MIT Libraries | DSpace@MIT

MIT Open Access Articles

Inventory Balancing with Online Learning

The MIT Faculty has made this article openly available. ***Please share*** how this access benefits you. Your story matters.

Citation: Cheung, Wang Chi, Ma, Will, Simchi-Levi, David and Wang, Xinshang. 2022. "Inventory Balancing with Online Learning." Management Science, 68 (3).

As Published: 10.1287/MNSC.2021.4216

Publisher: Institute for Operations Research and the Management Sciences (INFORMS)

Persistent URL: https://hdl.handle.net/1721.1/148653

Version: Original manuscript: author's manuscript prior to formal peer review

Terms of use: Creative Commons Attribution-Noncommercial-Share Alike

Massachusetts Institute of Technology

Inventory Balancing with Online Learning

Wang Chi Cheung
National University of Singapore, NUS Engineering, Department of Industrial Systems Engineering and Management, Singapore, SG 117576, isecwc@nus.edu.sg

Will Ma
Graduate School of Business, Columbia University, New York, NY 10027, wm2428@gsb.columbia.edu

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

Xinshang Wang
Institute for Data, Systems, and Society, Massachusetts Institute of Technology, Cambridge, MA 02139, xinshang@mit.edu

We study a general problem of allocating limited resources to heterogeneous customers over time under model uncertainty. Each type of customer can be serviced using different actions, each of which stochastically consumes some combination of resources, and returns different rewards for the resources consumed. We consider a general model where the resource consumption distribution associated with each (customer type, action)-combination is not known, but is consistent and can be learned over time. In addition, the sequence of customer types to arrive over time is arbitrary and completely unknown.

We overcome both the challenges of model uncertainty and customer heterogeneity by judiciously synthesizing two algorithmic frameworks from the literature: inventory balancing, which "reserves" a portion of each resource for high-reward customer types that could later arrive, based on competitive ratio analysis; and online learning, which "explores" the resource consumption distributions for each customer type under different actions, based on regret analysis. We define an auxiliary problem, which allows for existing competitive ratio and regret bounds to be seamlessly integrated. Furthermore, we propose a new variant of UCB, dubbed LazyUCB, which conducts less exploration in a bid to focus on "exploitation", in view of the resource scarcity. Finally, we construct an information-theoretic family of counterexamples to show that our integrated framework achieves the best possible performance guarantee.

We demonstrate the efficacy of our algorithms both on synthetic instances generated for the online matching with stochastic rewards problem under unknown probabilities, and on a publicly available hotel data set. Our framework is highly practical in that it requires no historical data (no fitted customer choice models, nor forecasting of customer arrival patterns) and can be used to initialize allocation strategies in fast-changing environments.

1. Introduction

Online resource allocation is a fundamental topic in many applications of operations research, such as revenue management, display advertisement allocation, and appointment scheduling. In each of these settings, an online platform needs to allocate limited resources to a heterogeneous pool of

Electronic copy available at: https://ssrn.com/abstract=3236533

customers arriving in real time, while maximizing the cumulative reward. The starting amount of each resource is exogenous, and these resources cannot be replenished during the time horizon.

In many applications, the online platform can observe a list of feature values associated with each arriving customer, which allows for allocation decisions to be customized in real time. For example, a display advertising platform operator is usually provided with the internet cookie from a website visitor, upon the visitor's arrival. Consequently, the operator is able to display relevant advertisements to each website visitor based on this cookie, in a bid to maximize the total revenue earned from clicks on these advertisements.

To achieve an optimal allocation in the presence of resource constraints, the platform's allocation decision at any moment has to take into account the features of both the current customer as well as the customers who will arrive in the future. In the preceding example, advertisements have daily budgets on how often they can be shown, making it suboptimal for the operator to behave myopically for the current visitor (Mehta et al. 2007, Buchbinder et al. 2007). In another example of selling airline tickets, it is profitable to judiciously reserve a number of seats for business class customers, who often purchase tickets close to departure time (Talluri and van Ryzin 1998, Ball and Queyranne 2009). Finally, in healthcare applications, when making advance appointments for out-patients, it is critical to reserve certain physicians' hours for urgent patients (Feldman et al. 2014, Truong 2015). In all of these examples, the platform's central task is to *reserve* the right amount of each resource for future customers so as to maximize the total reward.

While resource reservation is vital for optimizing online resource allocations, the implementation of resource reservation is hindered by the following two challenges. First, the online platform often lacks an accurate forecast about the arrival patterns of future demand. Second, the online platform is often uncertain about the relationship between an arriving customer's expected behavior, e.g. click-through rate on an ad, and their observed features.

These challenges in implementing resource reservation raise the following research question: *Can the online platform perform resource reservation effectively, in the absence of any demand forecast model and under uncertain customer behavior?*

1.1. Description of Model and Contributions

Initially there is a finite and discrete amount of inventory for each of multiple resources. Resources can be converted to rewards when they are consumed by a customer. Customers arrive sequentially, each of whom is characterized by a context vector that describes the customer's features. Upon the arrival of each customer, an action is selected, after which there is a stochastic consumption of resources, which determines the reward collected. For example, the action can represent offering a specific item to the customer at a particular price, and the stochastic consumption can correspond

Electronic copy available at: https://ssrn.com/abstract=3236533

to whether the customer chooses to purchase. The distribution of this stochastic consumption depends both on the customer's features and the action selected. The objective is to maximize the total expected reward collected from the resources during a finite time horizon of unknown length.

We highlight two salient aspects of our model:

1. The number of future customers and their context vectors are unknown and chosen by an adversary. As a result, historical observations do not provide any information about future arrivals.

2. For each potential combination of context vector and action, there is a fixed unknown distribution over the consumption outcome. That is, two customers arriving at different time periods with identical context vectors will have the same consumption distribution. As a concrete example, in e-commerce, the context vector represents the characteristics (e.g., age, location) of an online shopper. We are assuming that the conversion rate only depends on the characteristics of the shopper and the product offered, but not the time. The platform needs to learn these conversion rates in an online fashion.

Each of these two aspects has been studied extensively, but only separately, in the literature (reviewed in Section 1.3). In models with the first aspect alone, model parameters on customer behavior such as purchase probabilities are known, and the difficulty is in conducting resource reservation without any demand forecast. The conventional approach is to set an opportunity cost for each resource which is increasing in how quickly it has already been consumed, using these to ideally "balance" the consumption rates of the different resources. We call such techniques *Inventory Balancing*. Meanwhile, in models with the second aspect alone, the trade-off is between "exploring" the probabilities from playing different actions on different customers, and "exploiting" actions which are known to yield desirable outcomes. *Online Learning* techniques are designed for managing this trade-off. However, in the presence of resource constraints, work on online learning has assumed that the context vectors are drawn i.i.d. from a known distribution, and there is no element of "hedging" against an adversarial input sequence.

In our work, we present a unified analysis of the online allocation problem in the presence of both of these aspects. We proceed to describe our contributions.

IBOL algorithmic framework with performance guarantees. We propose a framework that integrates the Inventory Balancing technique with a broad class of Online Learning algorithms, which we dub IBOL, short for "Inventory Balancing with Online Learning". Our framework produces online allocation algorithms with performance guarantees of the form

$$\mathbb{E}[\mathsf{ALG}] \geq \alpha \cdot \mathsf{OPT} - \mathsf{REG}, \tag{1}$$

where ALG is the total reward earned by IBOL; OPT is an LP-based upper bound on the expected revenue of an optimal algorithm which knows both the arrival sequence and the unknown probabilities in advance; and REG represents the *regret*, i.e., the loss from having to explore the unknown

Electronic copy available at: https://ssrn.com/abstract=3236533

probabilities. REG in fact represents the optimality gap in an *auxiliary problem* we define, which is a non-stationary stochastic multi-armed bandits problem. The non-stationarity in our auxiliary problem arises from the adversarial uncertainty in customers' arrivals. The factor $\alpha \in (0,1)$ in our guarantee (1) can be viewed as the *competitive ratio* when the probabilities are known, i.e., when $\mathsf{REG} = 0$.

Asymptotically-tight guarantee for online matching with unknown stochastic rewards. As an application of our framework, we analyze an online bipartite matching problem in which edges, upon being selected, only get matched with an unknown probability. We first apply the IBOL algorithm with an Upper Confidence Bound (UCB) oracle, which is based on the optimistic estimation approach in Auer et al. (2002a).[1] We establish the performance guarantee

$$\mathbb{E}[\mathsf{ALG}] \geq \left(1 - \frac{1}{e}\right) \mathsf{OPT} - \tilde{O}(\sqrt{\mathsf{OPT}}). \tag{2}$$

The $\tilde{O}(\cdot)$ notation hides the logarithmic dependence on T, the number of time rounds in the problem, as well as the dependence on model parameters other than T. A consequence of (2) is that $\mathbb{E}[\mathsf{ALG}]/\mathsf{OPT}$ is bounded from below by $1 - 1/e - \tilde{O}(1/\sqrt{\mathsf{OPT}})$, which approaches the best-possible competitive ratio of $1 - 1/e$ as OPT becomes large (i.e. as the regret from learning the matching probabilities becomes negligible).

Importantly, we also show the guarantee in (2), which can be re-expressed as $\mathsf{OPT} - \mathbb{E}[\mathsf{ALG}] \leq \mathsf{OPT}/e - \tilde{O}(\sqrt{\mathsf{OPT}})$, to be tight. That is, the loss of OPT/e is unavoidable due to not knowing the arrival sequence in advance, and the loss of $\tilde{O}(\sqrt{\mathsf{OPT}})$ is unavoidable due to not knowing the matching probabilities in advance. The fact that these losses *accumulate* instead of alleviating each other was surprising to us, and to our knowledge, requires a non-trivial new analysis combining Yao's minimax principle with information theory. We elaborate further when we present our counterexample that demonstrates this tightness.

ϵ-perturbed potential function and $(1+\epsilon)$-relaxed regret. Our IBOL framework also has the flexibility of an additional parameter $\epsilon \in [0,1]$, which allows the Online Learning algorithm to "borrow" an ϵ-share of the reward from the Inventory Balancing algorithm, with both algorithms then re-optimized for the worst case under this new accounting scheme. It leads to the notion an "ϵ-perturbed potential function Ψ", which extends the typical inventory balancing function from online matching by placing a *steeper penalty* on almost-depleted resources when $\epsilon > 0$. On the other hand, this new accounting also leads to the notion of "$(1+\epsilon)$-relaxed regret" in our auxiliary multi-armed bandits problem, and we propose a new "LazyUCB" oracle for minimizing it, which ends up performing *less exploration and more exploitation* than traditional UCB oracles when $\epsilon > 0$.

[1] Essentially, under the optimistic estimation approach for multi-armed bandits, the decision maker adds an *optimistic bonus* to the maximum likelihood estimate on each arm's latent reward, which encourages the exploration of the under-explored arms.

Electronic copy available at: https://ssrn.com/abstract=3236533

Both of these changes brought by $\epsilon > 0$ are intuitive, in our problem setting with both adversarial contexts and unknown probabilities. On one hand, Ψ has less reason to assign almost-depleted resources, because the unknown probabilities for an almost-depleted resources are less worth learning. On the other hand, LazyUCB has less reason to explore, because the adversarial contexts mean there is no guarantee that an arm can be legally pulled again in the future.

For the online matching application, we show that by using IBOL with our LazyUCB oracle optimized for $(1+\epsilon)$-relaxed regret, we can obtain a guarantee of

$$\mathbb{E}[\mathsf{ALG}] \geq \left(1 - \frac{1}{e}\right) \mathsf{OPT} - O(\epsilon \cdot \mathsf{OPT}) - \min\left\{\tilde{O}(\sqrt{\mathsf{OPT}}), \tilde{O}\left(\frac{1}{\epsilon}\right)\right\}, \tag{3}$$

which captures (2) as a special case when $\epsilon = 0$. Although this does not improve the asymptotic guarantee in the worst case, given an estimate of OPT, parameter ϵ can be tuned to maximize the bound in (3) based on the particular constants suppressed by the big-O notation.

LazyUCB: the empirical benefit of UCB with less exploration. We show in numerical simulations that our LazyUCB oracle empirically outperforms traditional UCB; meanwhile, (3) shows that it has a worst-case guarantee parameterized by ϵ that is identical to (2) when $\epsilon = \Theta(1/\sqrt{\mathsf{OPT}})$. This echoes the results in a recent line of work (Bastani et al. 2021, Kannan et al. 2018), who show that (mostly) exploration-free algorithms improve empirical performance while maintaining an asymptotically-optimal theoretical guarantee, for bandits under stochastic contexts. In contrast to these works, we allow for adversarial contexts, and the driving force behind our result is the inventory constraints.

Further simulations on hotel data set. To demonstrate the flexibility of our framework, we also apply it to a dynamic assortment optimization problem in which each resource can be sold at different reward rates. We use the same setup as Ma and Simchi-Levi (2020), except now the choice probabilities must be learned, and we test on the same hotel data set (Bodea et al. 2009).

1.2. Roadmap

In Section 2 we present our general online resource allocation model as well as specific Applications 1 and 2. In Section 3 we define our general IBOL (Inventory Balancing with Online Learning) algorithmic framework, including the parameter $\epsilon \in [0, 1]$. In Section 4 we provide a general performance guarantee for IBOL which depends on ϵ. In Section 5 we derive MAB oracles for the specific Applications 1 (Sections 5.1–5.2) and 2 (Section 5.4), including a proof that these oracles lead to a tight overall performance guarantee for IBOL (Section 5.3). In Section 6 we present experimental results on synthetic instances of Application 1 (Section 6.2) and on a real-world hotel data set (Section 6.1).

Electronic copy available at: https://ssrn.com/abstract=3236533

Table 1 Breakdown of the literature on resource-constrained online allocation. See Section 1.3 for a review.

		Sequence of customer contexts $x^1, x^2, \ldots$	
		(Distributionally) Known	*Unknown Adversarial (must hedge)*
Decisions of customer with context x	*(Distributionally) Known*	Approximation Algorithms	Competitive Analysis
	Unknown i.i.d. (can learn)	Online Learning	**[this paper]**

1.3. Literature Review

We summarize the positioning of our paper in Table 1. Our analysis incorporates the loss from two unknown aspects: the adversarial sequence of customer contexts, and the probabilistic decision for a given customer context. When one or both of these aspects are known, many papers have analyzed the corresponding metrics of interest (competitive ratio, regret, approximation ratio). To our understanding, we are the first to give a unified analysis for online algorithms involving (i) resource constraints, (ii) learning customer behavior, and (iii) adversarial customer arrivals. We now review past work which has considered some subset of these aspects, as outlined in Table 1.

1.3.1. Approximation algorithms. When both the arrival sequence and customer decisions are distributionally known, many algorithms have been proposed for overcoming the "curse of dimensionality" in solving the corresponding dynamic programming problem. Performance guarantees of bid-pricing algorithms were initially analyzed in Talluri and van Ryzin (1998). Later, Alaei et al. (2012) and Wang et al. (2015) proposed new algorithms with improved bounds, for models with time-varying customer arrival probabilities. These performance guarantees are relative to a deterministic LP relaxation (see Section 4) instead of the optimal dynamic programming solution, and hence still represent a form of "competitive ratio" relative to a clairvoyant which knows the arrival sequence in advance (see Wang et al. (2015)).

In addition, the special case in which customer arrival probabilities are time-invariant has been studied in Feldman et al. (2009) and its subsequent research. We refer to Brubach et al. (2016) for discussions of recent research in this direction.

1.3.2. Competitive analysis. We briefly review the literature analyzing the competitive ratio for resource allocation problems under adversarial arrivals. This technique is often called *competitive analysis*, and for a more extensive background, we refer the reader to Borodin and El-Yaniv (2005). For more on the application of competitive analysis in online matching and allocation problems, we refer to Mehta (2013a). For more on the application of competitive analysis in airline revenue management problems, we refer to the discussions in Ball and Queyranne (2009).

Electronic copy available at: https://ssrn.com/abstract=3236533

Our work is focused on the case where competitive analysis is used to manage the consumption of resources. The prototypical problem in this domain is the Adwords problem (Mehta et al. 2007). Often, the resources are considered to have large starting capacities—this assumption is equivalently called the "small bids assumption" (Mehta et al. 2007), "large inventory assumption" (Golrezaei et al. 2014), or "fractional matching assumption" (Kalyanasundaram and Pruhs 2000). In our work, we use the best-known bound that is parametrized by the starting inventory amounts (Ma and Simchi-Levi 2020). The Adwords problem originated from the classical online matching problem (Karp et al. 1990)—see Devanur et al. (2013) for a recent unified analysis. The competitive ratio aspect of our analysis uses ideas from this analysis as well as the primal-dual analysis of Adwords (Buchbinder et al. 2007). We also refer to Devanur and Jain (2012), Kell and Panigrahi (2016), Ma and Simchi-Levi (2020) for recent generalizations of the Adwords problem.

Our model also allows for probabilistic resource consumption, resembling many recent papers in the area starting with Mehta and Panigrahi (2012). We incorporate the *assortment* framework of Golrezaei et al. (2014), where the probabilistic consumption comes in the form of a random customer choice—see also Chen et al. (2016), Ma and Simchi-Levi (2020). However, unlike these papers on assortment planning, our model does not require the substitutability assumption on the choice model, since we allow resources which have ran out to still be consumed for zero reward.

1.3.3. Online learning. The problem of learning customer behavior is conventionally studied in the field of *online learning*. For a comprehensive review on recent advances in online learning, we refer the reader to Bubeck and Cesa-Bianchi (2012), Slivkins (2017).

Our research focuses on online learning problems with resources constraints. Badanidiyuru et al. (2014), Agrawal and Devanur (2014) incorporate resource constraints into the standard multi-armed bandit problem, and propose allocation algorithms with provable upper bounds on the regret. Badanidiyuru et al. (2013), Agrawal and Devanur (2016), Agrawal et al. (2016) study extensions in which customers are associated with independently and identically distributed context vectors; the values of reward and resource consumption are determined by the customer context. Besbes and Zeevi (2009, 2012), Babaioff et al. (2015), Wang et al. (2014), Ferreira et al. (2016) study pricing strategies for revenue management problems, where a resource-constrained seller offers a price from a potential infinite price set to each arriving customer. Customers are homogeneous, in the sense that each customer has the same purchase probability under the same offered price.

Those models with resource constraints in the current literature assume that the type (if there is any) of each customer is drawn from a fixed distribution that does not change over time. As a result, there exists an underlying fixed randomized allocation strategy (typically based on an optimal linear programming solution) that converges to optimality as the number of customers

Electronic copy available at: https://ssrn.com/abstract=3236533

becomes large. The idea of the online learning techniques involved in the above-mentioned research works is to try to converge to that fixed allocation strategy. In our model, however, there is no such fixed allocation strategy that we can discover over time. For instance, the optimal algorithm in our model may reject all the low-fare customers who arrive first and reserve all the resources for high-fare customers who arrive at the end. As a result, the optimal algorithm does not earn any reward at first, and thus cannot be identified as the best strategy by any learning technique. Our analysis is innovative as we construct learning algorithms with strong performance guarantees without trying to converge to any benchmark allocation strategy.

Finally, the LazyUCB oracle proposed in the paper is related to, and inspired by, a recent body of research (Bastani et al. 2021, Kannan et al. 2018) on (mostly) exploration-free approaches for the stochastic contextual multi-armed bandit problem. These works highlight the observation that, in stochastic contextual bandit settings, exploration free algorithms often *empirically* out-perform traditional algorithms such as Upper-Confidence Bound (UCB) and Thompson Sampling (TS). These research works propose theoretical justifications by establishing regret bounds based on certain regularity assumptions on the contextual vectors and the latent parameters. While the theoretical guarantees for the (mostly) exploration-free approaches established in Bastani et al. (2021), Kannan et al. (2018) are no better than the best-known theoretical guarantee for the stochastic contextual bandit problem, the authors demonstrate that their proposed algorithms are consistently superior to traditional algorithms in terms of the empirical performance.

Similar to these works, our proposed LazyUCB oracle reduces the amount of exploration in existing UCB algorithms. However, our work differ from Bastani et al. (2021), Kannan et al. (2018) in three ways. First, our LazyUCB oracle still includes an exploration bonus in its computation of upper-confidence intervals, while Bastani et al. (2021), Kannan et al. (2018) require full exploitation and no exploration. Second, we allow the contextual information of different customers to vary arbitrarily and adversarially without any assumption on how the contextual information varies among customers. By contrast, Bastani et al. (2021), Kannan et al. (2018) require the contextual information of different customers to be drawn i.i.d. from a latent probability distribution, satisfying certain regularity assumptions, in order for the theoretical guarantees to hold. Third, we consider an inventory-constrained setting, while Bastani et al. (2021), Kannan et al. (2018) consider settings without any constraint on the choices of arms.

2. Model Formulation

Throughout this paper, we let $\mathbb{N}$ denote the set of positive integers. For any $n \in \mathbb{N}$, let $[n]$ denote the set $\{1, 2, \ldots, n\}$.

Electronic copy available at: https://ssrn.com/abstract=3236533

We consider the following class of online resource allocation problems. An online platform has a collection of $n \in \mathbb{N}$ resources, denoted $[n]$, to be allocated to $T \in \mathbb{N}$ customers who arrive sequentially. For each $i \in [n]$, the platform has $b_i \in \mathbb{N}$ units of resource i, that are not replenishable during the allocation period. Each unit of resource i is associated with reward r_i normalized to lie in $(0, 1]$. In Sections 6.1 and E, we consider a generalized setting where each resource is associated with multiple reward values as in Ma and Simchi-Levi (2020).

We now define the notation regarding an allocation to a customer. Each customer is associated with a context x, and a context carries personal information about the customer. We denote $\mathcal{X}$ as the set of all possible contexts. The set $\mathcal{X}$ is finite and is known to the online platform. The variation among contexts models the heterogeneity among the customers, and the context sequence is generated adversarially. There is an action set $\mathcal{A}$, which represents the set of allocations decisions. Each pair of $x \in \mathcal{X}$ and $a \in \mathcal{A}$ is associated with an outcome distribution $\rho_{x,a}$, which is a probability distribution over $\{0, 1\}^n$. For each $\mathbf{y} \in \{0, 1\}^n$, we let $\rho_{x,a}(\mathbf{y})$ denote the probability that the outcome if $\mathbf{y}$.

Dynamics. The platform interacts with the customers in T discrete time steps. For each $t \in \{0, 1, \ldots, T\}$ and each $i \in [N]$, we denote N_i^t as the number of units of resource i that have been consumed by the *end* of time t. In particular, we have $N_i^0 = 0$ for all i. At time step $t \in [T]$, four events happen. First, customer t arrives, and their context x^t is revealed to the platform. Second, the platform selects an action $a^t \in \mathcal{A}$, based on x^t and the observations in time steps $1, \ldots, t-1$. Third, the platform observes the vectorial outcome $\mathbf{y}^t = (\mathbf{y}_i^t)_{i \in [n]} \in \{0, 1\}^n$, which is distributed according to the distribution ρ_{x^t, a^t}.[2] Fourth, if $\mathbf{y}_i^t = 1$ and resource i is not yet depleted ($N_i^{t-1} < b_i$), then one unit of the inventory of resource i is consumed ($N_i^t = N_i^{t-1} + 1$), and a reward of r_i is earned. If $\mathbf{y}_i^t = 1$ but resource i is depleted, or if $\mathbf{y}_i^t = 0$, then no resource i is consumed ($N_i^t = N_i^{t-1}$) and no reward is earned.

It is worth noting that the feedback $\mathbf{y}^t$ at each time t is a partial feedback, which is more precisely known as *bandit feedback* in the online learning literature. The feedback is partial in the sense that the platform only observes $\mathbf{y}^t$ under the action a^t, but it does not observe the feedback under any other actions.

Model Uncertainty. The online allocation problem involves model uncertainty in two dimensions. First, the sequence of contexts $\{x^t\}_{t=1}^T$ is generated by an oblivious adversary, who cannot see any information related to $\{a^t\}_{t=1}^T, \{\mathbf{y}^t\}_{t=1}^T$. In particular, the contexts $x^1, x^2, \ldots, x^T$ do not

[2] The outcome $\mathbf{y}^t$ is described more precisely as follows. Before the online process, for each $x \in \mathcal{X}, a \in \mathcal{A}$ the nature generates i.i.d. samples $\{\mathbf{y}_{x,a}^s\}_{s=1}^\infty$. At time t, when the context is x^t and action a^t is chosen, the nature reveals $\mathbf{y}_{x^t,a^t}^{n^t}$ as the outcome, where $n^t = \sum_{s=1}^t \mathbf{1}(x^s = x, a^s = a)$ is the number of occurrences of (x, a) from time 1 to t.

Electronic copy available at: https://ssrn.com/abstract=3236533

generally come from any fixed distribution. Instead, they could vary arbitrarily. The adversarial uncertainty models the volatile and the unpredictable nature of customer arrivals in e-service operations settings.

Second, for each $x \in \mathcal{X}, a \in \mathcal{A}$, the probability distribution $\rho_{x,a}$ is not known to the platform. Rather, the platform has to learn the distribution for each x, a during the online process. It is of interest to learn the latent parameter

$$p_{x,a,i} = \mathbb{E}_{\mathbf{y}_{x,a} \sim \rho_{x,a}}[\mathbf{y}_{x,a,i}] = \sum_{\mathbf{y} \in \{0,1\}^n} \rho_{x,a}(\mathbf{y}) \mathbf{y}_i,$$

which is the probability that the outcome for resource i is 1, when the context is x and the action is a. The Bernoulli random variables $\mathbf{y}_{x,a,1}, \mathbf{y}_{x,a,2}, \ldots, \mathbf{y}_{x,a,n}$ can be correlated in general.

Altogether, our online resource allocation model requires the platform to *hedge* against the adversarial uncertainty of customers' contexts $\{x^t\}_{t=1}^T$, while simultaneously balancing the *explore vs. exploit* tradeoff on the uncertainty in the stochastic model $\{\rho_{x,a}\}_{x \in \mathcal{X}, a \in \mathcal{A}}$. The main thesis of this work is about how the platform manages the three-way trade-off among hedging, exploration, and exploitation.

Objective. The platform's objective is to maximize the total expected revenue. Mathematically, the platform maximizes

$$\mathbb{E}\left[\sum_{i \in [n]} r_i \sum_{t \in [T]} \mathbf{y}_i^t \cdot \mathbf{1}(N_i^{t-1} < b_i)\right] = \mathbb{E}\left[\sum_{i \in [n]} r_i \min\left\{b_i, \sum_{t \in [T]} \mathbf{y}_i^t\right\}\right].$$

The platform is subject to the inventory constraints that at most b_i units of resource i are consumed for each $i \in [n]$. The expectation is taken over the randomness in the actions $a^1, \ldots, a^T$ and the stochastic outcomes $\mathbf{y}^1, \ldots, \mathbf{y}^T$.

Finally, we relate our online resource allocation model to the existing literature. If the probability distributions $\{\rho_{x,a}\}_{x \in \mathcal{X}, a \in \mathcal{A}}$ are known to the platform, then we essentially recover the setting in Golrezaei et al. (2014).[3] If in addition the outcome $\mathbf{y}$ is deterministic given x, a (that is, $\rho_{x,a}$ is the distribution for a deterministic random variable for each x, a), we recover the Adwords problem of Mehta et al. (2007).

Applications. Our problem represents a generic resource allocation model with general context set $\mathcal{X}$ and action set $\mathcal{A}$. For a concrete discussion, we consider the following two specializations of $[n], [T], \mathcal{X}, \mathcal{A}, \rho$, which capture important applications in e-service operations. We elaborate on these applications in Sections 2.1, 2.2, and summarize these applications in Table 2.

[3] The model in Golrezaei et al. (2014) uses the language of assortment optimization, but it is not hard to abstract it to match our resource allocation setting.

Electronic copy available at: https://ssrn.com/abstract=3236533

2.1. Application 1: Internet advertising / Crowd-sourcing

This model is based on the Online Matching with Stochastic Rewards problem of Mehta and Panigrahi (2012), except there could be $K > 1$ probabilities associated with each offline vertex, and these probabilities must be learned.

Internet advertising. The first application concerns the dynamic allocation of internet advertisement from advertisers to web surfers, with the objective of maximizing the total pay-per-click (Mehta and Panigrahi 2012, Mehta 2013b, Goyal and Udwani 2019). The advertisers are modeled as the resources $[n]$, and the web-surfers are modeled as the customers $[T]$, who arrive sequentially at the platform during a certain planning horizon, say during a day. Each advertiser $i \in [n]$ is willing to spend at most $b_i \cdot r_i$ dollars for receiving clicks on their advertisements.

The context set is $\mathcal{X} = \{0,1\}^n$. A customer with context x only clicks on an advertisements from advertisers in $\{i : x_i = 1\}$. Therefore, it is sensible to match a customer with context x with advertiser i only if $x_i = 1$. For example, with a customer who is known to have recently purchased an android phone, it is sensible for the platform to allocate an advertisement on complementary products such as phone accessory, but not an advertisement on another android phone.

Next, we describe the action set $\mathcal{A}$. Each advertiser has $K \in \mathbb{N}$ different advertisements, e.g., K videos/banners. The action set is $\mathcal{A} = \{(i,k) : i \in [n], k \in [K]\}$. When the action (i,k) is taken, it means that the platform allocates the k'th advertisement of advertiser i to the customer. The resulting click probability is $\mathbf{1}(x_i = 1)q_{(i,k)}$. The quantity $q_{(i,k)}$ is latent, whereas the quantity $\mathbf{1}(x_i = 1)$ is not since the context is revealed before an action is chosen. Collectively, the probability model $\{\rho_{x,(i,k)}\}_{x\in\{0,1\}^n, i\in[n], k\in[K]}$ is defined as

$$\begin{aligned} &\rho_{x,(i,k)}(\mathbf{e}_i) = \mathbf{1}(x_i = 1) \cdot q_{(i,k)}, \\ &\rho_{x,(i,k)}(\mathbf{0}) = 1 - \rho_{x,(i,k)}(\mathbf{e}_i), \\ &\rho_{x,(i,k)}(\mathbf{y}) = 0 \text{ for all other outcomes } \mathbf{y} \text{ in } \{0,1\}^n. \end{aligned} \tag{4}$$

There are Kn many latent terms $\{q_{(i,k)}\}_{i\in[n],k\in[K]}$ to be learned.

The platform earns a revenue of r_i when an advertisement from advertiser i is clicked, and the platform earns nothing if there is no click. The objective is to maximize the total expected revenue based on the pay-per-click, subject to the budget constraints of the advertisers. The adversarial uncertainty on $\{x^t\}_{t=1}^T$ reflects that web-surfers arrivals are highly volatile, and they are influenced by so many different factors that they are hard to be precisely forecast. The model uncertainty on $\{\rho_{x,a}\}_{x\in\mathcal{X},a\in\mathcal{A}}$ reflects that customers' tastes have to be learned during the planning horizon.

Crowd-sourcing. The same mathematical model on $[n], [T], \mathcal{X}, \mathcal{A}, \rho$ captures a class of crowd-sourcing problems (Ho and Vaughan 2012, Karger et al. 2014).[4] We interpret $[n]$ as a collection

[4] Nevertheless, we still refer to the above model as the online advertisement allocation problem.

Electronic copy available at: https://ssrn.com/abstract=3236533

of task owners, who pose their tasks on an online crowd-sourcing platform, for example Amazon Mechanical Turk. Each task owner $i \in [n]$ has K tasks to be completed. The workers, represented as $[T]$, arrive at the online platform sequentially. A worker with context $x \in \mathcal{X} = \{0,1\}^n$ is only capable of accomplishing tasks from task owners in $\{i : x_i = 1\}$.

When the action $(i,k) \in \mathcal{A} = \{(i,k)\}_{i\in[n],k\in[K]}$ is taken with a customer of context x, it means that task k from task owner i is assigned to the customer. The outcome $\mathbf{y}_{x,(i,k)}$ is either the task owner i has their task accomplished ($\mathbf{y}_{x,(i,k)} = \mathbf{e}_i$) or no task is accomplished ($\mathbf{y}_{x,(i,k)} = \mathbf{0}$), according to the probability distribution ρ in equations (4). If a worker successfully accomplishes a task from owner i, the worker earns a reward r_i, otherwise the worker does not earn any reward. The crowd-sourcing platform acts as a welfare maximizer, who aims to maximize the total amount of revenue earns by the workers in order to encourage participation into the platform.

2.2. Application 2: Personalized Product Recommendation with Customer Segmentation.

Our model also applies when an inventory constrained seller conducts sales to a pool of heterogeneous customers. In contrast to Application 1, here the probabilities for successfully allocating the resources depend on the customer segment. These can be learned over time as there are repeated customers from the same segments. The seller has n types of products, denoted $[n]$. They have b_i units of product i for each $i \in [n]$, which are not replenishable during the planning horizon. There are T customers, collectively denoted as $[T]$, who arrives at the seller's platform sequentially. The customers are heterogeneous, and they are segmented in terms of the customers' characteristics, such as their gender, age and occupation.

The context set $\mathcal{X}$ denotes the set of all customer segments. The action set $\mathcal{A} = [n]$ corresponds to the set of products. Based on the observed customer segment x, the seller recommends to the customer a product i, which corresponds to action i. The outcome $\mathbf{y} \in \{0,1\}^n$ is equal to $\mathbf{e}_i$ (the customer buys the recommended product) with probability $p_{x,i}$, and is equal to $\mathbf{0}$ (the customer does not buy) with the complementary probability $1 - p_{x,i}$. Altogether, $\rho_{x,i}(\mathbf{e}_i) = p_{x,i} = 1 - \rho_{x,i}(\mathbf{0})$. The resulting expected revenue is $r_i p_{x,i}$. The platform's objective is to maximize the total expected revenue, subject to the inventory constraints and the model uncertainty on $\{x^t\}_{t\in[T]}$, $\{\rho_{x,i}\}_{x\in\mathcal{X},i\in[n]}$.

3. Online Allocation Algorithm: IBOL

We present a framework in Algorithm 1, called IBOL ("Inventory Balancing with Online Learning"), for solving our online resource allocation problem. The IBOL algorithm involves two inputs: a potential function Ψ, and a multi-armed bandit (MAB) oracle $\{\mathcal{O}^t\}_{t=1}^T$. They are respectively used to hedge against the adversarial uncertainty on $\{x^t\}_{t=1}^T$ and to learn the uncertain model on $\{\rho_{x,a}\}_{x\in\mathcal{X},a\in\mathcal{A}}$. The IBOL algorithm also requires the input of the *exploitation parameter* $\epsilon \in [0,1]$,

Electronic copy available at: https://ssrn.com/abstract=3236533

Application	Internet Ad (§ 2.1)	Crowd-sourcing (§ 2.1)	Personalized OM (§ 2.2)
$[n]$	Advertisers	Task owners	Products
$[T]$	Web-surfers	Workers	Customers
$\mathcal{X}$	Compatibility	Compatibility	Customer segments
$\mathcal{A}$	Ad allocation	Task allocation	Product recommendation
ρ	Click probability	Success probability	Purchase probability
Objective	Pay-per-click	Workers' reward	Platform's revenue
To estimate	$\{q_{(i,k)}\}_{i\in[n],k\in[K]}$	$\{q_{(i,k)}\}_{i\in[n],k\in[K]}$	$\{p_{x,i}\}_{x\in\mathcal{X},i\in[n]}$

Table 2 Applications of our online model

Algorithm 1 Inventory-Balancing with Online Learning (IBOL)

Inputs: Inventory levels $\{b_i\}_{i\in[n]}$, exploitation parameter $\epsilon\in[0,1]$, ϵ-perturbed potential function Ψ, MAB oracle $\{\mathcal{O}^t\}_{t\geq 1}$ (see Section 5 for concrete examples of $\{\mathcal{O}^t_\epsilon\}_{t\geq 1}$ on Applications 1, 2).

Initialize: $N_i^0=0$ for all $i\in[n]$.

for $t=1,\ldots,T$ **do**

Observe the feature vector x^t of the customer at time t.

Compute the discounted rewards r_i^t for all i, where

$$r_i^t = r_i\left(1-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right), \tag{5}$$

with Ψ being our ϵ-perturbed potential function in (7).

Compute the action using the online learning algorithm $\mathcal{O}^t$:

$$a^t=\mathcal{O}^t(\mathcal{F}^{t-1},x^t,(r_i^t)_{i\in[n]};U). \tag{6}$$

Observe the feedback $\mathbf{y}^t_{x^t,a^t}=(\mathbf{y}^t_{x^t,a^t,i})_{i\in[n]}$.

For each $i\in[n]$, if $\mathbf{y}^t_{x^t,a^t,i}=1$ and $N_i^{t-1}<b_i$, the platform earns r_i, and depletes 1 unit of resource i: $N_i^t=N_i^{t-1}+1$. Otherwise, the platform earns nothing, and $N_i^t=N_i^{t-1}$.

Update the history $\mathcal{F}^{t+1}=\mathcal{F}^t\cup(x^t,a^t,\mathbf{y}^t)$.

end for

which affects both our potential function Ψ and MAB oracle $\{\mathcal{O}^t\}_{t\geq 1}$. On a high level, when the exploitation parameter ϵ increases, the algorithm conducts more exploitation but less exploration on the latent model $\{\rho_{x,a}\}_{x\in\mathcal{X},a\in\mathcal{A}}$. A precise description about the role of ϵ and how to set it is deferred to Section 4, where we discuss the performance guarantee for IBOL.

Definition of ϵ-perturbed Potential Function Ψ. A potential function is a standard tool used in online resource allocation to generate discounted rewards which guide an algorithm's optimization. More specifically, $\Psi:[0,1]\to[0,1]$ is a non-decreasing function satisfying $\Psi(0)=0,\Psi(1)=$

Electronic copy available at: https://ssrn.com/abstract=3236533

1. The rewards of resources are discounted by a factor of $1-\Psi(N_i^{t-1}/b_i)$, to penalize the allocation of resources which have been overutilized relative to their starting amounts (recall that N_i^{t-1} denotes the units of resource i allocated by the start of a time t), in anticipation of the adversarial contexts $x^1, x^2, \ldots$. Note that if $N_i^{t-1} = b_i$, i.e. resource i is depleted, then the discounted reward is 0, which is consistent with the assumption that no reward is earned from depleted resources.

For $\epsilon \in [0,1]$, we will be using a new "ϵ-perturbed" potential function

$$\Psi(x) = \frac{e^{(1+\epsilon)x} - 1}{e^{1+\epsilon} - 1}, \tag{7}$$

which recovers the classical potential function of $\Psi(x) = \frac{e^x - 1}{e-1}$ from Mehta et al. (2007) when $\epsilon = 0$. Our ϵ-perturbed potential function is designed to maximize the competitive ratio of the online algorithm when its reward has been reduced by ϵ. For $\epsilon > 0$, our ϵ-perturbed Potential Function is steeper than the classical potential function, i.e. it places a relatively greater penalty on almost-depleted resources. This is intuitive, because in our generalized problem where probabilities must be learned, there is relatively less reason to select almost-depleted resources, since there is less benefit to learning the probabilities for such resources.

Given a potential function Ψ, a standard approach (see e.g. Golrezaei et al. 2014) for an online algorithm is to play at each period t an action $a_*^t \in \mathcal{A}$ which maximizes the discounted reward

$$\begin{aligned} R^t(a) &= \sum_{i\in[n]} r_i \left[1 - \Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right] \sum_{\mathbf{y}\in\{0,1\}^{[n]}} \rho_{x^t,a}(\mathbf{y})\mathbf{y}_i \\ &= \sum_{i\in[n]} r_i \left[1 - \Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right] \cdot p_{x^t,a,i}. \end{aligned}$$

However, in our generalized problem the probabilities $p_{x,a,i}$ are unknown, making the existing approach unapplicable. Instead, the platform needs to simultaneously: explore to learn the probabilities $p_{x,a,i}$, while maximizing $R^t(a)$, and also hedging against the adversarial uncertainty on $\{x^t\}_{t=1}^T$. We now formulate an *auxiliary problem*, new to our work, which captures this three-way trade-off between hedging, exploration, and exploitation.

Definition of Auxiliary Problem. Our auxiliary problem is a contextual stochastic bandit problem with the same context set $\mathcal{X}$, action set $\mathcal{A}$, and distributions $\{\rho_{x,a}\}_{x\in\mathcal{X},a\in\mathcal{A}}$ as the online resource allocation problem. The distributions $\{\rho_{x,a}\}_{x\in\mathcal{X},a\in\mathcal{A}}$ are still unknown from the beginning and have to be learned. An MAB oracle $\{\mathcal{O}^t\}_{t=1}^T$ is a learning algorithm designed for solving the auxiliary problem, where $\mathcal{O}^t$ is used for the decision at time t. In our auxiliary problem, we think about four events happening at each time t. First, customer t arrives, and the platform is provided with the context x^t and the discounted reward defined as

$$r_i^t = r_i\Big(1 - \Psi(\frac{N_i^{t-1}}{b_i})\Big).$$

Electronic copy available at: https://ssrn.com/abstract=3236533

Second, the platform chooses an action $a^t \in \mathcal{A}$ using the oracle function $\mathcal{O}^t$. Mathematically, it is expressed as $a^t = \mathcal{O}^t(\mathcal{F}^{t-1}, x^t, r^t)$, where $\mathcal{F}^{t-1} = \{(x^s, a^s, \mathbf{y}^s)\}_{s=1}^{t-1}$ consists of the observations in time steps $1, \ldots, t-1$. Third, the platform observes the vectorial outcome $\mathbf{y}^t$ that is distributed according to ρ_{x^t, a^t}. Fourth, the platform receives the reward $\sum_{i \in [n]} r_i^t \cdot \mathbf{y}_i^t$. Overall, the platform aims to design an MAB oracle that maximizes the total expected reward

$$\mathbf{E}\left[\sum_{t\in[T]}\sum_{i\in[n]} r_i^t \cdot \mathbf{y}_i^t\right] = \mathbf{E}\left[\sum_{t\in[T]} \mathbf{E}\left[\sum_{i\in[n]} r_i^t \cdot \mathbf{y}_i^t \mid x^t, a^t, r_i^t\right]\right] = \mathbf{E}\left[\sum_{t=1}^{T} R^t(a^t)\right].$$

Justification for Auxiliary Problem. Our auxiliary problem can be seen as a way of "abstracting" the inventory constraints away from a typical contextual bandit problem, by introducing the discounted rewards r_i^t. In contrast to traditional "Bandits with Knapsacks" approaches (reviewed in Section 1.3), which do not have adversarial contexts, we define these rewards based on the potential function and the current resource consumption at time t, to hedge against the adversarial contexts. This leads to a contextual bandit problem with *non-stationary* rewards, in which the optimal action a_*^t changes across time not only because of changes in the context x^t, but also because of the non-stationarity of r_i^t over time. Moreover, this change in r_i^t is adaptively influenced by the platform's decisions in time $1, \ldots, t-1$, and is difficult to control. Therefore, in our abstracted contextual bandit problem, we simply allow the values of r_i^t to be generated by an adaptive adversary, who can decide them based on historical information. A key part of our analysis is then to show that this is the "correct" learning problem to focus on, where inventory is unconstrained but regret is measured with respect to these non-stationary rewards r_i^t, instead of the the original problem where inventory was constrained but the rewards r_i were fixed.

A priori, it might appear that, for the auxiliary problem, a learning algorithm needs to deviate from the traditional stochastic MAB framework and to adapt to the reward non-stationarity, in the same vein as the existing literature on non-stationary stochastic bandits (e.g. Garivier and Moulines 2011, Gur et al. 2014). Nevertheless, in subsequent Sections, we demonstrate that it is still possible to adapt the existing stationary stochastic MAB tools, despite the auxiliary problem's non-stationarity, by decoupling the non-stationarity of r_i^t from the learning problem on ρ, in the contexts of Applications 1, 2. This is important for our solution of the auxiliary problem.

$(1+\epsilon)$-Relaxed Regret in the Auxiliary Problem. The rewards for our auxiliary problem were generated by a "perturbed" potential function Ψ which aimed to maximize the competitive ratio of an online algorithm whose reward has been reduced by a factor of ϵ. To compensate, in the auxiliary problem, the algorithm's reward is boosted by a factor of ϵ and we aim to minimize the notion of $(1+\epsilon)$-*relaxed regret:*

$$\mathrm{Reg}_\epsilon = \mathbb{E}\left[\sum_{t=1}^{T} R^t(a_*^t) - (1+\epsilon)R^t(a^t)\right]. \tag{8}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

Recall that a_*^t denotes the action that maximizes the discounted reward $R^t(a)$ among all actions a. When we set $\epsilon = 0$ in the definition of $(1+\epsilon)$-relaxed regret in (8), we recover the classical notion of regret, Reg_0. We elaborate on how ϵ affects the performance of IBOL in Section 4.

The value of ϵ affects our choice of MAB oracle, because the goal of the MAB oracle is to make Reg_ϵ in (8) as small as possible in the worst case. A salient difference in our case when $\epsilon > 0$ is that a sublinear Reg_ϵ can be attained by identifying an action that ϵ-optimal, instead of needing to eventually learn what the exact optimal action is. For a given input $\epsilon \in [0,1]$, we design MAB oracles that are variants of UCB optimized for Reg_ϵ, in the context of Applications 1, 2. These variants coincide with a traditional UCB algorithm when $\epsilon = 0$. On the other hand, when $\epsilon \in (0,1]$, these variants end up being more greedy than a traditional UCB algorithm, in the sense that they conduct less exploration but more exploitation. Thus, we refer to our variants using "LazyUCB".

The design of learning algorithms with less exploration than traditional approaches is similar in spirit to the recent papers by Bastani et al. (2021), Kannan et al. (2018), who show that less exploration leads to empirically better algorithms. As shown in our numerical experiments in Section 6, our proposed greedy variants also achieve better empirical performances than the traditional approaches, in our setting where there are inventory constraints under unknown contexts. Altogether, the motivation for perturbing both our potential function and notion of regret by ϵ is that the overall performance can be improved, when the MAB algorithm is essentially "borrowing" an ϵ-share of the reward from the potential function, and then both of these are re-optimized.

4. Analysis of the IBOL Algorithm

In this section, we bound the performance of our IBOL algorithm, with parameter ϵ, in terms of the $(1+\epsilon)$-relaxed regret incurred by its underlying MAB oracle. In the next section we develop MAB oracles which specifically minimize $(1+\epsilon)$-relaxed regret for Applications 1, 2.

LP Upper Bound. We compare the performance of the IBOL algorithm to a benchmark defined by a linear program (LP) called **Primal**. Our benchmark is the optimal value OPT of LP **Primal**, which upper bounds the total expected reward of any algorithm that knows both $\{\rho_{x,a}\}_{x\in\mathcal{X},a\in\mathcal{A}}$ and $\{x^t\}_{t=1}^T$ before the process begins. The formulation of this linear program benchmark for resource allocation is standard in the revenue management literature, and we formulate it below:

$$\textbf{Primal:} \quad \max \sum_{t\in[T]}\sum_{a\in\mathcal{A}} s_{a,t}\cdot\left[\sum_{i\in[n]} r_i p_{x^t,a,i}\right] \tag{9}$$

$$\text{s.t.} \sum_{t\in[T]}\sum_{a\in\mathcal{A}} s_{a,t}\cdot p_{x^t,a,i} \le b_i \qquad \forall i\in[n] \tag{10}$$

$$\sum_{a\in\mathcal{A}} s_{a,t} = 1 \qquad \forall t\in[T] \tag{11}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

$$s_{a,t} \geq 0 \qquad \forall a \in \mathcal{A}, t \in [T]. \tag{12}$$

The LP **Primal** serves as a fluid relaxation of the constrained online problem. As modeled by the constraints (11, 12), the variable $s_{a,t}$ represents the unconditional probability of an algorithm taking action a in period t. Consequently, the objective (9) of **Primal** is to maximize the total expected revenue. The set of constraints (10) only requires the resource constraints to be satisfied in expectation, which is a relaxation to the online problem.

LEMMA 1. *For any online algorithm that satisfies the resource constraints* $\sum_{t=1}^{T} \mathbf{y}_i^t \leq b_i$ *for all* $i \in [n]$ *with certainty, its total expected reward is at most* OPT.

For completeness we provide a proof of Lemma 1 in Appendix A.1.

Performance Guarantee. Equipped with Lemma 1, we are now ready to compare the total expected reward $\mathbb{E}[\mathsf{ALG}]$ collected by the IBOL algorithm to the benchmark OPT. Theorem 1 below maintains the generality of the algorithmic framework in Section 3, in that the performance guarantee holds for the general online resource allocation problem, not just Applications 1, 2, and the performance guarantee also holds for any MAB oracle.

THEOREM 1. *For any* $\epsilon \in [0,1]$*, the total reward* ALG *earned by the IBOL algorithm, using our* ϵ*-perturbed potential function* $\Psi(x) = \frac{e^{(1+\epsilon)x}-1}{e^{1+\epsilon}-1}$*, satisfies*

$$\mathbb{E}[\mathsf{ALG}] \geq f_\Psi(\{b_i\}_{i\in[n]}, \epsilon) \cdot \left(\mathsf{OPT} - \mathbb{E}\left[\sum_{t\in[T]} R^t(a_*^t) - (1+\epsilon) R^t(a^t) \right] \right), \tag{13}$$

where

$$f_\Psi(\{b_i\}_{i\in[n]}, \epsilon) = \min_{i\in[n]} \left\{ \frac{1-e^{-(1+\epsilon)}}{(b_i+1+\epsilon)(1-e^{-(1+\epsilon)/b_i})} \right\}. \tag{14}$$

The proof of Theorem 1 is deferred to Appendix A.2. The expected reward $\mathbb{E}[\mathsf{ALG}]$ of the algorithm is smaller than OPT in two ways: first, it is scaled down by the competitive ratio $f_\Psi(\{b_i\}_{i\in[n]}, \epsilon)$ which is less than 1; there is also an additive loss of the term $\mathbb{E}\left[\sum_{t\in[T]} R^t(a_*^t) - (1+\epsilon)R^t(a^t)\right]$ which denotes the $(1+\epsilon)$-relaxed regret of the MAB oracle. We note that choosing a larger ϵ in [0,1] for our IBOL algorithm will cause the competitive ratio $f_\Psi(\{b_i\}_{i\in[n]}, \epsilon)$ to decrease, but in return, the $(1+\epsilon)$-relaxed regret will be smaller.

Justification for Form of Performance Guarantee. Our guarantee (13) measures the regret in comparison to OPT after it has been scaled down by $f_\Psi(\{b_i\}_{i\in[n]}, \epsilon)$. We now explain why a meaningful (i.e. sublinear) regret is impossible in our setting if we do not scale down OPT.

The competitive ratio $f_\Psi(\{b_i\}_{i\in[n]}, \epsilon)$ is at its maximum for any vector $\{b_i\}_{i\in[n]}$ when $\epsilon = 0$, in which case it can be re-expressed based on $b_{\min} = \min_{i\in[n]} b_i$ as

$$f_\Psi(\{b_i\}_{i\in[n]}, 0) = \min_{i\in[n]} \left\{ \frac{1-e^{-1}}{(b_i+1)(1-e^{-1/b_i})} \right\} = \frac{1-1/e}{(1+b_{\min})(1-e^{-1/b_{\min}})}. \tag{15}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

Importantly, expression (15) represents the *maximum fraction of* OPT *that can be obtained by any online algorithm* when there are both adversarial contexts $x^1, x^2, \ldots$ and capacity limits $\{b_i\}_{i\in[n]}$. This fraction increases from $1/2$ to $1-1/e$ as $b_{\min}$ increases from 1 to ∞. The asymptotic ratio of $1-1/e$ has been shown to be best-possible by Mehta et al. (2007), with the expression given in (15) denoting the best-known dependence on $b_{\min}$ due to Ma and Simchi-Levi (2020). That is, even if all of the underlying probabilities $p_{x,a,i}$ are known and there is *nothing to learn*, an online algorithm still cannot earn a fraction of OPT greater than (15), which lies in [0.5, 0.632]. Consequently, if one attempts to directly measure the regret $\mathsf{OPT} - \mathbb{E}[\mathsf{ALG}]$, then a regret sub-linear in OPT is impossible.

This is why we measure regret in comparison to $f_\Psi(\{b_i\}_{i\in[n]}, \epsilon) \cdot \mathsf{OPT}$. In fact, we show this form of performance guarantee, with both a multiplicative and additive loss term like in (13), to be *tight* for our Application 1 corresponding to online matching, in Section 5.3.

Tuning the ϵ Parameter. We let ϵ to be a parameter in [0,1], instead of fixing $\epsilon = 0$, because it allows our algorithmic framework IBOL to balance between the two aforementioned losses caused by the competitive ratio $f_\Psi(\{b_i\}_{i\in[n]}, \epsilon)$ and the $(1+\epsilon)$-relaxed regret. When ϵ increases, IBOL focuses on maximizing a more stringent competitive ratio but minimizing a $(1+\epsilon)$-relaxed regret, which causes its Inventory Balancing part to place a greater penalty on almost-depleted resources (through an ϵ-perturbed potential function), and its Online Learning part to explore less (through our LazyUCB oracle).

Although there is no notion of "optimal ϵ" given a problem instance due to the unknown probabilities and adversarial contexts, our Theorem 1 provides a plausible method for setting ϵ, based on maximizing its worst-case guarantee on $\mathbb{E}[\mathsf{ALG}]$. Denote $\mathsf{BD}(\epsilon)$ as the upper bound on the $(1+\epsilon)$-relaxed regret of the underlying MAB oracle. An appropriate value of ϵ can then be found by solving the following tuning optimization problem, formulated below based on equation (13):

$$\max_{\epsilon\in[0,1]} \left\{ f_\Psi(\{b_i\}_{i\in[n]}, \epsilon) \cdot (\mathsf{OPT} - \mathsf{BD}(\epsilon)) \right\}. \tag{16}$$

Although we have stated the tuning optimization problem for a general MAB oracle, in the next section we show how the regret bound $\mathsf{BD}(\epsilon)$ materializes for different values of ϵ, and we define the optimization problem over ϵ for Application 1 at the end of Section 5.2. We also remark that the formulation of (16) involves knowing the value of OPT, and our upper bounds $\mathsf{BD}(\epsilon)$ on regret will involve knowing the value of T. The assumptions of knowing OPT, T could be justified when the optimal total reward and the number of customers can be estimated based on historical instances. This provides a method for optimizing ϵ against the worst case, using less information than the full knowledge of $\rho, \{x^t\}_{t=1}^T$. Of course, if one had full knowledge of $\rho, \{x^t\}_{t=1}^T$, then they could tune ϵ

Electronic copy available at: https://ssrn.com/abstract=3236533

using simulation instead of using our bound, but the full knowledge assumption is much stronger than only needing an estimate of OPT and T.

Furthermore, we empirically find that setting ϵ to be larger, usually 1, will improve performance, justifying the benefit of having the tunable parameter ϵ in our IBOL framework. When ϵ is larger, the Online Learning part of IBOL ends up focusing more on exploitation than exploration. Such an insight is in line with the findings in a recent stream of papers (Bastani et al. 2021, Kannan et al. 2018) which show that reducing the amount of exploration in conventional MAB algorithms (more specifically, UCB algorithms) leads to better empirical performance, even though these (almost) exploration-free variants do not have a better theoretical performance guarantee than the conventional algorithms.

Re-designing UCB for Worst-case $(1+\epsilon)$-Relaxed Regret. In addition to the tradeoff between the two sources of error, the notion of $(1+\epsilon)$-relaxed regret inspires the design of MAB oracles that differ significantly from the classical approach of UCB. Let's revisit definition (8), and multiply both sides by $1/(1+\epsilon)$:

$$\frac{1}{1+\epsilon}\cdot \mathrm{Reg}_\epsilon = \mathbb{E}\left[\sum_{t=1}^{T} \frac{\max_{\bar{a}_t\in\mathcal{A}} R^t(\bar{a}^t)}{1+\epsilon} - R^t(a^t)\right]. \tag{17}$$

The benchmark $\frac{1}{1+\epsilon}\sum_{t=1}^{T}\max_{\bar{a}_t\in\mathcal{A}} R^t(\bar{a}^t)$ only requires the decision maker to identify an action that is $1/(1+\epsilon)$-optimal, i.e. an action a^t such that $R^t(a^t)\geq \frac{1}{1+\epsilon}\max_{\bar{a}^t\in\mathcal{A}} R^t(\bar{a}^t) = \frac{1}{1+\epsilon}R^t(a^t_*)$, which is an easier task than solving $\max_{\bar{a}^t\in\mathcal{A}} R^t(\bar{a}^t)$. In particular, the former task requires less exploration on ρ than the latter, and suggests that the decision maker could potentially perform less exploration on ρ for achieving near-optimality for the online resource allocation problem.

Altogether, the main message of Theorem 1 is as follows. While we can adapt existing tools such as UCB to construct an MAB oracle for solving the online resource allocation problem, the problem in fact admits a much wider class of MAB oracles for achieving near-optimality. In particular, an MAB oracle that achieves a low $(1+\epsilon)$-relaxed regret for some $\epsilon>0$, which potentially involves less exploration than UCB, also leads us to near-optimality for the online resource allocation problem.

5. MAB Oracles for Applications 1, 2

In the previous sections, we proposed the IBOL algorithm that hedges against adversarial contexts $\{x^t\}_{t=1}^T$ while learning the outcome distribution ρ. In addition, we provided Theorem 1, which related the expected reward of IBOL to the $(1+\epsilon)$-relaxed regret of the underlying MAB oracle. In this section we complete the picture by constructing MAB oracles, which conduct simultaneous exploration-exploitation, to solve the auxiliary problem and to overcome the uncertainty on the outcome distribution ρ. We specialize to the settings of $\rho, \mathcal{X}, \mathcal{A}$ under Applications 1, 2 (as defined in Sections 2.1–2.2) for our construction of MAB oracles.

Electronic copy available at: https://ssrn.com/abstract=3236533

In Section 5.1, we construct the UCB oracle for our Application 1 in the case where $\epsilon = 0$. This UCB oracle is based on the classical UCB approach (Auer et al. 2002a), for which we upper-bound the unrelaxed regret Reg_0 in the auxiliary problem. In Section 5.2, we construct our LazyUCB oracle, which performs less exploration than the UCB oracle, in the case where $\epsilon \in (0, 1]$. We then demonstrate an upper bound to the $(1+\epsilon)$-relaxed regret for LazyUCB, hence showing that it is possible to achieve near-optimality with less exploration than the classical UCB approach. In Section 5.3, we present our negative result establishing tightness in the context of Application 1. In Section 5.4, we show how the machinery developed in Sections 5.1, 5.2 for Application 1 can be generalized to Application 2.

5.1. UCB Oracles for Application 1 ($\epsilon = 0$)

We start with a reminder on Application 1. The action set is $\mathcal{A} = \{(i,k)\}_{i\in[n],k\in[K]}$. When the action (i,k) is taken at time t, where the customer has feature x^t, the feedback $\mathbf{y} \in \{0,1\}^n$ is equal to $\mathbf{1}(x_i^t = 1)\mathbf{e}_i$ with latent probability $q_{i,k}$, and equal to $\mathbf{0}$ with latent probability $1 - q_{i,k}$ (there is only any reason to take an action (i,k) at time t if context $x_i^t = 1$). For the auxiliary problem, the discounted reward at time t under action $a = (i,k)$ is

$$R^t(a) = \sum_{i\in[n]} r_i^t p_{x^t,a,i} = r_i^t \cdot \mathbf{1}(x_i^t = 1) \cdot q_{(i,k)}. \tag{18}$$

The UCB oracle at time t is provided in Algorithm 2. This oracle is to be used on Application 1 when $\epsilon = 0$. The oracle inputs the information $\mathcal{F}^{t-1}, x^t, (r_i^t)_{i\in[n]}$ that are known at the start of time t, and output the action $a^t = (i^t, k^t)$ for the time step. For estimating the latent probability $q_{(i,k)}$ for each $i \in [n], k \in [K]$, we consider

$$M_{(i,k)}^t = \sum_{s=1}^{t-1} \mathbf{1}(a^s = (i,k)), \qquad \bar{q}_{(i,k)}^t = \frac{\sum_{s=1}^{t-1} \mathbf{1}(a^s = (i,k), \mathbf{y}_i^s = 1)}{\max\{M_{(i,k)}^t, 1\}}.$$

The parameter $M_{(i,k)}^t$ counts the number of times the algorithm takes the action (i,k) during time steps $1, \ldots, t-1$. The statistic $\bar{q}_{(i,k)}^t$ serves to estimate the latent parameter $q_{(i,k)}$. For each and every (i,k), the quantities $M_{(i,k)}^t, \bar{q}_{(i,k)}^t$ can be constructed based on the observations $\mathcal{F}^{t-1}$ during time $1, \ldots, t-1$.

While the empirical mean $\bar{q}_{(i,k)}^t$ is a natural estimate to $q_{(i,k)}$, the decision maker needs to quantify the accuracy of the estimate $\bar{q}_{(i,k)}^t$, in order to decide if it wishes to explore other actions' probabilities, or if it wishes to use the estimate $\bar{q}_{(i,k)}^t$ for exploitation. The accuracy of the estimate $\bar{q}_{(i,k)}^t$ is quantified by a confidence radius for the estimate.

In the forthcoming UCB and LazyUCB Oracle, the decision maker conducts simultaneous exploration and exploitation by replacing the latent probability $q_{(i,k)}$ with an *optimistic estimate*, which

Electronic copy available at: https://ssrn.com/abstract=3236533

Algorithm 2 UCB oracle at time t for Application 1, in the case where $\epsilon = 0$

1: Input: observation $\mathcal{F}^{t-1}$ from time 1 to $t-1$, context x^t and discounted rewards $(r_i^t)_{i\in[n]}$.

2: Use $\mathcal{F}^{t-1}$ to compute the statistics $\{M^t_{(i,k)}\}_{i\in[n],k\in[K]}, \{\bar{q}^t_{(i,k)}\}_{i\in[n],k\in[K]}$.

3: Set $\delta_t = \frac{1}{(1+t)^2}$.

4: For each action $a=(i,k)$, compute a UCB for the discounted revenue with action (i,k):

$$\mathsf{UCB}^t(a) = r_i^t \cdot \mathbf{1}(x_i^t = 1) \cdot \left[\bar{q}^t_{(i,k)} + \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}, \delta^{(t)})\right].$$

5: Output an action $a^t = (i^t, k^t)$ which satisfies

$$a^t \in \operatorname*{argmax}_{a=(i,k)\in\mathcal{A}} \mathsf{UCB}^t(a).$$

is equal to the sum of the the empirical mean $\bar{q}^t_{(i,k)}$ (exploitation) and a confidence radius (exploration). The use of an optimistic estimate embodies the famous "optimism in the face of uncertainty" principle in the multi-armed bandit literature. For the UCB oracle, we follow the approach by Auer et al. (2002a), Kleinberg et al. (2008) to define the confidence radius. For $p>0, M\in\mathbb{Z}_{\geq 0}, \delta\in(0,1)$, let

$$\mathsf{rad}(p, M; \delta) := \sqrt{\frac{2p\log(1/\delta)}{\max\{M,1\}}} + \frac{3\log(1/\delta)}{\max\{M,1\}}. \tag{19}$$

In Line 4 in the UCB oracle, we replace the latent $q_{(i,k)}$ with the optimistic estimate $\bar{q}^t_{(i,k)} + \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}, \delta_t)$, where $\delta_t\in(0,1)$ is a confidence parameter defined in Line 3. The definition of rad is justified by the following Lemma.

LEMMA 2 **(Kleinberg et al. (2008))**. *For each $t\in[T]$, consider the event $\mathcal{E}^t = \cap_{i\in[n],k\in[K]}\mathcal{E}^t_{i,k}$, where $\mathcal{E}^t_{(i,k)}$ is defined as*

$$\mathcal{E}^t_{(i,k)} = \left\{\left|\bar{q}^t_{(i,k)} - q_{(i,k)}\right| \leq \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}, \delta_t) \leq 3\cdot\mathsf{rad}(q_{(i,k)}, M^t_{(i,k)}, \delta_t)\right\}.$$

Then we have $\Pr(\mathcal{E}^t) \geq 1 - \frac{4nK}{1+t}$.

Lemma is proved in Appendix B.2. While the Lemma is first proposed in Kleinberg et al. (2008), we still provide the proof to make the constants involved in rad explicit. Lemma 2 is crucial for justifying the UCB in step 4. Indeed, if the event $\mathcal{E}^t$ holds, then for any $a=(i,k)$,

$$\begin{aligned} R^t(a) &= r_i^t \cdot \mathbf{1}(x_i^t = 1)\cdot q_{i,k} \\ &\leq r_i^t \cdot \mathbf{1}(x_i^t = 1)\cdot \left[\bar{q}^t_{(i,k)} + \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}, \delta_t)\right] \\ &= \mathsf{UCB}^t(a). \end{aligned} \tag{20}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

Therefore, the quantity $\mathsf{UCB}^t(a)$ defined in Line 4 is a bona fide upper bound of the discounted reward $R^t(a)$ with high probability. En route, we show that even though the auxiliary problem involves non-stationary rewards, we can still harness existing machinery on UCB algorithms.

Finally, in Line 5 we choose an action a^t that maximizes the UCB. Without loss of generality, we assume that $x^t_{it} = 1$. To demonstrate the salience of the UCB oracle, we bound its regret $\mathrm{Reg}_0 = \mathbb{E}\left[\sum_{t=1}^T R^t(a^t_*) - R^t(a^t)\right]$ in the theorem below.

THEOREM 2. *Consider the UCB oracle (Algorithm 2) for Application 1. The oracle has regret*

$$Reg_0 = O\left(\sqrt{nK \cdot \mathsf{OPT} \cdot \log(T)} + nK \log(T) \log \frac{T}{nK}\right) = \tilde{O}\left(\sqrt{nK \cdot \mathsf{OPT}}\right),$$

where OPT *is the optimal value of the LP* ***Primal****, and the notation* $\tilde{O}(\cdot)$ *hides the logarithmic dependence on* n, K, T*.*

Theorem 2 is proved in Appendix C.1. On a high level, the Theorem is proved by incorporating the analytical tools on UCB algorithms from Auer et al. (2002a), with extra care that yields to dependence on OPT. Clearly, we know that $\mathsf{OPT} = O(T)$, and by replacing OPT with the upper bound $O(T)$, we have $\mathrm{Reg}_0 = \tilde{O}(\sqrt{nKT})$, which coincides with the $\tilde{O}(\cdot)$ bound for an nK-armed bandits problem with T time steps. In passing, we remark that in the large-volume regime (Besbes and Zeevi 2011) where $b_{\min}$ grows linearly with T, we do have $\mathsf{OPT} = cT$ for some constant c that depends on the model but is independent of T. The dependence on OPT provides a more refined guarantee than the dependence on T when $b_{\min} = o(T)$.

Combined with the IBOL algorithm (Algorithm 1), we achieve the following performance guarantee for Application 1.

COROLLARY 1. *For Application 1, the IBOL algorithm (Algorithm 1) with the UCB oracle (Algorithm 2) yields expected reward* $\mathbb{E}[\mathsf{ALG}]$ *that satisfies*

$$\mathbb{E}[\mathsf{ALG}] \geq \frac{1 - 1/e}{(1 + b_{min})(1 - e^{-1/b_{min}})} \cdot \left[\mathsf{OPT} - \tilde{O}\left(\sqrt{nK \cdot \mathsf{OPT}}\right)\right]. \tag{21}$$

As an illustration of Theorem 1, the Corollary illustrates the two sources of error for the online resource allocation problem in Application 1. The competitive ratio $\frac{1-1/e}{(1+b_{\min})(1-e^{-1/b_{\min}})}$ is due to the adversarial uncertainty on $x^1, \ldots, x^T$, and the regret bound $\tilde{O}\left(\sqrt{nK \cdot \mathsf{OPT}}\right)$ is due to the model uncertainty on the probabilities $q_{(i,k)}$.

5.2. LazyUCB Oracles for Application 1 ($\epsilon \in (0, 1]$)

After the construction of the UCB oracle for Application 1, which is for the case where $\epsilon = 0$, we construct the LazyUCB oracle, which is for the case where $\epsilon \in (0, 1]$. The LazyUCB oracle, which involves ϵ, is exhibited in Algorithm 3.

Electronic copy available at: https://ssrn.com/abstract=3236533

Algorithm 3 LazyUCB Oracle at time t for Application 1, in the case where $\epsilon \in (0,1]$

1: Input: exploitation parameter $\epsilon \in (0,1]$, observation $\mathcal{F}^{t-1}$ from time 1 to $t-1$, context x^t and discounted rewards $(r_i^t)_{i\in[n]}$.

2: Use $\mathcal{F}^{t-1}$ to compute the statistics $\{M_{(i,k)}^t\}_{i\in[n],k\in[K]}, \{\bar{q}_{(i,k)}^t\}_{i\in[n],k\in[K]}$.

3: Set $\delta_t = \frac{1}{(1+t)^2}$.

4: For each action $a=(i,k)$, compute a *lazy* UCB for the discounted revenue with action (i,k):

$$\mathsf{LazyUCB}^t(a) = r_i^t \cdot \mathbf{1}(x_i^t = 1) \cdot \left[\bar{q}_{(i,k)}^t + \mathsf{LazyRad}^t(i,k)\right].$$

5: Output an action $a^t = (i^t, k^t)$ which satisfies

$$a^t \in \operatorname*{argmax}_{a=(i,k)\in\mathcal{A}} \mathsf{LazyUCB}^t(a).$$

Similar to the UCB oracle, the LazyUCB oracle also hinges on constructing an optimistic estimate for each latent probability $q_{(i,k)}$. Different from the UCB oracle, however, the LazyUCB oracle employs a smaller confidence radius. Hence, the LazyUCB oracle focuses more on exploitation, and less on exploration in comparison to the UCB oracle. The confidence radius employed by the LazyUCB oracle is shown in Line 4 in the algorithm.

To construct the confidence radii for the LazyUCB oracle, for $\epsilon \in [0,1], M \in \mathbb{Z}_{\geq 0}, \delta \in (0,1)$, we define

$$\mathsf{lad}(\epsilon, M; \delta) = \frac{2+\epsilon}{\epsilon} \cdot \frac{\log(1/\delta)}{\max\{M,1\}}. \tag{22}$$

The optimistic estimate for $q_{(i,k)}$ at time t under the LazyUCB oracle is

$$\bar{q}_{(i,k)}^t + \mathsf{LazyRad}^t(i,k),$$

where we define

$$\mathsf{LazyRad}^t(i,k) = \min\left\{\mathsf{lad}(\epsilon, M_{(i,k)}^t; \delta_t), \mathsf{rad}(\bar{q}_{(i,k)}^t, M_{(i,k)}^t, \delta^{(t)})\right\},$$

with rad as defined in (19). The other confidence radius $\mathsf{lad}(\epsilon, M_{(i,k)}^t; \delta_t)$ can be can be interpreted as follows. The confidence radius lad involves an *exploitation parameter* $\epsilon \in [0,1]$, which controls the amount of exploitation conducted by the LazyUCB oracle. As ϵ increases, $\mathsf{lad}(\epsilon, M; \delta)$ decreases. In particular, when $\epsilon = 1$, we have

$$\mathsf{lad}(\epsilon, M; \delta) < \mathsf{rad}(p, M; \delta)$$

for any p, M, δ. For any $\epsilon > 0$, it is critical to observe that we still have $\mathsf{lad}(\epsilon, M; \delta) \leq \mathsf{rad}(p, M; \delta)$ as long as $p > 0$ and M is sufficiently large, since the dominant term in $\mathsf{rad}(p, M; \delta)$ is of order $\sqrt{p/M}$ while $\mathsf{lad}(\epsilon, M; \delta)$ scales as $1/(\epsilon M)$.

Electronic copy available at: https://ssrn.com/abstract=3236533

In the extreme case where we set $\epsilon = 0$, we have $\mathsf{lad}(\epsilon, M^t_{(i,k)}; \delta) = 1$, and the LazyUCB oracle is reduced to the UCB oracle. In another extreme case when we set ϵ to be 1, we have $\mathsf{lad}(\epsilon, M; \delta) = \frac{3\log(1/\delta)}{\max\{M,1\}}$. It is worth noting that the lazy confidence radius $\mathsf{lad}(\epsilon, M; \delta)$ does not shrink to zero, as we define $\mathsf{lad}(\epsilon, M; \delta)$ in a way that still induces a minute amount of optimistic exploration when ϵ is large. Finally, similar to the UCB oracle, in Line 5 we chooses an action a^t that maximizes the optimistic estimate.

We justify the definition of the lazy confidence radius rad in the following Lemma:

LEMMA 3. *For each t, consider the event $\mathcal{E}^t = \left(\cap_{i\in[n],k\in[K]}\mathcal{U}^t_{(i,k)}\right) \cap \left(\cap_{i\in[n],k\in[K]}\mathcal{L}^t_{(i,k)}\right)$, where $\mathcal{U}^t_{(i,k)}$ is the event*

$$q_{(i,k)} \le \left(1+\frac{\epsilon}{2}\right)\cdot\left[\bar{q}^t_{(i,k)} + \mathsf{LazyRad}^t(i,k)\right], \tag{23}$$

and $\mathcal{L}^t_{(i,k)}$ is the event

$$\bar{q}^t_{(i,k)} \le \left(1+\frac{\epsilon}{2+\epsilon}\right)\left[q_{(i,k)} + \mathsf{LazyRad}^t(i,k)\right]. \tag{24}$$

Then we have $\Pr(\mathcal{E}^t) \ge 1 - \frac{7nK}{1+t}$.

Lemma 3 is proved in Appendix B.4. Inequality (23) shows that the lazy optimistic estimate $\bar{q}^t_{(i,k)} + \mathsf{LazyRad}^t(i,k)$ is "optimistic" in an approximate sense, captured by the multiplicative factor $1+\epsilon/2$. Inequality (24) shows that, despite the approximate nature, the lazy optimistic estimate is still close to the actual latent probability, as quantified in the inequality. It is useful to note that when we set $\epsilon = 0$, we recover Lemma 2 from the original UCB algorithm.

If the event $\mathcal{E}^t$ holds, then for any action $a = (i,k)$, the LazyUCB in Line 4 satisfies

$$\begin{aligned} R^t(a) &= r^t_i \cdot \mathbf{1}(x^t_i = 1)\cdot q_{i,k} \\ &\le r^t_i \cdot \mathbf{1}(x^t_i = 1)\cdot\left(1+\frac{\epsilon}{2}\right)\cdot\left[\bar{q}^t_{(i,k)} + \mathsf{LazyRad}^t(i,k)\right] \\ &= \left(1+\frac{\epsilon}{2}\right)\cdot \mathsf{LazyUCB}^t(a). \end{aligned} \tag{25}$$

We provide the following performance guarantee on the LazyUCB oracle for the auxiliary problem, with the metric of $(1+\epsilon)$-relaxed regret Reg_ϵ.

THEOREM 3. *Consider the LazyUCB oracle (Algorithm 3) for Application 1. The oracle has $\frac{1}{1+\epsilon}Reg_\epsilon$ at most*

$$\begin{aligned} &\min\left\{O\left(\sqrt{nK\cdot\mathsf{OPT}\cdot\log T}\right), O\left(\left(1+\frac{1}{\epsilon}\right)\cdot nK\log(T)\log\left(\frac{T}{nK}\right)\right)\right\} + O\left(nK\log(T)\log\left(\frac{T}{nK}\right)\right) \\ =&\min\left\{\tilde{O}\left(\sqrt{nK\mathsf{OPT}}\right), \tilde{O}\left(\left(1+\frac{1}{\epsilon}\right)\cdot nK\right)\right\}. \end{aligned}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

Theorem 3 is proved in Appendix C.2. While the regret bound in Theorem 3 is smaller than the regret bound for the UCB oracle in Theorem 2, it does not mean that the LazyUCB oracle earns a greater reward on the auxiliary problem than the UCB oracle. It is important to note that Theorems 2, 3 involve different notions of regret. These Theorems together suggest that the auxiliary problem can be solved by a variety of MAB oracles, such as UCB or LazyUCB, but they do not suggest that one oracle is better than the other.

In conjunction with Theorem 1, we arrive at the following performance guarantee for the IBOL algorithm using the LazyUCB oracle.

COROLLARY 2. *For Application 1, the IBOL algorithm (Algorithm 1) with the LazyUCB oracle (Algorithm 3) yields expected reward* $\mathbb{E}[\mathsf{ALG}]$ *that satisfies*

$$\mathbb{E}[\mathsf{ALG}] \geq f_\Psi(\{b_i\}_{i\in[n]}, \epsilon) \cdot \left[\mathsf{OPT} - \min\left\{\tilde{O}\left(\sqrt{nK\mathsf{OPT}}\right), \tilde{O}\left(\left(1+\frac{1}{\epsilon}\right)\cdot nK\right)\right\}\right], \tag{26}$$

where $f_\Psi(\{b_i\}_{i\in[n]}, \epsilon)$ *is defined in equation (14).*

To this end, it is important to note that when we specify $\epsilon = 0$ in Corollary 2, we arrive at the bound in Corollary 1. Indeed, it is crucial to recall that when we set $\epsilon = 0$ in the LazyUCB oracle, we recover the UCB oracle.

We conclude our discussion with two remarks. First, it is useful to compare the performance guarantee under the LazyUCB oracle in Corollary 2 (where $\epsilon \in (0, 1]$) with that under the UCB oracle in Corollary 1 (where $\epsilon = 0$). On one hand, the competitive ratio $f_\Psi(\{b_i\}_{i\in[n]}, 0)$ in Corollary 1 is greater than or equal to the competitive ratio $f_\Psi(\{b_i\}_{i\in[n]}, \epsilon)$ in Corollary 2. On the other hand, the regret term in (26) is less than or equal to the regret term in (21).

Second, going back to the question of tuning ϵ, when OPT, T are known, an appropriate choice for ϵ can be made by solving the optimization problem

$$\min_{\epsilon\in[0,1]}\left\{f_\Psi(\{b_i\}_{i\in[n]}, \epsilon) \cdot \left[\mathsf{OPT} - \min\left\{\tilde{O}\left(\sqrt{nK\mathsf{OPT}}\right), \tilde{O}\left(\left(1+\frac{1}{\epsilon}\right)\cdot nK\right)\right\}\right]\right\}. \tag{27}$$

Although the optimization problem (27) is not convex in ϵ in general, an optimal ϵ can still be identified by a one-dimensional line search on $[0, 1]$. While the exact expressions of $\tilde{O}\left(\sqrt{nK\mathsf{OPT}}\right), \tilde{O}\left(\left(1+\frac{1}{\epsilon}\right)\cdot nK\right)$ are suppressed due to the use of $\tilde{O}(\cdot)$ notation, the optimization problem (27) can be explicitly defined by replacing $\tilde{O}\left(\sqrt{nK\mathsf{OPT}}\right), \tilde{O}\left(\left(1+\frac{1}{\epsilon}\right)\cdot nK\right)$ respectively with their explicit expressions (79), (81) in Appendix C.2.

5.3. Tightness of our Guarantee for Application 1

We conclude our discussion on Application 1 by providing the following negative result on the expected reward achieved by any feasible online algorithm.

Electronic copy available at: https://ssrn.com/abstract=3236533

THEOREM 4. *Let n, b, K be any positive integers satisfying $b \geq K \geq 3$. For any online algorithm that is feasible to Application 1, there exists a problem instance under which*

$$\mathbb{E}[\mathsf{ALG}] \leq \left(1 - \frac{1}{e}\right) \mathsf{OPT} - \Omega(\sqrt{K\mathsf{OPT}}).$$

Theorem 4 is proved in Appendix D. The main message of the theorem is that any feasible online algorithm must suffer a loss in reward from both the adversarial uncertainty on $\{x^t\}_{t=1}^T$ and the model uncertainty on $\{q_{(i,k)}\}_{i\in[n],k\in[K]}$. Our paper is the first to study online problems with both sources of uncertainty in a resource constrained setting. The proof involves crafting a special class of problem instances.

To elaborate, the adversarial uncertainty construction requires having an upper-triangular graph whose ordering of offline vertices is hidden to the online algorithm, in which case it is impossible to do better than arbitrarily "guessing" an offline vertex to probe at each stage. In our combined construction, each offline vertex actually corresponds to $2b$ arms, one of which is a "secret" arm which successfully matches with probability $1/2+\varepsilon$ (instead of $1/2-\varepsilon$) upon a probe. The online algorithm also suffers from not being able to learn the secret arm for each offline vertex, and hence loses an additional ε in the matches made at each stage. However, this means that more offline vertices remain unmatched, making the online algorithm less likely to get stuck in the future. To see that this ε-loss is not later recouped by the online algorithm (in terms of first-order regret) requires an intricate analysis, combining the information-theoretic framework in Auer et al. (2002b) with the Yao's-minimax proof in Mehta et al. (2007). To the best of our understanding, such an analysis is new to our paper, and our negative result is not possible to confirm without this detailed analysis.

5.4. UCB and LazyUCB Oracles for Application 2

Analogous versions of the UCB and LazyUCB oracles for Application 1 can be constructed for Application 2. We start with a reminder on the mathematical model of Application 2. The action set $\mathcal{A}$ is $[n]$. When the context (customer segment) is x and the action is i, the outcome $\mathbf{y}$ is equal to $\mathbf{e}_i$ with probability $p_{x,i}$ and is equal to $\mathbf{0}$ with probability $1-p_{x,i}$. The probability terms in $\{p_{x,i}\}_{x\in\mathcal{X},i\in[n]}$ are not known but are to be learned. We consider the statistics

$$L_{x,i}^t = \sum_{s=1}^{t-1} \mathbf{1}(x^s = x, a^s = i), \qquad \bar{p}_{x,i}^t = \frac{\sum_{s=1}^{t-1} \mathbf{1}(x^s = x, a^s = i, \mathbf{y}_i^s = 1)}{\max\{L_{x,i}^t, 1\}}.$$

The statistics $L_{x,i}^t, \bar{p}_{x,i}^t$ can be constructed from the observations $\mathcal{F}^{t-1}$ during time $1, \ldots, t-1$. The UCB and LazyUCB oracles for Application 2 are provided in Algorithms 4, 5 respectively.

Note that Algorithms 4, 5 are analogous to Algorithms 2, 3 respectively. Their performance guarantees are also analogous. Let $K = |\mathcal{X}|$ denote the total number of customer segments.

Electronic copy available at: https://ssrn.com/abstract=3236533

Algorithm 4 UCB oracle at time t for Application 2

1: Input: observation $\mathcal{F}^{t-1}$ from time 1 to $t-1$, context x^t and discounted rewards $(r_i^t)_{i\in[n]}$.

2: Use $\mathcal{F}^{t-1}$ to compute the statistics $\{L_{x,i}^t\}_{x\in\mathcal{X},i\in[n]}, \{\bar{p}_{x,i}^t\}_{x\in\mathcal{X},i\in[n]}$.

3: Set $\delta_t = \frac{1}{(1+t)^2}$.

4: For each action $i\in[n]$, compute a UCB for associated the discounted revenue:

$$\mathsf{UCB}^t(x^t,i) = r_i^t \cdot \left[\bar{p}_{x^t,i}^t + \mathsf{rad}(\bar{p}_{x^t,i}^t, L_{x^t,i}^t, \delta_{(t)})\right].$$

5: Output an action i^t which satisfies

$$i^t \in \operatorname*{argmax}_{i\in[n]} \mathsf{UCB}^t(x^t,i).$$

Algorithm 5 LazyUCB Oracle at time t for Application 2

1: Input: exploitation parameter $\epsilon \geq 0$, observation $\mathcal{F}^{t-1}$ from time 1 to $t-1$, context x^t and discounted rewards $(r_i^t)_{i\in[n]}$.

2: Use $\mathcal{F}^{t-1}$ to compute the statistics $\{L_{x,i}^t\}_{x\in\mathcal{X},i\in[n]}, \{\bar{p}_{x,i}^t\}_{x\in\mathcal{X},i\in[n]}$.

3: Set $\delta_t = \frac{1}{(1+t)^2}$.

4: For each action $i\in[n]$, compute a *lazy* UCB for the discounted revenue with action (i,k):

$$\mathsf{LazyUCB}^t(x^t,i) = r_i^t \cdot \left[\bar{p}_{x^t,i}^t + \mathsf{LazyRad}^t(x^t,i)\right],$$

where

$$\mathsf{LazyRad}^t(x^t,i) = \min\left\{\mathsf{lad}(\epsilon, L_{x^t,i}^t; \delta_t), \mathsf{rad}(\bar{p}_{x^t,i}^t, L_{x^t,i}^t, \delta_t)\right\}.$$

5: Output an action i^t which satisfies

$$i^t \in \operatorname*{argmax}_{i\in[n]} \mathsf{LazyUCB}^t(x^t,i).$$

Corollary 3. *For Application 2, the IBOL algorithm (Algorithm 1) with the UCB oracle (Algorithm 4) yields expected reward* $\mathbb{E}[\mathsf{ALG}]$ *that satisfies*

$$\mathbb{E}[\mathsf{ALG}] \geq \frac{1-1/e}{(1+b_{min})(1-e^{-1/b_{min}})} \cdot \left[\mathsf{OPT} - \tilde{O}\left(\sqrt{nK\cdot\mathsf{OPT}}\right)\right].$$

Corollary 4. *For Application 2, the IBOL algorithm (Algorithm 1) with the LazyUCB oracle (Algorithm 5) yields expected reward* $\mathbb{E}[\mathsf{ALG}]$ *that satisfies*

$$\mathbb{E}[\mathsf{ALG}] \geq f_\Psi(\{b_i\}_{i\in[n]},\epsilon) \cdot \left[\mathsf{OPT} - \min\left\{\tilde{O}\left(\sqrt{nK\mathsf{OPT}}\right), \tilde{O}\left(\left(1+\frac{1}{\epsilon}\right)\cdot nK\right)\right\}\right],$$

where $f_\Psi(\{b_i\}_{i\in[n]},\epsilon)$ *is defined in equation (14).*

Electronic copy available at: https://ssrn.com/abstract=3236533

The proofs for these corollaries hinges on proving bounds on $\mathrm{Reg}_0, \mathrm{Reg}_\epsilon$ for the UCB, LazyUCB oracles respectively, and these proofs can be reproduced by replacing $\{q_{(i,k)}\}_{i\in[n],k\in[K]}$ in Appendices C.1, C.2 with $\{p_{x,i}\}_{x\in\mathcal{X},i\in[n]}$. The comparison between the UCB oracle and the LazyUCB oracle, as well as the tuning of ϵ, are similar to that in Application 1, so we do not repeat the discussion here.

6. Numerical Studies

In this section, we conduct numerical experiments to demonstrate the performance of the proposed algorithms. First, in Section 6.1, we use synthetic data to test the three-way trade-off between hedging, exploration, and exploitation, using our LazyUCB oracle for Application 1. Then in Section 6.2, we simulate a dynamic assortment optimization problem using a real-world dataset.

6.1. Experiments on Synthetic Data

We conduct experiments for the online matching with unknown matching probabilities model described in Section 2.1. We test the role of ϵ in our algorithmic framework IBOL by using our LazyUCB oracle with ϵ ranging from 0 to 1. Recall that $\epsilon = 0$ corresponds to the classical UCB oracle, while $\epsilon = 0.1, \ldots, 1$ corresponds to a LazyUCB oracle that does progressively less exploration.

In all test cases, we set the number of unknown arms per resource to be $K = 5$, independently draw their unknown probabilities $q_{(i,k)}$ from $[0.2, 0.5]$ uniformly at random, and independently draw the resource adjacencies $x_i^t = 1$ from $\{0, 1\}$ uniformly at random. We set the reward values r_i to be identical for all resources i. We consider different scales of the problem, with the number of resources n lying in $\{5, 50\}$, the number of times steps T lying in $\{10^5, 10^6, 10^7\}$, and the capacity b_i of each resource i being identical to some B which varies depending on the combination of n and T.

We report the simulation results in Figures 1 to 4. For each test case, the expected total reward $\mathbb{E}[\mathsf{ALG}]$ is an average value based on 500 simulation replications.

Discussion of results. The empirical performance is consistently best when ϵ takes its maximum value of 1, i.e. when LazyUCB does the least exploration. This is consistent with recent findings (Bastani et al. 2021, Kannan et al. 2018) that reducing forced exploration in contextual bandit settings will generally improve practical performance, despite not having a better worst-case guarantee. Interestingly, in our setting there is a drop in performance for ϵ between 0 and 1, because it is better to either fully explore (which is optimal if the arrivals x^t were to continue on indefinitely) or minimize exploration (which is optimal if the arrivals x^t were to suddenly end).

In our graphs, all performances are worse for higher n because there is more uncertainty in the resource adjacencies, and there are more unknown probabilities to learn. On the other hand, all performances are better for higher T because there is more time to learn the unknown probabilities.

Electronic copy available at: https://ssrn.com/abstract=3236533

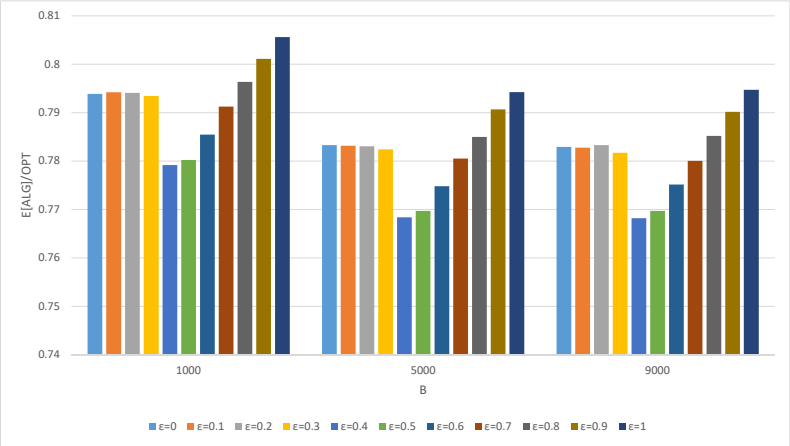

Figure 1 **Performance ratios of algorithms using UCB and LazyUCB oracles.** $n = 5$. $b_i = B$ **for all resources** $i \in [n]$. $T = 10000$.

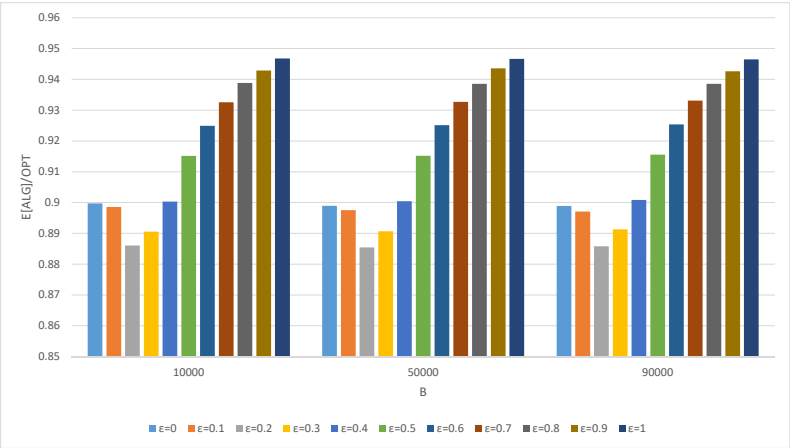

Figure 2 **Performance ratios of algorithms using UCB and LazyUCB oracles.** $n = 5$. $b_i = B$ **for all resources** $i \in [n]$. $T = 100000$.

The dependence on the capacities B varies based on its relation to n and T, but generally the performances are worse for higher B. This is because when B is small, it is less punishing to waste attempts on low-probability arms, since the capacity is the bottleneck and most of it will end up being exhausted anyway.

6.2. Experiments on Real-World Data

We conduct numerical experiments using dataset Hotel 1 of Bodea et al. (2009). Our numerical setting is a dynamic assortment planning problem, similar to that in Ma and Simchi-Levi (2020), but we consider their extension in which customer purchase probabilities are not observable.

We focus on a dynamic assortment planning problem where each room could be sold at multiple different prices. Our results can be extended to this setting (see Appendix E). We consider a hotel with $n = 4$ room categories: King rooms, Queen rooms, Suites, and Two-double rooms. Each room

Electronic copy available at: https://ssrn.com/abstract=3236533

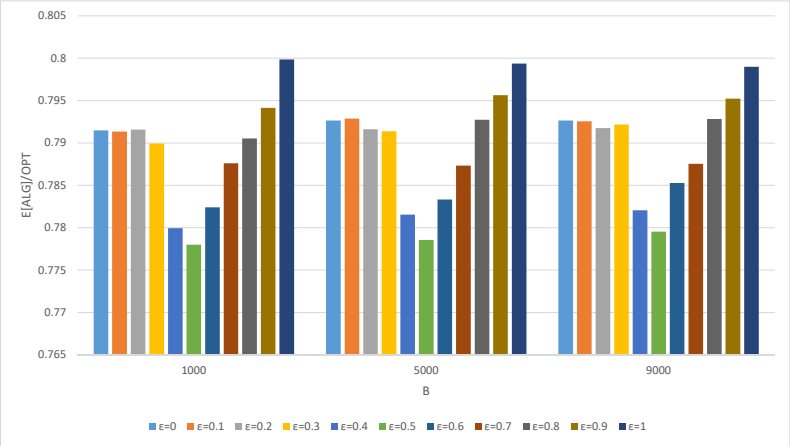

Figure 3 **Performance ratios of algorithms using UCB and LazyUCB oracles.** $n = 50$. $b_i = B$ **for all resources** $i \in [n]$. $T = 100000$.

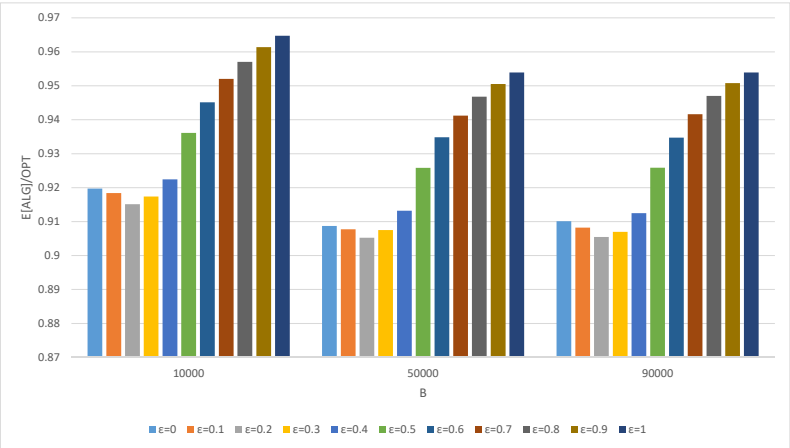

Figure 4 **Performance ratios of algorithms using UCB and LazyUCB oracles.** $n = 50$. $b_i = B$ **for all resources** $i \in [n]$. $T = 1000000$.

category is a resource, indexed by $i = 1, 2, 3, 4$. The inventory level of each of these resources is the number of available rooms in the corresponding category.

Rooms of each category i can be offered at two prices $\mathcal{P}_i = \{P_{i,1}, P_{i,2}\}$, for $i = 1, 2, 3, 4$. Each of the $m = 8$ combinations, indexed by $j = 1, 2, ..., 8$, of room category and price is a product. Table 3 summarizes the prices of all the $m = 8$ products from the data set. In the experiments, we double the higher price $P_{i,2}$ of each room category i in order to differentiate the performance of different algorithms.

Each customer has a feature (context) vector $x \in \mathcal{X} \subseteq \mathbb{R}^9$. $x_1 = 1$ is a constant feature. Features $x_2, ..., x_9$ represent the customer's personal information, such as the party size and the VIP level. (See Appendix F for a more detailed discussion on feature selection.) Each product $j \in \{1, 2, ..., 8\}$ has a latent vector $\beta_j^* \in \mathbb{R}^9$. We assume that customers follow the MNL choice model. For each

Electronic copy available at: https://ssrn.com/abstract=3236533

Table 3 Prices of 8 products from the dataset.

Category	$P_{i,1}$	$P_{i,2}$
King	307	361
Queen	304	361
Suites	384	496
Two-double	306	342

customer $x \in \mathcal{X}$, the personalized attraction value of product j is $e^{x^\top \beta_j^*}$. The action set $\mathcal{A}$ consists of all the possible assortments formed by the 8 products. When assortment $a \subseteq \{1, 2, ..., 8\}$ is offered to customer $x \in \mathcal{X}$, the customer will purchase product $j \in a$ with probability

$$\frac{e^{x^\top \beta_j^*}}{v_0 + \sum_{j' \in a} e^{x^\top \beta_{j'}^*}},$$

where v_0 is the attraction value for the no-purchase option. We vary v_0 in the experiments.

We consider a Bayesian environment. The prior distribution for each β_j^*, $j \in \{1, 2, ..., 8\}$, is generated as follows. First, calculate the maximum likelihood estimator $\bar{\beta}_j$ for β_j^* from all the transactions in the dataset. Then, we assume that each element $\beta_{j,k}^*$, for $k = 1, 2, ..., 9$, of β_j^* is an independent uniform random variable over $[\bar{\beta}_{j,k} - \epsilon, \bar{\beta}_{j,k} + \epsilon]$. We vary the uncertainty level ϵ in the tests. $\epsilon = 0$ corresponds to the model of Ma and Simchi-Levi (2020), in which the algorithms know the true values of β_j^*.

This numerical setting essentially follows Cheung and Simchi-Levi (2017) except that we impose inventory constraints here. The Thompson sampling algorithm in Cheung and Simchi-Levi (2017) solves the auxiliary problem of this setting.

PROPOSITION 1 **(Cheung and Simchi-Levi (2017))**. *Suppose that $\beta = (\beta_1, \beta_2, ..., \beta_8)$ is drawn from a known prior distribution π_0. For the auxiliary problem, there is a Thompson sampling algorithm with Bayesian regret*

$$\mathbb{E}_{\beta \sim \pi_0}[\mathsf{REG}(\mathcal{F}_T)] = \mathbb{E}_{\beta \sim \pi_0}[\mathbb{E}[\mathsf{REG}(\mathcal{F}_T)|\beta]] = \tilde{O}(Dm\sqrt{BT}).$$

In our numerical model, $D = 9$ is the length of feature vectors, $m = 8$ is the number of products, $B = 8$ is the maximum size of any assortment, and $T = 231$ is the number of customers.

Applying this Thompson sampling algorithm to our framework, and letting $b_{\min} \to \infty$, we can obtain the following performance guarantee by Theorem 8

$$\mathbb{E}_{\beta \sim \pi_0}[\mathsf{OPT}] \leq \frac{1}{1 - \exp(-\min_{i \in [n]} \alpha_i^{(1)})} \cdot \mathbb{E}_{\beta \sim \pi_0}[\mathsf{ALG}] + \tilde{O}(Dm\sqrt{BT}).$$

Based on the prices in Table 3, we can easily calculate $1 - \exp(-\min_{i \in [n]} \alpha_i^{(1)}) \approx 0.58$. For details of the calculation, we refer to Ma and Simchi-Levi (2020).

Electronic copy available at: https://ssrn.com/abstract=3236533

For each test case, we simulate 500 replicates and report the average performance of each algorithm. For each replicate, we uniformly draw a sample path of customer arrivals, i.e., a sequence of feature vectors, from 31 different instances constructed in Ma and Simchi-Levi (2020). Each sample path contains about 200 customers. For each replicate, we also randomly draw the latent vectors β_j^* for all products $j \in \{1, 2, ..., 8\}$ from their prior distributions.

We compare the following algorithms

- IB-TS: the inventory-balancing algorithm generated by our framework using the Thompson sampling algorithm in Cheung and Simchi-Levi (2017) as the oracles.
- Gdy-TS: same as IB-TS but the framework uses the original reward values, instead of the virtual rewards, as the input for the oracles.
- Conserv-TS: same as IB-TS but the algorithm assumes that there are only 4 higher-price products, i.e., products with prices $P_{\cdot,2}$.

Tables 4 to 8 report the performance of these algorithms under different test parameters. In particular, the first column of each table is a parameter that scales the initial inventory levels of all the four resources. In general, Gdy-TS performs better when inventory is more abundant. This is because the greedy algorithm is the optimal algorithm when there is no need to reserve resources. On the other hand, Conserv-TS has better performance when inventory is more scarce. This is because there is no need to sell resources at lower prices when we can sell all of them. Overall, our IB-TS algorithm performs much better when total inventory is close to total demand.

7. Conclusion

We study a general class of resource allocation problems, which involve both uncertainty on the contextual information of each customer, as well as on the functional relationship between a customer's contextual information to their behavior. We propose the Inventory Balancing with Online Learning (IBOL) algorithm that handles both sources of uncertainty simultaneously. In addition, we harness existing tools from the online learning literature to construct the Upper Confidence Bound (UCB) oracle, and we also design a new LazyUCB oracle that conducts substantially less exploration and more exploitation than the LazyUCB oracle. The performance guarantees of our algorithms are shown to be near optimal, and they are corroborated by numerical experiments on both synthetic and actual datasets.

To finish off, we would like to discuss the benefit of describing our resource allocation problem using generic "actions", especially in the context of the Inventory Balancing literature. Previously, the most general description of an Inventory Balancing algorithm under adversarial arrivals was that of offering an *assortment* of multiple resources, introduced by Golrezaei et al. (2014). However, our treatment allows for even more general actions, such as offering a *sequence* of resources to

Electronic copy available at: https://ssrn.com/abstract=3236533

Table 4 **Performance of algorithms relative to** OPT. $v_0 = 5$, $\epsilon = 1$.

Inventory scale	IB-TS	Gdy-TS	Conserv-TS
0.1	93.6%	90.3%	99.3%
0.15	95.5%	90.7%	98.2%
0.2	95.7%	90.8%	98.0%
0.25	96.1%	91.4%	97.0%
0.3	95.8%	92.1%	96.1%
0.35	95.1%	92.6%	96.0%
0.4	94.2%	92.5%	95.1%
0.45	93.5%	92.8%	94.5%
0.5	93.2%	93.2%	94.2%
0.55	91.5%	92.9%	92.9%
0.6	91.0%	92.9%	93.2%

Table 5 **Performance of algorithms relative to** OPT. $v_0 = 40$, $\epsilon = 1$.

Inventory scale	IB-TS	Gdy-TS	Conserv-TS
0.1	87.7%	84.1%	92.8%
0.15	90.4%	85.5%	91.4%
0.2	91.8%	87.2%	89.3%
0.25	91.5%	87.5%	88.3%
0.3	92.0%	88.4%	87.8%
0.35	91.6%	89.2%	86.5%
0.4	91.3%	89.0%	86.4%
0.45	91.4%	89.8%	86.1%
0.5	92.8%	90.8%	86.5%
0.55	91.8%	90.1%	86.1%
0.6	92.2%	90.7%	86.7%

each online customer, as in the online matching with timeouts problem (Bansal et al. 2012). Our Theorems 1 and 8 directly imply that an online algorithm can be 1/2-competitive in general, and $(1 - 1/e)$-competitive as resource capacities approach ∞, in the online vertex-weighted matching with timeouts problem of Bansal et al. (2012), in which matching probabilities are known exactly (i.e. the regret from learning is 0). This further demonstrates the benefit of our unified and generic framework for online resource allocation.

References

Agrawal, Shipra, Nikhil R. Devanur. 2014. Bandits with concave rewards and convex knapsacks. *Proceedings of the fifteenth ACM conference on Economics and computation - EC '14* 989–1006.

Agrawal, Shipra, Nikhil R. Devanur. 2016. *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*. 3450–3458.

Agrawal, Shipra, Nikhil R. Devanur, Lihong Li. 2016. An efficient algorithm for contextual bandits with knapsacks, and an extension to concave objectives. *Proceedings of the 29th Conference on Learning Theory, COLT 2016, New York, USA, June 23-26, 2016* 4–18.

Electronic copy available at: https://ssrn.com/abstract=3236533

Table 6 **Performance of algorithms relative to** OPT. $v_0 = 100$, $\epsilon = 1$.

Inventory scale	IB-TS	Gdy-TS	Conserv-TS
0.1	87.1%	86.2%	87.7%
0.15	89.9%	87.6%	86.4%
0.2	90.5%	88.0%	86.2%
0.25	91.9%	90.2%	85.6%
0.3	91.7%	90.2%	84.3%
0.35	91.5%	91.1%	84.3%
0.4	92.1%	90.8%	83.5%
0.45	92.4%	91.5%	84.7%
0.5	93.3%	91.4%	85.1%
0.55	93.2%	92.2%	84.7%
0.6	92.4%	92.6%	84.2%

Table 7 **Performance of algorithms relative to** OPT. $v_0 = 40$, $\epsilon = 0.01$.

Inventory scale	IB-TS	Gdy-TS	Conserv-TS
0.1	93.3%	91.9%	99.2%
0.15	93.5%	89.9%	97.4%
0.2	92.7%	88.5%	95.3%
0.25	93.2%	89.6%	93.5%
0.3	92.9%	91.1%	92.7%
0.35	94.9%	95.0%	92.4%
0.4	96.4%	95.7%	93.1%
0.45	96.8%	97.4%	93.7%
0.5	98.4%	98.4%	95.2%
0.55	98.3%	99.6%	95.2%
0.6	97.9%	99.0%	95.0%

Table 8 **Performance of algorithms relative to** OPT. $v_0 = 40$, $\epsilon = 5$.

Inventory scale	IB-TS	Gdy-TS	Conserv-TS
0.1	84.8%	82.0%	91.4%
0.15	87.5%	84.2%	89.9%
0.2	88.5%	83.7%	89.6%
0.25	88.9%	84.3%	88.5%
0.3	89.3%	84.7%	87.7%
0.35	89.4%	86.1%	86.3%
0.4	89.9%	86.2%	86.0%
0.45	89.1%	85.3%	85.4%
0.5	88.9%	85.8%	84.6%
0.55	88.6%	85.3%	84.5%
0.6	88.6%	85.6%	84.7%

Alaei, Saeed, MohammadTaghi Hajiaghayi, Vahid Liaghat. 2012. Online prophet-inequality matching with applications to ad allocation. *Proceedings of the 13th ACM Conference on Electronic Commerce*. ACM, 18–35.

Audibert, Jean-Yves, Remi Munos, Csaba Szepesvri. 2009. Explorationexploitation tradeoff using variance estimates in multi-armed bandits. *Theor. Comput. Sci.* **410** 1876–1902.

Electronic copy available at: https://ssrn.com/abstract=3236533

Auer, Peter, Nicolò Cesa-Bianchi, Paul Fischer. 2002a. Finite-time analysis of the multiarmed bandit problem. *Machine Learning* **47**(2) 235–256.

Auer, Peter, Nicolo Cesa-Bianchi, Yoav Freund, Robert E Schapire. 2002b. The nonstochastic multiarmed bandit problem. *SIAM journal on computing* **32**(1) 48–77.

Babaioff, Moshe, Shaddin Dughmi, Robert Kleinberg, Aleksandrs Slivkins. 2015. Dynamic Pricing with Limited Supply. *ACM Trans. Economics and Comput.* **3**(1) 4:1–4:26.

Badanidiyuru, Ashwinkumar, Robert Kleinberg, Aleksandrs Slivkins. 2013. Bandits with knapsacks. *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*. IEEE, 207–216.

Badanidiyuru, Ashwinkumar, John Langford, Aleksandrs Slivkins. 2014. Resourceful contextual bandits. Maria Florina Balcan, Vitaly Feldman, Csaba Szepesvri, eds., *Proceedings of The 27th Conference on Learning Theory*, *Proceedings of Machine Learning Research*, vol. 35. PMLR, Barcelona, Spain, 1109–1134. URL `http://proceedings.mlr.press/v35/badanidiyuru14.html`.

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

Bansal, Nikhil, Anupam Gupta, Jian Li, Julián Mestre, Viswanath Nagarajan, Atri Rudra. 2012. When lp is the cure for your matching woes: Improved bounds for stochastic matchings. *Algorithmica* **63**(4) 733–762.

Bastani, Hamsa, Mohsen Bayati, Khashayar Khosravi. 2021. Mostly exploration-free algorithms for contextual bandits. *Management Science* **67**(3) 1329–1349.

Besbes, Omar, Assaf Zeevi. 2009. Dynamic Pricing Without Knowing the Demand Function: Risk Bounds and Near-Optimal Algorithms. *Operations Research* **57**(6) 1407–1420. doi:10.1287/opre.1080.0640. URL `http://pubsonline.informs.org/doi/abs/10.1287/opre.1080.0640`.

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

Besbes, Omar, Assaf Zeevi. 2012. Blind Network Revenue Management. *Operations Research* **60**(6) 1537–1550. doi:10.1287/opre.1120.1103. URL `http://pubsonline.informs.org/doi/abs/10.1287/opre.1120.1103`.

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

Borodin, Allan, Ran El-Yaniv. 2005. *Online computation and competitive analysis*. cambridge university press.

Brubach, Brian, Karthik Abinav Sankararaman, Aravind Srinivasan, Pan Xu. 2016. New algorithms, better bounds, and a novel model for online stochastic matching. *24th Annual European Symposium on Algorithms, ESA 2016, August 22-24, 2016, Aarhus, Denmark*. 24:1–24:16.

Electronic copy available at: https://ssrn.com/abstract=3236533

Bubeck, Sébastien, Nicolò Cesa-Bianchi. 2012. Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. *Foundations and Trends in Machine Learning* **5**(1) 1–122.

Buchbinder, Niv, Kamal Jain, Joseph Seffi Naor. 2007. Online primal-dual algorithms for maximizing ad-auctions revenue. *European Symposium on Algorithms*. Springer, 253–264.

Chen, Xi, Will Ma, David Simchi-Levi, Linwei Xin. 2016. Assortment planning for recommendations at checkout under inventory constraints. *Available at SSRN 2853093* .

Cheung, Wang Chi, David Simchi-Levi. 2017. Thompson sampling for online personalized assortment optimization problems with multinomial logit choice models. *Manuscript* URL `https://ssrn.com/abstract=3075658`.

Chung, Fan, Linyuan Lu. 2006. Concentration inequalities and martingale inequalities: a survey. *Internet Math.* **3**(1) 79–127.

Devanur, Nikhil R, Kamal Jain. 2012. Online matching with concave returns. *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*. ACM, 137–144.

Devanur, Nikhil R, Kamal Jain, Robert D Kleinberg. 2013. Randomized primal-dual analysis of ranking for online bipartite matching. *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 101–107.

Feldman, Jacob, Nan Liu, Huseyin Topaloglu, Serhan Ziya. 2014. Appointment scheduling under patient preference and no-show behavior. *Operations Research* **62**(4) 794–811.

Feldman, Jon, Aranyak Mehta, Vahab Mirrokni, S. Muthukrishnan. 2009. Online stochastic matching: Beating 1-1/e. *Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science*. FOCS '09, IEEE Computer Society, Washington, DC, USA, 117–126. doi:10.1109/FOCS.2009.72. URL `http://dx.doi.org/10.1109/FOCS.2009.72`.

Ferreira, Kris Johnson, David Simchi-Levi, He Wang. 2016. Online network revenue management using thompson sampling. *Accepted by Operations Research* .

Garivier, Aurélien, Eric Moulines. 2011. On upper-confidence bound policies for switching bandit problems. *Algorithmic Learning Theory - 22nd International Conference*. 174–188.

Golrezaei, Negin, Hamid Nazerzadeh, Paat Rusmevichientong. 2014. Real-time optimization of personalized assortments. *Management Science* **60**(6) 1532–1551.

Goyal, Vineet, Rajan Udwani. 2019. Online matching with stochastic rewards: Optimal competitive ratio via path based formulation. *CoRR* **abs/1905.12778**. URL `http://arxiv.org/abs/1905.12778`.

Gur, Yonatan, Assaf J. Zeevi, Omar Besbes. 2014. Stochastic multi-armed-bandit problem with non-stationary rewards. *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*. 199–207.

Electronic copy available at: https://ssrn.com/abstract=3236533

Ho, Chien-Ju, Jennifer Wortman Vaughan. 2012. Online task assignment in crowdsourcing markets. *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada*.

Janson, Svante. 1999. On concentration of probability. *Combinatorics, Probability and Computing* **11** 2002.

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

Kannan, Sampath, Jamie H Morgenstern, Aaron Roth, Bo Waggoner, Zhiwei Steven Wu. 2018. A smoothed analysis of the greedy algorithm for the linear contextual bandit problem. *Advances in Neural Information Processing Systems 31*. Curran Associates, Inc., 2227–2236.

Karger, David R., Sewoong Oh, Devavrat Shah. 2014. Budget-optimal task allocation for reliable crowdsourcing systems. *Operations Research* **62**(1) 1–24.

Karp, Richard M, Umesh V Vazirani, Vijay V Vazirani. 1990. An optimal algorithm for on-line bipartite matching. *Proceedings of the twenty-second annual ACM symposium on Theory of computing*. ACM, 352–358.

Kell, Nathaniel, Debmalya Panigrahi. 2016. Online budgeted allocation with general budgets. *Proceedings of the 2016 ACM Conference on Economics and Computation*. ACM, 419–436.

Kleinberg, Robert, Aleksandrs Slivkins, Eli Upfal. 2008. Multi-armed Bandits in Metric Spaces. *Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing*. STOC '08, ACM, New York, NY, USA, 681–690. doi:10.1145/1374376.1374475. URL `http://doi.acm.org/10.1145/1374376.1374475`.

Ma, Will, David Simchi-Levi. 2020. Algorithms for online matching, assortment, and pricing with tight weight-dependent competitive ratios. *Operations Research* **68**(6) 1787–1803.

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

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

Mehta, Aranyak. 2013b. Online matching and ad allocation. *Foundations and Trends in Theoretical Computer Science* **8**(4) 265–368.

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

Mehta, Aranyak, Amin Saberi, Umesh Vazirani, Vijay Vazirani. 2007. Adwords and generalized online matching. *Journal of the ACM (JACM)* **54**(5) 22.

Mitzenmacher, Michael, Eli Upfal. 2005. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis.*. Cambridge University Press.

Slivkins, Aleksandrs. 2017. *Introduction to Multi-Armed Bandits*. September. URL `http://slivkins.com/work/MAB-book.pdf`.

Electronic copy available at: https://ssrn.com/abstract=3236533

Talluri, Kalyan, Garrett van Ryzin. 1998. An analysis of bid-price controls for network revenue management. *Management Science* **44**(11-part-1) 1577–1593.

Truong, Van-Anh. 2015. Optimal advance scheduling. *Management Science* **61**(7) 1584–1597.

Wang, Xinshang, Van-Anh Truong, David Bank. 2015. Online advance admission scheduling for services, with customer preferences. *ArXiv preprint arXiv:1805.10412*.

Wang, Zizhuo, Shiming Deng, Yinyu Ye. 2014. Close the gaps: A learning-while-doing algorithm for single-product revenue management problems. *Operations Research* **62**(2) 318–331.

Electronic copy available at: https://ssrn.com/abstract=3236533

Appendix A: Proofs for Section 4

A.1. Proof of Lemma 1

For an online algorithm, let's denote $\gamma_{a,t}$ as the probability that the algorithm chooses action a at time t. We first claim that $\{\gamma_{a,t}\}_{a\in\mathcal{A},t\in[T]}$ is a feasible solution to the LP **Primal**. Indeed, for each t, $\{\gamma_{a,t}\}_{a\in\mathcal{A}}$ forms a probability distribution over the action set $\mathcal{A}$, therefore the constraints (11, 12) are satisfied. To check the constraints (10), we observe that $\sum_{t=1}^{T}\mathbf{y}_i^t \le b_i$ for all $i\in[n]$ with certainty, since we assume that the online algorithm is feasible. In particular, we also have $\mathbb{E}\left[\sum_{t=1}^{T}\mathbf{y}_i^t\right] \le b_i$ for all $i\in[n]$. Observe that we have

$$\mathbb{E}\left[\mathbf{y}_i^t\right] = \sum_{a\in\mathcal{A}} \Pr(y_i^t = 1 | a^t = a)\Pr(a^t = a) = \sum_{a\in\mathcal{A}} p_{x^t,a,i}\gamma_{a,t}$$

for all $i\in[n]$, by our model definition of $p_{x^t,a,i}$. Altogether, the constraints (10) are also satisfied, which shows that $\{\gamma_{a,t}\}_{a\in\mathcal{A},t\in[T]}$ is feasible to the LP **Primal**.

To finish the proof, observe that the expected total reward is equal to the objective value:

$$\begin{aligned}
&\mathbb{E}\left[\sum_{i\in[n]} r_i \sum_{t\in[T]} \mathbf{y}_i^t \cdot \mathbf{1}(N_i^{t-1} < b_i)\right] \\
=&\mathbb{E}\left[\sum_{i\in[n]} r_i \sum_{t\in[T]} \mathbf{y}_i^t\right] &(28)\\
=&\sum_{i\in[n]} r_i \sum_{a\in\mathcal{A}} p_{x^t,a,i}\gamma_{a,t}. &(29)
\end{aligned}$$

Step (28) is by the Lemma's assumption that the online algorithm is feasible. Observing that (29) is the objective value of the LP **Primal** under the solution $\{\gamma_{a,t}\}_{a\in\mathcal{A},t\in[T]}$, we have altogether shown that $\{\gamma_{a,t}\}_{a\in\mathcal{A},t\in[T]}$ is feasible to the LP, and the expected total reward under the algorithm is at most OPT. □

A.2. Proof of Theorem 1

Throughout the proof, we fix $\epsilon \ge 0$ as a constant. We demonstrate the Theorem by showing the following inequality:

$$\mathsf{OPT} \le \frac{e}{e-1} f_\Psi(b_{\min},\epsilon)\mathbb{E}[\mathsf{ALG}] + \mathbb{E}\left[\sum_{t=1}^{T} R^t(a_*^t) - (1+\epsilon)R^t(a^t)\right], \tag{30}$$

The proof of Theorem 1 begins by considering a dual formulation of the LP **Primal**:

$$\begin{aligned}
\textbf{Dual: } &\min \sum_{t\in[T]}\gamma_t + \sum_{i\in[n]} b_i\lambda_i && &(31)\\
&\text{s.t. } \gamma_t \ge \sum_{i\in[n]} p_{x^t,a,i}(r_i - \lambda_i) && a\in\mathcal{A}, t\in[T] &(32)\\
&p_{x^t,a,i}\lambda_i \ge 0 && \forall i\in[n]. &(33)
\end{aligned}$$

We prove the performance guarantee using a primal dual approach. More precisely, we construct a solution (Λ,Γ) feasible to **Dual**, where (Λ,Γ) are constructed based on the dynamics of Algorithm 1. Then, we relate the algorithm's performace to the expected value of the solution (Λ,Γ) under objective (31), which upper bounds the benchmark by the linear duality.

Electronic copy available at: https://ssrn.com/abstract=3236533

We define the solution $\Lambda = (\Lambda_i)_{i\in[n]}, \Gamma = (\Gamma_t)_{t\in[T]}$ as

$$\Lambda_i = r_i \cdot \Psi\left(\frac{N_i^T}{b_i}\right) \tag{34}$$

$$\Gamma_t = \max_{a\in\mathcal{A}}\left\{\sum_{i\in[n]} p_{x^t,a,i} r_i \left[1-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right]\right\} = \max_{a\in\mathcal{A}}\left\{\sum_{i\in[n]} p_{x^t,a,i} r_i^t\right\} = R^t(a_*^t). \tag{35}$$

Recall that a_*^t and Γ^t are respectively an optimal action and the optimal reward at time t in the auxiliary problem. We first claim the feasibility of (Λ, Γ) to **Dual**.

Claim 1. *For any realization of $\{N_i^t\}_{t\in[T],i\in[n]}$, the solution (Λ,Γ) defined in (34, 35) is feasible to **Dual**. Moreover, we have $\mathsf{OPT} \le \mathbb{E}[\sum_{t\in[T]}\Gamma_t + \sum_{i\in[n]} b_i\Lambda_i]$.*

Proof of Claim 1. The constraints in (33) are clearly satisfied by Λ, since $\Psi(x)\ge 0$ for all $x\in[0,1]$. To verify the feasibility to the constraints (32), for any $a\in\mathcal{A}, t\in[T]$ we check that

$$\Gamma_t \ge \sum_{i\in[n]} p_{x^t,a,i} r_i \left[1-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right] \tag{36}$$

$$\ge \sum_{i\in[n]} p_{x^t,a,i} r_i \left[1-\Psi\left(\frac{N_i^{T}}{b_i}\right)\right] \tag{37}$$

$$= \sum_{i\in[n]} p_{x^t,a,i}(r_i - \Lambda_i). \tag{38}$$

Step (36) is by the first equation (35) in the definition of Γ^t, step (37) is by the fact that Ψ is an increasing function and $N_i^T \ge N_i^{t-1}$, step (38) is by the definition of Λ_i in (34). Altogether, the Claim is proved. □

First, we use Claim 1 to argue that

$$\begin{aligned}\mathsf{OPT} \le& \mathbb{E}\left[\sum_{t\in[T]}\Gamma_t + \sum_{i\in[n]} b_i\Lambda_i\right]\\ =& \sum_{i\in[n]} b_i r_i \cdot \sum_{t\in[T]} \mathbb{E}\left[\Psi\left(\frac{N_i^t}{b_i}\right) - \Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right] + \mathbb{E}\left[\sum_{t\in[T]} R^t(a_*^t)\right].\end{aligned} \tag{39}$$

Step (39) is by stating the first summation in (39) as a telescoping sum.

To proceed, recall that $\mathcal{F}_t$ is the input to the auxiliary problem at the start of time t, which determines the values of N_i^{t-1}. Conditioned on $\mathcal{F}_t$, the algorithm's action $a^t = \mathcal{O}^t(\mathcal{F}_t)$ is determined. Thus, for any resource i,

$$\mathbb{E}\left[\Psi\left(\frac{N_i^t}{b_i}\right)\middle|\mathcal{F}_t\right] - \Psi\left(\frac{N_i^{t-1}}{b_i}\right) = \sum_{\mathbf{y}\in\{0,1\}^n} \rho_{x^t,a^t}(\mathbf{y})\mathbf{y}_i\left[\Psi\left(\frac{\min\{b_i, N_i^{t-1}+1\}}{b_i}\right) - \Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right]. \tag{40}$$

We explain equation (40). We claim that, conditioned on $\mathcal{F}_t$, we have $N_i^t = \min\{b_i, N_i^{t-1}+1\}$. Indeed, the vector of outcomes $\mathbf{y}$ is distributed according to ρ_{x^t,a^t}. If $\mathbf{y}_i = 1$ and resource i is not yet depleted, i.e. $N_i^{t-1} < b_i$, then a unit of resource i is consumed, leading to $N_i^t = N_i^{t-1}+1 \le b_i$. If $\mathbf{y}_i = 1$ but resource i is depleted, i.e. $N_i^{t-1} = b_i$, then resource i cannot be consumed further, leading to $N_i^t = b_i$. Altogether, equation (40) is justified.

Electronic copy available at: https://ssrn.com/abstract=3236533

Using, (40) and the towering property of conditional expectation, we can express the summands in the first sum in (39) as

$$\begin{aligned}
&\mathbb{E}\left[\Psi\left(\frac{N_i^t}{b_i}\right)-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right]=\mathbb{E}\left[\mathbb{E}\left[\Psi\left(\frac{N_i^t}{b_i}\right)|\mathcal{F}_t\right]-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right]\\
=&\mathbb{E}\left[\sum_{\mathbf{y}\in\{0,1\}^n}\rho_{x^t,a^t}(\mathbf{y})\mathbf{y}_i\left[\Psi\left(\frac{\min\{b_i,N_i^{t-1}+1\}}{b_i}\right)-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right]\right]\\
=&\mathbb{E}\left[p_{x^t,a^t,i}\left[\Psi\left(\frac{\min\{b_i,N_i^{t-1}+1\}}{b_i}\right)-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right]\right] \qquad (41)
\end{aligned}$$

Next, we continue with (39, 41):

$$\begin{aligned}
\mathsf{OPT}&\le\sum_{i\in[n]}b_ir_i\sum_{t\in[T]}\mathbb{E}\left[p_{x^t,a^t,i}\left[\Psi\left(\frac{\min\{b_i,N_i^{t-1}+1\}}{b_i}\right)-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right]\right]+\mathbb{E}\left[\sum_{t\in[T]}R^t(a_*^t)\right]\\
&=\sum_{i\in[n]}r_i\cdot\mathbb{E}\left[\sum_{t\in[T]}p_{x^t,a^t,i}\cdot\left\{\frac{\Psi\left(\frac{\min\{b_i,N_i^{t-1}+1\}}{b_i}\right)-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)}{1/b_i}+(1+\epsilon)\left[1-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right]\right\}\right]\\
&\quad+\mathbb{E}\left[\sum_{t\in[T]}R^t(a_*^t)-(1+\epsilon)R^t(a^t)\right] \qquad (42)
\end{aligned}$$

Step (42) uses the definition of $R^t(a^t)=\sum_{i\in[n]}r_ip_{x^t,a^t,i}[1-\Psi(N_i^{t-1}/b_i)]$.

We now need to derive our ϵ-perturbed potential function Ψ, and establish the following guarantee.

LEMMA 4 **(Guarantee for ϵ-perturbed potential function)**. *As long as the ϵ-perturbed potential function $\Psi(x)=\frac{e^{(1+\epsilon)x}-1}{e^{1+\epsilon}-1}$ is used, for any resource i, time t, and possible value of N_i^{t-1} in $\{0,\dots,b_i\}$,*

$$\frac{\Psi\left(\frac{\min\{b_i,N_i^{t-1}+1\}}{b_i}\right)-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)}{1/b_i}+(1+\epsilon)\left[1-\Psi\left(\frac{N_i^{t-1}}{b_i}\right)\right]\le\mathbf{1}(N_i^{t-1}<b_i)\frac{(b_i+1+\epsilon)(1-e^{-(1+\epsilon)/b_i})}{1-e^{-(1+\epsilon)}}. \qquad (43)$$

Proof. Consider a generic i and t. We omit scripts i,t and let $s\in\{0,1/b,\dots,1\}$ denote N/b. Both sides of the desired inequality are 0 when $s=1$, so in the sequel we assume $s<1$. We would like to show

$$\frac{\Psi(s+1/b)-\Psi(s)}{1/b}+(1+\epsilon)(1-\Psi(s))\le\frac{(b+1+\epsilon)(1-e^{-(1+\epsilon)/b})}{1-e^{-(1+\epsilon)}}. \qquad (44)$$

This difference between (44) and the typical constraint from primal-dual analysis (Buchbinder et al. 2007) is the multiplication on the LHS by the term $1+\epsilon$. Consequently, the RHS has also been relaxed by an expression dependent on ϵ. The LHS of (44) can be analyzed as follows:

$$\begin{aligned}
&b\left(\frac{e^{(1+\epsilon)(s+1/b)}-1}{e^{1+\epsilon}-1}-\frac{e^{(1+\epsilon)s}-1}{e^{1+\epsilon}-1}\right)+(1+\epsilon)\left(1-\frac{e^{(1+\epsilon)s}-1}{e^{1+\epsilon}-1}\right)\\
&=\frac{be^{(1+\epsilon)(s+1/b)}-be^{(1+\epsilon)s}+(1+\epsilon)e^{1+\epsilon}-(1+\epsilon)e^{(1+\epsilon)s}}{e^{1+\epsilon}-1}\\
&=\frac{e^{(1+\epsilon)s}(be^{(1+\epsilon)/b}-b-1-\epsilon)+(1+\epsilon)e^{1+\epsilon}}{e^{1+\epsilon}-1}\\
&\le\frac{e^{(1+\epsilon)(1-1/b)}(be^{(1+\epsilon)/b}-b-1-\epsilon)+(1+\epsilon)e^{1+\epsilon}}{e^{1+\epsilon}-1}\\
&=\frac{be^{1+\epsilon}(1-e^{-(1+\epsilon)/b})+(1+\epsilon)e^{1+\epsilon}(1-e^{-(1+\epsilon)/b})}{e^{1+\epsilon}-1}
\end{aligned}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

$$= \frac{(b+1+\epsilon)(1-e^{-(1+\epsilon)/b})}{1-e^{-(1+\epsilon)}}$$

where the inequality holds because the maximum possible value of s is $1-1/b$ and the expression $be^{(1+\epsilon)/b} - b - 1 - \epsilon$ is non-negative. This completes the proof of Lemma 4. □

Substituting the result from Lemma 4 back into (42), we get

$$\mathsf{OPT} \le \sum_{i\in[n]} r_i \cdot \mathbb{E}\left[\sum_{t\in[T]} p_{x^t,a^t,i}\cdot \mathbf{1}(N_i^{t-1} < b_i)\frac{(b_i+1+\epsilon)(1-e^{-(1+\epsilon)/b_i})}{1-e^{-(1+\epsilon)}}\right] + \mathbb{E}\left[\sum_{t\in[T]} R^t(a_*^t) - (1+\epsilon)R^t(a^t)\right].$$

The RHS can be upper-bounded by $\mathbb{E}[\mathsf{ALG}]\max_i \frac{(b_i+1+\epsilon)(1-e^{-(1+\epsilon)/b_i})}{1-e^{-(1+\epsilon)}} + \mathbb{E}[\sum_{t\in[T]} R^t(a_*^t) - (1+\epsilon)R^t(a^t)]$, by definition of $\mathbb{E}[\mathsf{ALG}]$. This implies

$$\mathbb{E}[\mathsf{ALG}] \ge \left(\mathsf{OPT} - \mathbb{E}\left[\sum_{t\in[T]} R^t(a_*^t) - (1+\epsilon)R^t(a^t)\right]\right)\min_i \frac{1-e^{-(1+\epsilon)}}{(b_i+1+\epsilon)(1-e^{-(1+\epsilon)/b_i})}.$$

Appendix B: Concentration Inequalities and Their Proofs

B.1. Lemma 5 for UCB and its Proof

LEMMA 5 **(Kleinberg et al. (2008))**. *Let $Y_1,\dots,Y_M$ be i.i.d. Bernoulli random variables with mean p. Denote $\bar{Y} = \frac{1}{M}\sum_{i=1}^M Y_i$. For any $0<\delta<1$, it holds that*

$$\Pr\left(\left|\bar{Y}-p\right| \le \mathsf{rad}(\bar{Y},M;\delta) \le 3\cdot\mathsf{rad}(p,M;\delta)\right) \ge 1-4\delta.$$

We first recall that for $p>0, M\in\mathbb{Z}_{\ge 0}, \delta\in(0,1)$, we have defined

$$\mathsf{rad}(p,M;\delta) = \sqrt{\frac{2p\log(1/\delta)}{\max\{M,1\}}} + \frac{3\log(1/\delta)}{\max\{M,1\}}.$$

We prove Lemma 5 with the following two concentration inequalities.

THEOREM 5 **(Theorem 1 in (Audibert et al. 2009))**. *Let $Y_1,\dots,Y_M$ be i.i.d. Bernoulli random variables with mean p. Denote $\bar{Y} = \frac{1}{M}\sum_{i=1}^M Y_i$. For any $0<\delta<1$, it holds that*

$$\Pr\left(\left|\bar{Y}-p\right| \le \mathsf{rad}(\bar{Y},M;\delta)\right) \ge 1-3\delta.$$

THEOREM 6 **(Theorem 4.4 (3) in (Mitzenmacher and Upfal 2005))**. *Let $Y_1,\dots,Y_M$ be i.i.d. Bernoulli random variables with mean p. Denote $\bar{Y} = \frac{1}{M}\sum_{i=1}^M Y_i$. For any $R>6p$, it holds that*

$$\Pr(\bar{Y} > R) \le 2^{-nR}.$$

Proof of Lemma 5 We have

$$\begin{aligned}&\Pr\left(\left|\bar{Y}-p\right| \le \mathsf{rad}(\bar{Y},M;\delta) \le 3\cdot\mathsf{rad}(p,M;\delta)\right)\\ \ge& 1 - \Pr\left(\left|\bar{Y}-p\right| \ge \mathsf{rad}(\bar{Y},M;\delta)\right) - \Pr\left(\mathsf{rad}(\bar{Y},M;\delta) \ge 3\cdot\mathsf{rad}(p,M;\delta)\right).\end{aligned} \tag{45}$$

By Theorem 5, we know that

$$\Pr\left(\left|\bar{Y}-p\right| \ge \mathsf{rad}(\bar{Y},M;\delta)\right) \le 3\delta. \tag{46}$$

Next, we have

$$\Pr\left(\mathsf{rad}(\bar{Y},M;\delta) \ge 3\cdot\mathsf{rad}(p,M;\delta)\right)$$

Electronic copy available at: https://ssrn.com/abstract=3236533

$$
\begin{aligned}
&= \Pr\left(\sqrt{\frac{2\bar{Y}\log(1/\delta)}{M}} + \frac{3\log(1/\delta)}{M} \geq 3\left(\sqrt{\frac{2p\log(1/\delta)}{M}} + \frac{3\log(1/\delta)}{M}\right)\right) \\
&= \Pr\left(\frac{2\bar{Y}\log(1/\delta)}{M} > \frac{18p\log(1/\delta)}{M} + 36\sqrt{\frac{2p\log(1/\delta)}{M}}\cdot\frac{\log(1/\delta)}{M} + \frac{36\log^2(1/\delta)}{M^2}\right) \\
&\leq \Pr\left(\bar{Y} > 9p + \frac{18\log(1/\delta)}{M}\right) \\
&\leq 2^{-9pM-18\log(1/\delta)} \qquad (47) \\
&< \delta. \qquad (48)
\end{aligned}
$$

Step (47) is by applying Theorem 6. Combining bounds (46, 48) and applying to (45), the Lemma is proved. □

B.2. Proof of Lemma 2 for the UCB Oracle for Application 1

While Lemma 2 is first discovered by (Kleinberg et al. 2008), the explicit constants in their confidence radii are not expressed explicitly, and they are instead hidden in $O(\cdot)$. We re-derive Lemma 2 in order to uncover those constants. Lemma 2 is proved by a direct application of Lemma 5 and the union bound.

Proof of Lemma 2 To this end, let $Y_1, \ldots, Y_{t-1}$ be i.i.d. Bernoulli random variables with mean $q_{(i,k)}$. Then

$$
\begin{aligned}
&\Pr(\mathcal{E}^t_{(i,k)}) \\
&= \Pr\left(\left|\bar{q}^t_{(i,k)} - q_{(i,k)}\right| \leq \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}, \delta_t) \leq 3\cdot\mathsf{rad}(q_{(i,k)}, M^t_{(i,k)}, \delta_t)\right) \\
&\geq \Pr\left(\left|\frac{1}{m}\sum_{s=1}^{m} Y_s - q_{(i,k)}\right| \leq \mathsf{rad}(\frac{1}{m}\sum_{s=1}^{m} Y_s, m, \delta_t) \leq 3\cdot\mathsf{rad}(q_{(i,k)}, m, \delta_t) \text{ for all } m \in \{0,\ldots,t-1\}\right) \qquad (49) \\
&\geq 1 - 4t\delta_t \qquad (50) \\
&\geq 1 - \frac{4}{1+t}.
\end{aligned}
$$

Step (49) is by a union bound over the possible values of $M^t_{(i,k)} \in \{0, 1, \ldots, t-1\}$. Step (50) is by the application of Lemma 5. Finally, by a union bound over all $i \in [n], k \in [K]$, the Lemma is proved. □

B.3. Lemma 6 for LazyUCB and its Proof

LEMMA 6. *Let $Y_1, \ldots, Y_M$ be i.i.d. Bernoulli random variables with mean p. Denote $\bar{Y} = \frac{1}{M}\sum_{m=1}^{M} Y_m$. For any $\epsilon > 0$, we have*

$$
\Pr\left(p \leq \left(1 + \frac{\epsilon}{2}\right)\left[\bar{Y} + \frac{2+\epsilon}{\epsilon}\cdot\frac{\log(1/\delta)}{M}\right]\right) \geq 1 - \delta, \qquad (51)
$$

$$
\Pr\left(\bar{Y} \leq \left(1 + \frac{\epsilon}{2+\epsilon}\right)\left[p + \frac{2+\epsilon}{\epsilon}\cdot\frac{\log(1/\delta)}{M}\right]\right) \geq 1 - 2\delta. \qquad (52)
$$

The proof of the Lemma crucially uses the following concentration inequalities:

PROPOSITION 2 **(Theorem 1 in (Janson 1999), Theorem 4 in (Chung and Lu 2006))**. *Let $Y_1, \ldots, Y_M$ be i.i.d. Bernoulli random variables with mean p. Denote $\bar{Y} = \frac{1}{M}\sum_{m=1}^{M} Y_m$. For any $\varepsilon \geq 0$, the following inequalities hold:*

$$
\Pr\left(\bar{Y} \geq p + \varepsilon\right) \leq \exp\left[-\frac{M\varepsilon^2}{2(p + \varepsilon/3)}\right], \qquad (53)
$$

$$
\Pr\left(\bar{Y} \leq p - \varepsilon\right) \leq \exp\left[-\frac{M\varepsilon^2}{2p}\right]. \qquad (54)
$$

Electronic copy available at: https://ssrn.com/abstract=3236533

We start with proving (51). First, by unraveling (53) in Proposition 2, we deduce that

$$\begin{aligned}\Pr\left(\bar{Y} \geq p+\varepsilon\right) &\leq \exp\left[-\frac{M\varepsilon^2}{2(p+\varepsilon/3)}\right] \\ &\leq \exp\left[-\frac{M\varepsilon^2}{\max\{4p, 4\varepsilon/3\}}\right] \\ &\leq \exp\left[-\frac{M\varepsilon^2}{4p}\right] + \exp\left[-\frac{3M\varepsilon}{4}\right].\end{aligned}$$

Now,

$$\begin{aligned}&\Pr\left(p \geq \left(1+\frac{\epsilon}{2}\right)\left[\bar{Y} + \frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]\right) \\ &= \Pr\left(\bar{Y} \leq p - \left[\frac{\epsilon}{2+\epsilon}\cdot p + \frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]\right) \\ &\leq \exp\left\{-\frac{M}{2p}\left[\frac{\epsilon}{2+\epsilon}\cdot p + \frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]^2\right\} \qquad (55) \\ &\leq \exp\left\{-\frac{M}{2p}\left[4\cdot\frac{\epsilon}{2+\epsilon}\cdot p\cdot\frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]\right\} \qquad (56) \\ &\leq \exp(-2\log(1/\delta)) = \delta^2 < \delta.\end{aligned}$$

Step (55) is by applying (54) from Proposition (2). Step (56) is by the inequality that $(a+b)^2 \geq 4ab$ for any $a, b \geq 0$.

To complete the proof, we prove (52).

$$\begin{aligned}&\Pr\left(\bar{Y} \geq \left(1+\frac{\epsilon}{2+\epsilon}\right)\left[p + \frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]\right) \\ &= \Pr\left(\bar{Y} \geq p + \left[\frac{\epsilon}{2+\epsilon}\cdot p + \left(1+\frac{\epsilon}{2+\epsilon}\right)\frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]\right) \\ &\leq \exp\left\{-\frac{M}{4p}\cdot\left[\frac{\epsilon}{2+\epsilon}\cdot p + \left(1+\frac{\epsilon}{2+\epsilon}\right)\frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]^2\right\} \qquad (57) \\ &\qquad + \exp\left\{-\frac{3M}{4}\cdot\left[\frac{\epsilon}{2+\epsilon}\cdot p + \left(1+\frac{\epsilon}{2+\epsilon}\right)\frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]\right\}. \qquad (58)\end{aligned}$$

The term (57) is bounded as

$$\begin{aligned}&\exp\left\{-\frac{M}{4p}\cdot\left[\frac{\epsilon}{2+\epsilon}\cdot p + \left(1+\frac{\epsilon}{2+\epsilon}\right)\frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]^2\right\} \\ &\leq \exp\left\{-\frac{M}{4p}\cdot\left[4\cdot\frac{\epsilon}{2+\epsilon}\cdot p\cdot\left(1+\frac{\epsilon}{2+\epsilon}\right)\frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]\right\} \\ &= \exp\left\{-\left(1+\frac{\epsilon}{2+\epsilon}\right)\log(1/\delta)\right\} \leq \delta.\end{aligned}$$

The term (58) is bounded as

$$\begin{aligned}&\exp\left\{-\frac{3M}{4}\cdot\left[\frac{\epsilon}{2+\epsilon}\cdot p + \left(1+\frac{\epsilon}{2+\epsilon}\right)\frac{(2+\epsilon)\log(1/\delta)}{\epsilon M}\right]\right\} \\ &= \exp\left\{-\frac{3Mp\epsilon}{4(2+\epsilon)}\right\}\cdot\exp\left\{-\frac{3+3\epsilon}{2\epsilon}\log\frac{1}{\delta}\right\} \\ &< \delta.\end{aligned}$$

Combining these bounds for (57, 58), the inequality (52) is proved. □

Electronic copy available at: https://ssrn.com/abstract=3236533

B.4. Proof of Lemma 3

First, note that by Lemma 2, we have derived that , for any t, the inequalities

$$q_{(i,k)} \le \bar{q}^t_{(i,k)} + \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}; \delta_t) \le \left(1+\frac{\epsilon}{2}\right)\left[\bar{q}^t_{(i,k)} + \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}; \delta_t)\right]$$
$$\bar{q}^t_{(i,k)} \le q_{(i,k)} + \mathsf{rad}(q_{(i,k)}, M^t_{(i,k)}; \delta_t) \le \left(1+\frac{\epsilon}{2+\epsilon}\right)\left[q_{(i,k)} + \mathsf{rad}(q_{(i,k)}, M^t_{(i,k)}; \delta_t)\right]$$

hold for each an every $i \in [n], k \in [K]$ with probability at least $1 - \frac{4nK}{1+t}$. Therefore, it suffices to show that the inequalities

$$q_{(i,k)} \le \left(1+\frac{\epsilon}{2}\right)\left[\bar{q}^t_{(i,k)} + \mathsf{lad}(\epsilon, M^t_{(i,k)}; \delta_t)\right] \tag{59}$$
$$\bar{q}^t_{(i,k)} \le \left(1+\frac{\epsilon}{2+\epsilon}\right)\left[q_{(i,k)} + \mathsf{lad}(\epsilon, M^t_{(i,k)}; \delta_t)\right] \tag{60}$$

hold simultaneously for all $i \in [n], k \in [K]$ with probability at least $1 - \frac{3nK}{1+t}$.

To show this, we fix a pair (i,k). By applying (51, 52) with the union bout on all the possible values of $M^t_{(i,k)} \in \{0,1,2,\ldots,t-1\}$, we see that for the fixed pair (i,k) the inequalities (59, 60) hold with probability at least $1 - t\delta_t, 1 - 2t\delta_t$ respective. Finally, by a union bound on all possible $i \in [n], k \in [K]$, we show that the inequalities (59, 60) hold for all i,k with probability at least $1 - 3nKt\delta_t \ge 1 - \frac{3nK}{1+t}$, hence the Lemma is proved. □

Appendix C: Regret Analysis of the MAB Oracles for Application 1

C.1. Proof of Theorem 2, Bounding Reg_0 of the UCB Oracle

Denote the event $\bar{\mathcal{E}}^t$ as the complement of the event $\mathcal{E}^t$. Denote $a^t_* = (i^t_*, k^t_*)$ as an optimal action for the auxiliary problem at time t. We have

$$\begin{aligned}
\mathrm{Reg}_0 &= \mathbb{E}\left[\sum_{t=1}^T R^t(a^t_*) - R^t(a^t)\right] \\
&= \sum_{t=1}^T \left\{\mathbb{E}\left[(R^t(a^t_*) - R^t(a^t)) \cdot \mathbf{1}(\mathcal{E}^t)\right] + \mathbb{E}\left[(R^t(a^t_*) - R^t(a^t)) \cdot \mathbf{1}(\bar{\mathcal{E}}^t)\right]\right\} \\
&\le \sum_{t=1}^T \left\{\mathbb{E}\left[(R^t(a^t_*) - R^t(a^t)) \cdot \mathbf{1}(\mathcal{E}^t)\right] + \frac{4nK}{1+t}\right\}. \qquad (61)
\end{aligned}$$

Step (61) is by the model assumption that $r^t_i \in [0,1]$ for all i,t. We focus on upper bounding the first term:

$$\begin{aligned}
&\mathbb{E}\left[R^t(a^t_*)\mathbf{1}(\mathcal{E}^t)\right] \\
\le&\mathbb{E}\left[\mathsf{UCB}^t(a^t_*)\mathbf{1}(\mathcal{E}^t)\right] \qquad (62)\\
\le&\mathbb{E}\left[\mathsf{UCB}^t(a^t)\mathbf{1}(\mathcal{E}^t)\right] \qquad (63)\\
=&\mathbb{E}\left[r^t_{i^t} \cdot \mathbf{1}(x^t_{i^t} = 1) \cdot \bar{q}^t_{(i^t,k^t)} \cdot \mathbf{1}(\mathcal{E}^t)\right] + \mathbb{E}\left[r^t_{i^t} \cdot \mathbf{1}(x^t_{i^t} = 1) \cdot \mathsf{rad}(\bar{q}^t_{(i^t,k^t)}, M^t_{(i^t,k^t)}, \delta_t) \cdot \mathbf{1}(\mathcal{E}^t)\right] \\
\le&\mathbb{E}\left[r^t_{i^t} \cdot \mathbf{1}(x^t_{i^t} = 1) \cdot \left(q_{i^t,k^t} + 3 \cdot \mathsf{rad}(q_{(i^t,k^t)}, M^t_{(i^t,k^t)}, \delta_t)\right)\right] \\
&\quad + \mathbb{E}\left[r^t_{i^t} \cdot \mathbf{1}(x^t_{i^t} = 1) \cdot 3\mathsf{rad}(q_{(i^t,k^t)}, M^t_{(i^t,k^t)}, \delta_t) \cdot \mathbf{1}(\mathcal{E}^t)\right] \qquad (64)\\
\le&\mathbb{E}\left[R^t(a^t)\mathbf{1}(\mathcal{E}^t)\right] + 6\mathbb{E}\left[r^t_{i^t} \cdot \mathsf{rad}(q_{(i^t,k^t)}, M^t_{(i^t,k^t)}, \delta_t)\right]. \qquad (65)
\end{aligned}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

Step (62) is by the property of UCB as shown in (20). Step (63) is by the choice of a^t in Line 5 in the UCB oracle Algorithm 2. Step (64) is by applying Lemma 2 twice. Step (65) is by the definition of r_i^t.

Applying the bound in (65) to the intermediate step (61), we continue bounding Reg_0 as follows:

$$\begin{aligned}
\mathrm{Reg}_0 &\le 6\mathbb{E}\left[\sum_{t=1}^{T} r_{i^t}^t \cdot \mathsf{rad}(q_{(i^t,k^t)}, M^t_{(i^t,k^t)}, \delta_t)\right] + 4nK\log(1+T) \\
&= 6\mathbb{E}\left[\sum_{t=1}^{T} r_{i^t}^t \cdot \left\{\sqrt{\frac{2q_{(i^t,k^t)}\log(1/\delta_t)}{\max\{M^t_{(i^t,k^t)},1\}}} + \frac{3\log(1/\delta_t)}{\max\{M^t_{(i^t,k^t)},1\}}\right\}\right] + 4nK\log(1+T) \\
&\le 6\mathbb{E}\left[\sum_{t=1}^{T} \left\{\sqrt{\frac{4r_{i^t}^t q_{(i^t,k^t)}\log(1+T)}{\max\{M^t_{(i^t,k^t)},1\}}} + \frac{6\log(1+T)}{\max\{M^t_{(i^t,k^t)},1\}}\right\}\right] + 4nK\log(1+T) \qquad (66)
\end{aligned}$$

Step (66) is by invoking the definition of δ_t and the fact that $r_i^t \in [0,1]$. Let's examine the two sums in the expectation.

The first sum.

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T}\sqrt{\frac{4r_{i^t}^t q_{(i^t,k^t)}\log(1+T)}{\max\{M^t_{(i^t,k^t)},1\}}}\right] \\
=&\mathbb{E}\left[\sum_{i\in[n],k\in[K]}\sum_{t=1}^{T}\sqrt{\frac{4r_{i^t}^t q_{(i^t,k^t)}\log(1+T)}{\max\{M^t_{(i^t,k^t)},1\}}}\cdot\mathbf{1}((i^t,k^t)=(i,k))\right] \\
=&\mathbb{E}\left[\sum_{i\in[n],k\in[K]}\sqrt{4r_i q_{(i,k)}\log(1+T)}\sum_{t=1}^{T}\sqrt{\frac{\mathbf{1}((i^t,k^t)=(i,k))\cdot\mathbf{1}(N_i^{t-1}>0)}{\max\{M^t_{(i,k)},1\}}}\right]. \qquad (67)
\end{aligned}$$

Step (67) is by the fact that $r_i^t \le r_i\mathbf{1}(N_i^{t-1}>0)$. To proceed from (67), note that the summand $\sqrt{\frac{\mathbf{1}((i^t,k^t)=(i,k))\cdot\mathbf{1}(N_i^{t-1}>0)}{\max\{M^t_{(i,k)},1\}}}$ is positive only when action (i,k) is taken at time t, and the amount of inventory of i at time t is still positive. Denote $\tau_i = \mathrm{argmax}\left\{t\in[T] : N_i^{t-1}>0\right\}$. Then we know that $M^{\tau_i}_{(i,k)}$ is the number of time steps when action (i,k) is taken, and there is still remaining inventory for item i. By the fact that $\sum_{i=1}^{n} 1/\sqrt{i} \le \sqrt{2n}$ for all n, we have

$$\sum_{t=1}^{T}\sqrt{\frac{\mathbf{1}((i^t,k^t)=(i,k))\cdot\mathbf{1}(N_i^{t-1}>0)}{\max\{M^t_{(i,k)},1\}}} \le 1+\sqrt{2M^{\tau_i}_{(i,k)}},$$

and we can proceed from step (67) as

$$\begin{aligned}
&\mathbb{E}\left[\sum_{i\in[n],k\in[K]}\sqrt{4r_i q_{(i,k)}\log(1+T)}\sum_{t=1}^{T}\sqrt{\frac{\mathbf{1}((i^t,k^t)=(i,k))\cdot\mathbf{1}(N_i^{t-1}>0)}{\max\{M^t_{(i,k)},1\}}}\right] \\
\le& 2nK\sqrt{\log(1+T)} + \mathbb{E}\left[\sum_{i\in[n],k\in[K]}\sqrt{4r_i q_{(i,k)}\log(1+T)}\cdot\sqrt{2M^{\tau_i}_{(i,k)}}\right] \\
\le& 2nK\sqrt{\log(1+T)} + \sqrt{8nK\cdot\mathbb{E}\left[\sum_{i\in[n],k\in[K]} r_i q_{(i,k)} M^{\tau_i}_{(i,k)}\right]\log(1+T)} \qquad (68) \\
=& 2nK\sqrt{\log(1+T)} + \sqrt{8nK\cdot\mathbb{E}\left[\mathsf{ALG}\right]\log(1+T)} \qquad (69) \\
\le& 2nK\sqrt{\log(1+T)} + \sqrt{8nK\cdot\mathsf{OPT}\log(1+T)} \qquad (70)
\end{aligned}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

Step (68) is by the Cauchy Schwartz inequality, and step (69) is by the fact that $r_i q_{(i,k)} M^{\tau_i}_{(i,k)}$ is the amount of reward earned in the T rounds by the algorithm in taking action (i,k), and hence the total reward ALG earned by the algorithm is

$$\mathsf{ALG} = \sum_{i\in[n],k\in[K]} r_i q_{(i,k)} M^{\tau_i}_{(i,k)}.$$

Step (70) is by the fact that the offline benchmark OPT, which is the optimal value of the **Primal LP**, upper bounds $\mathbf{E}[\mathsf{ALG}]$. **The second sum.** We first re-express the sum:

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T}\frac{6\log(1+T)}{\max\{M^t_{(i^t,k^t)},1\}}\right]\\
=&6\log(1+T)\cdot\mathbb{E}\left[\sum_{i\in[n],k\in[K]}\sum_{t=1}^{T}\frac{\mathbf{1}((i^t,k^t=(i,k)))}{\max\{M^t_{(i,k)},1\}}\right]\\
\le&6\log(1+T)\cdot\mathbb{E}\left[\sum_{i\in[n],k\in[K]}(2+\log(M^T_{i,k}))\right] &(71)\\
\le&6nK\log(1+T)\left(2+\log\frac{T}{nK}\right). &(72)
\end{aligned}$$

Step (71) is by the fact that $\sum_{i=1}^{N}\frac{1}{i}\le 1+\log n$, and step (72) is by the Jensen's inequality and the fact that $\sum_{i\in[n],k\in[K]} M^T_{i,k} = T$.

Combining the bounds (69) and (72), we finally have

$$\begin{aligned}
\mathrm{Reg}_0 &\le 12\sqrt{2nK\cdot\mathsf{OPT}\log(1+T)}+12nK\sqrt{\log(1+T)}+36nK\log(1+T)\left(2+\log\frac{T}{nK}\right)+4nK\log(1+T)\\
&\le 12\sqrt{2nK\cdot\mathsf{OPT}\log(1+T)}+52nK\log(1+T)\left(2+\log\frac{T}{nK}\right)\\
&= O\left(\sqrt{nK\mathsf{OPT}\log(T)}+nK\log(T)\log(T/nK)\right). \quad \square
\end{aligned}$$

C.2. Proof of Theorem 3, Bounding $\frac{1}{1+\epsilon}\mathbf{Reg}_\epsilon$ for the LazyUCB Oracle

Similar to the analysis for the UCB oracle, we denote $a^t_* = (i^t_*, k^t_*)$.

$$\begin{aligned}
\frac{1}{1+\epsilon}\cdot\mathrm{Reg}_\epsilon &= \frac{1}{1+\epsilon}\cdot\mathbb{E}\left[\sum_{t=1}^{T}R^t(a^t_*)\right]-\mathbb{E}\left[\sum_{t=1}^{T}R^t(a^t)\right]\\
&= \frac{1}{1+\epsilon}\cdot\mathbb{E}\left[\sum_{t=1}^{T}R^t(a^t_*)\mathbf{1}(\mathcal{E}^t)\right]+\frac{1}{1+\epsilon}\cdot\mathbb{E}\left[\sum_{t=1}^{T}R^t(a^t_*)\mathbf{1}(\bar{\mathcal{E}}^t)\right]-\mathbb{E}\left[\sum_{t=1}^{T}R^t(a^t)\right]\\
&\le \frac{1}{1+\epsilon}\cdot\mathbb{E}\left[\sum_{t=1}^{T}R^t(a^t_*)\mathbf{1}(\mathcal{E}^t)\right]-\mathbb{E}\left[\sum_{t=1}^{T}R^t(a^t)\right]+\frac{1}{1+\epsilon}\sum_{t=1}^{T}\frac{7nK}{1+t}. &(73)
\end{aligned}$$

Step (73) is by applying Lemma 3.

To proceed, we focus on the first term:

$$\begin{aligned}
&\frac{1}{1+\epsilon}\cdot\mathbb{E}\left[\sum_{t=1}^{T}R^t(a^t_*)\cdot\mathbf{1}(\mathcal{E}^t)\right]\\
\le&\frac{1+\epsilon/2}{1+\epsilon}\cdot\mathbb{E}\left[\sum_{t=1}^{T}\mathsf{LazyUCB}^t(a^t_*)\cdot\mathbf{1}(\mathcal{E}^t)\right] &(74)
\end{aligned}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

$$\leq \frac{1+\epsilon/2}{1+\epsilon} \cdot \mathbb{E}\left[\sum_{t=1}^{T} \mathsf{LazyUCB}^t(a^t) \cdot \mathbf{1}(\mathcal{E}^t)\right] \tag{75}$$

$$\leq \frac{1+\epsilon/2}{1+\epsilon} \cdot \mathbb{E}\left[\sum_{t=1}^{T} r^t_{i^t} \cdot \mathbf{1}(x^t_{i^t}=1) \cdot \left[\bar{q}^t_{(i^t,k^t)} + \mathsf{LazyRad}^t(i^t,k^t)\right] \cdot \mathbf{1}(\mathcal{E}^t)\right] \tag{76}$$

$$\leq \frac{1+\epsilon/2}{1+\epsilon} \cdot \mathbb{E}\left[\sum_{t=1}^{T} r^t_{i^t} \cdot \mathbf{1}(x^t_{i^t}=1) \cdot \left[\left(1+\frac{\epsilon}{2+\epsilon}\right)\left[q_{(i^t,k^t)} + \mathsf{LazyRad}^t(i^t,k^t)\right] + \mathsf{LazyRad}^t(i^t,k^t)\right] \cdot \mathbf{1}(\mathcal{E}^t)\right] \tag{77}$$

$$\leq \mathbb{E}\left[\sum_{t=1}^{T} r^t_{i^t} \cdot \mathbf{1}(x^t_{i^t}=1) \cdot q_{(i^t,k^t)}\right] + 2 \cdot \mathbb{E}\left[\sum_{t=1}^{T} \mathsf{LazyRad}^t(i^t,k^t)\right]$$

$$= \mathbb{E}\left[\sum_{t=1}^{T} R^t(a^t)\right] + 2 \cdot \mathbb{E}\left[\sum_{t=1}^{T} \mathsf{LazyRad}^t(i^t,k^t)\right]. \tag{78}$$

Step (74) is by Lemma 3, step (75) is by Line 5 in the LazyUCB oracle. Step (76) is by applying inequality (23) in Lemma 3, and step (77) is by applying inequality (24) in Lemma 3. We next proceed with bounding the confidence radii in (78):

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T} \mathsf{LazyRad}^t(i^t,k^t)\right] &= \mathbb{E}\left[\sum_{t=1}^{T} \min\left\{\mathsf{lad}(\epsilon, M^t_{(i,k)}; \delta_t), \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}, \delta^{(t)})\right\}\right] \\
&\leq \mathbb{E}\left[\min\left\{\sum_{t=1}^{T} \mathsf{lad}(\epsilon, M^t_{(i,k)}; \delta_t), \sum_{t=1}^{T} \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}, \delta^{(t)})\right\}\right] \\
&\leq \min\left\{\mathbb{E}\left[\sum_{t=1}^{T} \mathsf{lad}(\epsilon, M^t_{(i,k)}; \delta_t)\right], \mathbb{E}\left[\sum_{t=1}^{T} \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}, \delta^{(t)})\right]\right\}.
\end{aligned}$$

By the analysis in the proof of Theorem 2, we see that

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathsf{rad}(\bar{q}^t_{(i,k)}, M^t_{(i,k)}, \delta^{(t)})\right] \leq 2\sqrt{2nK \cdot \mathsf{OPT} \log(1+T)} + 8nK\log(1+T)\left(2+\log\frac{T}{nK}\right) \tag{79}$$

$$= O\left(\sqrt{nK \cdot \mathsf{OPT} \cdot \log(T)} + nK\log(T)\log\frac{T}{nK}\right). \tag{80}$$

Next, we can upper bound the sum on lad as follows:

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T} \mathsf{lad}(\epsilon, M^t_{(i^t,k^t)}; \delta_t)\right] \\
\leq &\mathbb{E}\left[\sum_{i\in[n],k\in[K]} \sum_{t=1}^{T} \mathsf{lad}(\epsilon, M^t_{(i,k)}; \delta_t)\mathbf{1}((i^t,k^t)=(i,k))\right] \\
\leq &\frac{2(2+\epsilon)\log(1+T)}{\epsilon} \cdot \mathbb{E}\left[\sum_{i\in[n],k\in[K]} \sum_{t=1}^{T} \frac{1}{M^t_{(i,k)}} \cdot \mathbf{1}((i^t,k^t)=(i,k))\right] \\
\leq &\frac{2(2+\epsilon)\log(1+T)}{\epsilon} \cdot \mathbb{E}\left[\sum_{i\in[n],k\in[K]} \left(1+\log(M^T_{i,k})\mathbf{1}(M^T_{(i,k)}>0)\right)\right] \\
\leq &\frac{2(2+\epsilon)\log(1+T)}{\epsilon} \cdot nK \cdot \log\frac{T}{nK} \qquad (81) \\
= &O\left(\frac{(1+\epsilon)nK\log(T)\log(T/nK)}{\epsilon}\right). \qquad (82)
\end{aligned}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

Step (81) is by the Jensen inequality and the concavity of log. Finally, applying (77, 80, 82) back to (73), we have

$$\begin{aligned}
&\frac{1}{1+\epsilon}\cdot \mathrm{Reg}_\epsilon = \min\left\{O\left(\sqrt{nK\cdot \mathsf{OPT}\cdot \log T} + nK\log(T)\log\frac{T}{nK}\right), O\left(\frac{(1+\epsilon)nK\log(T)\log(T/nK)}{\epsilon}\right)\right\}\\
&= \min\left\{O\left(\sqrt{nK\cdot \mathsf{OPT}\cdot \log T}\right), O\left(\frac{(1+\epsilon)nK\log(T/nK)\log(T)}{\epsilon}\right)\right\} + O\left(nK\log(T)\log\left(\frac{T}{nK}\right)\right)\\
&= \min\left\{\tilde{O}(\sqrt{nK\mathsf{OPT}}), \tilde{O}\left(\frac{(1+\epsilon)nK}{\epsilon}\right)\right\}. \quad \square
\end{aligned}$$

Appendix D: Proof of Theorem 7 (Lower Bound on Regret of Algorithm)

In this section, we establish a lower bound on the overall loss of any online algorithm for the online matching problem. Specifically, we prove that the performance guarantee in Corollary 1 is tight in the sense that both of the loss terms OPT/e, $\tilde{O}(\sqrt{\mathbb{E}[\mathsf{OPT}]})$ are unavoidable due to the uncertainty on the probabilities $q_{(i,k)}$ and the uncertainty on the sequence of customer contexts.

We construct a randomized worst-case instance as follows. The capacity values are the same $b_i = b$ for all $i \in [n]$. Let π be a random permutation of $[n]$. There are $T = 2bn$ customers, split into n "groups" of $2b$ customers each. The customers in each group $j \in [n]$ all have the same context (feature) vector $x^{(j)}$, where

$$x_i^{(j)} = 1 \text{ if and only if } \pi(i) \geq j.$$

In other words, if we view $\pi(i)$ as a random score of resource i, then the customers become increasingly selective as customers in group j are only interested in resources i with scores higher than j.

Let $\boldsymbol{\ell} = (\ell_1, \ldots, \ell_n) \in [K]^n$ be a random vector of "secret arms". The distribution $\rho_{x,(i,k)}$ is given by

$$\begin{aligned}
&\rho_{x,(i,k)}(\mathbf{e}_i) = \mathbb{1}(x_i = 1)\left(\frac{1-\varepsilon}{2} + \mathbb{1}(k = \ell_i)\cdot\varepsilon\right)\\
&\rho_{x,(i,k)}(\mathbf{0}) = 1 - \rho_{x,(i,k)}(\mathbf{e}_i)\\
&\rho_{x,(i,k)}(\mathbf{y}) = 0 \text{ for all other outcomes } \mathbf{y} \text{ in } \{0,1\}^n
\end{aligned}$$

Here, $\varepsilon \in (0, 1/2]$ will be defined in our analysis. We choose $\varepsilon \leq 1/2$ just for technical convenience.

This problem instance is a randomized one because we draw both π and $\boldsymbol{\ell}$ uniformly at random. Note that for all realization of π and $\boldsymbol{\ell}$, OPT will be bn.

A *deterministic policy* is a mapping, for any $t \in \mathbb{N}$, from any history of observed contexts and outcomes, $(x^1, \mathbf{y}^1, \ldots, x^t)$ in $\mathcal{X}^t \times \{0,1\}^{n\times(t-1)}$, to an action to play on context x^t, in $\mathcal{A}$. Our proof strategy is to upper-bound the performance of any deterministic policy on this randomized instance (it suffices to consider deterministic policies because when given the randomized instance, there always exists an optimal policy which is deterministic).

THEOREM 7 **(Lower Bound)**. *Let n, b, K be any positive integers satisfying $b \geq K \geq 3$. Then there exists a randomized instance (with a random arrival sequence and a random mapping from contexts to outcomes) such that for any deterministic or randomized algorithm,*

$$\mathsf{OPT} - \mathbb{E}[\mathsf{ALG}] \geq \frac{\mathsf{OPT}}{e} + \Theta(\sqrt{K\mathsf{OPT}}).$$

Electronic copy available at: https://ssrn.com/abstract=3236533

We prove this theorem through Lemmas 7, 8, 9, and Proposition 3. The proof is based on an information-theoretic analysis.

Let $\mathcal{T}_j = \{2b(j-1)+1, \ldots, 2bj\}$ denote the indices of the customers in group j, for all $j \in [n]$. Let $\mathcal{A}_i = \{(i,k) : k \in [K]\}$ denote the set of actions that correspond to resource i, for all $i \in [n]$. Let Y_t be the indicator random variable for whether customer t accepted her offer, for all $t \in [T]$.

We can write ALG, the random variable for the total reward earned by the deterministic policy, as

$$\mathsf{ALG} = \sum_{i=1}^{n} \min\Big\{ \sum_{j=1}^{i} \sum_{t \in \mathcal{T}_j} \mathbb{1}(Y_t = 1 \cap a^t \in \mathcal{A}_{\pi^{-1}(i)}), b \Big\}. \tag{83}$$

To upper-bound $\mathbb{E}[\mathsf{ALG}]$, we need to upper-bound $\mathbb{E}[\sum_{j=1}^{i} \sum_{t \in \mathcal{T}_j} \mathbb{1}(Y_t = 1 \cap a^t \in \mathcal{A}_{\pi^{-1}(i)})]$. Thus, we will focus on analyzing $\Pr[Y_t = 1 \cap a^t \in \mathcal{A}_{\pi^{-1}(i)}]$ for an arbitrary $i \in [n]$, $j \le i$, and $t \in \mathcal{T}_j$.

$$\begin{aligned}
&\Pr[Y_t = 1 \cap a^t \in \mathcal{A}_{\pi^{-1}(i)}] \\
=&\Pr[Y_t = 1 | a^t = (\pi^{-1}(i), \ell_{\pi^{-1}(i)})] \cdot \Pr[a^t = (\pi^{-1}(i), \ell_{\pi^{-1}(i)})] \\
&+ \Pr[Y_t = 1 | a^t \in \mathcal{A}_{\pi^{-1}(i)} \cap a^t \neq (\pi^{-1}(i), \ell_{\pi^{-1}(i)})] \cdot \Pr[a^t \in \mathcal{A}_{\pi^{-1}(i)} \cap a^t \neq (\pi^{-1}(i), \ell_{\pi^{-1}(i)})] \\
=&\frac{1+\varepsilon}{2} \Pr[a^t = (\pi^{-1}(i), \ell_{\pi^{-1}(i)})] + \frac{1-\varepsilon}{2} \Pr[a^t \in \mathcal{A}_{\pi^{-1}(i)} \cap a^t \neq (\pi^{-1}(i), \ell_{\pi^{-1}(i)})] \\
=&\frac{1-\varepsilon}{2} \Pr[a^t \in \mathcal{A}_{\pi^{-1}(i)}] + \varepsilon \cdot \Pr[a^t = (\pi^{-1}(i), \ell_{\pi^{-1}(i)})]
\end{aligned} \tag{84}$$

The difficult term to analyze is $\Pr[a^t = (\pi^{-1}(i), \ell_{\pi^{-1}(i)})]$. Note that the distribution of a^t is affected by the entire realized vector of secret arms $\boldsymbol{\ell}$, as well as the realized values of $\pi^{-1}(1), \ldots, \pi^{-1}(j-1)$.

Now, consider an alternate universe where for each resource $m \in [n]$, all of the actions $(m,1), \ldots, (m,K)$ result in the customer accepting with probability $\frac{1-\varepsilon}{2}$, regardless of the value of ℓ_m. We can also consider the execution of the fixed, deterministic policy in this alternate universe, where we will use random variables $\overline{a}^t, \overline{Y}_t$ to refer to its execution.

LEMMA 7 **(Using information theory to get an initial bound)**. *Let $j \in [n]$ be any customer group and let t be any customer from $\mathcal{T}_j$. Let $S \subseteq [n]$ be any set of resources. Condition on any sequence of $j-1$ resources with lowest scores*

$$\pi^{-1}([j-1]) := (\pi^{-1}(1), \ldots, \pi^{-1}(j-1))$$

and vector of secret arms $\boldsymbol{\ell}$. Then

$$\begin{aligned}
\sum_{m \in S} \Pr[a^t = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}] \le &\sum_{m \in S} \Pr[\overline{a}^t = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}] \\
&+ \varepsilon \sqrt{\sum_{s=1}^{t-1} \sum_{m \notin \pi^{-1}([s-1])} \Pr[\overline{a}^s = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}]}.
\end{aligned} \tag{85}$$

Proof. For brevity, we will omit the conditioning on $\pi^{-1}(1), \ldots, \pi^{-1}(j-1)$ and $\boldsymbol{\ell}$ throughout the proof. We will also use $\mathbf{Z}^s$ to denote the vector of random variables $(Y_1, \ldots, Y_s)$ and $\mathbf{z}^s$ to denote a vector in $\{0,1\}^s$, for any $s \in [t-1]$.

First, note that a^t is the rule of the deterministic policy for choosing the action at time t, dependent on sequence of observations $\mathbf{Z}^{t-1}$ and the sequence of contexts $x^1, \ldots, x^t$ (which is captured by $\pi^{-1}(1), \ldots, \pi^{-1}(j-1)$).

Electronic copy available at: https://ssrn.com/abstract=3236533

$$\begin{aligned}\sum_{m\in S}\Pr[a^t=(m,\ell_m)] &= \sum_{\mathbf{z}^{t-1}\in\{0,1\}^{t-1}}\Pr[\mathbf{Z}^{t-1}=\mathbf{z}^{t-1}]\sum_{m\in S}\Pr[a^t=(m,\ell_m)|\mathbf{Z}^{t-1}=\mathbf{z}^{t-1}]\\ &\le \sum_{\mathbf{z}^{t-1}\in\{0,1\}^{t-1}}\Pr[\overline{\mathbf{Z}}^{t-1}=\mathbf{z}^{t-1}]\sum_{m\in S}\Pr[\overline{a}^t=(m,\ell_m)|\overline{\mathbf{Z}}^{t-1}=\mathbf{z}^{t-1}]\\ &\quad+\delta(\overline{\mathbf{Z}}^{t-1},\mathbf{Z}^{t-1})\\ &\le \sum_{m\in S}\Pr[\overline{a}^t=(m,\ell_m)]+\sqrt{\frac{1}{2}\mathsf{KL}(\overline{\mathbf{Z}}^{t-1}\|\mathbf{Z}^{t-1})},\end{aligned}\tag{86}$$

where the first inequality is from the definition that

$$\delta(\overline{\mathbf{Z}}^{t-1},\mathbf{Z}^{t-1})=\sum_{\mathbf{z}^{t-1}\in\{0,1\}^{t-1}}|\Pr[\overline{\mathbf{Z}}^{t-1}=\mathbf{z}^{t-1}]-\Pr[\mathbf{Z}^{t-1}=\mathbf{z}^{t-1}]|,$$

and the second inequality is due to Pinsker's inequality.

$$\begin{aligned}&\mathsf{KL}(\overline{\mathbf{Z}}^{t-1}\|\mathbf{Z}^{t-1})\\ =&\sum_{\mathbf{z}^{t-1}\in\{0,1\}^{t-1}}\Pr[\overline{\mathbf{Z}}^{t-1}=\mathbf{z}^{t-1}]\cdot\ln\frac{\Pr[\overline{\mathbf{Z}}^{t-1}=\mathbf{z}^{t-1}]}{\Pr[\mathbf{Z}^{t-1}=\mathbf{z}^{t-1}]}\\ =&\sum_{s=1}^{t-1}\sum_{\mathbf{z}^{s-1}\in\{0,1\}^{t-1}}\Pr[\overline{\mathbf{Z}}^{s-1}=\mathbf{z}^{s-1}]\left(\sum_{y_s\in\{0,1\}}\Pr[\overline{Y}_s=y_s|\overline{\mathbf{Z}}^{s-1}=\mathbf{z}^{s-1}]\cdot\ln\frac{\Pr[\overline{Y}_s=y_s|\overline{\mathbf{Z}}^{s-1}=\mathbf{z}^{s-1}]}{\Pr[Y_s=y_s|\mathbf{Z}^{s-1}=\mathbf{z}^{s-1}]}\right),\end{aligned}$$

where the second equality comes from the Chain Rule for KL-divergences. Now, consider the term inside the parentheses. Conditioned on $\mathbf{z}^{s-1}$ (and $\pi^{-1}([j-1])$, which have been omitted in the notation), actions $\overline{a}^s$ and a^s are deterministic and equal. If this action is (m,ℓ_m) for some $m\in[n]$ and $m\notin\pi^{-1}([s-1])$, then $\overline{Y}_s$ is 1 w.p. $\frac{1-\varepsilon}{2}$ while Y_s is 1 w.p. $\frac{1+\varepsilon}{2}$, and the term inside the parentheses is the KL-divergence of $\mathsf{Ber}(\frac{1+\varepsilon}{2})$ from $\mathsf{Ber}(\frac{1-\varepsilon}{2})$, equal to $\varepsilon\cdot\ln\frac{1+\varepsilon}{1-\varepsilon}$. Otherwise, $\overline{Y}_s$ and Y_s are identically distributed, and the term inside the parentheses is zero.

Therefore,

$$\begin{aligned}&\mathsf{KL}(\overline{\mathbf{Z}}^{t-1}\|\mathbf{Z}^{t-1})\\ =&\sum_{s=1}^{t-1}\sum_{m\notin\pi^{-1}([s-1])}\Pr[\overline{a}^s=(m,\ell_m)]\left(\varepsilon\cdot\ln\frac{1+\varepsilon}{1-\varepsilon}\right)\\ \le&\sum_{s=1}^{t-1}\sum_{m\notin\pi^{-1}([s-1])}\Pr[\overline{a}^s=(m,\ell_m)]\left(2\varepsilon^2\right)\end{aligned}$$

(the inequality is because $\varepsilon\le 1/2$) and substituting into (86) completes the proof of the lemma.

□

Definition 1. Define the following random variables for all $i,j\in[n]$:

- $Q_{i,j}=\sum_{t\in\mathcal{T}_j}\mathbb{1}(a^t\in\mathcal{A}_{\pi^{-1}(i)})$ is the total number of group-j customers on whom an action corresponding to resource $\pi^{-1}(i)$ is played;
- $Q^*_{i,j}=\sum_{t\in\mathcal{T}_j}\mathbb{1}(a^t=(\pi^{-1}(i),\ell_{\pi^{-1}(i)}))$ is the total number of group-j customers on whom action $(\pi^{-1}(i),\ell_{\pi^{-1}(i)})$ is played.

Electronic copy available at: https://ssrn.com/abstract=3236533

Let $q_{i,j}, q^*_{i,j}$ denote the expected values of $Q_{i,j}, Q^*_{i,j}$, respectively. We will use $\overline{Q}_{i,j}, \overline{Q}^*_{i,j}, \overline{q}_{i,j}, \overline{q}^*_{i,j}$ to refer to the respective quantities under the alternate universe.

LEMMA 8 **(Removing dependence on t, π, and $\boldsymbol{\ell}$).** *Let $D \subseteq [n]$ be any set of scores, and $\pi^{-1}(D)$ be the corresponding set of resources with scores D. For any group $j \in [n]$,*

$$\sum_{i \in D} \mathbb{E}[Q^*_{i,j}] \leq \frac{1}{K} \sum_{i \in D} \mathbb{E}[\overline{Q}_{i,j}] + 2b\varepsilon\sqrt{\frac{2bj}{K}}.$$

Proof. Consider the probability

$$\Pr[\overline{a}^s = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}]$$

from the RHS of inequality (85). Since $\overline{a}^s$, which refers to the alternate universe, is unaffected by the value of ℓ_m, the probability is identical after removing the conditioning on ℓ_m. We can do this for all $s = 1, \ldots, t$.

Let $\boldsymbol{\ell}_{-m}$ denote the fixed vector of secret arms for resources other than m. We take an average over the randomness in ℓ_m (drawn uniformly from $[K]$) and apply the law of total probability to obtain:

$$\begin{aligned}
&\mathbb{E}[\Pr[\overline{a}^s = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}]] \\
=&\mathbb{E}[\Pr[\overline{a}^s = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}_{-m}]] \\
\leq&\mathbb{E}[\frac{1}{K} \Pr[\overline{a}^s \in \mathcal{A}_m | \pi^{-1}([j-1]), \boldsymbol{\ell}_{-m}]] \\
=&\frac{1}{K} \Pr[\overline{a}^s \in \mathcal{A}_m],
\end{aligned}$$

where the inequality is because the probability that $\overline{a}^s$ turns out to be the "secret arm" ℓ_m of resource m is $1/K$ if $\overline{a}^s \in \mathcal{A}_m$, and 0 otherwise.

Then, for any set $S \subseteq [n]$ of resources, we apply inequality (85) to obtain:

$$\begin{aligned}
&\sum_{m \in S} \Pr[a^t = (m, \ell_m)] \\
&= \sum_{m \in S} \mathbb{E}[\Pr[a^t = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}]] \\
&\leq \sum_{m \in S} \mathbb{E}[\Pr[\overline{a}^t = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}]] + \varepsilon \cdot \mathbb{E}\left[\sqrt{\sum_{s=1}^{t-1} \sum_{m \notin \pi^{-1}([s-1])} \Pr[\overline{a}^s = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}]}\right] \\
&\leq \sum_{m \in S} \mathbb{E}[\Pr[\overline{a}^t = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}]] + \varepsilon \cdot \sqrt{\sum_{s=1}^{t-1} \mathbb{E}\left[\sum_{m \notin \pi^{-1}([s-1])} \Pr[\overline{a}^s = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}]\right]} \\
&\leq \sum_{m \in S} \mathbb{E}[\Pr[\overline{a}^t = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}]] + \varepsilon \cdot \sqrt{\sum_{s=1}^{t-1} \mathbb{E}\left[\sum_{m \in [n]} \Pr[\overline{a}^s = (m, \ell_m) | \pi^{-1}([j-1]), \boldsymbol{\ell}]\right]} \\
&\leq \frac{1}{K} \sum_{m \in S} \Pr[\overline{a}^t \in \mathcal{A}_m] + \varepsilon \sqrt{\frac{1}{K} \sum_{s=1}^{t-1} \sum_{m \in [n]} \Pr[\overline{a}^s \in \mathcal{A}_m]} \\
&\leq \frac{1}{K} \sum_{m \in S} \Pr[\overline{a}^t \in \mathcal{A}_m] + \varepsilon \sqrt{\frac{t}{K}}.
\end{aligned}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

The second inequality is Jensen's inequality (the square root function is concave).

By the definition of $Q_{i,j}$ and $Q^*_{i,j}$, we sum over the $2b$ values of t in $\mathcal{T}_j$ to obtain

$$\begin{aligned}
&\sum_{i\in D}\mathbb{E}[Q^*_{i,j}]\\
=&\sum_{t\in\mathcal{T}_j}\mathbb{E}\left[\sum_{m\in\pi^{-1}(D)}\Pr[a^t=(m,\ell_m)]\right]\\
=&\sum_{t\in\mathcal{T}_j}\mathbb{E}\left[\mathbb{E}\left[\sum_{m\in S}\Pr[a^t=(m,\ell_m)]\middle|\pi^{-1}(D)=S\right]\right]\\
\le&\sum_{t\in\mathcal{T}_j}\mathbb{E}\left[\mathbb{E}\left[\frac{1}{K}\sum_{m\in S}\Pr[\overline{a}^t\in\mathcal{A}_m]+\varepsilon\sqrt{\frac{t}{K}}\middle|\pi^{-1}(D)=S\right]\right]\\
=&\frac{1}{K}\sum_{t\in\mathcal{T}_j}\mathbb{E}\left[\sum_{m\in\pi^{-1}(D)}\Pr[\overline{a}^t\in\mathcal{A}_m]\right]+\sum_{t\in\mathcal{T}_j}\varepsilon\sqrt{\frac{t}{K}}\\
=&\frac{1}{K}\sum_{i\in D}\mathbb{E}[\overline{Q}_{i,j}]+\sum_{t\in\mathcal{T}_j}\varepsilon\sqrt{\frac{t}{K}}\\
\le&\frac{1}{K}\sum_{i\in D}\mathbb{E}[\overline{Q}_{i,j}]+\varepsilon 2b\sqrt{\frac{2bj}{K}}.
\end{aligned}$$

The last inequality uses the fact that $t\le 2bj$ for all $t\in\mathcal{T}_j$.

□

LEMMA 9 **(Argument for randomized permutation)**. *For any customer group $j\in[n]$ and compatible resource with score $i\ge j$, both $\mathbb{E}[Q_{i,j}]$ and $\mathbb{E}[\overline{Q}_{i,j}]$ are upper-bounded by $2b/(n-j+1)$.*

Proof. We prove the result for $\mathbb{E}[Q_{i,j}]$ (the proof for $\mathbb{E}[\overline{Q}_{i,j}]$ is identical):

$$\begin{aligned}
\mathbb{E}[Q_{i,j}]=&\sum_{t\in\mathcal{T}_j}\Pr[a^t\in\mathcal{A}_{\pi^{-1}(i)}]\\
=&\sum_{t\in\mathcal{T}_j}\sum_{\pi^{-1}([j-1])}\Pr[\pi^{-1}([j-1])]\cdot\Pr[a^t\in\mathcal{A}_{\pi^{-1}(i)}|\pi^{-1}([j-1])]\\
=&\sum_{t\in\mathcal{T}_j}\sum_{\pi^{-1}([j-1])}\Pr[\pi^{-1}([j-1])]\sum_{m\notin\pi^{-1}([j-1])}\Pr[\pi(m)=i|\pi^{-1}(i)]\cdot\Pr[a^t\in\mathcal{A}_m|\pi^{-1}([j-1]),\pi(m)=i]\\
=&\sum_{t\in\mathcal{T}_j}\sum_{\pi^{-1}([j-1])}\Pr[\pi^{-1}([j-1])]\sum_{m\notin\pi^{-1}([j-1])}\frac{1}{n-j+1}\Pr[a^t\in\mathcal{A}_m|\pi^{-1}([j-1])]\\
\le&\sum_{t\in\mathcal{T}_j}\sum_{\pi^{-1}([j-1])}\Pr[\pi^{-1}([j-1])]\cdot\frac{1}{n-j+1}(1)\\
=&\frac{2b}{n-j+1}.
\end{aligned}$$

The first equality is by definition and the linearity of expectation; the second and third equalities are by the law of total probability; and the fourth equality is by the fact that a^t is independent of $\pi^{-1}(i)$, which completes the proof of the lemma.

□

Electronic copy available at: https://ssrn.com/abstract=3236533

Now, combining (83), (84), and definitions, we get that

$$\mathbb{E}[\mathsf{ALG}] \le \sum_{i=1}^{n} \min\Big\{ \sum_{j=1}^{i} \big(\frac{1-\varepsilon}{2}\mathbb{E}[Q_{i,j}] + \varepsilon \cdot \mathbb{E}[Q^*_{i,j}]\big), b\Big\}, \tag{87}$$

where we have also used the fact that $\min\{\cdot, b\}$ is concave. For all $i \in [n]$, let

$$H^n_{n-i} := (1 + \frac{1}{2} + \ldots + \frac{1}{n}) - (1 + \frac{1}{2} + \ldots + \frac{1}{n-i}) = \sum_{j=1}^{i} \frac{1}{n-j+1}. \tag{88}$$

Now, let $n' \in [n]$ be the largest value such that $H^n_{n-n'} \le 1$.

$$\begin{aligned}
&\sum_{i=1}^{n'} \sum_{j=1}^{i} \big(\frac{1-\varepsilon}{2}\mathbb{E}[Q_{i,j}] + \varepsilon \cdot \mathbb{E}[Q^*_{i,j}]\big) \\
\le &\sum_{i=1}^{n'} (1-\varepsilon) b \cdot H^n_{n-i} + \varepsilon \cdot \sum_{j=1}^{n'} \sum_{i=j}^{n'} \mathbb{E}[Q^*_{i,j}] \\
&\quad \text{(by Lemma 9)} \\
\le &\sum_{i=1}^{n'} (1-\varepsilon) b \cdot H^n_{n-i} + \varepsilon \cdot \sum_{j=1}^{n'} \left(\frac{1}{K} \sum_{i=j}^{n'} \mathbb{E}[\overline{Q}_{i,j}] + \varepsilon 2b \sqrt{\frac{2bj}{K}} \right) \\
&\quad \text{(by Lemma 8)} \\
= &\sum_{i=1}^{n'} (1-\varepsilon) b \cdot H^n_{n-i} + \frac{\varepsilon}{K} \cdot \sum_{i=1}^{n'} \sum_{j=1}^{i} \mathbb{E}[\overline{Q}_{i,j}] + \varepsilon^2 2b \cdot \sum_{j=1}^{n'} \sqrt{\frac{2bj}{K}} \\
\le &\sum_{i=1}^{n'} (1-\varepsilon) b \cdot H^n_{n-i} + \frac{\varepsilon 2b}{K} \cdot \sum_{i=1}^{n'} H^n_{n-i} + \varepsilon^2 2b \cdot \sum_{j=1}^{n'} \sqrt{\frac{2bj}{K}} \\
&\quad \text{(by Lemma 9)} \\
= &b \cdot \sum_{i=1}^{n'} H^n_{n-i} \cdot \left[1 - \varepsilon \left(1 - \frac{2}{K} \right) \right] + \varepsilon^2 2b \cdot \sum_{j=1}^{n'} \sqrt{\frac{2bj}{K}} \\
\le &b \cdot \sum_{i=1}^{n'} H^n_{n-i} \cdot \left[1 - \varepsilon \left(1 - \frac{2}{K} \right) \right] + \varepsilon^2 2b \cdot n \sqrt{\frac{2bn}{K}}.
\end{aligned}$$

Since $\min\{x, y\} \le x$ and $\min\{x, y\} \le y$, we can obtain

$$\begin{aligned}
&\mathbb{E}[\mathsf{ALG}] \\
\le &\sum_{i=1}^{n} \min\Big\{ \sum_{j=1}^{i} \big(\frac{1-\varepsilon}{2}\mathbb{E}[Q_{i,j}] + \varepsilon \cdot \mathbb{E}[Q^*_{i,j}]\big), b\Big\} \\
\le &\sum_{i=1}^{n'} \sum_{j=1}^{i} \big(\frac{1-\varepsilon}{2}\mathbb{E}[Q_{i,j}] + \varepsilon \cdot \mathbb{E}[Q^*_{i,j}]\big) + \sum_{i=n'+1}^{n} b \\
\le &b \cdot \sum_{i=1}^{n'} H^n_{n-i} \cdot \left[1 - \varepsilon \left(1 - \frac{2}{K} \right) \right] + \varepsilon^2 2b \cdot n \sqrt{\frac{2bn}{K}} + \sum_{i=n'+1}^{n} b \\
= &b \cdot \sum_{i=1}^{n} \min(H^n_{n-i}, 1) - b \cdot \sum_{i=1}^{n'} H^n_{n-i} \cdot \varepsilon \left(1 - \frac{2}{K} \right) + \varepsilon^2 2b \cdot n \sqrt{\frac{2bn}{K}}.
\end{aligned} \tag{89}$$

The last equality is because $H^n_{n-i} \le 1$ for all $i \le n'$.

Electronic copy available at: https://ssrn.com/abstract=3236533

Make the technical assumptions $b \geq K \geq 3$, and set

$$\varepsilon := \frac{1}{34}\sqrt{\frac{K}{bn}}$$

which satisfies the condition that $\varepsilon \leq 1/2$.

Substituting back into (89), we obtain

$$\begin{aligned}
&\mathbb{E}[\mathsf{ALG}] \\
\leq& b \cdot \sum_{i=1}^{n} \min(H_{n-i}^{n}, 1) - b \cdot \sum_{i=1}^{n'} H_{n-i}^{n} \cdot \varepsilon \left(1 - \frac{2}{K}\right) + \varepsilon^2 2b \cdot n\sqrt{\frac{2bn}{K}} \\
=& b \cdot \sum_{i=1}^{n} \min(H_{n-i}^{n}, 1) - \sqrt{\frac{Kb}{n}} \left[\frac{1}{34}\left(1 - \frac{2}{K}\right)\sum_{i=1}^{n'} H_{n-i}^{n} - \frac{\sqrt{2}}{578}n\right] \\
\leq& b \cdot \sum_{i=1}^{n} \min(H_{n-i}^{n}, 1) - \sqrt{\frac{Kb}{n}} \left[\frac{1}{34}\left(1 - \frac{2}{3}\right)\sum_{i=1}^{n'} H_{n-i}^{n} - \frac{\sqrt{2}}{578}n\right] \\
=& b \cdot \sum_{i=1}^{n} \min(H_{n-i}^{n}, 1) - \sqrt{\frac{Kb}{n}} \left[\frac{1}{102}\sum_{i=1}^{n'} H_{n-i}^{n} - \frac{\sqrt{2}}{578}n\right]. \qquad (90)
\end{aligned}$$

To complete the analysis, we need elementary facts about the harmonic sums H_{n-i}^{n} defined in (88):

Proposition 3.

$$\sum_{i=1}^{n'} H_{n-i}^{n} \leq n - 2n/e + 2; \qquad (91)$$

$$\sum_{i=n'+1}^{n} \min(H_{n-i}^{n}, 1) \leq n/e + 1. \qquad (92)$$

Proof. Since n' was defined to be the largest value such that $H_{n-n}^{n} \leq 1$, it can be checked that $n' = \lfloor n(1-1/e) \rfloor$. For all $i = 1, \ldots, n'$, $\min(H_{n-1}^{n}, 1) = H_{n-1}^{n}$, while for all $i = n'+1, \ldots, n$, $\min(H_{n-1}^{n}, 1) = 1$.

Therefore, the LHS of inequality (92) equals $(n - \lfloor n(1-1/e) \rfloor) \cdot 1$, which is at most $n - (n(1-1/e) - 1) = n/e + 1$, which equals the RHS of inequality (92).

For inequality (91), note that its LHS is at most $\sum_{i=1}^{n'} \ln(n/(n-i))$. In turn,

$$\begin{aligned}
\sum_{i=1}^{n'} \ln \frac{1}{1 - i/n} &\leq \int_{1}^{n'+1} \ln \frac{1}{1 - x/n} dx \\
&\leq \int_{0}^{n(1-1/e)+1} \ln \frac{1}{1 - x/n} dx \\
&= n \int_{0}^{1-1/e+1/n} \ln \frac{1}{1-y} dy
\end{aligned}$$

where the first inequality uses the fact that the function $\ln \frac{1}{1-x/n}$ is increasing over $x \in [1, n'+1]$. The final integral can be evaluated to equal

$$1 - 1/e + 1/n + (1/e - 1/n)\ln(1/e - 1/n)$$

which is at most $1 - 1/e + 1/n + (1/e - 1/n)(-1) = 1 - 2/e + 2/n$ as long as $n \geq 3$. This completes the proof of inequality (91).

□

Electronic copy available at: https://ssrn.com/abstract=3236533

Applying Proposition 3 to expression (90) and using the fact that $b - \sqrt{Kb/n}/102 > 0$, we bound expression (90) from above by

$$\begin{aligned} & bn\left(1-\frac{1}{e}+\frac{3}{n}\right)-\sqrt{\frac{Kb}{n}}\left[\frac{1}{102}\left(1-\frac{2}{e}+\frac{2}{n}\right)n-\frac{\sqrt{2}}{578}n\right] \\ \leq & bn\left(1-\frac{1}{e}\right)+3b-\frac{\sqrt{nKb}}{C}. \end{aligned}$$

$C > 1$ is an absolute constant. As long as $b \leq n$ and K is sufficiently large, the inequality $3b < \sqrt{nKb}/C$ holds. Since $\mathsf{OPT} = bn$ for all realization of π and $\boldsymbol{\ell}$, this completes the proof of Theorem 7. □

Appendix E: Extension to Multiple Reward Rates per Resource

We consider the generalization to the setting where each resource i could be depleted (sold) at varying rates (prices), instead of a single rate r_i, following Ma and Simchi-Levi (2020). This is used for our simulations of assortment optimization on the hotel data set in Section 6.2.

We assume that for each resource i, its set of reward rates $\mathcal{P}_i$ is known in advance. This introduces an aspect of "admission control" to the problem, where sometimes it is desirable to completely reject a customer, who is only willing to purchase a resource at a low price, to reserve resources for higher-paying customers.

We impose additional structure on the mapping from contexts and actions to distributions over outcomes. We assume that each $\mathcal{P}_i$ is finite and that the action set $\mathcal{A}$ is a non-empty downward-closed set of *combinations* (i, P) of resources i and prices $P \in \mathcal{P}_i$. $\mathcal{A}$ can be thought of as the feasible assortments of (resource, price)-combinations that the firm can offer. For example, actions $a \in \mathcal{A}$ can be constrained so that $|\{(j, P) \in a : j = i\}| \leq 1$ for all i, which says that the firm can set at most one price for each resource, or alternatively constrained only in total cardinality, so that the firm can offer the same resource at multiple prices (where presumably additional benefits are attached with the higher price).

We only allow the firm to offer combinations (i, P)'s for which resource i has not ran out. Note that this is in contrast to the model described in Section 2, where actions can be arbitrarily chosen and resources which have ran out are not consumed. Since $\mathcal{A}$ is downward-closed, it always contains the empty assortment $\emptyset$, which the firm can offer if it has ran out of all resources. When the firm offers an assortment a, the outcome is described by a vector $\mathbf{y} \in \{0,1\}^{|\mathcal{P}_1|+\ldots+|\mathcal{P}_n|}$ describing which combinations (i, P) were consumed. Only combinations $(i, P) \in a$ could be consumed, and for each resource i, at most one combination corresponding to i could be consumed.

Assumption 1 **(Substitutability)**. *Consider any context $x \in \mathcal{X}$ and any two actions $a, a' \in \mathcal{A}$ with $a \subseteq a'$. Then for any combination $(i, P) \in a$, we have $\sum_{\mathbf{y}:\mathbf{y}_{(i,P)}=1} \rho_{x,a}(\mathbf{y}) \geq \sum_{\mathbf{y}:\mathbf{y}_{(i,P)}=1} \rho_{x,a'}(\mathbf{y})$.*

Colloquially, Assumption 1 reads that augmenting an assortment (from a to a') can only decrease the chances of selling the combinations already in the assortment. It is a very mild assumption, originating from Golrezaei et al. (2014), which holds under any random-utility choice model.

We still define OPT as the optimal objective value of the LP relaxation:

Primal:

$$\max \sum_{a\in\mathcal{A}} \sum_{t\in[T]} s_{a,t} \sum_{(i,P)\in a} \sum_{\mathbf{y}:\mathbf{y}_{(i,P)}=1} \rho_{x^t,a}(\mathbf{y}) P \tag{93}$$

Electronic copy available at: https://ssrn.com/abstract=3236533

$$\sum_{a\in\mathcal{A}}\sum_{t\in[T]} s_{a,t} \sum_{(i,P)\in a}\sum_{\mathbf{y}:\mathbf{y}_{(i,P)}=1} \rho_{x^t,a}(\mathbf{y}) \le b_i \qquad i\in[n]$$

$$\sum_{a\in\mathcal{A}} s_{a,t} \le 1 \qquad t\in[T]$$

$$s_{a,t} \ge 0 \qquad a\in\mathcal{A}, t\in[T]$$

We modify the IBOL algorithm from Section 3 for the current setting with multiple reward rates. The only change is in the definition of rewards in the auxiliary online learning problem.

In Section 3, at each point in time t, we defined a virtual reward r_i^t for each resource i, based on the fraction N_i^{t-1}/b_i of that resource depleted at that time. Earlier, r_i^t was defined as the product r_i and a *penalty factor* $(1-\Psi(N_i^{t-1}/b_i))$, where $\Psi(\cdot)$ increased from 0 to 1 as the fraction depleted increased from 0 to 1. Now that resource i has multiple reward rates in $\mathcal{P}_i$, the change from Ma and Simchi-Levi (2020) is that we instead subtract a *virtual cost*. Specifically, for each combination (i,P), its virtual reward at time t is defined to be

$$r_{(i,P)}^t = P - \Phi_{\mathcal{P}_i}\left(\frac{N_i^{t-1}}{b_i}\right), \tag{94}$$

where $\Phi_{\mathcal{P}_i}(\cdot)$ increases from 0 to $\max\mathcal{P}_i$ as the fraction of resource i depleted increases from 0 to 1. Note that it is possible for the virtual reward $r_{(i,P)}^t$ to be negative. The definition of $\Phi_{\mathcal{P}_i}(\cdot)$, which is defined in Section 2.1 in (Ma and Simchi-Levi 2020), is rather intricate. For completeness, we provide the definition of $\Phi_{\mathcal{P}_i}(\cdot)$, together with the definition of parameters $\{\alpha_i^{(1)}\}$, in Appendix E.1. Similar to the previous single reward rate setting, we define the discounted reward at time t as

$$R^t(a) = \sum_{(i,P)\in a}\sum_{\mathbf{y}:\mathbf{y}_{(i,P)}=1} \rho_{x^t,a}(\mathbf{y})\left[P - \Phi_{\mathcal{P}_i}\left(\frac{N_i^{t-1}}{b_i}\right)\right],$$

and denote $a_t^* = \operatorname{argmax}_{a\in\mathcal{A}} R^t(a)$.

THEOREM 8. *The total reward* ALG *earned by the algorithm that uses virtual costs (94) satisfies*

$$\mathsf{OPT} \le \frac{(1+b_{min})(1-e^{-1/b_{min}})}{1-\exp(-\min_i \alpha_i^{(1)})}\cdot\mathbb{E}[\mathsf{ALG}] + \mathbb{E}[\mathsf{REG}(\mathcal{F}_T)], \tag{95}$$

where $\mathsf{REG}(\mathcal{F}_T) = \sum_{t\in[T]}(R^t(a_*^t) - R^t(a^t))$.

Compared to Theorem 1, the only change in inequality (95) in Theorem 8 is in the denominator, where the denominator $1-e^{-1}$ in Theorem 1 has been replaced by denominator $\min_i(1-e^{-\alpha_i^{(1)}})$ in Theorem 8. For each resource i, the factor $1-e^{-\alpha_i^{(1)}}$ is the *competitive ratio associated with* price set $\mathcal{P}_i$, and the competitive ratio is equal to $1-1/e$ when $\mathcal{P}_i$ is a singleton.

Proof. We start with the formulation **Dual:**

$$\min \sum_{i\in[n]} b_i\lambda_i + \sum_{t\in[T]}\gamma_t \tag{96}$$

$$\gamma_t \ge \sum_{(i,P)\in a}\sum_{\mathbf{y}:\mathbf{y}_{(i,P)}=1} \rho_{x^t,a}(\mathbf{y})(P-\lambda_i) \qquad a\in\mathcal{A}, t\in[T] \tag{97}$$

$$\lambda_i, \gamma_t \ge 0 \qquad i\in[n], t\in[T]$$

Define dual variables to LP (96) as

$$\Lambda_i = \Phi_{\mathcal{P}_i}\left(\frac{N_i^T}{b_i}\right), \quad \Gamma_t = R^t(a_*^t).$$

Electronic copy available at: https://ssrn.com/abstract=3236533

These dual variables can be readily verified to be feasible for LP (96). Based on strong duality for linear program, we know that

$$\begin{aligned}\mathsf{OPT} &\le \mathbb{E}\left[\sum_{i\in[n]} b_i\Lambda_i + \sum_{t\in[T]}\Gamma_t\right] \\ &= \mathbb{E}\left[\sum_{i\in[n]} b_i\Phi_{\mathcal{P}_i}\left(\frac{N_i^T}{b_i}\right) + \sum_{t\in[T]} R^t(a^t) + \sum_{t\in[T]}(R^t(a_*^t) - R^t(a^t))\right] \\ &= \mathbb{E}\left[\sum_{i\in[n]} b_i\Phi_{\mathcal{P}_i}\left(\frac{N_i^T}{b_i}\right) + \sum_{t\in[T]} R^t(a^t)] + \mathbb{E}[\mathsf{REG}(\mathcal{F}_T)]\right]. \end{aligned} \tag{98}$$

The following is shown in Ma and Simchi-Levi (2020):

$$\mathbb{E}\left[\sum_{i\in[n]} b_i\Phi_{\mathcal{P}_i}\left(\frac{N_i^T}{b_i}\right) + \sum_{t\in[T]} R^t(a^t)\right] \le \frac{(1+b_{\min})(1-e^{-1/b_{\min}})}{1-\exp(-\min_i \alpha_i^{(1)})}\cdot\mathbb{E}[\mathsf{ALG}],$$

which completes the proof after combining with equation (98) and rearranging. □

E.1. Definition of $\alpha_i^{(1)}, \Phi_{\mathcal{P}_i}$

For a set of $\mathcal{P}$, consisting of m discrete prices $0 < r^{(1)} < \cdots < r^{(m)}$, the function $\Phi_{\mathcal{P}}$ is defined as follows. To define $\Phi_{\mathcal{P}}$, we first need to define m constants $\alpha^{(1)}, \alpha^{(2)}, \ldots, \alpha^{(m)}$, which constitute a unique set of positive real numbers that satisfies the following set of equations:

$$\begin{aligned} &\alpha^{(i)} > 0 \quad \text{for all } 1 \le i \le m \\ &\sum_{i=1}^{m} \alpha^{(i)} = 1 \\ &1 - e^{-\alpha^{(1)}} = \frac{1}{1-r^{(1)}/r^{(2)}}\cdot(1-e^{-\alpha^{(2)}}) = \ldots = \frac{1}{1-r^{(m-1)}/r^{(m)}}\cdot(1-e^{-\alpha^{(m)}}) \end{aligned}$$

By (Ma and Simchi-Levi 2020), the above set of equations has a unique solution. To define the function $\Phi_{\mathcal{P}}$, we still need to define one sets of parameters and a function:

- $L^{(0)} = 0$, and $L^{(j)} = \sum_{j'=1}^{j} \alpha^{(j')}$, and in particular $L^{(m)} = 1$.
- $\ell(\cdot)$: a function on $[0,1]$, where $\ell(w)$ is the unique $j \in [m]$ for which $w \in [L^{(j-1)}, L^{(j)})$.

The function $\Phi_{\mathcal{P}}$ for price set $\mathcal{P}$ is then defined over $w \in [0,1]$ by:

$$\Phi_{\mathcal{P}}(w) = r^{(\ell(w)-1)} + (r^{(\ell(w))} - r^{(\ell(w)-1)})\frac{\exp\left[w - L^{\ell(w)-1}\right] - 1}{\exp\left[\alpha^{(\ell)}\right] - 1}.$$

Finally, we can apply the above definition on each $\mathcal{P} = \mathcal{P}_i$, which yields the parameter $\alpha_i^{(1)} = \alpha^{(1)}$ for the Theorem.

Appendix F: Supplementary Details about Numerical Experiments

We provide additional details about our choice estimation from Section 6.2. We define 8 customer types, one for each combination of the 3 following binary features.

1. Group: whether the customer indicated a party size greater than 1.

2. CRO: whether the customer booked using the Central Reservation Office, as opposed to the hotel's website or a Global Distribution System (for details on these terms, see (Bodea et al. 2009)).

Electronic copy available at: https://ssrn.com/abstract=3236533

3. VIP: whether the customer had any kind of VIP status.

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

We estimate the mean MNL utilities for each of the 8 products separately for each customer type. The results are displayed in Table 3 in (Ma and Simchi-Levi 2020) (Page 52). The total share of each customer type (out of all the transactions) is also displayed in that table. We should point out that it is possible for a customer to choose the higher fare for a room, even if the lower fare was also offered. This is because the higher fares are often packaged with additional offers, such as airline services, city attractions, in-room services, etc. We have shifted the mean utilities so that for each customer type, the weights of both the no-purchase option, and the most-preferred purchase option, is equal to 0. (We synthetically set the weight of the no-purchase option because it is not possible to estimate from the data.) The large weights on the no-purchase options ensure that the revenue-maximizing assortments tend to include both the low and high fares.

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

Electronic copy available at: https://ssrn.com/abstract=3236533