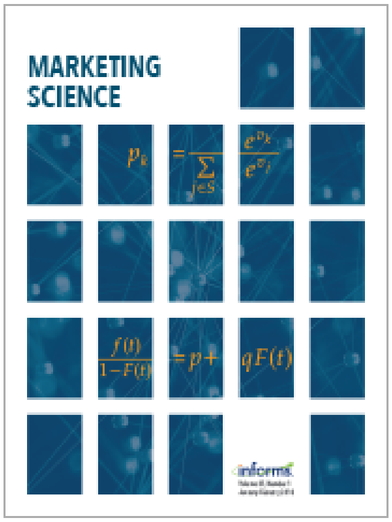


Marketing Science

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

Dynamic Online Pricing with Incomplete Information Using Multiarmed Bandit Experiments

Kanishka Misra, Eric M. Schwartz, Jacob Abernethy

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

MARKETING SCIENCE
Vol. 38, No. 2, March–April 2019, pp. 226–252
ISSN 0732-2399 (print), ISSN 1526-548X (online)

Dynamic Online Pricing with Incomplete Information Using Multiarmed Bandit Experiments

Kanishka Misra,[a] Eric M. Schwartz,[b] Jacob Abernethy[c]

[a] Rady School of Management, University of California, San Diego, La Jolla, California 92093; [b] Ross School of Business, University of Michigan, Ann Arbor, Michigan 48109; [c] School of Computer Science, College of Computing, Georgia Institute of Technologyy, Atlanta, Georgia 30332

Contact: kamisra@ucsd.edu, http://orcid.org/0000-0002-0106-1230 (KM); ericmsch@umich.edu (EMS); prof@gatech.edu, http://orcid.org/0000-0002-3115-6804 (JA)

Received: June 4, 2017
Revised: February 13, 2018; May 6, 2018
Accepted: May 25, 2018
Published Online in Articles in Advance: March 29, 2019

https://doi.org/10.1287/mksc.2018.1129

Abstract. Pricing managers at online retailers face a unique challenge. They must decide on real-time prices for a large number of products with incomplete demand information. The manager runs price experiments to learn about each product's demand curve and the profit-maximizing price. In practice, balanced field price experiments can create high opportunity costs, because a large number of customers are presented with suboptimal prices. In this paper, we propose an alternative dynamic price experimentation policy. The proposed approach extends multiarmed bandit (MAB) algorithms from statistical machine learning to include microeconomic choice theory. Our automated pricing policy solves this MAB problem using a scalable distribution-free algorithm. We prove analytically that our method is asymptotically optimal for any weakly downward sloping demand curve. In a series of Monte Carlo simulations, we show that the proposed approach performs favorably compared with balanced field experiments and standard methods in dynamic pricing from computer science. In a calibrated simulation based on an existing pricing field experiment, we find that our algorithm can increase profits by 43% during the month of testing and 4% annually.

History: K. Sudhir served as the editor-in-chief and Olivier Toubia served as associate editor for this article.

Supplemental Material: Data files and the online appendix are available at https://doi.org/10.1287/mksc.2018.1129.

Keywords: dynamic pricing • e-commerce • online experiments • machine learning • multiarmed bandits • partial identification • minimax regret • nonparametric econometrics • A/B testing • field experiments

1. Introduction

1.1. Overview

Consider a pricing decision for a manager at an online retailer. Two features of this setting make it different from a brick-and-mortar retailer. First, online sellers can vary prices nearly continuously and often randomize those changes for learning purposes (Furman and Simcoe 2015). This differs from a traditional retail setting where retailers face high costs, known as menu costs, to make price changes (Anderson et al. 2015). Second, large online retailers sell a far larger number of products than brick-and-mortar retailers (e.g., Amazon.com sells hundreds of millions of products with thousands of new products per day).[1] At the time of pricing, the manager is unlikely to have complete information about each product's demand curve. In these markets, a manager must consider an automated pricing policy (Baker et al. 2014) to set real-time retail prices despite incomplete demand information.

Consider how a firm typically tests prices when faced with incomplete demand information. For example, a firm is deciding among a set of 10 prices ($p \in \{\$0.10, \$0.20, \ldots, \$0.90, \$1.00\}$), but it does not have information about demand at each price. The firm may experiment with the prices one at a time, observe demand and profits at each level, and then select the price that leads to highest total profits. This approach is intuitive, often implemented in industry, and is a benchmark in the academic operations research literature (Besbes and Zeevi 2009). This form of price experimentation is described as "learn, then earn." In our example, the firm runs a *balanced experiment*, where the same number of consumers are shown each of the 10 prices. The experiment estimates the mean (with measurement error) demand $(D(p = \$0.10), \ldots, D(p = \$1.00))$ and profit at each price $(\pi(p = \$0.10), \ldots, \pi(p = \$1.00))$. The manager can then select the price with the highest profits.

In this paper, we build on the dynamic price experimentation literature. Instead of considering the two distinct phases of learning and earning, this literature frames the problem as a dynamic optimization problem with the goal of maximizing *earning while learning*. The objective of continuous price experimentation is to maximize long-run profits, where the firm wants to minimize the opportunity costs of experimentation.

The pricing algorithm sequentially sets prices to balance currently earning profits and learning about demand with future profits. The key difference here is that the firm observes real-time purchase decisions and incorporates this additional information for future pricing decisions. Therefore, to select the price for the tth round of the experiment, the firm will use profit estimates from the first $t-1$ rounds.

Multiarmed bandit (MAB) methods provide algorithms for adaptive experimentation. The MAB problem is a fundamental dynamic optimization problem in reinforcement learning (Sutton and Barto 1998). The problem has a long history in statistics and operations researchand more recently in marketing (Gittins et al. 2011); see also Schwartz et al. (2017) for an overview of MAB in the marketing literature). The algorithm is faced with a set of possible decisions, called "arms" (in our application, arms represent different prices). Each arm has a stable but unknown reward distribution (in our application, profit). In this setup, the algorithm selects arms sequentially in real time with the goal of optimizing cumulative reward.

We illustrate the difference between a MAB experiment and a balanced experiment for the example above. In the computer science literature, Auer (2002) propose the UCB1, an optimal nonparametric bandit algorithm that continuously balances learning and earning over time (we will build on this algorithm in our paper, and it is described in detail in Section 2). For our illustration, we assume a demand curve (here, $D(p) = 3{*}(1-p)$, with the profit-maximizing price at \$0.5) and run both algorithms over 1,000 experimental periods. In any period, the manager gets a noisy signal of the true profit with variance of 0.3. The results are shown in Figure 1. In panel (a), we show the prices played (by round and a histogram across all rounds) in each method. In a balanced experiment, by definition, each price is played an equal number of times. The bandit experiment, on the other hand, learns the profit-maximizing price and charges that price more often.

As a result (Figure 1, panel (b)), the bandit algorithm earns more profit than the balanced experiment (95% versus 66%). Moreover, the bandit continuously increases the profit achieved with experiments, whereas the balanced experiment is, by definition, constant over time.

A second important difference is the precision of the learning. Figure 1, panel (b) (right side) also illustrates that the confidence intervals for the balanced experiment are the same for all prices. However, the bandit experiment results in smaller confidence intervals for prices around the optimal price and much larger ones for prices that are suboptimal. In terms of profit learning, a balanced experiment focuses on *learning everywhere*, whereas the bandit focuses on *relevant learning*. Relevant learning (Aghion et al. 1991) refers to learning the true profit for the most profit-relevant price points. We use this example to illustrate the value of relevant learning and the value gained from MAB algorithms to optimize experiments, as we will use this MAB framework throughout the paper.

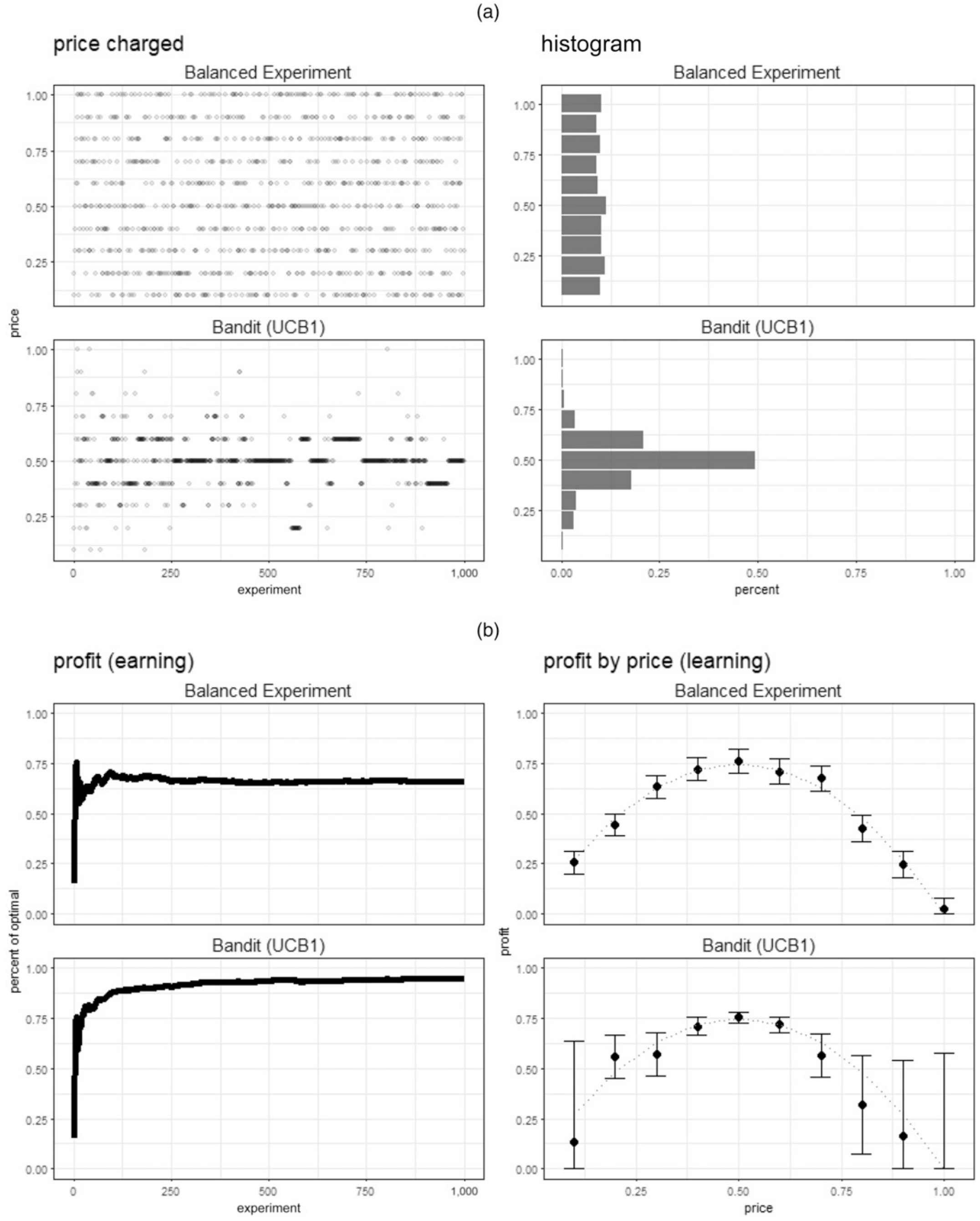
Our proposed solution extends common MAB algorithms to include choice theory from microeconomics. We account for the two key features of the online retail pricing problem setting: a wide variety of products and real-time changes for those millions of products. To ensure that our model can be used for a wide variety of products, we make minimal assumptions about the underlying demand curve for a particular product. Instead of assuming each product's demand curve to come from the same family of distributions, we opt for an approach that is driven by economic theory and flexible. By limiting parametric assumptions, our algorithm will be more robust to any arbitrary unknown downward-sloping demand system. Although traditional robust optimization solutions come with a relatively large computational cost (e.g., Berger 1985 note on p. 309 that the "minimax principle can be devilishly hard to implement"), our algorithm can be estimated in real time.

Our algorithm builds on the upper confidence bound (UCB) algorithm in the nonparametric MAB literature (Auer et al. 2002). The family of UCB algorithms works as follows. In each time period, the algorithm assigns each arm a so-called UCB value: the sum of the expected reward and an exploration bonus. That exploration bonus is the potential value from experimentation. Then the algorithm plays the arm with the highest UCB value. The algorithm observes a noisy reward and updates these values for each arm. The UCB algorithm is guaranteed to be the "best" nonparametric algorithm for any bounded payoff function in terms of achieving the theoretically optimal error rate in a finite-time setting (Lai 1987).

We extend this MAB algorithm to incorporate partially identified demand learning. In the typical UCB algorithm, when a particular price is charged (an arm is played), the firm's observations are limited to profits (reward) from that price (arm). We extend this to allow learning across prices (arms) based on economics. Formally, we assume that a consumer's choice structure is such that individual demand curves are weakly downward sloping. With this additional yet minimal assumption, when a consumer is exposed to any price, the manager can *partially identify* the consumer's underlying preference across different potential prices. For example, if a consumer purchases a good at \$3, the manager can infer she would have purchased at any price below \$3. And if a consumer does not purchase a good at \$3, the manager can infer she would not have purchased at any price above \$3.

We consider a setting where each consumer makes a single purchase decision, so we rely on cross-sectional identification to estimate aggregate demand. We assume

Figure 1. Balanced Field Experiment (Left) vs. Multiarmed Bandit Experiment (Right)



Notes. The bandit experiment is adaptive. This particular algorithm is UCB-tuned (Auer 2002). (a) Prices played: the left hand side plots show the price charged over time and the right hand side charts show the distribution of all prices charged; a balanced experiment (top) charges all prices equally while the bandit (bottom) learns in real time and charges the truly optimal price ($0.50) more often. (b) Profits learning and earning: the left hand side plots show the profits over time (earning); after 1,000 prices tested, the bandit earns 95% of optimal profits, while the balanced experiment earns only 66% of optimal profits; the right hand side chart shows the mean and 95% confidence intervals of the learned profit at each price; the dotted line represents the true profit curve; the balanced experiment learns the profit curve with the same precision at all prices, while the bandit learns the true profit with small confidence intervals around the optimal price ($0.5) and large confidence intervals at sub optimal prices.

the firm observes consumer characteristics and can use this to observe heterogeneity for segmentation. In practice, these segments may be based on many variables, including demographics and behavior. Because of the abundance of data, online retailers can have a large number of segments. For example, Google Analytics and Facebook Analytics offer over 1,000 segments to advertising brands.[2] We estimate the within-segment variation in customer valuations and allow the heterogeneity across segments to be fully nonparametric; that is, information about preferences in each segment are independent.

Because we impose minimal assumptions about the shape or structure of the demand model, our algorithm is quite robust to distributional assumptions about heterogeneity and context (Handel et al. 2013), and therefore it can be credibly applied for a variety of products. We will show that this includes situations where demand and profit functions are discontinuous or multimodal. It even allows for special price effects (e.g., contextual factors). This robustness to the structure of the demand model is important, as our pricing model can be used to estimate prices for a variety of products.

Finally, the other key feature of online retail is pricing in real time at large scale. We ensure that our algorithm can run in real time for millions of products. Because our proposed algorithm has minimal estimation requirements, it plays prices at speeds that are orders of magnitude faster than current solutions. In a two-period version of a pricing problem, the full solution in Handel and Misra (2015) plays first-period prices for one product in about 15 hours of computation time. Our proposed method can calculate about 2 million prices per minute, and it can be used for real-time online pricing.

We consider a monopolist's pricing, which is consistent with much of the literature on price experimentation—for example, the literature on MAB (Kleinberg and Leighton 2014), pricing under ambiguity (Besbes and Zeevi 2009, Bergemann and Schlag 2011), and field experiments (Dubé and Misra 2017). Our algorithm is therefore most relevant for products with limited competition.[3] This includes products with a monopolist seller and products with limited distribution early or late in their life cycles. The algorithm is also relevant when faced with a stable or predictable competitor. A stable competitor will not to respond to real-time price changes. A predictable competitor, for example, may use an automated real-time price-matching algorithm. With "nonstrategic" competition (see Hviid and Shaffer 1999 for a review of price matching), the product's demand curve is stable over time. We note that our algorithm and theoretical results are not guaranteed for products with strategic competition.

Our main results are as follows:

1. We analytically prove our proposed algorithm's performance using a novel analysis technique for MAB algorithms. Here, we show that our algorithm is guaranteed to be asymptotically optimal (see Section 3).

2. In a series of Monte Carlo simulation experiments across a range of settings, we outperform a set of benchmarks. We find our algorithm achieves a higher mean profit and a lower variance across simulation settings. We document the limiting condition (no observable heterogeneity) under which our proposed algorithm matches current machine learning methods (see Section 4).

3. We conduct a simulation experiment based on a randomized pricing experiment in Dubé and Misra (2017), run with ZipRecruiter.com. Approximately 8,000 consumers were shown one of 10 prices between $19 and $399. We use those experimental results as a basis for simulating demand curves. We find that our algorithm achieves 43% higher profits during the month of testing and about 4% higher annual profits (see Section 5).

1.2. Contributions

With the emergence of big data, we see an increase in machine learning applications in marketing (Chintagunta et al. 2016). But a natural critique is machine learning algorithms' absence of economic theory. This work illustrates how we can bridge this gap. We propose a novel combination of economic theory with machine learning. To marketing and economics, we adapt scalable reinforcement learning methods to address dynamic optimization problems. We propose a fast dynamic pricing algorithm rooted in economic theory.

To machine learning, we introduce distribution-free theory of demand to improve existing algorithms theoretically and empirically. Typically, the models of active learning in computer science often rely on stylized demand models, because they are amenable to formal analysis. But they may lack the economic theory, which, as we find, can improve the optimal pricing algorithms.

1.3. Related Literature

1.3.1. Literature on Pricing.

Our paper adds to a large literature on pricing and learning demand in marketing, economics, and operations research. Much of the current literature makes strong assumptions about the information that a firm has about demand for each product. In marketing and operations, for instance, the literature often assumes that firms make product pricing decisions based on knowing the demand curve (Oren et al. 1982, Rao and Bass 1985, Smith 1986, Wernerfelt 1986, Bayus 1992, Rajan et al. 1992, Acquisti and Varian 2005, Nair 2007, Akçay et al. 2010, Jiang et al. 2011). These methods assume that the firm has access to perfect information about the demand curve and considers the optimal dynamic pricing given this information. We argue that it is not feasible for large online retailers to know the demand curves for millions of products.

A second related literature assumes that firms know demand only up to a parameter (Rothschild 1974, Lodish 1980, Aghion et al. 1991, Braden and Oren 1994, Kalyanam 1996, Bergemann and Valimaki 2000, Aviv and Pazcal 2002, Biyalogorsky and Gerstner 2004, Hitsch 2006, Desai et al. 2010, Bonatti 2011, Biyalogorsky and Koenigsberg 2014). The modeling approach in these papers assumes that the manager knows the structure of demand and just learns the parameters. This could be a two-period model (Biyalogorsky and Koenigsberg 2014) or an infinite-time model (Aghion et al. 1991). In the infinite-time model, Aghion et al. (1991) consider a very general model where the manager knows the structure of demand up to a parameter (θ), and the firm sets prices and observes market outcomes. In subsequent periods, the firm updates the posterior belief distributions for the parameters, and then the firm sets prices. They show that under this structure, learning can be "inadequate" in cases where the profit function is multimodal or discontinuous. Inadequate learning is defined as when the agent never acquires adequate knowledge (i.e., asymptotically, adequate with probability zero). Adequate knowledge is defined when the agent knows enough about the true profit function to achieve ex post optimal profits. The Aghion et al. (1991) paper concludes: "Even when learning goes on forever, it does not result in adequate knowledge" (p. 642).

We argue that it is important for a robust pricing policy to incorporate all possible demand curves. Therefore, we make weaker assumptions than assuming a demand model to derive optimal prices. By making only weak assumptions about the demand curve, we sacrifice precision for credibility (Manski 2005). This is consistent with saying that it is not feasible for a firm to have both precise and credible demand information about every single product, so we have to make a tradeoff between precision and credibility.

Our nonparametric method builds on the robust pricing literature (Bergemann and Schlag 2008, 2011; Handel et al. 2013). The robust dynamic pricing literature provides a solution for two-period pricing (Handel and Misra 2015). Here, the authors consider a brick-and-mortar retail setting, where the retailer must keep a fixed price for the entire time period. We differ from these papers by building a solution based on the MAB literature in computer science. The advantage of our model is that it allows for continuous learning and real-time changes to suggested prices. The key simplifying assumption here is that we do not account for fully endogenous demand curve learning. In our model, we account for the fact that observed outcomes of all prior price experiments impact expected demand for all price points (through our demand estimation). However, when we estimate the value of experimentation at any price, we consider only the benefit from reducing uncertainty about profits at that price (e.g., an experiment at price = $3 will reduce the future uncertainty about profits at price = $3). A fully forward-looking experimenter (endogenous demand curve learning) would also consider the benefit from reducing uncertainty about profits at other prices (e.g., an experiment at price = $3 could reduce the future uncertainty about profits at price = $4). We believe endogenous demand curve learning is more relevant in settings with infrequent price changes. The advantage of not including endogenous learning is analytical tractability. As a result, whereas Handel and Misra (2015) can only consider a two-time period model, our model can generalize to longer-run experiments (e.g., we consider 200,000 experimental rounds in Section 4).

The current work also contributes to theoretical work in operations research—namely, dynamic pricing and revenue management. The area of revenue management has a large literature that considers dynamic pricing (see Elmaghraby and Keskinocak 2003 and den Boer 2015 for reviews of the literature). Our work falls into a category known as dynamic pricing without inventory constraints, where dynamics are due to incomplete information and learning. Within this literature, our paper fits with the less studied and more recent stream of nonparametric work. Nonparametric approaches (Besbes and Zeevi 2009) consider pricing policies in an incomplete information setting. Here, the authors consider algorithms that minimize the maximum ex post statistical regret from not charging the optimal static price. The proposed algorithm divides the sales horizon into an "exploration" phase during which the demand function is learned and an "exploitation" phase during which the estimated optimal price is used. The firm has to ex ante set the length of experimentation stage; therefore this algorithm corresponds to a balanced field experiment. In more recent additions to this literature, Wang and Hu (2014) and Lei et al. (2014) improve the convergence results, yet the algorithms proposed in these papers also consider distinct phases for exploration then exploitation, or, as we refer to it, "learning *then* earning." Instead, in our paper, we consider the learning and earning phases simultaneously, accounting for the potential value from learning at each point in time. This is consistent with the broader MAB literature from machine learning.

1.3.2. Literature on Multiarmed Bandits.

We consider the problem of pricing using MAB methods, which are not typically used for pricing but do stretch across computer science, statistics, operations research, and marketing. The MAB problem is the quintessential problem of the fields of active learning, referring to the sequential decision-making process, and reinforcement learning in the computer science literature (for an overview, see Sutton and Barto 1998).

A large part of this literature provides theoretical analysis and mathematical guarantees of algorithms. The algorithms are policies to adaptively select the arms to achieve the best profits. The objective function for these algorithms is to minimize the statistical regret. *Statistical regret* "is the loss due to the fact that the globally optimal policy is not followed all the times" (Auer et al. 2002, p. 235). That is the difference between the achieved profits and the ex post optimal profits, if the decision maker knew the true average profits for each arm. Algorithms are compared based on the bounds on regret. This bound represents the worst-case performance: the maximum possible regret for any possible distribution of the true rewards across the arms.

These UCB policies provide the backbone of a stream of MAB solutions in reinforcement learning coming from statistical machine learning (Agrawal 1955, Auer et al. 2002). Lai and Robbins (1985) first obtained these "nearly optimal index rules in the finite-horizon case" (Brezzi and Lai 2002, p. 89) where the indices can be interpreted as "upper confidence bounds for the expected rewards"—hence UCB (Brezzi and Lai 2002, p. 89). Although these index rules do not provide the exactly optimal solution to the optimization problem with a discounted infinite sum of expected rewards, these rules are asymptotically optimal for arbitrarily large finite-time horizons, T. As $T \to \infty$, the UCB-based index rule achieves optimal performance with respect to maximizing the expected sum of rewards through T periods "from both the Bayesian and frequentist viewpoints for moderate and small values of [T]" (Brezzi and Lai 2002, p. 89).[4]

Asymptotic theory links the finite-horizon undiscounted case and the infinite-horizon discounted multiarmed bandit problem (Brezzi and Lai 2002). Depending on the discount factor, *disc*, the UCB directly approximates the Gittins index (Lai 1987). When setting $T = (1 - disc)^{-1}$, as $disc \to 1$, then the UCB in Lai (1987) (p. 1113) is asymptotically optimal not only for the finite-horizon undiscounted problem but also for the infinite-horizon undiscounted problem from Gittins (1989). The link is strengthened as Brezzi and Lai (2002) derive an approximate Gittins index, with a structure exactly the same as that of the UCB, the sum of expected reward and an exploration bonus.

We add to this literature by allowing (partially identified) demand curve learning across the potential prices. Formally, we assume that each consumer's choice structure satisfies the weak axiom of revealed preference (WARP) (see proposition 3.D.3 in Mas-Colell et al. (1995) and discussed in the next section). Although in the UCB models the prices would typically be considered independent arms, adding an economic assumption allows us to use information about demand at a price (say, p_1) to make an inference about demand at other prices ($p \neq p_1$).

The learning-and-earning problem for pricing relates to a broader class of problems, optimizing marketing experiments or so-called A/B testing. The emerging framework is using MAB methods to optimally balance earning while learning. These approaches most commonly appear in online advertising or website design (Hauser et al. 2009, Urban et al. 2014, Schwartz et al. 2017). We argue that pricing is different from other marketing decisions in two key ways. First, economic theory gives strong predictions that individual indirect utility functions are nonincreasing in prices, but this is not true for other marketing decision variables. Without particular parametric assumptions, learning about one ad creative or website design does not inform predictions of others in predictable ways. Second, unlike advertising, the randomization of prices is imperfect. Retailers do not commonly offer the same product to different consumers at a given point in time.[5] Therefore, price changes can happen only across time and not across consumers.

2. Dynamic Multiperiod Monopoly Pricing

2.1. Model Setup and Maintained Assumptions

In this section, we state the main assumptions in our analysis. We first discuss our assumptions on the demand side and then our assumptions on the supply side.

We assume there are a large set of potential consumers with unit demand for each product. For each consumer we assume the following:

1. she has stable preferences,
2. she has a stable budget over time,
3. she faces a stable outside option, and
4. her choice structure satisfies the WARP.

We note that the first three of these assumptions, although typically unstated, are assumed in any field experiment; these ensure that the results of the field experiment can be used to understand demand after the experiment. With these assumptions, we represent the consumer's preference as v_i. In any purchase occasion, when facing a price p, her indirect utility can be written as $u_i = v_i - p$, and she will purchase the good if and only if $u_i \geq 0$; that is, $v_i \geq p$. The assumption of stable preference guarantees that v_i does not change over time. This rules out learning (Erdem and Keane 1996), stockpiling (Hendel and Nevo 2006), network externalities (Nair et al. 2004), reference price effect (Winer 1986, Kalyanaram and Winer 1995), and strategic consumers (Nair 2007). Unlike much of the prior work on dynamic prices (e.g., Nair 2007), in our paper, firms change prices for learning the demand curve as opposed to intertemporal price discrimination.

Our next set of assumptions considers the heterogeneity across consumers. We assume that the manager has access to demand-relevant descriptive variables for each consumer. These could include observable variables, such as demographics and behavioral patterns, or model-based criteria. As an example, in their application, Dubé and Misra (2017) consider geography, company type, and job benefits as relevant demand shifters in their pricing for ZipRecuriter.com. We assume heterogeneity among consumers can be separated as observable (by descriptive variables Z) and unobservable heterogeneity. Formally, $v_i = f(Z_i) + \nu_i$.

To allow for any functional form of the $f(Z_i)$, we assume that the firm assigns consumers into segments S, where each $s \in S$ represents a combination of the descriptor variables Z_i. We define the aggregate proportion of consumers in each segment, ψ_s. We note that the online retailers have the ability to use a large number of segments; for example, Google and Facebook offer more than 1,000 different segments to its advertisers (see Endnote 2).

With this formulation, we can add a fixed effect for each segment. Formally, for a consumer i in segment s, we have $v_i = v_s + \nu_i$, where v_s represents the midpoint of range of consumer valuations within segment s and ν_i represents the unobserved heterogeneity. Rather than assume a function form for ν_i, we assume it can be bounded by δ (or $\nu_i \in [-\delta, \delta]$).[6] Handel et al. (2013) show that this formulation sacrifices precision for credibility, in the sense that it is robust to violations to standard distributional and independence assumptions. With this assumption, the preferences of all consumers in a segment are within δ of the segment midpoint, v_s. Taken together,

$$v_i \in [v_s - \delta, v_s + \delta] \forall i \in s. \tag{1}$$

These demand assumptions are quite general. Within each segment, we allow for any distribution of preferences within this range; therefore, we note that v_s is not assumed to be the mean or the median of the segment valuations. Across segments, we allow for fully nonparametric heterogeneity across segments. This assumption allows cross-sectional learning of consumer preferences.

We will estimate v_s and δ in our empirical algorithm. Our estimate of δ can be interpreted as a measure of the "quality of segmentation." If the firm's ex ante segmentation does not group consumers with similar preferences, then the estimate of δ will be large. But if the firm's ex ante segmentation does group consumers with similar preferences, then δ will be small.

On the supply side, we assume that the firm is a monopolist that sets online prices to maximize profits for a constant marginal cost product. The main deviation we make from the standard pricing literature (e.g., Oren et al. 1982 and Rao and Bass 1985) is that we assume the firm does not know consumer valuations. We assume that the only information available to the firm at the time of initial pricing is that consumer valuations are between $[v_L, v_H]$. The interpretation here is that if the product is sold for v_L (can be zero), all consumers will purchase for sure, and if the product is sold for v_H, no consumers will buy. Consistent with the robust pricing literature (Bergemann and Schlag 2008, 2011; Besbes and Zeevi 2009; Handel et al. 2013; Lei et al. 2014; Wang and Hu 2014; Handel and Misra 2015), we assume within this range that the firm does not know the distribution of consumer preferences across or within segments. Our motivation for this assumption is that it is infeasible for the manager to have credible priors for millions of products.

We assume that the firm does not price discriminate across consumers. Formally, the firm observes a consumer's identity and segment membership; however, we assume the firm does not use this information to price discriminate. If we had full information, there exists an optimal static price that a monopolist would charge. However, because of the lack of information, the monopolist must experiment with prices. We assume that the firm can change prices quickly. Furman and Simcoe (2015) report that Amazon.com can change prices within 15 minutes. In our model, we will assume that prices can change after every N consumers who visit the product.[7]

2.2. Overview of Multiarmed Bandit Pricing

We begin by formulating the pricing problem as a dynamic optimization problem. We assume that there exists a finite set of K prices that the firm can choose from, $p \in \{p_1, \ldots, p_K\}$. For any price, p, the firm faces an unknown true demand $D(p)$. We assume a constant marginal cost (set to zero for ease of exposition). Given this setup, the true profit is given by $\pi(p) = pD(p)$.

Although the true profit function $\pi(p)$ is unknown, the firm observes realizations of profits for each price p_k. Suppose by time t, the firm has charged p_k a total of n_{kt} times. Let $\pi_{k,1}, \pi_{k,2}, \ldots, \pi_{k,n_{kt}}$ be realizations of profit per consumer from every time that price p_k has been charged. We assume that these are drawn from an unknown probability distribution with a mean at the true profit $\pi(p_k)$. We refer to the sample mean at time t as $\bar{\pi}_{kt} = 1/n_{kt} \sum_{\tau=1}^{n_{kt}} \pi_{k\tau}$. By definition, we must have $\sum_{k=1}^{K} n_{kt} = t$.

A pricing *policy or algorithm*, Ψ, selects prices based on the history of past prices and earned profits. From a mathematical perspective, this can be described as, $p_t = \Psi(\{p_\tau, \pi_\tau | \ \tau = 1, \ldots, t-1\})$. The policy maps data from all previous experiments onto price.

To evaluate a policy's performance, the literature considers statistical *regret*.[8] The key criterion to evaluate policies is minimizing maximum regret (i.e., minimax regret). Regret for a policy is defined as the expected profit loss as a result of not always playing the unknown ex post optimal profit-maximizing fixed price (Lai and Robbins 1985). The notion of regret is standard in the computer science and decision theory literature (Berger 1985, Lai and Robbins 1985, Auer 2002). This was first proposed by Wald (1950) and has been axiomatized in the economic literature (Milnor 1954, Stoye 2011). This criterion has been used to study pricing in economics (Bergemann and Schlag 2008, 2011; Handel et al. 2013) and marketing (Handel and Misra 2015). The economic interpretation of regret is the "forgone profits" as a result of price experimentation.

Formally, we represent regret as the distance to the optimal profits. We define the ex post profit-maximizing price to be p^* for all t, yielding an expected profit

$\mu^* = \mathbb{E}[\pi(p)] = p^* D(p^*)$ each time period. The regret of a policy Ψ through time t is

$$\begin{aligned}\text{Regret}(\Psi, \{\pi(\text{p}_\text{k})\}, t) &= \mathbb{E}\left[\sum_{\tau=1}^{\text{t}} \pi^* - \pi_{\text{p}_\tau}\right] \\ &= \sum_{\tau=1}^{t} (\pi^* - \pi(\text{p}_\tau)) \\ &= \pi^* t - \sum_{k=1}^{K} \pi(p_k)\mathbb{E}[n_{kt}], \qquad (2)\end{aligned}$$

where π_{p_τ} is the profit realized when the price p_τ was charged in time period τ, and then $\mathbb{E}[\pi_{p_\tau}] = \pi(p_\tau)$ is expected profit for that price. We note that regret can be simply stated in terms of unknown true average profits, $\{\pi(p_k)\} = \{\pi(p_1), \ldots, \pi(p_K)\}$, and the expected number of times each price is charged, $\mathbb{E}[n_{kt}]$. A summary of the notation is provided in Table 1.

But when considering the analysis of regret, we do not observe true profits, $\{\pi(p_1), \ldots, \pi(p_K)\}$, and therefore, we also do not know truly optimal profit, π^*. The analysis instead considers all possible realizations of profit, because the distribution of profits is not known before running the algorithm. Next we consider the possible realization that generates the "worst case" or maximum regret for a given policy, Ψ. The economic interpretation of this is to consider a feasible demand curve, $D(p)$, that results in the maximum regret given a pricing policy. The algorithm with the best *minimax regret* is one that can minimize the maximum regret.

Table 1. Summary of Notation

Notation	Description
p	Price, with $p \in \{p_1, \ldots, p_K\}$
$D(p)$	Demand at price p
$\pi(p)$	Profit at price p (i.e., $\pi(p) = pD(p)$))
v_{is}	Valuation of individual consumer i in segment s
v_s	Midpoint of valuations in segment s, with its estimated value $\hat{v}_{st}$ after t rounds
δ	Estimated range of valuations across with $v_{is} \in [v_s - \delta, v_s + \delta_s]$, $\forall s$
$\hat{\delta_{st}}$	Lowest value of δ that can rationalize all data for segment s after t rounds
$\hat{\delta}_t$	Estimated range of valuations across all segments at time t
Ψ	Policy for dynamic pricing (i.e., data-driven decision rule $p_{t+1} = \Psi(\{p_1, \pi_{p_1}, \ldots, p_t, \pi_{p_t}\}))$
n_{kt}	Number of times price p_k was tested through time t
n_{st}	Number of times any price was tested with segment s through time t
ψ_s	Percentage of consumers in segment s
$LB_t(x)$, $U, B_t(x)$	Estimated lower and upper bounds through t of a parameter x (e.g., either v_{is}, $D(p)$, or $\pi(p)$)
$H_t(x)$	Partially identified set through t of a parameter x (i.e., $H_t(x) \equiv [LB_t(x), U, B_t(x)]$)
UCB	Upper confidence bound, with its original implementations **UCB1**
UCB-PI	Upper confidence bound with partial identification (proposed in this paper)

In the MAB literature, according to the minimax regret criterion, the optimal solution for this nonparametric problem is a policy involving an index rule scoring each action with its UCB of expected rewards (Agrawal 1955, Lai and Robbins 1985, Lai 1987, Auer 2002). This policy is proven to be the asymptotically best possible performance in terms of achieving the lowest maximum regret.

The structure of the index assigned to each action in the UCB algorithm is the sum of expected reward and an exploration bonus. For instance, in the focal algorithm in Auer (2002), UCB1, the index for action k at time T, is based on only the sample mean reward of the arm and an exploration bonus. In our notation, this translates to

$$\text{UCB1}_{kt} = \bar{\pi}_{kt} + \sqrt{\frac{2\log t}{n_{kt}}}. \qquad (3)$$

Then the arm with the highest UCB value is selected to be played in the next round.

In Equation (3), the amount by which UCB exceeds the expected reward is called the *exploration bonus*, representing the value of information. The exploration bonus, $\sqrt{2\log t/n_{kt}}$, can be expressed more generally with a parameter $\alpha > 0$ as $\sqrt{\alpha \log t/n_{kt}}$. The particular structure inside the UCB1 exploration bonus follows from the proof of the algorithm's optimality. The proof in Auer (2002) leads to setting $\alpha = 2$ to obtain its bound on regret for UCB1. But we will discuss the details of UCB in greater detail in Section 3 and derive an alternative value of α informed by economic theory and the pricing setting.

To prove that it is asymptotically optimal, the exploration bonus, which quantifies the value of information, is defined to ensure that cumulative regret grows slowly at a logarithmic rate in time, with arbitrarily high probability. Cumulative regret is the difference between the cumulative reward and the optimal cumulative reward, and we refer to ex post profits. We illustrate this in our proof, using deviation bounds (concentration inequalities), for our algorithm later in the theoretical analysis (Section 3.1 and the appendix).

The UCB framework, spawning multiple versions since Auer (2002), has become a standard approach in the MAB literature in machine learning, but those UCB versions have had limited application—for instance, first appearing in marketing in Schwartz et al. (2017). Even starting in Auer (2002) itself, new versions emerge. The authors note that the UCB1 algorithm considers only the number of times each action is played and does not account for the variance in outcomes from the trial of each arm. They provide an additional algorithm called UCB-tuned, with which they report better performance but without analytical regret bounds. For this algorithm, they define

$$V_{kt} = \left(\frac{1}{n_{kt}}\sum_{\tau=1}^{n_{kt}}\pi_{k\tau}^2\right) - \bar{\pi}_{kt}^2 + \sqrt{\frac{2\log t}{n_{kt}}}$$

$$\text{UCB-tuned}_{kt} = \bar{\pi}_{kt} + \sqrt{\frac{\log t}{n_{kt}}\min\left(\frac{1}{4}, V_{kt}\right)}. \tag{4}$$

An assumption in nonparametric multiarmed bandit algorithms, including the UCB algorithms, is that the profit outcomes in any two actions are uncorrelated. That is, the realized profits when action k is played do not inform us of the possible profits when another action j is played. In many marketing applications, this is a valid assumption depending on the design of the experiment, such as website design (Hauser et al. 2009). In other marketing applications with correlated actions, parametric assumptions are required to capture those correlations, as shown in online advertising (Schwartz et al. 2017).

Pricing is different. But it typically enters into a parametric demand model. In our application to pricing, however, we can add nonparametric demand learning (Handel et al. 2013) to MAB algorithms. We will prove the regret convergence rates with adding demand learning for the UCB1 algorithm and will then adapt the UCB-tuned algorithm to account for variance in observed profits. In the next section, we discuss how we can add nonparametric demand learning, and then we will discuss an updated model.

2.3. Learning the Demand Curve from Price Experiments

In this section, we will discuss how an analyst can learn preferences across different price experiments. In this section, our key parameter of interest is the demand for each product at each price level, or $D(p_k)\ \forall k \in [1, K]$. This section is based on the demand-side assumptions we make in Section 2.1. The implication of these assumptions is that if a consumer is willing to purchase a product at price p_1, she will be willing to purchase the product for a price $p_k < p_1$. Similarly, if the consumer does not purchase the product at p_2, she will not be willing to purchase the product for any price $p_k > p_2$.[9] Formally, we can define a set of possible consumer preference as $\Theta \equiv \{\theta_1, \ldots, \theta_K\}$, where θ_k refers to a preference that satisfies (a) $\theta_k - p_k > 0$ and (b) $\theta_k - p_{k+1} < 0$. Here, θ_k represents a preference under which the highest amount the consumer will purchase this good for is p_k.

If we consider products that are purchased repeatedly, we can use this information to identify bounds for each consumer's valuations. Consider an example where we observe the following price experiments for a consumer i. She purchases at \$3, does not purchase at \$8, purchases at \$2, and does not purchase at \$6. We can say that the true preference for this consumer (v_i) must be between \$3 and \$6. We can then aggregate this nonparametrically across all consumers to identify the set to all feasible demand curves (see section 2 of Handel et al. 2013 for details).

Requiring many repeat-purchase data for each customer may be overly restrictive or not suited for most products. In particular, we believe the assumption of stable preferences is likely to be violated when a consumer is exposed to multiple prices for the same product (Kalyanaram and Winer 1995). Instead, we focus on cross-sectional learning across consumers.

2.3.1. Learning Segment-Level Demand with Partial Identification.

The firm has data on $n_{s,t}$ price experiments for segment s through time t. In each experiment, a consumer i in segment s is exposed to a price p_k and makes a purchase decision. In this section, we will first describe how one can estimate v_s (segment valuations) and δ (intrasegment heterogeneity) from observed price experiments.

First, we can define the possible set of valuations containing consumers of various types, θ, to be consistent with observing segment s demand at price p_k, $D(p_k)_{s,t}$. We explain this with three cases:

- $D(p_k)_{s,t} = 0$ is consistent with consumers being of types $\{\theta_1, \ldots, \theta_{k-1}\}$, where consumers will not purchase at price p_k;
- $D(p_k)_{s,t} = 1$ is consistent with consumers being of types $\{\theta_k, \ldots, \theta_K\}$, where consumers will purchase for sure at price p_k; and
- $D(p_k)_{s,t} \in (0, 1)$ is consistent with a mixture of consumer types $\{\theta_1, \ldots, \theta_{k-1}\}$ and $\{\theta_k, \ldots, \theta_K\}$, where some consumers will purchase and other consumers will not purchase.

For any segment s, we can take intersection of the identified types across all pricing experiments, t. In our application, to estimate this set, we define p_s^{min} as the highest price where all consumers in segment s purchase. From a mathematical perspective, we set $p_{s,t}^{min} \equiv \max\{p_k | D(p_k)_{s,t} = 1\}$. And similarly, we define p_s^{max} as the lowest price where no consumer from segment s purchased, so $p_{s,t}^{max} \equiv \min\{p_k | D(p_k)_{s,t} = 0\}$.

To illustrate our estimation of p_s^{min} and p_s^{max}, consider the following example. Suppose we have data for a segment, $s = A$, with the following observations:

- At a price \$1, 100% of consumers purchased.
- At a price \$2, 20% of consumers purchased.
- At a price \$3, 0% of consumers purchased.
- At a price \$4, 0% of consumers purchased.
- At a price \$5, 0% of consumers purchased.

Given these data, we would define $p_A^{min} = \$1$ and $p_A^{max} = \$3$ because we know for sure that all consumers purchase at prices lower than \$1 and that no consumers will purchase at prices higher than \$3. This example is graphically displayed in Figure 2, panel (a). Here, we introduce another example for purchases in another segment, labeled segment B. We will refer to these segments in the next section.

We know that given the information so far, $\forall i \in s$, $v_{i,s} \in [p^{min}_{s,t}, p^{max}_{s,t}]$. To formally express set identification, we define $H_t(x) := [LB_t(x), UB_t(x)]$ to be the partially identified set for any parameter of interest, x, after t observed price experiments. Here, $LB_t(x)$ and $UB_t(x)$ are the estimated lower and upper bounds, respectively. In set notation, $x \in H_t(x) \equiv [LB_t(x), UB_t(x)]$. In this notation, $H_t[v_{i,s}] = [p^{min}_{s,t}, p^{max}_{s,t}]$.

We can define an estimated midpoint of the segment valuations ($\hat{v}_s$) and the segment level $\hat{\delta}_{s,t}$ as

$$\hat{v}_{s,t} = \frac{p^{\max}_{s,t} + p^{\min}_{s,t}}{2}$$
$$\hat{\delta}_{s,t} = \frac{p^{\max}_{s,t} - p^{\min}_{s,t}}{2}.$$

The interpretation of $\hat{\delta}_{s,t}$ here is that it is the smallest value of δ that can rationalize the observed decisions for consumers in segment s after t observed price experiments. We note that this estimate will be consistent: in the limit as $t \to \infty$ and with enough price variation, we can identify the true δ, calling this δ^*. That is, $\lim_{t\to\infty} P(\hat{\delta}_{st} = \delta^*) = 1$. However, for any finite t, these $\hat{\delta}_{st}$ will be biased downward.

To correct for this downward bias for each segment, we rely on the assumption that all segments have the same δ. We note that in the case where all segments do have heterogeneous δ_s our assumption then equates to $\delta = \max(\delta_s)$. This is a conservation assumption, as the estimate of the individual segments' δ_s can be upward biased. In turn, this upward bias leads to less precisely estimated demand and therefore slower learning of optimal prices.[10]

After t price experiments, consider the set $\{\hat{\delta}_{1t}, \ldots, \hat{\delta}_{St}\}$. We then estimate the maximum of that set, $\max\{\hat{\delta}_{st}, s \in S\}$. However, in small samples, this will be also be biased downward relative to the true δ^*. We follow the econometric literature on estimating boundaries to correct for this bias (see Karunamuni and Alberts 2005 for a review). We then denote the bias as γ_t, and we define $f(.)$ as the empirical distribution of $\hat{\delta}_{st}$ across consumers. Formally, $f(x) = \sum_{s\in S} \psi_s \mathbf{1}(\hat{\delta}_{st} = x)$. Note that ψ_s represents the segment sizes.

Our estimator for the bias γ_t is:[11]

$$\hat{\gamma}_t = \sum_{\{\delta_{st}\}} (\max\{\hat{\delta}_{st}, s \in S\} - \hat{\delta}_{st}) f(\hat{\delta}_{st}).$$

Therefore $\hat{\delta}_t = \max\{\hat{\delta}_{st}, s \in S\} + \hat{\gamma}_t$. This estimator is consistent, by our assumption of a common δ^* across segments. Handel et al. (2013) show that $\hat{\delta}_t$ provides a reliable and conservative estimate for the true δ^*. In Online Appendix A3 (Figure 2, panel (b)), we empirically show how the estimated $\hat{\delta}_t$ converges from above to the true δ as we run more price experiments.

Then we define $\hat{v}^{\min}_{s,t} = \hat{v}_{s,t} - \hat{\delta}_t$ and $\hat{v}^{\max}_{s,t} = \hat{v}_{s,t} + \hat{\delta}_t$ to represent, respectively, the lowest and highest possible consumer valuations within segment s. Therefore, the key output from this analysis is that for each segment s, we express the identified set of consumer valuations to be $H_t[v_{i,s}]$ as follows:

$$H_t[v_{i,s}] = [\hat{v}^{\min}_{s,t}, \hat{v}^{\max}_{s,t}] = [\hat{v}_{s,t} - \hat{\delta}_t, \hat{v}_{s,t} + \hat{\delta}_t].$$

2.3.2. Learning Population-Level Demand with Partial Identification.

Using distribution-free partial identification, aggregated to the population level, we gain information to narrow the set of possible demand curves. As we accumulate data of demand for different prices, we aim to bound expected demand (and expected profit) more tightly. After gaining new data, we can update the bounds. For each price p_k, the true demand is $D(p_k)$. Without any data, we can define the identification region of demand at p_k as $H_t[D(p_k)] = [0, 1]$. Here, we will use the identified set of valuations within each segment, $H_t[v_{i,s}]$, to estimate the bounds on aggregate demand and profits.

The aggregate demand at a price p_k is the number of consumers in each segment that have valuations $v_{i,s} \geq p_k$. Define $F_s(.)$ to be the true cumulative density of all valuation with a segment s. Then, we can rewrite aggregate demand as

$$D(p_k) = \sum_{s\in S} \psi_s (1 - F_s(p_k)),$$

where ψ_s is the (known) proportion of consumers in segment s.

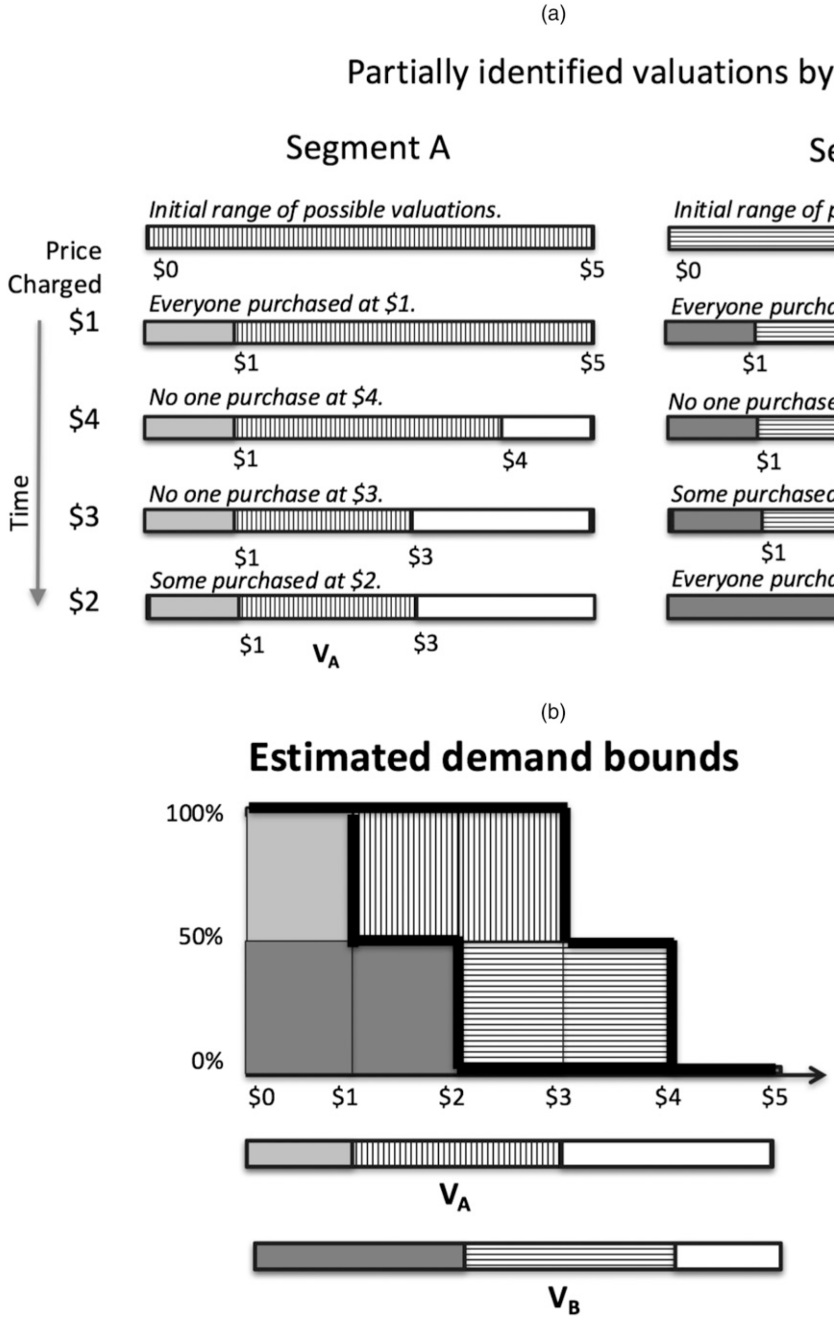
However, we do not observe $F_s(p_k)$. Therefore we can consider bounds of this distribution. From our estimation in the previous section, we know that $F_s(p_k) = 0$ if p_k is less than the lower bound of valuations for segment s, $\hat{v}^{\min}_{s,t}$. Similarly, $F_s(p_k) = 1$ if p_k is greater than the upper bound of valuations for segment s, $\hat{v}^{\max}_{s,t}$. Therefore, we can define the identified region for demand at price p_k as

$$H_t[D(p_k)] = [\sum_{s\in S} \psi_s \mathbf{1}(\hat{v}^{\min}_{s,t} \geq p_k), \sum_{s\in S} \psi_s \mathbf{1}(\hat{v}^{\max}_{s,t} \geq p_k)]. \tag{5}$$

That is, we aggregate the segment-level bounds of valuations, $H_t[v_{i,s}]$, to form the population-level bounds of aggregate demand, $H_t[D(p_k)]$.

This aggregation is best described in an example illustrated in Figure 2. Suppose we have two segments, A and B, of equal sizes. For segment A, we have identified preferences to be in the set [\$1, \$3], as described in the previous section. For segment B, suppose we have identified preference to be between [\$2, \$4]. We can identify the feasible demand sets as follows:

Figure 2. Partial Identification of Valuations by Segments and Aggregated Estimated Demand Bounds



Notes. (a) Identifying segment valuation: sample segment valuations are partially identified over four time periods of different prices charged; the dashed regions reflect ambiguity within a range of valuations in Segment A (vertical dashes) and Segment B (horizontal dashes); at prices below the identified set of valuations, everyone in the segment would purchase certainly (solid light gray, Segment A, or dark gray, Segment B); at prices above that set, no one in the segment would purchase (solid white). (b) Aggregate demand bounds: demand bounds (thick lines around dashed areas) come from aggregating segment-specific identified sets of valuations; solid colored areas represent prices where a segment would purchase certainly, and dashed regions reflect uncertainty; for instance, when the price is $1.50, all of Segment B will purchase, but some of Segment A may or may not purchase (hence, feasible demand is between 50% and 100%).

- $H_t[D(p \leq \$1)] = [1, 1]$ (point identified), as consumers in segments A and B will purchase for sure;
- $H_t[D(p \in (\$1, \$2])] = [0.5, 1]$, as consumers in segment A may or may not purchase and consumers in segment B will purchase for sure;
- $H_t[D(p \in (\$2, \$3])] = [0, 1]$, as consumers in segment A and segment B may or may not purchase;
- $H_t[D(p \in (\$3, \$4])] = [0, 0.5]$, as consumers in segment A will not purchase and consumers in segment B may or may not purchase; and

- $H_t[D(p > \$4)] = [0, 0]$ (point identified), as consumers in segments A and B will not purchase.

Using the identified aggregate demand bounds, we define profit bounds for each price as $H_t[\pi(p_k)] = p_k H_t[D(p_k)]$. This gives us the lower and upper bounds of true profit after t observations, which we define as $LB_t(\pi(p_k))$ and $UB_t(\pi(p_k))$, respectively. In summary, we have

$$\begin{aligned} H_t[\pi(p_k)] &= [LB_t(\pi(p_k)), UB_t(\pi(p_k))] \\ LB_t(\pi(p_k)) &= p_k \sum_{s \in S} \psi_s \mathbf{1}(\hat{v}_{s,t}^{\min} \geq p_k) \\ UB_t(\pi(p_k)) &= p_k \sum_{s \in S} \psi_s \mathbf{1}(\hat{v}_{s,t}^{\max} \geq p_k). \end{aligned} \tag{6}$$

3. Upper Confidence Bound with Partial Identification to Learn Demand

In this section, we extend the UCB1 algorithm to accommodate profit maximization by incorporating learning demand with partial identification. We define this upper confidence bound bandit algorithm incorporating learning partially identified demand (UCB-PI). The UCB-PI (untuned[12]) index is

$$\mathbf{UCB\text{-}PI}\text{-untuned}_{kt} = \begin{cases} \bar{\pi}_{kt} + p_k\sqrt{\frac{2\log t}{n_{kt}}} & \text{if } UB_t(\pi(p_k)) > \max_l LB_t(\pi(p_l)), \\ 0 & \text{if } UB_t(\pi(p_k)) \leq \max_l LB_t(\pi(p_l)). \end{cases} \tag{7}$$

There are two key differences between our proposed algorithm and the UCB1 algorithm (see Equation (3)) described in Auer (2002). First, we assign an action a nonzero value only if the upper bound of potential returns is higher than the highest lower bound across all actions. In a partial identification sense, we only consider an action if it is not dominated by another action. From an economic sense, there is no reason to explore an action if we know an alternative action will lead to higher profits with certainty. From an empirical standpoint, we will examine how the set of active prices varies over time, eliminating dominated prices and focusing on a set including the true optimal price.

Second, we scale the exploration bonus by price p_k. This is because we know $D(p_k) \in [0, 1]$, and therefore $\pi(p_k) \in [0, p_k]$. But the original UCB1 algorithm was defined where each action's reward had the same potential range (e.g., $[0, 1]$), regardless of action. By restricting demand, we impose a natural upper bound of profit that depends on price.

In the following sections, we first prove properties of the UCB-PI and show that regret is lower than UCB1. Then we define a tuned version of the UCB-PI algorithm analogous to the UCB-tuned algorithm in Audibert et al. (2009).

3.1. Theoretical Performance

In this section, we provide the key result of the theoretical analysis of our proposed algorithm. First, we derive the exploration bonus in the UCB-PI-untuned algorithm, given by Equation (7), and second, we prove performance guarantees for the algorithm. The complete proof is in the appendix. The derived performance guarantee is an upper bound on the regret for our algorithm, which describes performance in the worst-case scenario. From an economic perspective, this represents the maximum regret over all feasible demand curves for our algorithm.

We give the structure of the theoretical results. In Equation (2), we define regret as a function of (a) a feasible demand curve (and therefore, a profit curve), (b) an algorithm (i.e., UCB-PI-untuned), and (c) a time horizon (T). To understand theoretical performance, we can derive the implications of an algorithm across all feasible demand systems. Our main result shows that the UCB-PI-untuned algorithm (Equation (7)) provides a log-regret upper bound. That is, for any feasible demand model, the regret from UCB-PI-untuned increases, at most, on the scale of $\log(T)$ over time. Log-regret bounds are important; the computer science literature shows that these are asymptotically optimal for MAB problems (Agrawal 1955, Lai and Robbins 1985, Lai 1987, Auer 2002). Therefore, our proposed UCB-PI-untuned is an asymptotically optimal MAB algorithm.

From a mathematical perspective, suppose we consider a set of prices $\{p_k | k = 1, \ldots, K\}$ and any feasible demand curve that generates a set of profits $\{\pi(p_k)\}$. Without loss of generality, we label p_1 as the optimal prices for this demand curve (i.e., $p_1 = \text{argmax}_{p_k}\{\pi(p_k)\}$). The regret through time T, in terms of profit, is bounded from above as follows:

$$\begin{aligned} &\text{Regret}(\text{UCB-PI-untuned}, \{\pi(p_k)\}, T) \\ &\quad \leq 8 \sum_{k=2}^{K} \frac{p_i \log(T)}{\pi^* - \pi(p_i)} + O(1). \end{aligned} \tag{8}$$

Because all p_k are scaled to be in $[0, 1]$, this regret is guaranteed to be lower than that in the UCB1 algorithm, which is the standard UCB formulation in the computer science literature. We note that our proof is based on an alternative view of the original UCB analysis (Auer et al. 2002). But our proof differs from the standard proof, even in technique, because we use an argument based on the *potential function*, which we define formally in the proof. This argument using the potential function is a novel application of these tools for the formal analysis of learning algorithms. We choose this technique because we believe it provides a clearer economic intuition for UCB-style algorithms and their proofs.

3.2. UCB-PI-Tuned Algorithm

We will present a version of our algorithm where we tune the exploration bonus by considering both the variance of the observed outcomes and the size of the bound. This is analogous to the UCB-tuned algorithm presented in Auer et al. (2002). The V_{kt} represents an upper bound on the reward variance (as opposed to mean). It is also equal to its empirical variance plus an exploration bonus:

$$\mathrm{V}_{kt} = \left(\frac{1}{n_{kt}} \sum_{\tau=1}^{n_{kt}} \pi_{k\tau}^2\right) - \bar{\pi}_{kt}^2 + \sqrt{\frac{2\log t}{n_{kt}}}.$$

The upper bound on variance enters the UCB of the mean to control the size of the exploration bonus. We add an additional tuning factor, $2\hat{\delta}$, because it is the size of the range for our partially identified intervals. When δ is large, there is more uncertainty; when it is small, the intervals shrink and so does the exploration bonus:

$$\textbf{UCB-PI}\text{-tuned}_{kt} = \begin{cases} \bar{\pi}_{kt} + 2p_k\hat{\delta}\sqrt{\frac{\log t}{n_{kt}}}\min\left(\frac{1}{4}, \mathrm{V}_{kt}\right) & \\ \qquad \text{if } UB_t(\pi(p_k)) > \max_l LB_t(\pi(p_l)), & \\ 0 \qquad \text{if } UB_t(\pi(p_k)) \le \max_l LB_t(\pi(p_l)). & \end{cases} \tag{9}$$

The final novel aspect of the proposed algorithm is "shutting off" prices that are dominated. Dominated prices have an upper bound that is still worse than at least some other price's lower bound, $UB_t(\pi(p_k)) \le \max_l(LB_t(\pi(p_l)))$.

4. Empirical Performance: Simulation Study

We test our proposed algorithm in a series of simulation experiments. These show its robust performance across unknown true distributions of consumer valuations. Each simulation has the same structure of the data-generating process. Customers arrive, observe the price, and purchase only if their valuation is greater than the price. After the customers decide, the firm observes which customers arrived (in particular, the consumer's segment) and their choices. Then the firm sets the price for the next period.

In our simulation, we consider a firm with $K = 100$ potential prices from \$0 to \$1 in 1*c* increments. The firm can change prices after every 10 consumers visit. Each consumer belongs to one of $S = 1,000$ segments. We draw the segment probabilities (true ψ_s) from a simplex on the uniform distribution.

Each segment's valuation (true v_s) is drawn from a parametric distribution. We consider five possible distributions of valuations. Importantly, this distribution is the data-generating process and is unknown to the researcher, so it is not assumed in the estimation method:

1. right-skewed Beta distribution given by Beta(2, 9),
2. symmetric Beta distribution given by Beta(2, 2),
3. left-skewed Beta distribution given by Beta(9, 2),
4. bimodal continuous given by Beta(0.2, 0.3), and
5. discontinuous finite mixture model with each v_s equal to either \$0.2 (with 70% chance) or \$0.9 (30%).

The purpose of these settings is to consider a range of different possible distributions of consumer preferences, leading to range of aggregate demand and profit curves. The first three simulation settings involve unimodal continuous distributions of valuations. Varying the distribution of consumer preferences across these simulations will result in different levels of sparseness of purchases at each price point. The last two simulations have bimodal continuous and bimodal discontinuous distributions leading to bimodal profit functions. Aghion et al. (1991) show that Bayesian learning leads to inadequate learning in these settings. The distribution of the valuations, the aggregate demand curve, and the aggregate profit curve for each simulation setting are all shown in Appendix A2.

Within each segment, consumers' valuations can be 10*c* ($\delta = 0.1$) above or below the segment valuation. Alternatively, the range of within-segment heterogeneity is 20% of the range of across-segment heterogeneity (between 0 and 1).

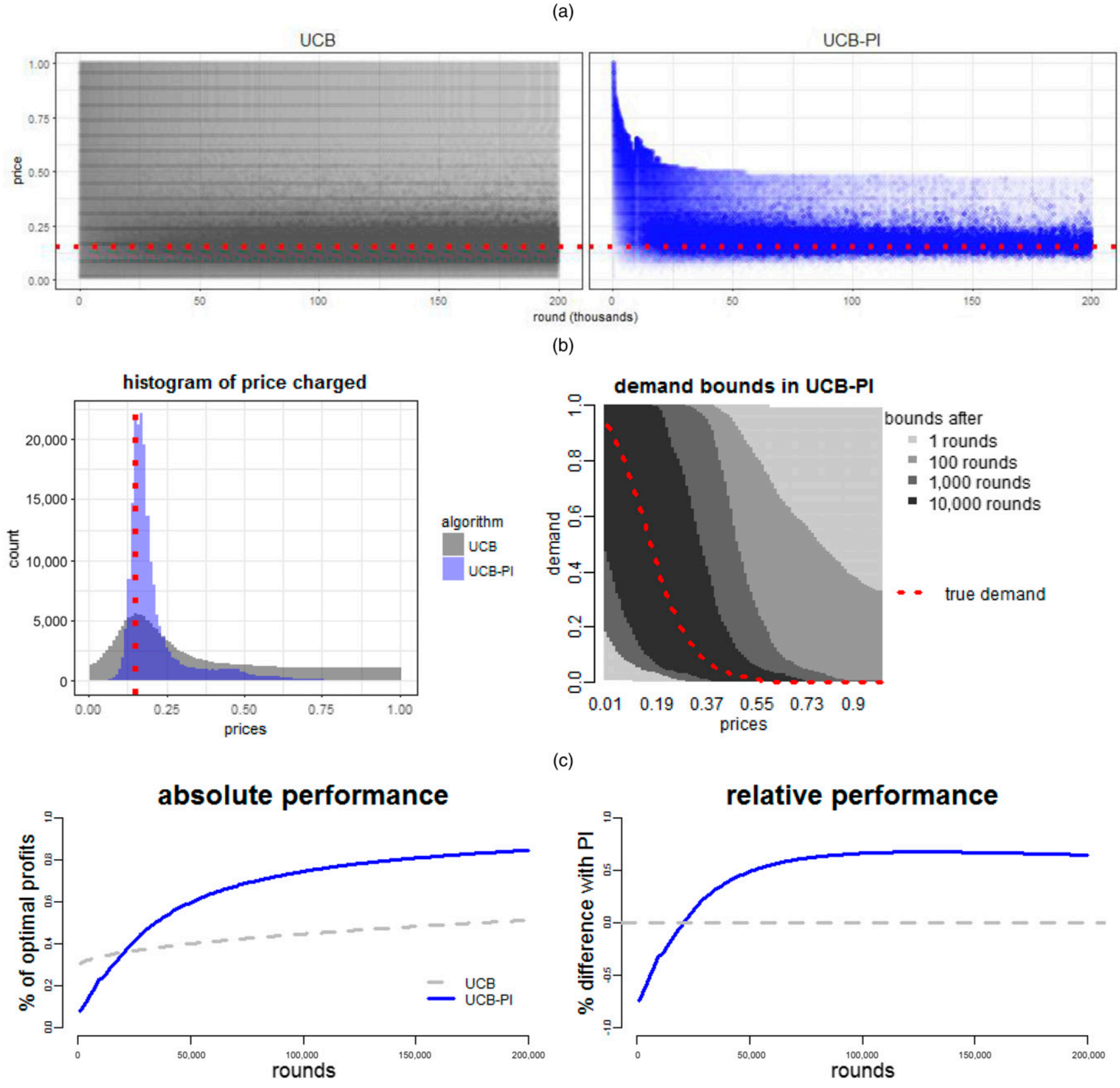
4.1. Estimating the Demand Model: Comparing Untuned Algorithms

To show the advantage of adding partial identification to UCB algorithms, we run simulations of prices and consumer decisions over 200,000 decision rounds. We note that the computer science literature has noted that tuned algorithms outperform untuned algorithms, although Auer (2002) feels that this is an important comparison to show the relative benefit of adding partial identification.

The results for the first simulation (segment valuations are right-skewed) are shown in Figure 3. In panel (a), we plot the prices charged in each of 200,000 rounds for UCB1 untuned (left) and UCB-PI untuned (right). We can see the set of prices tested each period by UCB-PI narrows, showing that partial identification allows us to reduce the number of prices with which to experiment. The algorithm narrows to focus only on prices near the true optimal price. This is summarized in panel (b) (left), which shows the number of times each possible price is charged by UCB and UCB-PI. Although UCB's modal price is near the true optimal price, the UCB-PI algorithm concentrates nearly all of its observations on prices at or close to optimal.

The key reason for the differences between the algorithms is partial identification. We illustrate the

Figure 3. (Color online) Value from Partial Identification: Comparison of the UCB1- and UCB-PI-Untuned Algorithms for the Simulation with True Segment Valuations from Right-Skewed Distribution, beta(2, 9), and Within-Segment Heterogeneity Set to $\delta = 0.1$



(a)

(b)

(c)

Notes. (a) Price charged; prices by experiment with UCB (left) and UCB-PI (right) to learn the true optimal price (red line). The UCB-PI learns the optimal price sooner and does not experiment with higher prices (compared to UCB). (b) Prices and demand: the left histogram of prices charged under UCB (gray) and UCB-PI (blue), with truly optimal price (vertical red line); the right figure shows the partially identified demand learning under UCB-PI; the shaded gray regions show the partially identified demand bounds after 1 round (start; lightest), 100 rounds, 1,000 rounds and 10,000 rounds (darkest). (c) Profits: implication of price learning on profits earned by round; the left chart consider profits relative to optimal profits, and the right chart shows the relative performance of the algorithms [defined as: (UCB-PI profits/UCB profits) – 1].

partially identified bounds of the aggregate demand for UCB-PI in Figure 3, panel (b), right. At each price the shaded areas represent the partially identified bounds for the demand curve after 100, 1,000, and 10,000 rounds (later rounds are shown in darker shades). The true demand (which is the dashed line) is always within the partially identified bounds. Notice that after only one round, we have very wide demand bounds, and as we get more data, the partially identified bounds get narrower. Furthermore, after 10,000 rounds we know that the demand above a price of \$0.7 is the point identified at 0. As a result, the UCB-PI algorithm no longer experiments with prices above $0.7 (see panel (a), right chart). In Appendix A3, we explicitly show the prices "turned off" by round. Because of this demand learning, the UCB-PI algorithm results in higher ex post profits than the UCB algorithm (panel (c)). Overall, the UCB-PI attains 90% of ex post optimal profits, whereas the UCB1 attains 50% of ex post optimal profits. We note that we will focus on profits when comparing the tuned algorithms in the next section.

We show this demand learning for the other four simulations in Figure 4. Each panel of this figure represents a simulation. The left column is the histogram of prices charged across 200,000 rounds. Here, we see that in each of our simulations the histogram for prices played under UCB-PI (blue) is tighter around the true optimal profit (vertical left line) than the prices played under UCB (gray). Therefore, with partial identification, the algorithm spends more rounds earning and fewer rounds learning.

Figure 4 (right column) represents demand learning over time. In all of our simulations, the demand bounds get narrower around the true demand curve as we get more data. This is the advantage of partial identification: for any data-generating process that satisfies the assumptions in Section 2.1, we can bound the true demand curve. Most importantly, consider Figure 4, panel (d), where demand is discontinuous. This is a case where Aghion et al. (1991) find that Bayesian learning almost surely leads to inadequate learning. With partial identification we do correctly learn the demand curve; moreover, after 10,000 rounds (the darkest shade), the model recovers with certainty that the true demand curve must be discontinuous.

4.2. Profit Implications of Adding PI: Comparison of Tuned Algorithms

In this section, we will consider the UCB-tuned and the UCB-PI-tuned algorithms. In this section, we will run our simulation for 20,000 rounds (as opposed to 200,000 rounds in the previous section), as learning is faster in the tuned algorithms. Figure 5 plots the prices played and profit earned in each of these five simulation settings based on different true demand curves. Each row of this figure corresponds to each simulation setting.

We will first focus on the prices charged. In panel (a) of Figure 5, we plot the price played each round for UCB-tuned (left column) and UCB-PI-tuned (middle column) algorithms. Across all simulations we find that adding partial identification results in the algorithm setting prices at the optimal levels more often, with more focused experimentation. This is consistent with our results from the untuned algorithms where we found that partial identification leads to faster learning of demand.

Turning our attention to profit achieved, Figure 5, panel (a) (right column) shows UCB-PI's relative profit improvement over UCB over time. We find that the UCB-PI outperforms UCB consistently. The maximum increase in profitability is between 15% and 90% across the different simulations. As the number of rounds goes to infinity, the UCB is guaranteed to achieve optimal profits (Auer 2002), and therefore the relative benefit is guaranteed to asymptote to zero. However, we find that adding partial identification increases the profits gained in smaller rounds, with the maximum benefit achieved between 1 and 5,000 rounds.

In an absolute sense, the algorithms can be compared with ex post optimal profit. The ex post optimal profit is the profit achieved assuming the firm had perfect demand information and set the optimal price in every period. Figure 5, panel (b) shows the ex post profits across all five settings. Again, we find the UCB-PI-tuned achieves higher profits than the UCB-tuned algorithm in each simulation. In particular, the UCB-PI algorithm achieves above 95% of ex post optimal profit in four of the five settings. In the finite mixture setting where learning is difficult (Aghion et al. 1991), the algorithm achieves 89% of ex post profits.

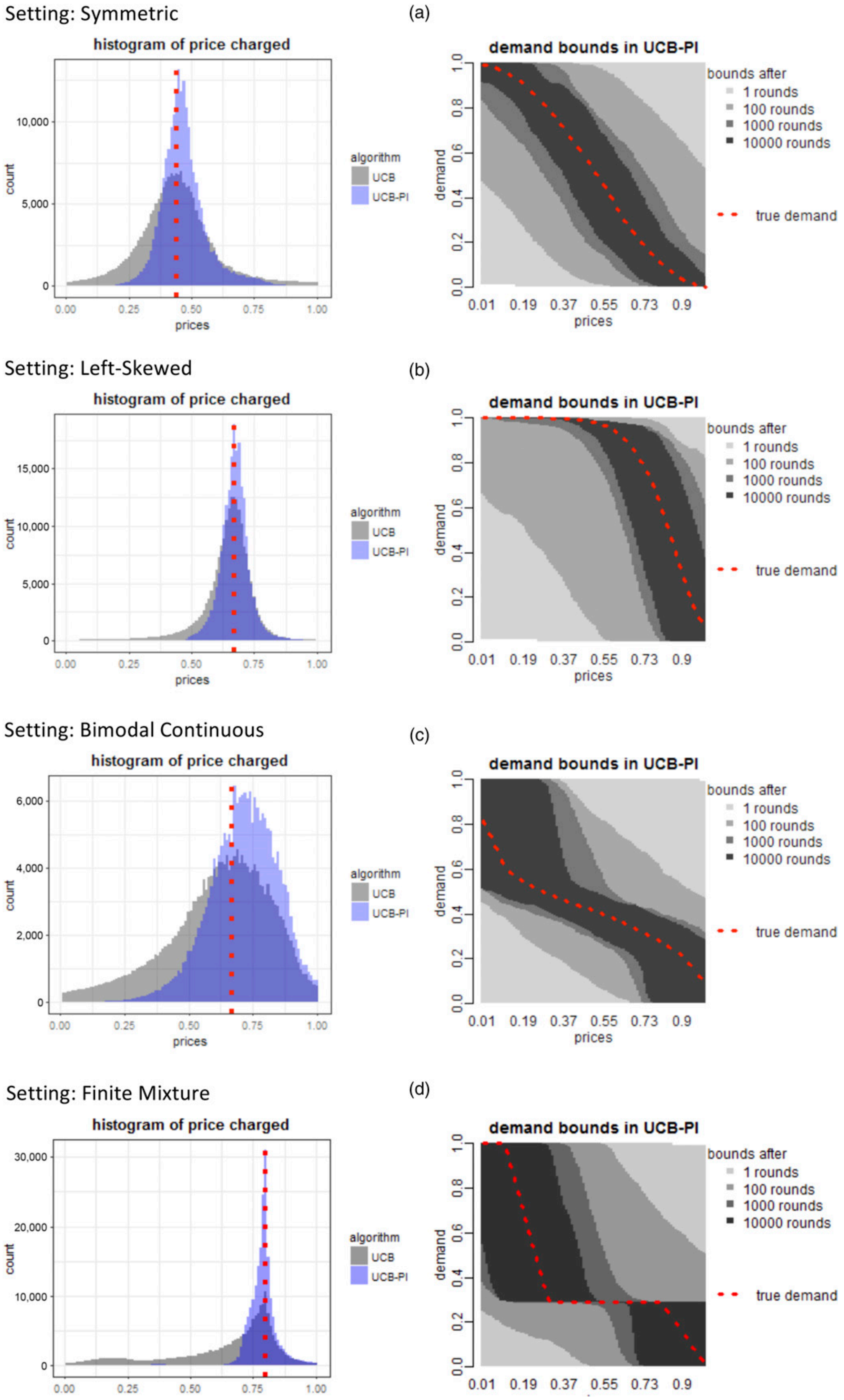
4.3. Monte Carlo Comparisons to Alternative Algorithms

Whereas we have just illustrated UCB-PI's superior performance over UCB in five settings, the algorithm performance is stochastic, so we now show a large Monte Carlo simulation across algorithms and problem settings. Here, we consider a broader set of algorithms; we consider the *learn-then-earn* algorithm, or a balanced field experiment. In these algorithms, the researcher has to ex ante set how long the algorithm should learn (experimental time) and how long the algorithm should use that learning in order to earn. In our simulation experiments, we consider five versions of the learn-and-then-earn algorithm where learning is set for 0.1%, 1%, 5%, 10%, and 25% of experiments. We also include the case where the learning period is 100% because it is exactly a *balanced experiment*, which matches typical research-driven field experiments and A/B or multivariate testing in industry. The performance of a balanced experiment is not optimizing because it only earns the average profit of all its arms.

In all, we consider 10 different algorithms: UCB untuned and tuned, UCB-PI untuned and tuned, and learn-then-earn with six different settings. We test each algorithm across each of our five simulation settings. We run each algorithm and setting pair for 1,000 independent Monte Carlo (MC) simulations. In each, we run each algorithm for 20,000 rounds, simulating time, with 10 consumers in the time period. In all, we simulate one billion prices in this Monte Carlo simulation, and it takes only about five hours of computation time.

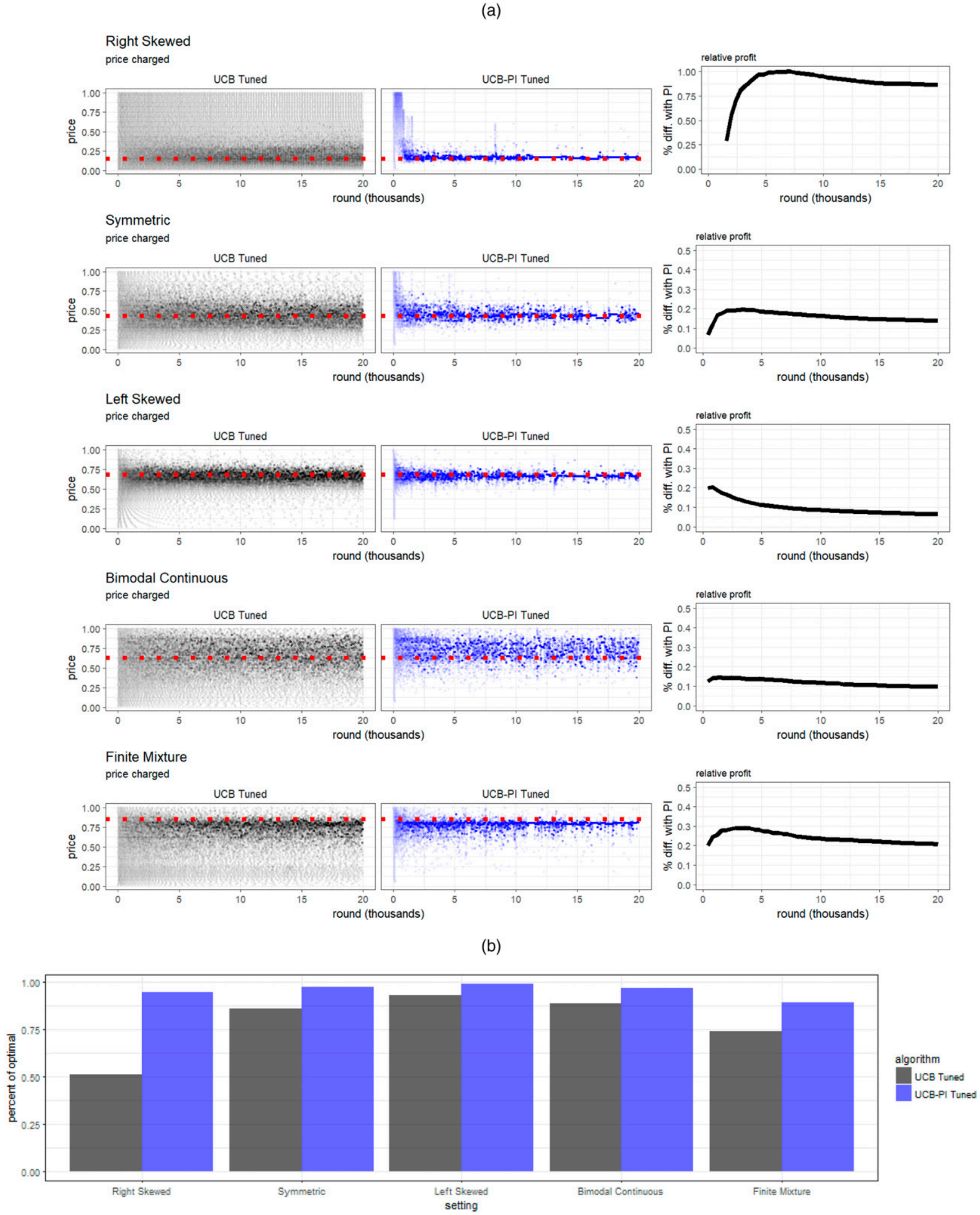
Our key measure of performance is profits as a percentage of optimal, for which a summary of ex post achieved profits for each setting and algorithm appears

Figure 4. (Color online) Value from Partial Identification: Demand Learning for the UCB-PI-Untuned Algorithm



Notes. The left panel is a histogram of prices charged under UCB (gray) and UCB-PI (blue), with truly optimal price (vertical red line). The right panel shows the partially identified demand learning under UCB-PI. The shaded gray regions show the partially identified demand bounds after 1 round (start; lightest), 100 rounds, 1,000 rounds, and 10,000 rounds (darkest). For all settings, the within-segment heterogeneity is set to $\delta = 0.1$.

Figure 5. (Color online) Value of Partial Identification for Tuned Algorithms



Notes. The UCB-tuned (gray) and UCB-PI-tuned (blue) algorithms differ across five settings, for one simulation each, by prices charged, relative profits, and percentage of ex post optimal profits. (a) Each row represents a simulation setting; the first (second) column contains the UCB Tuned (UCB-PI Tuned) prices by round; the dotted (red) line represent the optimal prices; the third column considers the percentage difference in profits. (b) Ex post profits achieved across the five (5) simulation settings.

in Figure 6. In panel (a), we display the distribution of performance for each algorithm across all five settings and 1,000 MC simulations. We display the mean and range (in brackets) of profits achieved. The bar charts represent the 100th (full range), 90th, and 75th percentiles of the profits achieved. Looking across all settings, the UCB-PI algorithm stands out, with the average of 96% of ex post optimal profit and a narrow range from 91% to 99%. The ex post optimal profit is the highest across all algorithms, and the range is the narrowest across all algorithms.

The learn-then-earn algorithms, with learning periods between 0.5% and 25%, achieve an average between 89% and 95% of ex post optimal profits depending on the time to learn. We find that in this setting, the highest mean profit is for the 5% learning period. However, a researcher cannot know ex ante (when setting the learning time) that 5% would be have been best in this setting. Consistent with the empirical bandit literature (Kuleshov and Precup 2014), we find that heuristic-based algorithms (learn-then-earn) can achieve higher ex post profits than UCB-tuned. We find that when we add economic theory to the UCB algorithm, the UCB-PI-tuned algorithm outperforms the heuristic-based algorithms across a wide range of settings.

The learn-then-earn experiments have a large range of ex post profit outcomes. A key advantage of the theory-based algorithms (such as UCB-PI) is that they have a lower range of outcome; this is consistent with the objective of minimax regret. To show the differences across rounds, in Figure 6, panel (b), we plot the outcomes across all simulations for the UCB-PI algorithm and the learn then earn (5%) policy (which had the highest ex post average profit among alternative algorithms) by experimental round (time). This shows that UCB-PI has higher profits, particularly in early time periods. The highest average percentage difference between the algorithms is at the end of the learn-then-earn policy's learning phase (5%). On average, the UCB-PI achieved 20% higher profits after 5% of experimental rounds. As both algorithms run for longer time periods, the mean difference is reduced to 1%. Beyond the mean difference, the UCB-PI has a lower variability across the 1,000 Monte Carlo experiments. This variability is desirable because a researcher could not predictably know ex ante, for any given setting, when the learn-then-earn algorithm will perform well. In Appendix A4, we show the distribution of outcomes after 10,000 rounds for each simulation setting.

5. Empirical Performance: Simulation Based on a Pricing Field Experiment (Dubé and Misra 2017)

In this section, we conduct a simulation study based on the field experiment in Dubé and Misra (2017) (henceforth referred to as DM). DM conducted a price experiment in collaboration with the business-to-business company ZipRecruiter.com, "a large, online recruiting firm" (DM, p. 2). In the experiment's phase I, conducted in September 2015, they ran a randomized price experiment where "the experiment randomly assigned each new customer arriving at the website's paywall to one of ten price buckets ranging from $19 to $399, including a control condition of $99, which was the firm's regular base price at that time" (DM, p. 2). In phase II, they "used the experimental data to estimate (i.e., 'train') a demand model" (DM, p. 3). In phase III, they implemented the best uniform price (DM, table 4). Using the terminology from our paper, it could be said that the pricing policy implemented in DM is learn-then-earn, where DM phases I and II were "learn" and phase III was "earn." In this section, we will consider a simulation to illustrate the benefits of our proposed bandit algorithm over a pricing experiment using distinct learn-and-earn phases.

We note that we consider only the uniform pricing in DM. Importantly, the objective (and title) of DM is to consider price discrimination with a targeting pricing model that is not considered in our algorithm.[13] This section is best viewed as considering the field experiment from DM with the objective function limited to finding the optimal uniform price.

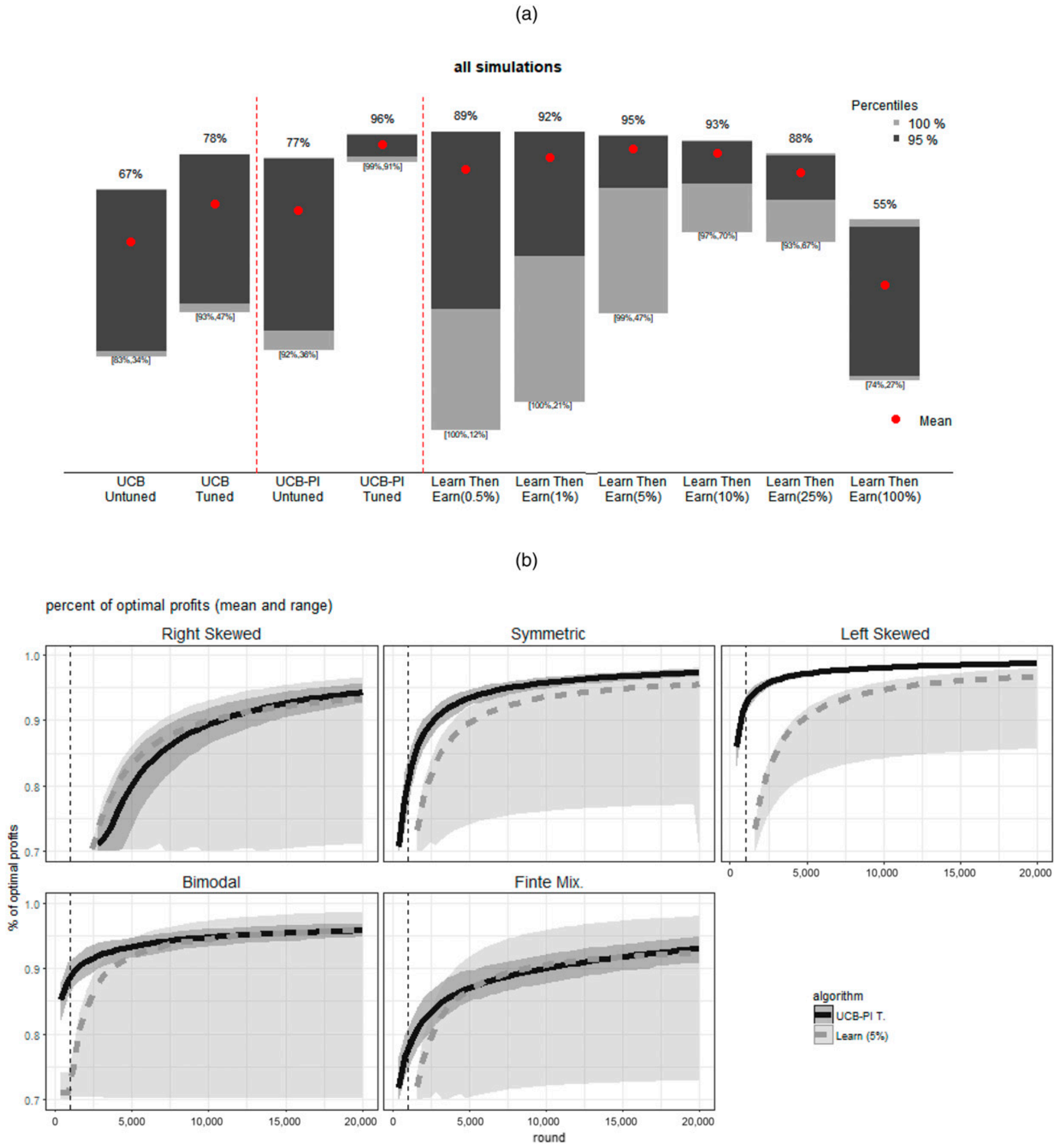
Two main features of this setting are consistent with our model. First, consistent with DM, we assume that the firm does not have additional information about the demand curve (formally, the firm does not have additional ex ante demand information such as, e.g., probability distribution over feasible demand curves, outside the experiment). Second, each consumer has to enter detailed characteristics data (e.g., type of job, type of firm). This allows the firm to assign consumers to different segments in our model.

Next, we will first discuss how we simulate data sets based on the DM study. Then we will compare our proposed method (UCB-PI) for optimizing pricing experiments with the computer science method (UCB-tuned) and a balanced field experiment (learn-then-earn). We find our approach achieves 43% higher profits in the first month and sustains positive gains throughout the year in 80% of simulations.

5.1. Simulation Setup: Data Generation Using Field Experiment Data

Our simulation is based on table 5 in DM, which reports the observed demand ("acquisition rates") based on phase I of the experiment. We use a simple data-generating process from a demand system consistent with the realized demand in the experiment. DM discusses that in its setting consumers have to reveal 11 different descriptive variables before they are shown a price. Because this is a business-to-business application, the consumers are firms paying for the services of ZipRecruiter.

Figure 6. (Color online) Monte Carlo Experiment


(a)

(b)

Notes. We compare the performance of the UCB, UCB-PI, and learn-then-earn algorithms across 1,000 MC simulations and five settings. We consider the ex post profits achieved, the average, and the variability in performance, summarizing the full distributions of outcomes overall (a) and over time by simulation setting (b). (a) Summary across all simulations: plot considers the ex post profits achieved by UCB, UCB-PI and Learn Then Earn (balanced experiments) across all 5 simulations and 1,000 Monte Carlo simulations; the bars represent the range of 100% (light gray; numerical values are shown below the bars), 95% (dark gray) of estimates; the dots (red) represent the mean profit achieved (numerical values are shown above the bars). (b) Profits over time: plot considers the mean and range of profits achieved by the UCB-PI tuned and the learn-then-earn (5%) algorithms by simulation settings and experimental round (time); the vertical line shows the end of the leaning period; this shows that the UCB-PI achieves the same or higher mean profits than the best performing learn-then-earn algorithm, and UCB-PI has a smaller range across Monte Carlo simulations.

These descriptive variables include company descriptors, job posting descriptors, and job benefits (for a complete list, see table 2 in DM). We assume that the firm can use these variables to create segments of consumers, and we assume there are 1,000 such ex ante equally sized segments the firm can create, just as we have previously assumed. We simulate the 1,000 segment mean valuations based on the following underlying piecewise-continuous distribution that is consistent with the aggregate demand reported for 10 prices in table 5 of DM.

Of all consumers viewing the lowest price of $19, 36% of them purchase, so we assume that 64% of segments have valuations drawn from a uniform distribution between $0 and $19. At the second-lowest price of $39, 32% of consumers purchase, so we assume that 4% (36% − 32%) of segments have valuations drawn from a uniform distribution between $19 and $39, and so on, to the highest price of $399.[14]

We now illustrate how our data-generating process is consistent with the data reported in DM. We consider 10,000 independent replications of the observed experiment with 7,870 simulated consumers. For each replication, we draw a set of 1,000 midpoints of segment valuations from the distribution described above. From all of the equally sized segments, we sample 7,870 consumers' segment membership labels. Each consumer's valuation is then a draw from a uniform distribution between $-\delta$ and $+\delta$, that midpoint of the consumer's segment. We set $\delta = \$5$ because we find that this value makes the true optimal price, averaged across replications of the experiment, to be $281, so we are consistent with DM. We randomize all consumers to the same 10 price points, with 787 consumers drawn for each price. This allows for small-sample upward-sloping demand curves. From the demand curve, and assuming zero marginal cost, we calculate the implied profit curve for each replication of the experiment.

The resulting empirical distributions of our simulated demand and profit curves are consistent with the realized demand and profit in DM, as we show in Figure 7, panel (a). For each price, we illustrate the distribution of simulated demand (panel (a), left) including the realized acquisition rates in DM's table 5 (red dots). The distribution of simulated per-consumer profits at each price (panel (a), right) shows that the optimal profit levels are achieved at prices $199–$399. For 60% of the experiment replications, the optimal fixed uniform price is $249. We also find $399 is optimal for 15% of small-sample experiments (as in DM's raw data). The average optimal uniform price is $281 (as in DM's phase II calculations). It is important to note that this implies, in an infinite sample setting, that the optimal uniform price would be $249 for all Monte Carlo draws (as in the implemented uniform price in DM).

5.2. Pricing Policies in Simulation

We now consider the firm's pricing decision, facing the previously described data and under the following assumptions. First, we assume that the firm sets prices with the objective to maximize total annual profits. In the DM experiment, ZipRecruiter.com had about 8,000 visitors in one month, so we assume 100,000 consumers will visit in a year. Second, the firm considers only a well-defined set of 10 prices to test, the same as in the DM field experiment. And third, at the start of the year, the firm does not know what the optimal price would be.

Under these assumptions, we will consider two type of experiment policies: learn-then-earn and bandit experiments. As we have defined, using the learn-then-earn policy, the firm runs a balanced field experiment, setting each price uniformly randomly for the predefined learning phase. Then, in the earn phase, the firm finds the optimal price and sets that for a remaining time. For example, a 10% learn-then-earn experiment runs a balanced experiment for 10% of the year, finds the optimal price, and sets that for the remaining 90% of the year. We will consider learning phases of various lengths: 0.5%, 1%, 5%, 7.9% (this represents the DM experiment), 10%, and 25%.

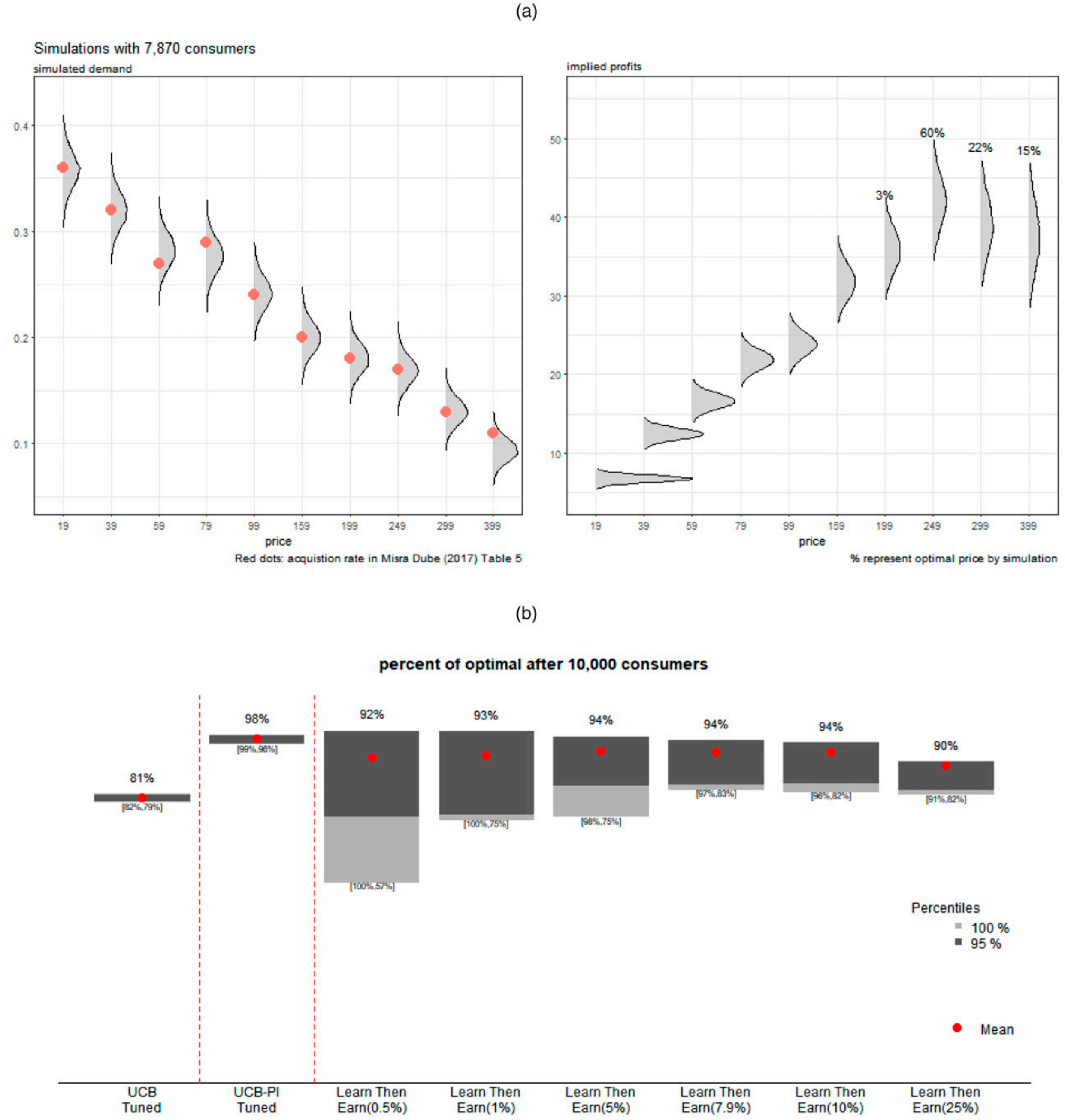
Using the bandit experiment, as opposed to the distinct phases of the learn-then-earn policy, the firm continually balances learning and earning across each time period. We assume that the firm can change prices after every 10 visitors, so there are 10,000 possible price changes (pricing rounds). In addition to our proposed policy, UCB-PI-tuned, we consider the standard MAB algorithm from computer science, UCB-tuned, described earlier.

5.3. Results of Simulation Based on Field Experiment

The resulting total annual profits, across different policies and all simulated replications, are shown in Figure 7, panel (b). The UCB-PI policy achieves the highest average profit and has the lowest variability of profits across simulations. To highlight the advantage of adding partial identification to this method, the UCB-PI-tuned increases ex post optimal profits to 98%. The computer science benchmark, UCB-tuned, does not perform as well and achieves only 81% of ex post optimal profits. Among the learn-then-earn policies tested, the highest ex post profit is achieved by the 7.9% version, which represents the DM experiment. This corresponds to about one month of balanced experimentation followed by setting the optimal price for the remaining 11 months of the year. However, even with this method, which attains 94% of optimal profits, there is a wide range of ex post profit achieved, as high as 97% and as low as 83%. The UCB-PI algorithm not only has a higher mean (98%) but also has a much tighter range across simulations, as high as 99% and only as low as 96%. Looking at profits across the year with 100,000 customers, compared with the 7.9% learn-then-earn, the UCB-PI does maintain higher profits, with a mean difference of about 4%. The range across all simulations is from −1% to 18%, with a positive difference in 88% of simulations.

Beyond looking at the annual profits, we specifically look at profits during the first month. Whereas annual profits combine the learn phase (first month) of the 7.9%

Figure 7. (Color online) Simulation Based on Dubé and Misra (2017): Setup and Results



(a)

(b)

Notes. We describe a data-generating process to recreate data from DM (a). We compare performance of the UCB, UCB-PI, and learn-then-earn algorithms across 1,000 Monte Carlo experiments (b). (a) Simulation set up: we consider the simulated demand (left) and profits (right) for 10,000 draws of 7,870 consumers; in each draw, we assume that an independent set of 787 consumers are exposed to each of the 10 prices. Consumers valuations are draw from a uniform distribution between −$5 and +$5 of the segment midpoint; the percentage of simulations in which each price is profit-maximizing appears in value above the denisty (right chart). (b) Ex post profits: we consider the ex post profits achieved by UCB, UCB-PI and Learn Then Earn (balanced experiments) across 1,000 Monte Carlo simulations; we consider the profit after 100,000 consumers (about 1 year); the bars represent the range of 100% (light gray, and the interval values are shown below the bars), and 95% (dark gray) of the profit estimates; the dots (red) represent the mean profit (the mean values are shown above the bars).

learn-then-earn policy and its earning (remaining 11 months), profits in the first month alone show UCB-PI enjoys its best relative performance, 43% higher profits (Figure 7, panel (b)). The UCB-PI quickly achieves 92% of optimal profits during those first 7,870 customer interactions. By contrast, the 7.9% learn-then-earn policy achieves 62% of optimal profits. Because it is a balanced experiment for the month, the policy just achieves an average of the profit levels for each price tested.

We consider the evolution of prices and profits by over time in both algorithms to better understand how the UCB-PI performs better than the learn-then-earn algorithm. In Figure 8, panel (a), we plot the evolution of prices by round, where each round refers to 10 customers visiting the website, and prices are shaded based on their true ex post profit level (i.e., higher profits are shown with less shading). In the learn-then-earn algorithm, by definition, the first 7,870 consumers are in a balanced field experiment and, therefore, are exposed to one of the 10 prices with equal probability (i.e., all prices appear as equally sized regions). After the end of the learning period (marked by the dotted vertical line, Figure 8), the algorithm picks the price with the best profits so far and charges that price for the remaining rounds. Reflecting the underlying data-generating process (Figure 7, panel (a)), the algorithm learns the true ex post optimal price of \$249 in about 60% of simulations.

The UCB-PI algorithm, however, balances learning and earning continuously over all rounds and brings about two important differences in the price paths. First, the UCB-PI tends not to charge low prices with lower profits (regions shaded by dark gray). This is driven by the partial identification (PI) in the algorithm. For example, if the estimated lower bound of demand at \$399 is 5%, then the lower bound of profit at \$399 is higher than the profit at \$19 even with 100% demand. Based on Figure 8, panel (a), the UCB-PI will "turn off" experimentation at \$19. Second, whereas learn-then-earn stops adjusting price after its experimentation phase, the UCB-PI method continues learning and continuously keeps increasing the percentage of rounds charging the ex post optimal price of \$249. These trends are also shown in the profits across rounds (Figure 8, panel (b)).

6. Conclusion and Future Research

With the emergence of big data, we see an increase in machine learning applications in marketing (Chintagunta et al. 2016). We study a realistic dynamic pricing problem for an online retailer. The goal is a pricing experimentation policy that can apply to many types of products (robust) and run in real time (fast).

We propose a novel combination of reinforcement learning with microeconomic theory. To marketing and economics, we bring these scalable reinforcement learning methods for the firm's pricing problem. Here, we show the benefit of earning while learning over balanced field experiments.

To the machine learning literature, we introduce distribution-free theory of demand to improve existing algorithms theoretically and empirically.

We provide strong evidence for the benefit of partial identification of demand in nonparametric bandit problems. We derive theoretically the rate of convergence for our algorithm. This shows that when faced with any weakly downward-sloping demand, our algorithm is guaranteed to converge faster than current algorithms.

Across a range of simulation settings, we show that our proposed algorithm, compared with alternatives, achieves a higher mean profit and a lower variance of outcomes across settings. This suggests that our proposed algorithm can be used for a variety of products and will predictably lead to higher profits. In a calibrated simulation based on a field experiment (Dubé and Misra 2017), we find that our algorithm achieves 43% higher profits during the month of testing and about 4% higher annual profits.

A limitation of our current work is that we consider a simple demand system, where each consumer has a stable valuation. We note that our demand model assumes that the firm has information about observed heterogeneity. In the limit, if the firm has no information about observed heterogeneity (or $\delta = 1$), our method (UCB-PI) will exactly replicate the computer science method (a version of UCB with scaling the exploration bonus by prices). We illustrate this in Online Appendix A5.3.

Further research can consider settings where consumer valuation can change over time. This could be in the form of prior prices creating reference prices or consumers with dynamic preferences. A challenge is that optimal profit will no longer be stationary and, moreover, will vary based on the experimentation method. Another avenue for further research is to analyze demand systems that consider more than one product; this includes both category management and competition. (We note that our algorithm and theoretical results are not guaranteed for markets with strategic competition.) Additionally, further research can consider optimal price discrimination in the form of targeted or personalized pricing.

Acknowledgments

The first two authors are listed alphabetically.

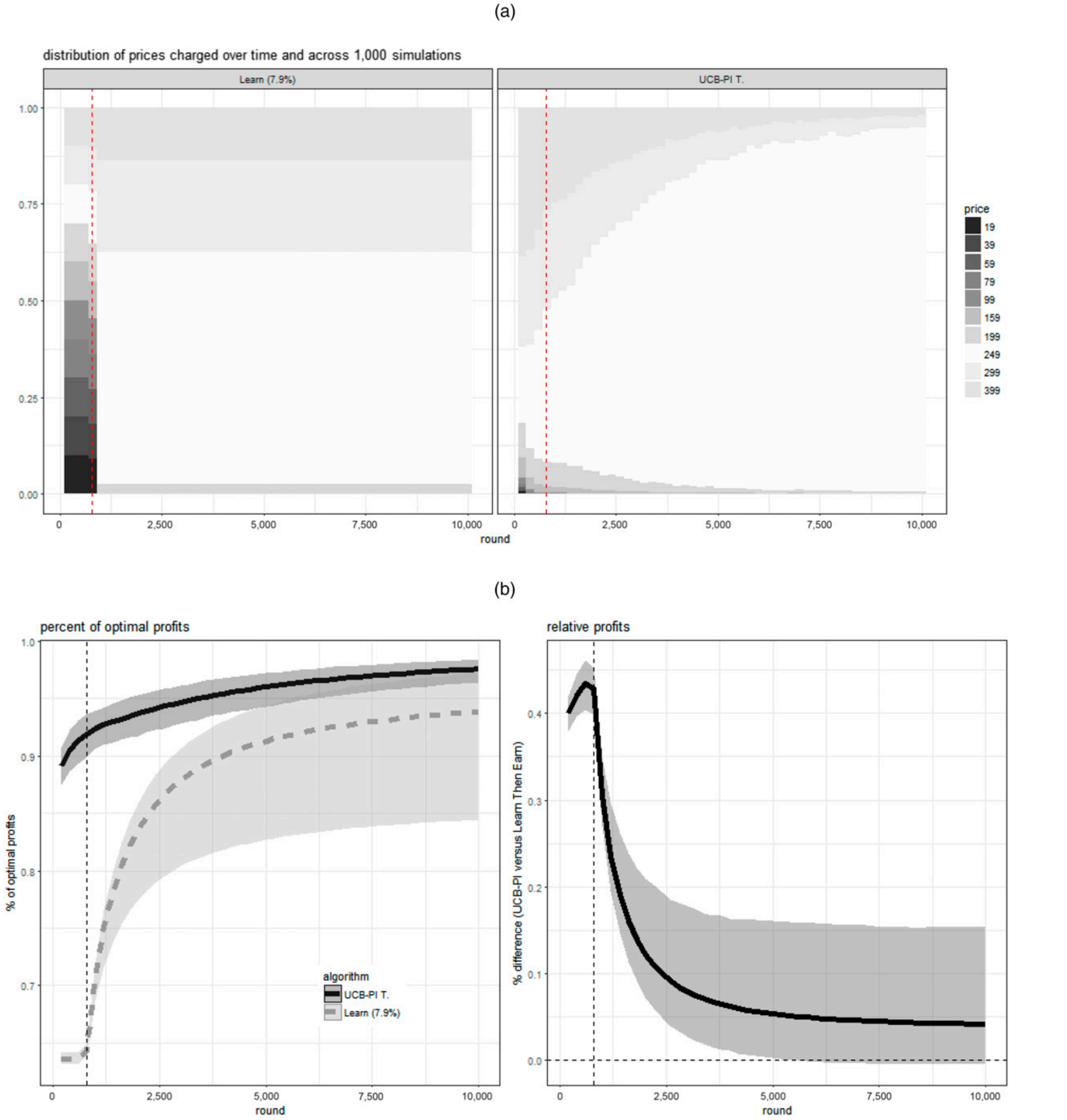
Appendix. Theoretical Performance of UCB-PI Algorithm

We provide theoretical guarantees for the UCB-PI index (Equation (7))[15]. The log-regret bound was first shown by Lai and Robbins (1985) for a particular stylized multiarmed bandit problem. Our proof is based on an alternative view of the original UCB analysis. The proof we present here differs from the standard UCB proof from Auer et al. (2002) because we use an argument based on *potential function*, which we define in the proof. This argument in the proof features a novel application of these tools for formally analysis of algorithms. We use this alternative approach, in part, because it permits a more general description of the exploration bonus.

Problem Definition and Preliminaries

We use more general notation here to go beyond price and profit to describe actions and rewards. Price p_k played at time

Figure 8. (Color online) Simulation Based on Dubé and Misra (2017): Tracking over Time



Notes. We compare the prices (a) and profits (b) of UCB-PI versus learn-then-earn (7.9%) algorithms over time. The 7.9% reflects the 7,870 customers, corresponding to the one month (or 787 rounds). (a) Distribution of prices: we plot the distribution of prices charged across 1,000 Monte Carlo simulations by UCB-PI tuned (right) and the learn-then-earn (7.9%) (left) algorithms by round (each round represents a price charged to 10 cosumers); the prices charged are color coded by optimal ex post profits (light is high and dark is low); the dashed (red) line indicates the end of the learning period in the learn-then-earn algorithm. (b) Profits over time: we plot the profits achieved by the UCB-PI tuned and the learn-then-earn (7.9%) algorithms by round (10 consumers); the lines represent the means and the shaded area represents the distribution across 1,000 Monte Carlo simulations; the left hand side plot shows the percent of ex post optimal achieved, and the right hand side plot show the relative profits; the vertical line indicates the end of the learning period in the learn then earn algorithm.

t is the action described by $\mathscr{A}^t$. Profit π_{p_k} at price p_k is described as a random variable R_i^t of reward for $i \in 1, \ldots, K$.

Let Q_i be a distribution on the reward R_i^t, with support on $[0, p_i]$. Then let the rewards $R_i^1, \ldots, R_i^T \overset{\text{iid}}{\sim} Q_i$, where mean $\mathbb{E}[R_i^t] = \mu_i$. We assume that the largest μ_i is unique and, without loss of generality, assume that the coordinates are permuted in order that μ_1 is the largest ex post mean reward. Define $\Delta_i := \mu_1 - \mu_i$ for $i = 2, \ldots, K$.

The *bandit algorithm* is a procedure that chooses an action $\mathscr{A}^t$ on round t as a function of the set of past observed action/reward pairs, $(\mathscr{A}^1, R^1_{\mathscr{A}^1}), \ldots, (\mathscr{A}^{t-1}, R^{t-1}_{\mathscr{A}^{t-1}})$.

On round t, the past data are summarized by the count, $N_i^t := \sum_{\tau=1}^{t-1} \mathbb{I}[\mathscr{A}^\tau = i]$, and the empirical mean estimator, $\hat{\mu}_i^t := \frac{\sum_{\tau=1}^{t-1} \mathbb{I}[\mathscr{A}^\tau = i] R^\tau_{\mathscr{A}^\tau}}{N_i^t}$.

Analysis Techniques: Concentration Inequalities and Potential Function

Much of the literature and techniques used to analyze finite time multiarmed bandit problems rely on a standard set of tools known as *deviation bounds* or *concentration inequalities*. Deviation bounds are used to reason about tail probabilities of averages of independent and identically distributed random variables and martingales, for instance. Perhaps the most basic deviation bound is Chebyshev's inequality, which says that for any random variable X with mean μ and variance σ^2 we have $\Pr(|X - \mu| > k\sigma) \leq \frac{1}{k^2}$. More advanced results are based on the *Chernoff bounds*, which provide much sharper guarantees on the decay of the tail probability. For example, the *Hoeffding–Azuma inequality* (Cesa-Bianchi and Lugosi 2006), which we present below, gives a probability bound on the order of $\exp(-k^2)$, which is much faster than $1/k^2$.

Let us assume we are given a particular deviation bound that provides the following guarantee:

$$\Pr(|\mu_i - \hat{\mu}_i^t| > p_i \epsilon \mid N_i^t \geq N) \leq f(N, \epsilon), \tag{10}$$

where $f(\cdot,\cdot)$ is a function, continuous in $\epsilon > 0$ and monotonically decreasing in both parameters, that controls the probability of a large deviation. Although UCB relies specifically on the Hoeffding–Azuma inequality, for now we leave the deviation bound in a generic form.

We define a pair of functions that allows us to convert between values of ϵ and N in order to guarantee that $f(N, \epsilon) \leq \nu$ for a given $\nu > 0$. To this end, define

$$\Lambda(\epsilon, \nu) := \min\{N \geq 1 : f(N, \epsilon/2) \leq \nu\};$$
$$\rho(N, \nu) := \begin{cases} \inf\{\epsilon : f(N, \epsilon) \leq \nu\} & \text{if } N > 0, \\ 1 & \text{otherwise.} \end{cases}$$

We omit the ν in the argument to $\Lambda(\cdot), \rho(\cdot)$. Note the property that $\rho(N, \nu) \leq \epsilon/2$ for any $N \geq \Lambda(\epsilon, \nu)$.

Note that $\hat{\hat{\delta}}$, which plays a role in the lower and upper bounds on reward, does not enter this proof, yet we can conclude the proof applies to our proposed algorithm. Indeed, δ is not known to the researcher and must be estimated. Consider the worst case (in the sense that this will lead to the maximum regret), where segmentation is useless; then $\delta = 1$. Then the credible intervals for every segment's feasible profit still are the entire possible range. This is the case presented in the proof here. But in practice, $0 \leq \delta \leq 1$, and δ can be smaller than its maximum value, making segmentation useful and narrowing the partially identified intervals. Therefore, the proposed UCB-PI algorithm does no worse than the performance described here.

Bounds for the UCB-PI Algorithm

Recall that the UCB-PI index is defined in Equation (7) by taking the mean estimated reward plus an exploration bonus for each price p_i. The precise form of the exploration bonus derives from the deviation bound, particularly from the form of $\rho(\cdot)$. In other words, for a fixed choice of $\nu > 0$, we can redefine the algorithm as follows:

$$\textbf{UCB-PI} \text{ Algorithm : on round } t, \text{play } \mathscr{A}^t = \arg\max_i \{\hat{\mu}_i^t + p_i \rho(N_i^t, \nu)\}. \tag{11}$$

A central piece of the analysis relies on the following potential function, which depends on the current number of plays of each arm $i = 2, \ldots, K$:

$$\Phi(N_2^t, \ldots, N_K^t) := 2 \sum_{i=2}^{K} \sum_{N=0}^{N_i^t - 1} p_i \rho(N, \nu).$$

With our notation, the expected regret can be expressed as

$$\mathbb{E}[\text{Regret}_T(\text{UCB})] = \sum_{t=1}^{\tau} \mu_1 - \mu_{\mathscr{A}^t}.$$

Lemma 1. *The expected regret of UCB is bounded as*

$$\mathbb{E}[\text{Regret}_T(\text{UCB})] \leq \mathbb{E}[\Phi(N_2^{T+1}, \ldots, N_K^{T+1})] + O(T\nu)$$

Proof. The additional (statistical) regret suffered on round t of UCB is exactly $\mu_1 - \mu_{\mathscr{A}^t}$. From our deviation bound (Equation (10)), we can consider two inequalities:[16]

$$\mu_1 \leq \hat{\mu}_1^t + p_i \rho(N_1^t, \nu) \quad \text{and} \quad \hat{\mu}^t_{\mathscr{A}^t} \leq \mu_{\mathscr{A}} + p_{\mathscr{A}^1} \rho(N^t_{\mathscr{A}^1}, \nu).$$

To analyze the two inequalities above, we let ξ^t be the indicator variable that one of the above two inequalities fails to hold. Note we chose $\rho(\cdot)$ so that $\mathbb{P}[\xi^t = 1] \leq 2\nu$.

Because the algorithms choose arm $\mathscr{A}^1$, we have

$$\hat{\mu}_1 + p_1 \rho(N_1^t, \nu) \leq \hat{\mu}^t_{\mathscr{A}^1} + p_{\mathscr{A}^1} \rho(N^t_{\mathscr{A}^1}, \nu).$$

If we combine the above two equations, and consider the event that $\xi^t = 0$, then we obtain

$$\begin{aligned} \mu_1 &\leq \hat{\mu}_1^t + p_1 \rho(N_1^t, \nu) \leq \hat{\mu}^t_{\mathscr{A}^1} + p_{\mathscr{A}^1} \rho(N^t_{\mathscr{A}^1}, \nu) \\ &\leq \mu_{\mathscr{A}^1} + 2 p_{\mathscr{A}^1} \rho(N^t_{\mathscr{A}^1}, \nu). \end{aligned}$$

Even in the event that $\xi^t = 1$, we have that $\mu_1 - \mu_{\mathscr{A}^1} \leq 1$. Hence, $\mu_1 - \mu_{\mathscr{A}^1} \leq 2 p_{\mathscr{A}^1} \rho(N^t_{\mathscr{A}^1}, \nu) + \xi^t$.

Finally, we observe that the potential function was chosen so that $\Phi(N_2^{t+1}, \ldots, N_K^{t+1}) - \Phi(N_2^t, \ldots, N_K^t) = 2 p_{\mathscr{A}^1} \rho(N_{\mathscr{A}^1}, \nu)$. Recalling that $\Phi(0, \ldots, 0) = 0$,

$$\begin{aligned} \mathbb{E}[\text{Regret}_T(\text{UCB})] &\leq \mathbb{E}\left[\Phi(N_2^{T+1}, \ldots, N_K^{T+1}) + \sum_{t=1}^{T} \xi^t\right] \\ &= \mathbb{E}[\Phi(N_2^{T+1}, \ldots, N_K^{T+1})] + 2T\nu. \quad \square \end{aligned}$$

The final piece we need to establish is that the number of pulls N_i of arm i, for $i = 2, \ldots, K$, is unlikely to exceed $\Lambda(\Delta_i, \nu)$.

Lemma 2. *For any $T > 0$, we have $\mathbb{E}[\Phi(N_2^{T+1}, \ldots, N_K^{T+1})] \leq \Phi(\Lambda(\frac{\Delta_2}{p_2}, \nu), \ldots, \Lambda(\frac{\Delta_K}{p_k}, \nu)) + O(T^2 \nu)$.*

Proof of Lemma 2. To obtain the inequality of the lemma, define for every $t = 1, \ldots, T$ and $i = 2, \ldots$ the indicator variable ζ_i^t, which returns 1 when $\mathscr{A}^t = i$ given that $N_i^t \geq \Lambda(\frac{\Delta_i}{p_i}, \nu)$ and returns 0 otherwise. We can show that $\zeta_i^t = 1$ with a probability smaller than 2ν.

Note that if $\mathscr{A}^t = i$, then the upper confidence estimate for i was larger than that of action 1. More precisely, it must be that $\hat{\mu}_i^t + p_i\rho(N_i^t) \geq \hat{\mu}_1^t + p_1\rho(N_1^t)$. For this to occur, either we had (a) a large underestimate on μ_1—that is $\hat{\mu}_1^t + p_1\rho(N_1^t) \leq \mu_1$—or (b) a large overestimate on μ_i—that is, $\hat{\mu}_i^t + p_i\rho(N_i^t) \geq \mu_1$. It is clear that (a) occurs with a probability of less than ν by construction of ρ.

To analyze (b), note that $\mu_1 = \mu_i + \Delta_i$, and we are also given that $N_i^t \geq \Lambda(\frac{\Delta_i}{p_i}, \nu)$, which implies that $p_i\rho(N_i^t) \leq \Delta_i/2$. Then

$$\hat{\mu}_i^t + p_i\rho(N_i^t) \geq \mu_1 \Rightarrow \hat{\mu}_i^t \geq \mu_i + p_i\rho(N_i^t),$$

which happens with a probability of no more than ν. Therefore,

$$\begin{aligned}\mathbb{E}[\Phi(N_2^{T+1}, \ldots, N_K^{T+1})] &\leq \Phi(\Lambda(\frac{\Delta_2}{p_2}, \nu), \ldots, \Lambda(\frac{\Delta_K}{p_k}, \nu)) \\ &\quad + \mathbb{E}[\sum_{i=2}^{K}\sum_{t=1}^{T}\zeta_i^t] \\ &\leq \Phi(\Lambda(\frac{\Delta_2}{p_2}, \nu), \ldots, \Lambda(\frac{\Delta_K}{p_k}, \nu)) + 2T^2\nu.\end{aligned}$$

□

We are now able to combine the above results for the final bound.

Theorem 3. *If we set $\nu = T^{-2}/2$, the deviation bound is given by*

$$\rho(N, \nu) = \sqrt{\frac{\log(2/\nu)}{2N}}.$$

And the expected regret of UCB is bounded as

$$\mathbb{E}[\text{Regret}_T(\text{UCB})] \leq 8\sum_{i=2}^{K}\frac{p_i\log(T)}{\Delta_i} + O(1).$$

Proof. A standard deviation bound that holds for *all* distributions supported on $[0, p_i]$ is the Hoeffding–Azuma inequality (Cesa-Bianchi and Lugosi 2006), where the bound is given by $f(N, \epsilon) = 2\exp(-2N\epsilon^2)$. Utilizing Hoeffding–Azuma, we have

$$\Lambda(\epsilon, \nu) = \left\lceil\frac{2\log 2/\nu}{\epsilon^2}\right\rceil$$

and

$$\rho(N, \nu) = \sqrt{\frac{\log(2/\nu)}{2N}}$$

for $N > 0$. If we utilize the fact that $\sum_{y=1}^{Y}\frac{1}{\sqrt{y}} \leq 2\sqrt{Y}$, then we see that

$$\begin{aligned}\Phi(\Lambda(\frac{\Delta_2}{p_2}, \nu), \ldots, \Lambda(\frac{\Delta_K}{p_k}, \nu)) &= 2\sum_{i=2}^{K}\sum_{N=1}^{\Lambda(\frac{\Delta_i}{p_i}, \nu)}\rho(N, \nu) \\ &= 2\sum_{i=2}^{K}\sum_{N=1}^{\Lambda(\frac{\Delta_i}{p_i}, \nu)}\sqrt{\frac{\log(2/\nu)}{2N}} \\ &\leq 2\sum_{i=2}^{K}2\sqrt{\frac{\log(2/\nu)\Lambda(\frac{\Delta_i}{p_i}, \nu)}{2}} \\ &= 4\sum_{i=2}^{K}\frac{p_i\log(2/\nu)}{\Delta_i}.\end{aligned}$$

Set $\nu = T^{-2}/2$ in the equation above. From Lemma 1 and 2, we prove that this provides an upper bound for the expected regret. □

Notice in Theorem 3 that we set

$$\rho(N, \nu) = \sqrt{\frac{\log(2/\nu)}{2N}}.$$

Substituting this into the UCB algorithm in Equation (11), we get the exploration bonus in our proposed algorithm (see Equation (7)). The bound for regret for this algorithm is strictly lower than that of Auer (2002) as all p_i's are scaled to be lower than 1.

Further adding the additional partial identification implies that an arm is played weakly less than $\Lambda(\epsilon, \nu)$ derived by the Hoeffding–Azuma inequality. Consider the proofs for Lemmas 1 and 2: as arms are "turned off," we get lower deviation bounds. As we discussed above, the number of arms turned off in an empirical application depends on the value of δ. The bounds derived are for δ at its maximum value, where no arms are turned off, and segmentation is not useful. However, the empirical performance of our algorithm should improve for any lower values of δ. The theoretical argument holds true as a worst-case analysis; that is $\delta = 1$.

Please see the online appendix for appendix Sections A2–A5.

Endnotes

[1] For example, industry reports suggest Amazon.com sells about 500 million products (https://www.scrapehero.com/many-products-amazon-sell-january-2018/, accessed February 11, 2018).

[2] We note that in practice such segmentation is used by advertisers using data from Facebook (https://www.facebook.com/help/analytics/1568220886796899/) or Google Analytics (https://support.google.com/analytics/answer/3123951?hl=en, accessed March 2016).

[3] Recent industry reports suggest that Amazon.com carries 14 times more products than the second-largest online retailer, Walmart.com (https://www.scrapehero.com/amazon-vs-walmart-products-sold-in-april-2017/).

[4] We note that if the decision maker is willing to make stronger parametric assumptions about the demand curve, alternative streams of MAB algorithms based on parametric models are more appropriate. One such algorithm is the earliest Bayesian formulation of the MAB problem, which led to Thompson sampling (Thompson 1933). A more prominent formulation with Bayesian learning led to the Gittins index (Gittins 1989) and Whittle index (Whittle 1980).

[5] Amazon.com ran price experiments in 2000 and then, as a result of consumer feedback, released a statement saying "random testing was a mistake, and we regret it because it created uncertainty and complexity for our customers, and our job is to simplify shopping for customers. That is why, more than two weeks ago, in response to customer feedback, we changed our policy to protect customers" (http://cnnfn.cnn.com/2000/09/28/technology/amazon/, accessed February 11, 2018).

[6] This is an assumption about the data-generating process and not the information that the firm has. In our empirical application, we assume that δ is not known to the firm and will be estimated from observed choice data.

[7] Using the websites http://camelcamelcamel.com and https://thetracktor.com (accessed February 11, 2018), we can track price changes.

[8] We note that regret is appropriate because of the active learning setting. We need an "ex ante" criteria to evaluate a pricing policy. By *ex ante*, we mean that the objective function must be one that can be calculated without knowing the true demand curve. Specifically, we cannot consider criteria such as total expected profits, as these cannot be used to evaluate a policy *before* the probability of outcomes is realized.

[9] In the treatment choice literature (Manski 2005), this corresponds to monotone treatment response.

[10] In estimating partially identified demand models, a downward biased estimate of δ leads to incorrect demand inferences, whereas an upward biased estimate of δ leads to imprecisely estimated demand. Consider our example in panel (a) of Figure 2. If δ is downward biased, then for segment A, we could estimate a set smaller than [$1, $3]; however, for an upward biased δ, we would estimate a set larger than [$1, $3].

[11] Our estimator is similar to that used in Handel et al. (2013), which is an adaption of the Hall and Park (2002) estimator. Formally, the Hall and Park (2002) boundary estimator considers a setup where the econometrician observes N draws from a continuous univariate distribution F with an unknown and finite upper boundary. The Handel et al. (2013) estimator is a discrete analog to these methods, because the distribution of $\hat{\delta}_{st}$ is discrete.

[12] Where we label this "untuned" in contrast to its tuned version defined later.

[13] Note that table 4 in DM suggests that optimal uniform pricing results in (a) a higher profit than the control condition and (b) the profits from uniform and targeted pricing being statistically indistinguishable.

[14] We make two further assumptions when simulating true population valuations. First, the acquisition rates in DM are upward sloping between $59 and $79. We assume that this is a small sample error, as opposed to the distribution of the population's preferences. In our simulation we assume 28% of consumers purchase at both $59 and $79 (instead of 27% at $59 and 29% at $79 as in DM, Table 5). Second, in table 5 of DM, the acquisition rate is 11% at $399. At this number, $399 is the optimal uniform price ($11\% * 399[43.89] > 17\% * 249[42.33]$). This is inconsistent with the population's optimal price from the analysis in DM ("We can rule out $399 as being too high since there is close to a 100% posterior probability that the own-elasticity is well above –1 at that point"; p. 17). We assume that this is because of a small sample error and assume demand at $399 is 10%.

[15] In the main text we refer to this as the UCB-PI-untuned, but we drop the "untuned" label here for easier reading because the theoretical literature does not focus on the "tuned" algorithms.

[16] Here, we can see that if δ were less than its maximum value, the partially identified intervals shrink, and the above probabilities are smaller.

References

Acquisti A, Varian HR (2005) Conditioning prices on purchase history. *Marketing Sci.* 24(3):367–381.

Aghion P, Bolton P, Harris C, Jullien B (1991) Optimal learning by experimentation. *Rev. Econom. Stud.* 58(4):621–654.

Agrawal R (1955) Sample mean based index policies with $O(\log n)$ regret for the multi-armed bandit problem. *Adv. Appl. Probab.* 27(4):1054–1078.

Akçay Y, Natarajan HP, Xu SH (2010) Joint dynamic pricing of multiple perishable products under consumer choice. *Management Sci.* 56(8):1345–1361.

Anderson E, Jaimovich N, Simester D (2015) Price stickiness: Empirical evidence of the menu cost channel. *Rev. Econom. Statist.* 97(4):813–826.

Audibert JY, Munos R, Szepesvári C (2009) Exploration-exploitation trade-off using variance estimates in multi-armed bandits. *Theoret. Comput. Sci.* 410(19):1876–1902.

Auer P (2002) Using confidence bounds for exploitation-exploration trade-offs. *J. Machine Learn. Res.* 3(November):397–422.

Auer P, Cesa-Bianchi N, Fischer P (2002) Finite-time analysis of the multiarmed bandit problem. *Machine Learn.* 47(2/3):235–256.

Aviv Y, Pazcal A (2002) Pricing of short life-cycle products through active learning. Working paper, Washington University of St. Louis, St. Louis.

Baker W, Kiewell D, Winkler G (2014) Using big data to make better pricing decisions. *McKinsey & Company* (June), http://www.mckinsey.com/business-functions/marketing-and-sales/our-insights/using-big-data-to-make-better-pricing-decisions.

Bayus BL (1992) The dynamic pricing of next generation consumer durables. *Marketing Sci.* 11(3):251–265.

Bergemann D, Schlag K (2008) Pricing without priors. *J. Eur. Econom. Assoc.* 6(2-3):560–569.

Bergemann D, Schlag K (2011) Robust monopoly pricing. *J. Econom. Theory* 146(6):2527–2543.

Bergemann D, Valimaki J (2000) Experimentation in markets. *Rev. Econom. Stud.* 67:213–234.

Berger JO (1985) *Statistical Decision Theory and Bayesian Analysis*, 2nd ed. (Springer, New York).

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

Biyalogorsky E, Gerstner E (2004) Contingent pricing to reduce price risks. *Marketing Sci.* 23(1):146–155.

Biyalogorsky E, Koenigsberg O (2014) The design and introduction of product lines when consumer valuations are uncertain. *Production Oper. Management* 23(9):1539–1548.

Bonatti A (2011) Menu pricing and learning. *Amer. Econom. J. Microeconom.* 3(3):124–163.

Braden DJ, Oren SS (1994) Nonlinear pricing to produce information. *Marketing Sci.* 13(3):310–326.

Brezzi M, Lai TL.(2002) Optimal learning and experimentation in bandit problems. *J. Econom. Dynam. Control* 27(1):87–108.

Cesa-Bianchi N, Lugosi G (2006) *Prediction, Learning, and Games* (Cambridge University Press, New York).

Chintagunta P, Hanssens DM, Hauser JR (2016) Marketing science and big data. *Marketing Sci.* 35(3):341–342.

den Boer AV (2015)Dynamic pricing and learning: Historical origins, current research, and new directions. *Surveys Oper. Res. Management Sci.* 20(1):1–18.

Desai PS, Koenigsberg O, Purohit D (2010) Forward buying by retailers. *J. Marketing Res.* 47(1):90–102.

Dubé J-P, Misra S (2017) Scalable price targeting. NBER Working Paper 23775, National Bureau of Economic Research, Cambridge, MA. http://www.nber.org/papers/w23775.

Elmaghraby W, Keskinocak P (2003) Dynamic pricing in the presence of inventory considerations: Research overview, current practices, and future directions. *Management Sci.* 49(10):1287–1309.

Erdem T, Keane MP (1996) Decision-making under uncertainty: Capturing dynamic brand choice processes in turbulent consumer goods markets. *Marketing Sci.* 15(1):1–20.

Furman J, Simcoe T (2015) The economics of big data and differential pricing. *The White House: President Barack Obama* (blog) (February 6), https://obamawhitehouse.archives.gov/blog/2015/02/06/economics-big-data-and-differential-pricing.

Gittins JC (1989) *Multi-Armed Bandit Allocation Indices*, 1st ed. (John Wiley & Sons, Chichester, UK).

Gittins JC, Glazebrook K, Weber R (2011) *Multi-Armed Bandit Allocation Indices*, 2nd ed. (John Wiley & Sons, New York).

Hall P, Park BU (2002) New methods for bias correction at endpoints and boundaries. *Ann. Statist.* 30(5):1460–1479.

Handel B, Misra K (2015) Robust new product pricing. *Marketing Sci.* 34(6):864–881.

Handel B, Misra K, Roberts K (2013) Robust firm pricing with panel data. *J. Econometrics* 174(2):165–185.

Hauser JR, Urban GL, Liberali G, Braun M (2009) Website morphing. *Marketing Sci.* 28(2):202–223.

Hendel I, Nevo A (2006) Measuring the implications of sales and consumer inventory behavior. *Econometrica* 74(6):1637–1673.

Hitsch G (2006) Optimal dynamic product launch and exit under demand uncertainty. *Marketing Sci.* 25(1):25–30.

Hviid M, Shaffer G (1999) Hassle costs: The achilles' heel of price-matching guarantees. *J. Econom. Management Strategy* 8(4):489–521.

Jiang Y, Shang J, Kemerer CF, Liu Y (2011) Optimizing e-tailer profits and customer savings: Pricing multistage customized online bundles. *Marketing Sci.* 30(4):737–752.

Kalyanam K (1996) Pricing decisions under demand uncertainty: A bayesian mixture model approach. *Marketing Sci.* 15(3):207–221.

Kalyanaram G, Winer RS (1995) Empirical generalizations from reference price research. *Marketing Sci.* 14(3, Part 2):G161–G169.

Karunamuni RJ, Alberts T (2005) On boundary correction in kernel density estimation. *Statist. Methodol.* 2(3):191–212.

Kleinberg R, Leighton FT (2014) The value of knowing a demand curve: Bounds on regret for on-line posted-price auctions, Working paper, Akamai Technologies, Cambridge, MA.

Kuleshov V, Precup D (2014) Algorithms for the multi-armed bandit problem Working paper, McGill University, Montréal. https://arxiv.org/abs/1402.6028.

Lai TL (1987) Adaptive treatment allocation and the multi-armed bandit problem. *Ann. Statist.* 15(3):1091–1114.

Lai TL, Robbins H (1985) Asymptotically efficient adaptive allocation rules. *Adv. Appl. Math.* 6(1):4–22.

Lei Y, Jasin S, Sinha A (2014) Near-optimal bisection search for nonparametric dynamic pricing with inventory constraint. Ross School of Business Working Paper 1252, University of Michigan, Ann Arbor. https://ssrn.com/abstract=2509425.

Lodish LM (1980) Applied dynamic pricing and production models with specific application to broadcast spot pricing. *J. Marketing Res.* 17(2):203–211.

Manski C (2005) *Social Choice with Partial Knowledge of Treatment Response* (Princeton University Press, Princeton, NJ).

Mas-Colell A, Whinston M, Green J (1995) *Microeconomic Theory* (Oxford University Press, New York).

Milnor J (1954) Games against nature. Thrall RM, Coombs CH, Davis RL, eds. *Decision Processes* (John Wiley & Sons, New York), 49–59.

Nair H (2007) Intertemporal price discrimination with forward-looking consumers: Application to the US market for console video-games. *Quant. Marketing Econom.* 5(3): 239–292.

Nair H, Chintagunta P, Dube J-P (2004) Empirical analysis of indirect network effects in the market for personal digital assistants. *Quant. Marketing Econom.* 2(1):23–58.

Oren SS, Smith SA, Wilson RB (1982) Nonlinear pricing in markets with interdependent demand. *Marketing Sci.* 1(3):287–313.

Rajan A, Steinberg R, Steinberg R (1992) Dynamic pricing and ordering decisions by a monopolist. *Management Sci.* 38(2): 240–262.

Rao RC, Bass FM (1985) Competition, strategy, and price dynamics: A theoretical and empirical investigation. *J. Marketing Res.* 22(3): 283–296.

Rothschild M (1974) A two-armed bandit theory of market pricing. *J. Econom. Theory* 9(2):185–202.

Schwartz EM, Bradlow ET, Fader PS (2017) Customer acquisition via display advertisements using multi-armed bandit experiments. *Marketing Sci.* 36(4):500–522.

Smith SA (1986) New product pricing in quality sensitive markets. *Marketing Sci.* 5(1):70–87.

Stoye J (2011) Axioms for minimax regret choice correspondences. *J. Econom. Theory* 146(11):2226–2251.

Sutton RS, Barto AG (1998) *Reinforcement Learning: An Introduction* (MIT Press, Cambridge, MA).

Thompson WR (1933) On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* 25(3):285–294.

Urban GL, Liberali G, MacDonald E, Bordley R, Hauser JR (2014) Morphing banner advertising. *Marketing Sci.* 33(1):27–46.

Wald A (1950) *Statistical Decision Functions* (John Wiley & Sons, New York).

Wang Z, Hu M (2014) Committed versus contingent pricing under competition. *Production Oper. Management* 23(11):1919–1936.

Wernerfelt B (1986) A special case of dynamic pricing policy. *Management Sci.* 32(12):1562–1566.

Whittle P (1980) Multi-armed bandits and the Gittins index. *J. Roy. Statist. Soc. Ser. B* 42(2):143–149.

Winer RS (1986) A reference price model of brand choice for frequently purchased products. *J. Consumer Res.* 13(2):250–256.