\Delta}{ab} = O(\epsilon)$, (b) $\frac{5\Delta}{\theta b} = O(\epsilon)$, (c) $\frac{\frac{k}{p^2}\log n}{ab} = O(\epsilon)$, and (d) $\frac{\frac{k}{p^2}\log n + 4\Delta}{\theta b} = O(\epsilon)$.*

This completes the proof of Theorem 1.

5. The Adaptive Algorithm

In the design of Algorithm 1, we used the observation that in the partially predictable model, the demand has a stochastic component that is uniformly spread over the entire horizon. This observation motivated us to define the evolving threshold rule. We remark that in Algorithm 1, neither the evolving threshold rule nor the fixed threshold rule adapts to the observed data, which makes Algorithm 1 a nonadaptive algorithm. As noted in Remark 4, when the initial inventory b is small compared with the horizon n, the competitive ratio of Algorithm 1, $p + \frac{1-p}{2-a}$, is in fact the best possible and can be achieved with our nonadaptive algorithm. Therefore, in this regime, adapting to the data, that is, setting thresholds based on the observed data, would not improve the performance. More precisely, when $b = o(\sqrt{n})$, the inventory is so small compared with the time horizon that there may not be enough time to effectively adapt to the observed data. The adversary can mislead us to allocate all the inventory before we can observe a sufficient portion of the data. However, as b becomes larger, we will have more chance to observe and adapt to the data before allocating a significant part of the inventory. In this section, in fact, we design an adaptive algorithm that achieves a better competitive ratio for large enough b (relative to n). In Section 5.1, we first present the ideas behind our adaptive algorithm along with its formal description. Then, in Section 5.2, we analyze the competitive ratio of our algorithm.

5.1. The Algorithm

In this section, we describe our adaptive algorithm, denoted by $ALG_{2,c}$, which takes $c \in [0,1]$ as a parameter. For a certain range of c, we show that $ALG_{2,c}$ attains a competitive ratio of c (up to an error term); however, if c becomes too large (for example if $c = 1$), then $ALG_{2,c}$ no longer guarantees a c fraction of the optimum offline solution. We call this algorithm adaptive because it makes decisions based on the sequence of arrivals it has observed so far. In particular, this algorithm repeatedly computes upper bounds on the total number of type-1/-2 customers based on the observed data and uses these upper bounds to decide whether to accept an arriving type-2 customer or not. Before proceeding with the algorithm, we first introduce two functions, $u_1(\lambda)$ and $u_{1,2}(\lambda)$, that will prove useful in constructing the aforementioned upper bounds. In particular we define

$$u_1(\lambda) \triangleq \begin{cases} b & \text{if } \lambda < \delta \text{ (not enough data observed).} \\ \min\left\{\frac{o_1(\lambda)}{\lambda p}, \frac{o_1(\lambda) + (1-\lambda)(1-p)n}{1-p+\lambda p}\right\} & \text{if } \lambda \geq \delta. \end{cases}$$

$$u_{1,2}(\lambda) \triangleq \begin{cases} b & \text{if } \lambda < \delta \text{ (not enough data observed).} \\ \min\left\{\frac{o_1(\lambda)+o_2(\lambda)}{\lambda p}, \frac{o_1(\lambda)+o_2(\lambda)+(1-\lambda)(1-p)n}{1-p+\lambda p}\right\} & \text{if } \lambda \geq \delta, \end{cases}$$

where $\delta \triangleq \frac{(1-c)b}{(1-a)n}$. Note that $u_1(\lambda)$ and $u_{1,2}(\lambda)$ are functions of the observed data $o_1(\lambda)$ and $o_2(\lambda)$. In the following lemma, we show how $u_1(\lambda)$ and $u_{1,2}(\lambda)$ provide upper bounds on n_1 and $n_1 + n_2$ when the realized sequence, $\vec{v}$, belongs to event $\mathcal{E}$ and the number of type-1 customers as well as the initial inventory b are large enough (as specified in the lemma's statement). Recall that we defined Δ to be $\alpha\sqrt{b\log n}$, where the constant α itself is defined in Lemma 1.

Lemma 9. *Under event $\mathcal{E}$, suppose $n_1 \geq \frac{k}{p^2}\log n$ and $b > \left(\frac{1}{\bar{\epsilon}}\frac{n\sqrt{\log n}}{(1-c)^2 ap^{3/2}}\right)^{\frac{2}{3}}$, where constants k and $\bar{\epsilon}$ are defined in Lemma 1. Then for all $\lambda \in \{1/n, 2/n, \ldots, 1\}$,*

$$u_1(\lambda) \geq \min\left\{b, n_1 - \frac{2\Delta}{\delta p}\right\}, \text{ and} \tag{11a}$$

$$u_{1,2}(\lambda) \geq \min\left\{b, n_1 + n_2 - \frac{2\Delta}{\delta p}\right\}. \tag{11b}$$

Lemma 9 is proven in Online Appendix EC.5.

Having defined $u_1(\lambda)$ and $u_{1,2}(\lambda)$, now we describe how the adaptive algorithm determines whether to accept an arriving type-2 customer when there is remaining inventory. In the following, $q_j(\lambda)$, $j = 1,2$ represents the number of type-j customers accepted by the algorithm up to time λ (for a particular realization $\vec{v}$). Suppose the arriving customer at time λ is of type-2. If $u_{1,2}(\lambda) < b$, then we accept the customer, because (11b) implies that the total number of type-1 and type-2 customers will not exceed b (neglecting the error term); thus, we will have extra inventory at the end. On the other hand, if $u_{1,2}(\lambda) \geq b$, we may want to reject this customer to reserve inventory for a future type-1 customer. The decision of whether to accept the customer is based on the following two observations:

Observation 1. If $u_1(\lambda) \geq n_1$, then

$$\begin{aligned} OPT(\vec{v}) &\leq \min\{n_1, b\} + a(b - n_1)^+ \\ &= (1-a)\min\{n_1, b\} + ab \leq \min\{u_1(\lambda), b\} \\ &\times (1-a) + ab. \end{aligned}$$

Observation 2. If we accept the current type-2 customer, then the maximum revenue we can get is $(b - (q_2(\lambda - 1/n) + 1)) + a(q_2(\lambda - 1/n) + 1)$.

To have a competitive ratio of at least c, Observations 1 and 2 motivate us to accept the type-2 customer only if

$$\frac{\big(b - \big(q_2(\lambda - 1/n) + 1\big)\big) + a\big(q_2(\lambda - 1/n) + 1\big)}{\min\{u_1(\lambda), b\}(1-a) + ab} \geq c. \quad (12)$$

After rearranging terms, we get the following threshold for accepting the type-2 customer:

$$q_2(\lambda - 1/n) + 1 \leq \frac{1-c}{1-a} b + c(b - u_1(\lambda))^+. \quad (13)$$

Thus, when $u_{1,2}(\lambda) \geq b$, we use condition (13) to accept/reject a type-2 customer. For notational convenience, we define $\phi \triangleq \frac{1-c}{1-a}$. We point out the right-hand side of (13) may not be an integer; thus, in our algorithm, we use a slightly modified version of it, defined as follows:

$$q_2(\lambda - 1/n) \leq \Big\lfloor \frac{1-c}{1-a} b + c(b - u_1(\lambda))^+ \Big\rfloor. \quad (14)$$

Note that by the definition of the threshold given in (14), we always accept the first $\lfloor \phi b \rfloor$ type-2 customers. The formal definition of our algorithm is presented in Algorithm 2. In Algorithm 2, q_j represents the counter for the number of accepted customers of type-j so far.

Algorithm 2 (Online Adaptive Algorithm ($ALG_{2,c}$))

1. Initialize $q_1, q_2 \leftarrow 0$, and define $\phi \triangleq \frac{1-c}{1-a}$, and $\delta \triangleq \frac{\phi b}{n}$.
2. Repeat for time $\lambda = 1/n, 2/n, \ldots, 1$:
   a. Calculate functions $u_1(\lambda)$ and $u_{1,2}(\lambda)$ (to construct upper bounds for n_1 and $n_1 + n_2$):

$$u_1(\lambda) \triangleq \begin{cases} b & \text{if } \lambda < \delta \text{ (not enough data observed).} \\ \min\Big\{\frac{o_1(\lambda)}{\lambda p}, \frac{o_1(\lambda) + (1-\lambda)(1-p)n}{1-p+\lambda p}\Big\} & \text{if } \lambda \geq \delta. \end{cases}$$

$$u_{1,2}(\lambda) \triangleq \begin{cases} b & \text{if } \lambda < \delta \text{ (not enough data observed).} \\ \min\Big\{\frac{o_1(\lambda) + o_2(\lambda)}{\lambda p}, \frac{o_1(\lambda) + o_2(\lambda) + (1-\lambda)(1-p)n}{1-p+\lambda p}\Big\} & \text{if } \lambda \geq \delta. \end{cases}$$

   b. Accept customer $i = \lambda n$ arriving at time λ if there is remaining inventory and one of the following conditions holds:

$v_i = 1$; update $q_1 \leftarrow q_1 + 1$.
$v_i = a$ and $u_{1,2}(\lambda) < b$; $update q_2 \leftarrow q_2 + 1$.
$v_i = a$ and $q_2 \leq \lfloor \phi b + c(b - u_1(\lambda))^+ \rfloor$; update$q_2 \leftarrow q_2 + 1$.

We prioritize the second condition if both the second and the third ones hold.

Before we analyze the algorithm, we highlight two key properties of threshold $\lfloor \phi b + c(b - u_1(\lambda))^+ \rfloor$: (i) The threshold is decreasing in $u_1(\lambda)$; the smaller $u_1(\lambda)$ is, the less inventory we reserve for future type-1 customers. (ii) The threshold is decreasing in c as well (the right-hand side of (14) can be expressed as $\lfloor \frac{1}{1-a} b - c(\frac{b}{1-a} - (b - u_1(\lambda))^+) \rfloor$). When c is too large, we may reject too many type-2 customers, which in turn hurts the revenue in a certain class of instances. Said another way, note that inequality (14) only gives a "necessary" condition for achieving c-competitiveness. We identify the sufficient condition for c-competitiveness by solving the *factor-revealing* mathematical program presented in (MP1). We will explain the construction of this program in the analysis of the competitive ratio (in Section 5.2). On a high level, we construct the feasible region such that it contains any valid instance that can violate the c-competitiveness; by minimizing over c, we find the smallest value of c for which the feasible region is not empty.

$$\underset{(l, n_1, n_2, \eta_1, \eta_2, c)}{\text{Minimize}} \quad c \quad \text{(MP1)}$$

subject to

$$c \geq \frac{a\big(n_2 - \tilde{o}_2 + \frac{b}{1-a}\big) + n_1}{a \min\{n_1 + n_2, b\} + (1-a)n_1 + \frac{a^2 b}{1-a} + a \min\{\tilde{u}_1, b\}}, \quad (15a)$$

$$\tilde{u}_{1,2} \geq b, \quad (15b)$$

$$l \leq 1, \quad (15c)$$

$$\eta_1 + \eta_2 \leq ln, \quad (15d)$$

$$\eta_1 \leq n_1, \quad (15e)$$

$$\eta_2 \leq n_2, \quad (15f)$$

$$n_1 \leq b, \quad (15g)$$

$$n_1 + n_2 \leq n, \quad (15h)$$

$$n_1 + n_2 \leq \eta_1 + \eta_2 + (1-l)n, \quad (15i)$$

where $\tilde{o}_1 \triangleq (1-p)\eta_1 + pn_1 l$, $\tilde{o}_2 \triangleq (1-p)\eta_2 + pn_2 l$, $\tilde{u}_1 \triangleq \min\{\frac{\tilde{o}_1}{lp}, \frac{\tilde{o}_1+(1-l)(1-p)n}{(1-p+lp)}\}$, and $\tilde{u}_{1,2} \triangleq \min\{\frac{\tilde{o}_1+\tilde{o}_2}{lp}, \frac{\tilde{o}_1+\tilde{o}_2+(1-l)(1-p)n}{(1-p+lp)}\}$. Before we analyze Algorithm 2, we also evaluate the solution of (MP1). Denote the optimal objective value of (MP1) by c^*. As will be stated in Theorem 2, ALG_{2,c^*} achieves a competitive ratio of c^* (minus an error term). First, we solve (MP1) numerically for the regime where $b = \kappa n$ (where $0 < \kappa \le 1$ is a constant) and show that if $b/n > 0.5$, then Algorithm 2 achieves a better competitive ratio than Algorithm 1.

In Figure 2, we fix $a = 0.5, 0.7$ and plot c^* for $p = 0.05, 0.1, \ldots, 0.95$ for three cases of $b/n = 0.9$, 0.7, and 0.5. Figure 2 leads us to make the following observation: The competitive ratio of ALG_{2,c^*} is at least that of ALG_1, and it is significantly larger when (i) p is small and (ii) b/n is large. This observation highlights the power of adapting to the data, even though it contains an adversarial component: Consider $a = 0.7$, $b = 0.7n$, and $p = 0.2$; this means that 80% of the demand belongs to the adversarial group. Our adaptive algorithm guarantees 10% more revenue than the nonadaptive algorithm does. In addition, we note that as the initial inventory b becomes larger (for a fixed time horizon n), the adversary's power naturally declines. Thus, one would expect that a "smart" algorithm achieves a higher competitive ratio. Our adaptive algorithm indeed attains a higher competitive ratio as the initial inventory increases. In contrast, the competitive ratio of our nonadaptive algorithm remains the same. We conclude our study of (MP1) by establishing a lower bound on its optimum solution. The following proposition states that c^* is at least $p + \frac{1-p}{2-a}$, which is the competitive ratio of Algorithm 1 (ignoring the error term).

Proposition 1. *For any $b \le n$, we have $c^* \ge p + \frac{1-p}{2-a}$. Further, if $b = n$, then $c^* = 1$.*

5.2. Competitive Analysis

In this section, we analyze the competitive ratio of Algorithm 2 and prove the following theorem:

Theorem 2. *For $p \in (0,1)$, let c^* be the optimal objective value of (MP1). For any $c \le c^*$ such that $c < 1$, $ALG_{2,c}$ is $c - O(\frac{1}{(1-c)^2 ap^{3/2}}\sqrt{\frac{n^2 \log n}{b^3}})$ competitive in the partially predictable model.*

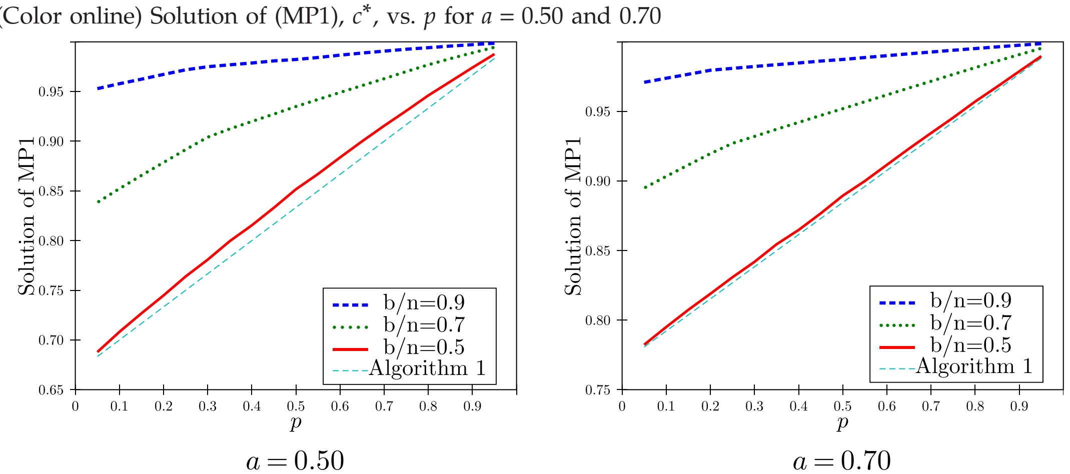
Theorem 2 implies that if $c^* < 1$, then ALG_{2,c^*} is $c^* - O(\frac{1}{(1-c^*)^2 ap^{3/2}}\sqrt{\frac{n^2 \log n}{b^3}})$ competitive. However, the same does not hold when $c^* = 1$. For this special case, we have the following corollary of Theorem 2:

Corollary 1. *When $c^* = 1$, for $c = 1 - \sqrt[3]{\frac{1}{ap^{3/2}}\sqrt{\frac{n^2 \log n}{b^3}}}$ the competitive ratio of $ALG_{2,c}$ is $1 - O(\sqrt[3]{\frac{1}{ap^{3/2}}\sqrt{\frac{n^2 \log n}{b^3}}})$.*

Remark 5. Theorem 2 combined with Proposition 1 shows that in the asymptotic regime (where n and b both grow), if the scaling factor $\sqrt{\frac{n^2 \log n}{b^3}}$ (which appears in the error term of the competitive ratio) is vanishing (i.e., order of $o(1)$), then our adaptive algorithm outperforms our nonadaptive one. For instance, the aforementioned condition holds if $b = \kappa n$ where $0 < \kappa \le 1$ is a constant.

Proof of Theorem 2. Similar to the proof of Theorem 1, we start by making the observation that Theorem 2 is nontrivial only if $\sqrt{\frac{n^2 \log n}{b^3}}$ is small enough such that the approximation term $O(\cdot)$ is negligible. Therefore, without loss of generality, we can restrict attention to the case where $\sqrt{\frac{n^2 \log n}{b^3}}$ is small. In particular, if $\frac{1}{(1-c)^2 ap^{3/2}}\sqrt{\frac{n^2 \log n}{b^3}} \ge \bar{\epsilon}$, then $O(\frac{1}{(1-c)^2 ap^{3/2}}\sqrt{\frac{n^2 \log n}{b^3}})$ becomes $O(1)$ and Theorem 2 becomes trivial (recall that constant $\bar{\epsilon} = 1/24$ is defined in Lemma 1). Therefore, without

Figure 2. (Color online) Solution of (MP1), c^*, vs. p for $a = 0.50$ and 0.70



loss of generality, we assume $\frac{1}{(1-c)^2 ap^{3/2}}\sqrt{\frac{n^2\log n}{b^3}} < \bar{\epsilon}$ or, equivalently,

$$b^{\frac{3}{2}} > \frac{1}{\bar{\epsilon}}\frac{n\sqrt{\log n}}{(1-c)^2 ap^{3/2}}. \tag{16}$$

We remark that we impose the same condition on b in Lemma 9. We denote the random revenue generated by Algorithm 2 by $ALG_{2,c}(\vec{V})$. Similar to the proof of Theorem 1, we define an appropriate ϵ that allows us to focus on the realizations that belong to event $\mathcal{E}$. In particular, let $\epsilon = \frac{1}{(1-c)^2 ap^{3/2}}\sqrt{\frac{n^2\log n}{b^3}}$. For b that satisfies condition (16), and assuming that $n > 24$, we have $\frac{1}{n} \le \epsilon \le \bar{\epsilon}$. Therefore, we can apply Lemma 1 to get

$$\frac{\mathbb{E}\left[ALG_{2,c}\left(\vec{V}\right)\right]}{OPT(\vec{v}_I)} \ge \frac{\mathbb{E}\left[ALG_{2,c}\left(\vec{V}\right)|\mathcal{E}\right]\mathbb{P}\,\mathcal{E}}{OPT(\vec{v}_I)}$$
$$\ge \frac{\mathbb{E}\left[ALG_{2,c}\left(\vec{V}\right)|\mathcal{E}\right]}{OPT(\vec{v}_I)}(1-\epsilon).$$

In the main part of the proof, we show that for any realization $\vec{v}$ belonging to event $\mathcal{E}$,

$$\frac{ALG_{2,c}(\vec{v})}{OPT(\vec{v}_I)} \ge c - O(\epsilon).$$

To analyze the competitive ratio, we analyze three cases separately.

Case (i). When $n_1 \ge \frac{k}{p^2}\log n$, and Algorithm 2 exhausts the inventory. When $n_1 \ge \frac{k}{p^2}\log n$, we can apply (2a) from Lemma 1 and Lemma 9. Because Algorithm 2 exhausts the inventory, we know that $n_1 + n_2 \ge b$. Now we have either (a) $n_1 + n_2 - \frac{2\Delta}{\delta p} \le b$ or (b) $n_1 + n_2 - \frac{2\Delta}{\delta p} > b$. If (a) happens, then (according to Lemma 9) we may have $u_{1,2}(\lambda) < b$, which may result in accepting a type-2 customer through the second condition that we should have rejected. However, in this case, we also have a tight upper bound on the optimum offline solution. As shown in the proof of Lemma 11—which analyzes the competitive ratio of the two cases (a) and (b) separately—such a bound allows us to establish the desired lower bound on the competitive ratio. Case (b) is the more interesting case, which accepts type-2 customers through the third condition of Algorithm 2. It is possible that the algorithm accepts *too many* type-2 customers through this condition, resulting in rejecting type-1 customers and thus in revenue loss. In the following lemma, we control for this loss by establishing an upper bound on the number of accepted type-2 customers. The proof of the lemma, which uses similar ideas to those in Lemma 2, is deferred to Online Appendix EC.5.

Lemma 10. *Under event $\mathcal{E}$, if $n_1 \ge \frac{k}{p^2}\log n$, then one of the following conditions holds:*
a. $n_1 + n_2 - \frac{2\Delta}{\delta p} \le b$ or
b. $n_1 + n_2 - \frac{2\Delta}{\delta p} > b$
and $q_2(1) \le \frac{1-c}{1-a}b + c(b-n_1)^+ + c\frac{2\Delta}{\delta p} + 1$.

Using Lemma 10 and the discussion before the lemma, in Online Appendix EC.5, we prove the following lemma, which gives a lower bound on the competitive ratio for Case (i):

Lemma 11. *Under event $\mathcal{E}$, if $n_1 \ge \frac{k}{p^2}\log n$ and $q_1(1) + q_2(1) = b$, then*

$$\frac{ALG_{2,c}(\vec{v})}{OPT(\vec{v})} \ge c - \frac{3\Delta}{ab\delta p}.$$

Case (ii). When $n_1 \ge \frac{k}{p^2}\log n$, and Algorithm 2 does not exhaust the inventory. First note that in this case $OPT(\vec{v}) = n_1 + a\min\{b-n_1, n_2\}$. Also, in this case, we accept all type-1 customers. Therefore, $q_1(1) = n_1$. To lower-bound the competitive ratio, we need to show only that we do not reject too many type-2 customers, that is, $q_2(1)$ is large enough. Note that if for all $\lambda \in \{1/n, 2/n, \ldots, 1\}$, condition (14) holds, then all type-2 customers are accepted, and we have $q_2(1) = n_2$. This implies that $ALG_{2,c}(\vec{v}) = OPT(\vec{v})$. The more interesting case is when there exists at least one time step for which condition (14) is violated. Let l be the last time that we reject a type-2 customer. This means that at time l, we have

$$u_{1,2}(l) \ge b, \tag{17}$$
$$q_2(l) \ge \frac{1-c}{1-a}b + c(b-u_1(l))^+. \tag{18}$$

This also provides the following lower bound on the number of accepted type-2 customers:

$$q_2(1) = q_2(l) + [n_2 - o_2(l)] \ge \frac{1-c}{1-a}b + c(b-u_1(l))^+ + \left[n_2 - o_2\left(\bar{\lambda}\right)\right]. \tag{19}$$

Therefore, when $q_1(1) + q_2(1) < b$,

$$\frac{ALG_{2,c}(\vec{v})}{OPT(\vec{v})} \ge \frac{n_1 + a\left(\frac{1-c}{1-a}b + c(b-u_1(l))^+ + [n_2 - o_2(l)]\right)}{n_1 + a\min\{b-n_1, n_2\}}. \tag{20}$$

For a fixed c, if for all possible instances the right-hand side of (20) is greater than c, then $ALG_{2,c}$ would be c-competitive. However, if c is too large, then there will be instances for which the right-hand side of (20) will be less than c. We identify a superset of these instances by all possible combinations of $(l, n_1, n_2, \eta_1(l), \eta_2(l))$ that satisfy certain constraints to ensure they correspond to valid instances. As a reminder,

$\eta_j(l)$ represents the number of type-j customers by time l in the initial sequence (determined by the adversary, i.e., $\vec{v}_I$). As we describe these constraints, it becomes clear that (1) any instance of the problem would satisfy all these constraints, and (2) these constraints correspond to the feasible region of the mathematical program in (MP1). We start with the straightforward constraints: for every instance, $n_1 + n_2 \le n$. Also, $\eta_1(l) \le n_1$, and $\eta_2(l) \le n_2$. Further, in the initial customer sequence $\vec{v}_I$, at time l, we cannot have more than ln customers, thus $\eta_1(l) + \eta_2(l) \le ln$. Similarly, after time l, we cannot have more than $(1-l)n$ customers, and therefore $n_1 + n_2 - [\eta_1(l) + \eta_2(l)] \le (1-l)n$. By definition of l, we have $l \le 1$. We also add the condition $n_1 \le b$, which is always true under the case when $q_1(1) + q_2(1) < b$. Note that these are constraints (15c)–(15i) in (MP1), where, in (MP1), with a slight abuse of notation, we simplify by substituting η_j for $\eta_j(l)$. For a moment, suppose $o_j(l) = \tilde{o}_j(l)$. First, we remind the reader that $\tilde{o}_j(l) = (1-p)\eta_j(l) + pln_j$ is the deterministic approximation of $o_j(l)$ that we introduced in Section 3 and also is redefined in (MP1) (at the bottom). Further, note that this is just to explain the idea behind constructing (MP1). Later in the proof, we address the difference between $\tilde{o}_j(l)$ and $o_j(l)$. In this case, we have

$$\tilde{u}_{1,2}(l) \triangleq \min\left\{\frac{\tilde{o}_1(l)+\tilde{o}_2(l)}{lp}, \frac{\tilde{o}_1(l)+\tilde{o}_2(l)+(1-l)\ \times(1-p)n}{(1-p+lp)}\right\} = u_{1,2}(l) \ge b, \tag{21}$$

where the last inequality is the same as inequality (17). Further note that rejecting a customer at time l implies that $l \ge \frac{\phi b}{n} = \delta$ and thus by definition $u_{1,2}(l) = \min\{\frac{o_1(l)+o_2(l)}{lp}, \frac{o_1(l)+o_2(l)+(1-l)(1-p)n}{1-p+lp}\}$.[11] Note that inequality (21) is constraint (15b), where in (MP1), again with a slight abuse of notation, we simplify by substituting $\tilde{u}_{1,2}$ for $\tilde{u}_{1,2}(l)$ and $\tilde{o}_j$ for $\tilde{o}_j(l)$. Further, the most interesting constraint, constraint (15a), comes from condition (20). By rearranging terms, we can show that the right-hand side of (20) being smaller or equal to c is equivalent to

$$c \ge \frac{a\left(n_2 - o_2(l) + \frac{b}{1-a}\right) + n_1}{a\min\{n_1+n_2, b\} + (1-a)n_1 + \frac{a^2 b}{1-a} + a\min\{u_1(l), b\}}, \tag{22}$$

which is constraint (15a) after substituting $o_2(l)$ with $\tilde{o}_2$ and $u_1(l)$ with $\tilde{u}_1$. Overall, the above conditions define the feasible region of the math program (MP1). By minimizing c, we find the threshold for making (MP1) infeasible: Let c^* be the solution of (MP1); for any $c < c^*$, (MP1) is infeasible, and the only constraint that $(l, n_1, n_2, \eta_1(l), \eta_2(l), c)$ can violate is (15a) (same as (22)). This implies that $ALG_{2,c}$ is c-competitive. We now go back and address the issue that $\tilde{o}_j(l)$ and $o_j(l)$ are not equal. Because of the difference between $o_j(l)$ and $\tilde{o}_j(l)$, (1) constraint (15b) might be violated (even though (17) is satisfied) and (2) violating constraint (15a) does not imply violating (22). To address these issues, first in Lemma 12, we give a slightly modified tuple that satisfies constraints (15b)–(15i); then, in Lemma 13, we prove that for any $c \le c^*$, if constraint (15a) is violated, then $\frac{ALG_{2,c}(\vec{v})}{OPT(\vec{v})} \ge c - \frac{4\Delta n}{\phi^2 b^2 p}$. The proofs of both lemmas are deferred to Online Appendix EC.5, and they amount to applying the concentration results of Lemma 1 and carefully analyzing the error terms. These two lemmas complete the analysis of competitive ratio in Case (ii).

Lemma 12. Under event $\mathcal{E}$, if $n_1 \ge \frac{k}{p^2}\log n$ and $q_1(1) + q_2(1) < b$, then the tuple $(l', n_1', n_2', \eta_1', \eta_2', c') \triangleq (l, n_1, n_2 + \xi, \eta_1(l), \eta_2(l) + \bar{\xi}, c)$ satisfies constraints (15b)–(15i), where

$$\xi \triangleq \begin{cases} 0 & \text{if } n_1 + n_2 \ge b, \\ \min\left\{n - (n_1+n_2), \frac{\Delta n}{\phi b p}\right\} & \text{if } n_1 + n_2 < b; \end{cases}$$

$$\bar{\xi} \triangleq \begin{cases} 0 & \text{if } n_1 + n_2 \ge b, \\ \min\left\{\xi, ln - \left(\eta_1(l) + \eta_2(l)\right)\right\} & \text{if } n_1 + n_2 < b, \end{cases}$$

and where $\Delta = \alpha\sqrt{b\log n}$, $\phi = \frac{1-c}{1-a}$, and l is the last time that we reject a type-2 customer.

Lemma 13. *Under event $\mathcal{E}$, if $n_1 \ge \frac{k}{p^2}\log n$ and $q_1(1) + q_2(1) < b$, then*

$$\frac{ALG_{2,c}(\vec{v})}{OPT(\vec{v})} \ge c - \frac{4\Delta n}{\phi^2 b^2 p}.$$

Case (iii). When $n_1 < \frac{k}{p^2}\log n$. The competitive ratio analysis for this case uses ideas similar to those in the previous two cases, and it follows from the next two lemmas. The proofs are deferred to Online Appendix EC.5.

Lemma 14. *Under event $\mathcal{E}$, if $n_1 < \frac{k}{p^2}\log n$, then one of the following three conditions holds:*

a. *$q_1(1) + q_2(1) = b$;*
b. *$q_1(1) = n_1$ and $q_2(1) = n_2$; or*
c. *$q_1(1) = n_1$ and $q_2(1) \ge cb$.*

Using Lemma 14, in the following lemma, we establish a lower bound on the competitive ratio for Case (iii):

Lemma 15. *Under event $\mathcal{E}$, if $n_1 < \frac{k}{p^2}\log n$, then $\frac{ALG_{2,c}(\vec{v})}{OPT(\vec{v})} \ge c$.*

Having Lemmas 11, 13, and 15, we have lower bounds on the competitive ratio of Algorithm 2 for all possible cases. We complete the proof of the theorem by the following lemma (proven in Online Appendix EC.5) that ensures that the error terms in Lemmas 11 and 13 are $O(\epsilon)$.

Lemma 16. *The error terms in Lemmas 11 and 13 are $O(\epsilon)$, that is, (a) $\frac{3\Delta}{ab\delta p} = O(\epsilon)$ and (b) $\frac{4\Delta n}{\phi^2 b^2 p} = O(\epsilon)$.*

This completes the proof of Theorem 2.

6. Discussion of the Model

In this section, we further study the performance of online algorithms in our demand model. First, in Section 6.1, we present an upper bound on the competitive ratio achievable by any online algorithm under our demand model when the initial inventory b is small—more precisely, $b = o(\sqrt{n})$. Next, in Section 6.2, we highlight the need for our new online algorithms by presenting a problem instance for which our algorithms outperform existing ones in our partially predictable model.

6.1. Upper Bounds

In this section, we present an upper bound on the competitive ratio of any online algorithm when $b = o(\sqrt{n})$. We start with a warm-up example that illustrates a fundamental limit of any online algorithm in the partially predictable model. Figure 3 shows two instances with $n = 8$. The bottom row shows the sequence that the online algorithm will see; as a reminder, we represent the nodes of the stochastic group as filled (even though the online algorithm cannot distinguish between the two groups of customers). Suppose $b = 4$; in the instance presented on the left, the optimum offline solution rejects all type-2 customers, and in the instance on the right, it accepts all of them. Now, by time $\lambda = b/n = 4/8$, online algorithms cannot distinguish between these two instances and, hence, cannot perform as well as the optimal offline algorithm on *both* of these instances. Similar to this example, in the following proposition, we establish the upper bound by constructing two problem instances that are "difficult" for online algorithms to distinguish between up to time $\frac{b}{n}$, and show that the trade-off between accepting too many or too few type-2 customers limits the competitive ratio of any online algorithm.

Proposition 2. *Under the partially predictable arrival model, and for any $p \in (0, 1)$, no online algorithm, deterministic and randomized, can achieve a competitive ratio better than $\frac{1-p}{2-a} + p + O(\frac{pb^2}{n})$. Therefore, when $b = o(\sqrt{n})$, no online algorithm can achieve a competitive ratio better than $\frac{1-p}{2-a} + p + o(1)$.*

The details of the proof are deferred to Online Appendix EC.6. As explained above, the main idea of the proof is to construct two instances that are almost indistinguishable up to time $\frac{b}{n}$ to any online algorithm. In the proof, we show that the following two instances $\vec{v}_I$ and $\vec{w}_I$ serve our purpose:

$$v_{I,j} = \begin{cases} a, & 1 \le j \le b, \\ 0, & b < j \le 2b, \\ 0, & j > 2b. \end{cases} \quad w_{I,j} = \begin{cases} a, & 1 \le j \le b, \\ 1, & b < j \le 2b, \\ 0, & j > 2b. \end{cases}$$

6.2. Comparison with Existing Algorithms

In this section, we show that, under our demand arrival model, there exists a class of instances for which our algorithms achieve higher revenue than algorithms designed for either the worst-case (Ball and Queyranne 2009) or the random-order model (Devanur and Hayes 2009, Agrawal et al. 2014), which respectively correspond to $p = 0$ and $p = 1$ in our model. To this end, we consider instance $\vec{v}_I$ where

$$v_{I,j} = \begin{cases} a & \text{for } 1 \le j \le b, \\ 0 & \text{for } j > b. \end{cases} \tag{23}$$

Table 2 presents the ratio between the expected revenue of different online algorithms and that of the optimum offline solution for the instance defined in (23). In the following, we will explain how we compute these bounds. Before that, we discuss the implications of this example. This instance class shows that, for any $p \in (0, 1)$, when $b = \omega(\sqrt{\log n})$ and $b = o(n)$ the ratio for both of our algorithms is better than existing ones. Further, note that the ratio for Algorithm 1 is in fact its competitive ratio; thus, the same ratio holds for any other instance as well. This implies that the competitive ratio of our nonadaptive algorithm is higher than those of Ball and Queyranne (2009) and Agrawal et al. (2014) under the partially predictable model. Also note that for the same instance, randomizing between the algorithm of Ball and Queyranne (2009) (with probability $1-p$) and that of Agrawal et al. (2014) (with probability p) leads to a ratio of $\frac{1-p}{2-a} + p^2 + o(1)$, which is not the convex combination of the competitive ratios of these two algorithms (as also pointed out in Remark 4). Next, we calculate the ratios listed in Table 2. The offline solution is $OPT(\vec{v}_I) = ab$. The algorithm of Ball and Queyranne (2009), proposed for the adversarial model, has a fixed threshold of $\frac{1}{2-a}b$ for

Figure 3. Two Problem Instances Between Which Online Algorithms Cannot Distinguish at Time $\frac{b}{n}$, Where $b = 4$ and $n = 8$

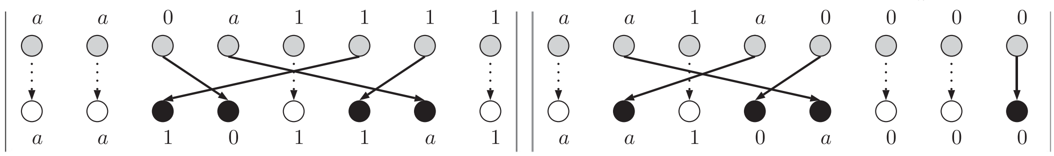


Table 2. Ratio Between the Expected Revenue of Different Algorithms and the Optimum Offline Solution for the Instance Defined in (23)

Algorithm	Worst case	Random order	Algorithm 1	Algorithm 2
Ratio	(Ball and Queyranne 2009) $\frac{1}{2-a}$	(Idea of Agrawal et al. 2014) At most $p+\frac{b}{n}(1-p)$	(Nonadaptive algorithm) $p+\frac{1-p}{2-a}-O(\frac{1}{a(1-p)p}\sqrt{\frac{\log n}{b}})$	(Adaptive algorithm) 1

accepting type-2 customers and, hence, accepts $\frac{1}{2-a}b$ type-2 customers. Next we compute the ratio for algorithms designed for the random-order model (e.g., Devanur and Hayes 2009, Agrawal et al. 2014, and Kesselheim et al. 2014). We note that, for the sake of brevity, we present an analysis based on the idea of these papers, which is allocating inventory at a roughly uniform rate over the entire horizon. In particular, these algorithms accept roughly λb customers at any time $\lambda \in [0,1]$. As a result, for this instance, they accept at most b^2/n type-2 customers up to time $\lambda = b/n$. According to our model, in the arriving instance $\vec{v}$, there are approximately $(1-b/n)bp$ type-2 customers arriving after time b/n. Therefore, these algorithms can accept at most $b^2/n+(1-b/n)bp$ type-2 customers, which corresponds to a ratio of at most $p+\frac{b}{n}(1-p)$. Note that $p+\frac{b}{n}(1-p) < p+\frac{1-p}{2-a}$ for any $b<\frac{n}{2-a}$. Our Algorithm 1 achieves a ratio of at least its competitive ratio as given in Theorem 1, and the ratio is tight for this instance (up to an additive error term of $O(\frac{1}{a(1-p)p}\sqrt{\frac{\log n}{b}})$). For Algorithm 2, let $c \in (0,1)$ be an arbitrary constant. We show that $ALG_{2,c}$ achieves the ratio of one because the third condition in Algorithm 2, that is, the dynamic threshold, is never violated. To see this we compute the threshold as follows:

$$\lfloor \phi b + c(b-u_1(\lambda))^+ \rfloor = \begin{cases} \lfloor \phi b \rfloor & \lambda < \delta = \frac{\phi b}{n} \\ \lfloor \phi b + cb \rfloor & \lambda \geq \delta \end{cases},$$

where we use the fact that $u_1(\lambda)=b$ for $\lambda<\delta$ and $u_1(\lambda)=0$ for $\lambda \geq \delta$. In both cases, we have $\lfloor \phi b + c(b-u_1(\lambda))^+ \rfloor > \lambda$, which implies that the algorithm never rejects a type-2 customer because $o_2(\lambda) \leq \lambda < \lfloor \phi b + c(b-u_1(\lambda))^+ \rfloor$.

7. The Secretary Problem Under Partially Predictable Demand

In this section, we study the online secretary problem under our new arrival model. In our setting, the secretary problem corresponds to having one unit of inventory, that is, $b=1$, and n customers, where $v_{I,j} \in \mathbb{R}^+$ for $1 \leq j \leq n$, that is, we relax the assumption that there are only two types. The objective is to maximize the probability of selecting the highest-revenue customer in the asymptotic regime, where $n \to \infty$. In the classical setting, the arrival sequence is assumed to be a uniformly random permutation of n customers, which corresponds to the extreme case of $p=1$ under our partially predictable model. In this setting, it is well known that the best-possible online algorithm is the following *deterministic* algorithm (Lindley 1961, Dynkin 1963, Freeman 1983, Ferguson 1989): Observe the first $\lfloor \gamma n \rfloor$ customers, where $\gamma = \frac{1}{e}$; then accept the next one that has the highest revenue so far (if any). The success probability of this algorithm approaches $\frac{1}{e} \approx 0.37$ as $n \to \infty$. We generalize the classical setting by studying the problem under our demand model. First, we analyze the success probability of a similar class of algorithms for any $p \in (0,1]$. Next, we show that under our demand model where $p<1$—that is, in the presence of an adversarial component—this class of algorithms is not necessarily the best possible. For any $\gamma \in (0,1)$, we define the Observation-Selection Algorithm (OSA_γ), which works similarly to the classical algorithm described earlier in this paragraph. The formal definition of the algorithm is presented in Algorithm 3.

Algorithm 3 (Observation-Selection Algorithm (OSA_γ, $\gamma \in (0,1)$))

1. Initialize $v_{\max} \leftarrow 0$.
2. *Observation period:* Repeat for customer $i = 1,2,\ldots,\lfloor \gamma n \rfloor$: reject customer i and update $v_{\max} \leftarrow \max\{v_{\max}, v_i\}$.
3. *Selection period:* Repeat for customer $i = \lfloor \gamma n \rfloor + 1, \lfloor \gamma n \rfloor + 2 \ldots, n$:
   - If $v_i \geq v_{\max}$, then select customer i and stop the algorithm.
   - Otherwise, reject customer i.

In Online Appendix EC.7, we analyze the success probability of Algorithm 3 and prove the following theorem:

Theorem 3. *Under the partially predictable model, in the limit $n \to \infty$, the success probability of OSA_γ approaches $\gamma p \log \frac{1}{\gamma p + 1 - p}$.*

By optimizing over γ, we obtain the following corollary:

Corollary 2. *Let $\gamma^* \in (0,1)$ be the unique solution to*

$$\log(\gamma^* p + 1 - p) + \frac{\gamma^* p}{\gamma^* p + 1 - p} = 0;$$

then OSA_{γ^*} achieves the highest success probability among OSA_γ for all $\gamma \in (0,1)$.

Table 3 presents the optimal length of the observation period, γ^* and the success probability of OSA_{γ^*} for different values of p. We observe that as the size of the stochastic component increases, that is, as p increases, the length of the observation period decreases, whereas the success probability increases. Next, in the following proposition, we establish a lower bound on the success probability when we randomize over the length of the observation period (γ); further, we present an example that shows such randomization increases the success probability for $p < 1$. This illustrates the benefit of employing randomized algorithms in the presence of an adversarial component in the arrival sequence.

Proposition 3. *Under the partially predictable model, for any $0 < \gamma_1 < \gamma_2 < 1$ and $0 < q < 1$, the randomized algorithm that runs OSA_{γ_1} with probability q and OSA_{γ_2} with probability $1 - q$ has an asymptotic success probability of at least*

$$qs_1 + (1-q)s_2 + \min\Big\{(1-q)p(1-p)(1-\gamma_2), \times q(1-p)\frac{\gamma_2 - \gamma_1}{1-\gamma_1}s_1\Big\}$$

where for $i = 1, 2$, s_i denotes the success probability of OSA_{γ_i}.

The proposition is proven in Online Appendix EC.7. Suppose $p = 0.5$; randomizing over $\gamma_1 = 0.427$ and $\gamma_2 = 0.69$ with $q = 0.824$ results in a success probability of at least 0.083 (utilizing the result of Proposition 3). On the other hand, the success probability of the best possible deterministic observation period OSA_γ, given in Theorem 3 and Corollary 2, is 0.072.

8. Conclusion

Online resource allocation is a central problem in the operations of numerous online platforms ranging from airline booking systems to hotel booking systems to internet advertising. Despite advances in information technology, demand arrival processes are rarely perfectly predictable. The presence of unpredictable patterns limits the performance of most allocation algorithms that rely on fully accurate prediction of future demand based on observed data. At the same time, ignoring available information and taking a completely worst-case approach usually leads to online allocation policies that are too conservative. In this paper, we take a middle-ground approach and introduce the first arrival model that contains both adversarial (thus unpredictable) and stochastic (predictable) components. Our demand model requires no forecast of demand; however, the stochastic component allows us to partially predict future demand as the sequence of arrivals unfolds. In our model, the relative size of the stochastic component, p, represents the level of predictability of the demand. Under our proposed demand model, we study the basic yet fundamental problem of allocating a single resource with an arbitrary initial inventory to a sequence of customers that belong to two types, with type-1 generating higher revenue. For this problem, we design a nonadaptive algorithm as well as an adaptive one. We analyze the competitive ratios of our algorithms and show that they outperform existing ones under our proposed demand model. The first implication of our analysis is that, by employing our algorithms, we can take advantage of limited available information (because of the presence of the stochastic component) to improve the revenue of the firm compared with a fully conservative approach. Indeed, the competitive ratios of our algorithms are parameterized by p; for both algorithms, the ratio increases with p (the relative size of the stochastic component), which highlights the value of even partial predictability. Further, we show that our adaptive algorithm—which repeatedly computes upper bounds on the total number of customers of each type based on observed data and makes online decisions based on those bounds—achieves a higher competitive ratio when the initial inventory b is sufficiently large. This underlines the significance of adapting to the data, even though it contains an adversarial component. Analyzing the adaptive algorithm, however, is considerably more challenging. We establish a lower bound on the competitive ratio by constructing a novel factor-revealing mathematical program. On the other hand, when b is small (more precisely, when $b = o(\sqrt{n})$), we prove an upper bound on the competitive ratio of any deterministic or randomized online algorithm that matches the competitive ratio of our nonadaptive algorithm (up to an error term). This implies (1) our nonadaptive algorithm is the best possible in this regime and (2) when the initial inventory is small relative to the time horizon, we may not be able to effectively adapt to observed data before

Table 3. The Optimal Length of the Observation Period, γ^*, and the Success Probability of OSA_{γ^*} vs. p

p	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
γ^*	0.4935	0.4863	0.4784	0.4696	0.4597	0.4482	0.4348	0.4184	0.3975	0.3679
OSA_{γ^*}	0.0026	0.0105	0.0244	0.0448	0.0724	0.1081	0.1533	0.2095	0.2796	0.3679

allocating most of the inventory. We also have heuristic arguments—in which we do not characterize the error terms—that indicate that (1) our adaptive algorithm achieves the best possible competitive ratio in the regime where $b = \kappa n$ (where $\kappa \in (0, 1]$ is a constant) and (2) underestimating parameter p does not affect the competitive ratio of our adaptive algorithm, whereas (3) if we overestimate p by (a small amount), its competitive ratio decreases only slightly. Because making the above results rigorous will make the paper prohibitively long, these results are not included in the paper. To illustrate the application of our model to other online allocation problems, we study the secretary problem under our demand model. We analyze the celebrated policy of selecting the highest revenue customer after an observation period with a deterministic length of γ under our new model and find the optimum value of γ (which is parameterized by p). We further show that, in the presence of an adversarial component and unlike the classical setting, randomizing over the length of the observation period may increase the probability of selecting the highest revenue customer. In this paper, we use a discrete time model and also assume that the arrival times of customers from the stochastic group are randomly permuted among their predetermined positions. We believe similar results can be obtained for a model where a total of n customers from the two groups (i.e., the stochastic and adversarial group) arrive according to independent Poisson processes with rates p and $1 - p$. We leave the rigorous treatment of this alternative model for future research. Studying other online allocation problems under our new demand model is a promising direction for future research. Our consequential concentration result from Lemma 1 can be extended to any finite number of types. Further, we believe that by combining our ideas for adaptively computing bounds on the demand of each type with those of Lan et al. (2008), and utilizing the concentration results, one can generalize our algorithms to a setting with any finite number of types. Such extensions are, however, beyond the scope of this paper.

Acknowledgments

The authors thank Lê Nguyên Hoang for fruitful discussions on the concentration result.

Endnotes

[1] A few papers consider arrival models outside these two categories. We carefully review them and compare them with our model in Section 2.

[2] In fact, these papers assume a more general model, the random order model, that we discuss in Section 2.

[3] We call an algorithm "adaptive" if it makes decisions based on the sequence of arrivals it has observed so far.

[4] We always accept a type-1 customer if there is remaining inventory.

[5] The condition on n is needed for technical reasons as it becomes clear later in Lemma 1.

[6] We present the values of the constants, defined in the statement of the lemma, only to clarify that they exist and do not depend on n; however, they are not optimized.

[7] The first inequality follows from definition of $\eta_j(\lambda)$. The second one also follows from definition of $\eta_j(\lambda)$ and from the observation that the number of type-j customers arriving between λ and one cannot be more than the number of remaining time steps, that is, $(1 - \lambda)n$.

[8] Algorithms developed in other papers for random order model, such as Devanur and Hayes (2009) and Kesselheim et al. (2014), are also adaptive.

[9] Note that we already have an upper bound on the number of type-2 customers accepted by the fixed threshold: $q_{2,f}(1) \leq \theta b$.

[10] This follows from condition (5) and the fact that $\frac{1}{a^2(1-p)^2} > 1$ and by definition (given in Lemma 1) $\frac{1}{\epsilon^2} \geq k$, which imply $\frac{1}{\epsilon^2}\frac{1}{a^2(1-p)^2} \geq k$.

[11] Note that when $l < \delta$, Algorithm 2 never rejects a customer, because $q_2(l) \leq ln < \delta n = \phi b$.

References

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

Araman VF, Caldentey R (2009) Dynamic pricing for nonperishable products with demand learning. *Oper. Res.* 57(5):1169–1188.

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

Belobaba PP (1987) Survey paper-airline yield management an overview of seat inventory control. *Transportation Sci.* 21(2): 63–73.

Belobaba PP (1989) OR practice-application of a probabilistic decision model to airline seat inventory control. *Oper. Res.* 37(2):183–197.

Ben-Tal A, Nemirovski A (2002) Robust optimization: Methodology and applications. *Math. Programming* 92(3):453–480.

Bertsimas D, Pachamanova D, Sim M (2004) Robust linear optimization under general norms. *Oper. Res. Lett.* 32(6):510–516.

Besbes O, Zeevi A (2009) Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms. *Oper. Res.* 57(6):1407–1420.

Brumelle SL, McGill JI (1993) Airline seat allocation with multiple nested fare classes. *Oper. Res.* 41(1):127–137.

Buchbinder N, Naor J (2009) The design of competitive online algorithms via a primal: Dual approach. *Foundations Trends® Theoret. Comput. Sci.* 3(2–3):93–263.

Cheapair (2016) What the airlines never tell you about airfares. Accessed March 10, 2016, https://www.cheapair.com/blog/travel-tips/what-the-airlines-never-tell-you-about-airfares/.

Ciocan DF, Farias V (2012) Model predictive control for dynamic resource allocation. *Math. Oper. Res.* 37(3):501–525.

Cooper WL (2002) Asymptotic behavior of an allocation policy for revenue management. *Oper. Res.* 50(4):720–727.

CWT (2016) Gender differences in booking business travel. Report, CWT, Minneapolis.

Devanur NR, Hayes TP (2009) The adwords problem: Online keyword matching with budgeted bidders under random permutations. *Proc. 10th ACM Conf. Electronic Commerce* (ACM, New York), 71–78.

Dynkin EB (1963) The optimum choice of the instant for stopping a Markov process. *Soviet Math.* 4:627–629.

Esfandiari H, Korula N, Mirrokni V (2015) Online allocation with traffic spikes: Mixing adversarial and stochastic models. *Proc. 16th ACM Conf. Econom. Comput.* (ACM, New York), 169–186.

Ferguson TS (1989) Who solved the secretary problem? *Statist. Sci.* 4(3):282–289.

Freeman PR (1983) The secretary problem and its extensions: A review. *Internat. Statist. Rev.* 51(2):189–206.

Jasin S (2015) Performance of an LP-based control for revenue management with unknown demand parameters. *Oper. Res.* 63(4):909–915.

Kesselheim T, Kleinberg R, Niazadeh R (2015) Secretary problems with non-uniform arrival order. *Proc. 47th Annual ACM Sympos. Theory Comput.* (ACM, New York), 879–888.

Kesselheim T, Tönnis A, Radke K, Vöcking B (2014) Primal beats dual on online packing LPs in the random-order model. *Proc. 46th Annual ACM Sympos. Theory Comput.* (ACM, New York), 303–312.

Kleinberg R (2005) A multiple-choice secretary algorithm with applications to online auctions. *Proc. 16th Annual ACM-SIAM Sympos. Discrete Algorithms* (SIAM, Philadelphia), 630–631.

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

Lautenbacher CJ, Stidham S Jr (1999) The underlying Markov decision process in the single-leg airline yield-management problem. *Transportation Sci.* 33(2):136–146.

Lee TC, Hersh M (1993) A model for dynamic airline seat inventory control with multiple seat bookings. *Transportation Sci.* 27(3): 252–265.

Lindley DV (1961) Dynamic programming and decision theory. *J. Royal Statist. Soc. Ser. C* 10(1):39–51.

Littlewood K (2005) Special issue papers: Forecasting and control of passenger bookings. *J. Revenue Pricing Management* 4(2):111–123.

Ma W, Simchi-Levi D (2019) Tight weight-dependent competitive ratios for online edge-weighted bipartite matching and beyond. *Proc. 2019 ACM Conf. Econom. Comput.* (ACM, New York), 727–728.

Mahdian M, Nazerzadeh H, Saberi A (2007) Allocating online advertisement space with unreliable estimates. *Proc. 8th ACM Conf. Electronic Commerce* (ACM, New York), 288–294.

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

Mirrokni V, Oveis Gharan S, Zadimoghaddam M (2012) Simultaneous approximations for adversarial and stochastic online budgeted allocation. *Proc. 23rd Annual ACM-SIAM Sympos. Discrete Algorithms* (SIAM, Philadelphia), 1690–1701.

Shamsi D, Holtan M, Luenberger R, Ye Y (2014) Online allocation rules in display advertising. Preprint, submitted July 24, https://arxiv.org/abs/1407.5710.

Stein C, Truong V-A, Wang X (2020) Advance service reservations with heterogeneous customers. *Management Sci.* 66(7):2929–2950.

Talluri K, Ryzin GV (1998) An analysis of bid-price controls for network revenue management. *Management Sci.* 44(11): 1577–1593.

Talluri KT, Van Ryzin GJ (2006) *The Theory and Practice of Revenue Management*, vol. 68 (Springer Science & Business Media, New York).

Wang X, Truong V-A (2015) Online advance admission scheduling for services, with customer preferences. Working paper, Columbia University, New York.

Wang H, Xie H, Qiu L, Yang YR, Zhang Y, Greenberg A (2006) Cope: Traffic engineering in dynamic networks. *Comput. Comm. Rev.* 36(4):99–110.

Zervas G, Proserpio D, Byers JW (2017) The rise of the sharing economy: Estimating the impact of Airbnb on the hotel industry. *J. Marketing Res.* 54(5):687–705.

Dawsen Hwang received a PhD degree in electrical engineering and computer science, advised by Professor P. Jaillet, from Massachusetts Institute of Technology in 2016. He is currently a data scientist at Google.

Patrick Jaillet is the Dugald C. Jackson Professor in the Department of Electrical Engineering and Computer Science and a member of the Laboratory for Information and Decision Systems at Massachusetts Institute of Technology (MIT). He is also co-director of the MIT Operations Research Center. His research interests include online optimization and learning, machine learning, and decision making under uncertainty. He is a Fellow of INFORMS.

Vahideh Manshadi is an associate professor of operations at the Yale School of Management, with a secondary appointment in the Department of Statistics and Data Science. Her research interests include sequential and dynamic decision making with applications to the design of online and matching marketplaces, as well as information platforms and networks.