Pricing a Low-regret Seller

Hoda Heidari HODA@CIS.UPENN.EDU
Mohammad Mahdian MAHDIAN@GOOGLE.COM
Umar Syed USYED@GOOGLE.COM
Sergei Vassilvitskii SERGEIV@GOOGLE.COM
Sadra Yazdanbod YAZDANBOD@GATECH.EDU

Abstract

As the number of ad exchanges has grown, publishers have turned to low regret learning algorithms to decide which exchange offers the best price for their inventory. This in turn opens the following question for the exchange: how to set prices to attract as many sellers as possible and maximize revenue. In this work we formulate this precisely as a learning problem, and present algorithms showing that by simply knowing that the counterparty is using a low regret algorithm is enough for the exchange to have its own low regret learning algorithm to find the optimal price.

1. Introduction

A display ad exchange (e.g. DoubleClick, AdECN, and AppNexus) is a platform that facilitates buying and selling of display advertising inventory connecting multiple publishers and advertisers. Publishers can select an exchange to serve an impression each time a user visits one of their websites. Upon receiving an ad slot, the exchange sells it to one of their advertisers—often by running an auction among real-time bidding agents—and pays the publisher an amount based on the revenue generated from the ad.

With the recent growth in the number of ad exchanges, one important decision a publisher has to make is which one of these exchanges to enlist in order to sell their inventory for the highest price. Unlike traditional settings where prices are posted, in display advertising the publisher cannot simply observe the offered prices in advance and choose the highest paying exchange. There are multiple reasons behind this constraint: First, on the exchange side each price check often involves running an auction and allocating the impression to the winner. As the result the publisher cannot send the same item to multiple exchanges at the same time. This combined with the fact that there is very limited time—on the order of a few milliseconds—to serve an ad to the user, forces the publisher to commit to using a particular exchange before observing the prices.

Given that prices cannot be observed in advance, in order to pick the highest paying exchange publishers have to rely on experimentation, utilizing different exchanges and seeing the payoffs realized from each over time. In recent years great progress has been made to automate decision making processes in such settings. The so called *bandit algorithms* automatically explore between the multitude of available options (here exchanges) and exploit the most profitable ones. These algorithms are easy to implement, are incredibly practical, and come with strong theoretical guarantees on the *regret* of the operator (here publisher). Therefore, from the point of view of the publisher, the situation is largely resolved.

From the point of view of an exchange, however, it is far from clear what strategy it must employ to maximize revenue. In an ideal world the exchange could look at the prices offered to the publisher by its competitors, and set the offering price ever so slightly higher. Any strategic publisher (e.g. one minimizing regret) would then shift their inventory towards this exchange, rewarding them for the higher prices. In practice, however, these prices are not publicly announced and there is no easy way to discover them. For instance, because of cookie based targeting, it is not possible for the exchange to simply find a 'similar' impression on one of the competing platforms and check its price. Given that the exchange cannot observe the competing prices directly, the only way to infer and react to them is through the actions of the publisher.

Faced with a publisher who selects among exchanges using a no-regret algorithm, the operator of an exchange must carefully decide what prices to offer. If the prices are too low, the publisher will never select the exchange, and if the prices are too high, the exchange is overpaying. Our goal in this work is to design a no regret pricing algorithm for the exchange.

Proceedings of the 33^{rd} International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W&CP volume 48. Copyright 2016 by the author(s).

We assume the prices offered by the competitors is drawn from an unknown distribution. As we will see in Section 5 this assumption is required for the existence of a no regret pricing algorithm. Furthermore, we believe that this assumption is in fact realistic: given that each exchange has a large number of competitors, the response of an individual exchange will not have a significant impact on the aggregate distribution of the competing prices (this is similar to the reasoning behind *mean-field equilibria*).

The first solution that comes to mind is to discretize the price space and run an off-the-shelf no regret algorithm in order to find the best price. As we will show in Section 7 this approach does not solve the problem. The main result of the current paper is a binary-search pricing algorithm that guarantees the exchange pays only a little more than the best price offered by its competitors, even though it never observes these prices directly (Section 3,4).

1.1. Related Work

We study a setting in which a seller repeatedly interacts with a group of buyers, deciding at each time step which buyer to sell the next item to. In this setting a natural choice for the seller is to employ low regret bandit learning algorithms (Lai & Robbins, 1985; Auer et al., 2002; 2003). Bandit algorithms are a popular solution to sequential decision making problems, as they require only limited feedback and have low regret, i.e., they guarantee performance comparable to the best single action in hindsight. While some of the earliest bandit algorithms were index-based (Lai & Robbins, 1985; Auer et al., 2002), the EXP family of algorithms (Auer et al., 2003) are designed for bandit problems where the feedback is generated arbitrarily, rather than stochastically. Bayesian bandit algorithms based on Thompson sampling (Thompson, 1933) have also been very successful empirically (Graepel et al., 2010; Chapelle & Li, 2011).

The focus of the present paper is to design a no-regret pricing scheme for a buyer who interacts with a strategic seller over multiple time periods. The most closely related to our work are the results in (Amin et al., 2013; 2014). These papers study a repeated posted-price auction setting consisting of a single strategic buyer and a price-setting seller. The main results in (Amin et al., 2013; 2014) are pricing algorithms for the seller that guarantee no regret if the buyer's discounting factor is small. Compared to our work, Amin et al. define regret with respect to different benchmarks. Also in contrast to our model, they assume buyer's valuation is subject to time discounting, with non-trivial regret achievable only when the discount rate is strictly less than 1.

Our work is also related to the broad literature on repeated auctions, where an auctioneer interacts with buyers and sellers over multiple time steps. Repeated auctions have been studied extensively and from various angles (Bikhchandan, 1988; Thomas, 1996; Chouinard, 2006). Both empirical (Edelman & Ostrovsky, 2007) and anecdotal evidence have suggested that in repeated auctions agents use sophisticated algorithms to induce better payoffs for themselves in the future. Indeed a growing part of the literature has been dedicated to designing various strategies and algorithms to improve the future payoff (Jofre-Bonet & Pesendorfer, 2000; Kitts & Leblanc, 2004; Kitts et al., 2005; Cary et al., 2007; Lucier, 2009; Gummadi et al., 2012). Our work is in particular concerned with the study of pricing in repeated auctions. Some of the previous papers on this topic are (Bar-Yossef et al., 2002; Kleinberg & Leighton, 2003; Blum et al., 2003; Cesa-Bianchi et al., 2013; Medina & Mohri, 2014). These papers mostly consider a simplified setting, focus on the buyer (and not the seller) side, and assume the buyer behaves in a naive manner. Our work is also related to the study intertemporal price discrimination, i.e. conditioning the price on buyer's past behavior in a repeated auction. Previous work, for instance (Acquisti & Varian, 2005; Kanoria & Nazerzadeh, 2014) examine the conditions under which it is profitable to engage in this form of pricing.

Finally, we remark that the current paper adds to the growing line of research in algorithmic game theory investigating the outcome of games in which players employ some form of no-regret learning (Roughgarden, 2012; Syrgkanis & Tardos, 2013; Nekipelov et al., 2015). As opposed to classic economics where players are assumed to have reached an equilibrium, this recent body of work relies on the weaker assumption that players utilize no regret learning to learn from their past observations and adjust their strategies. This idea is compelling especially in online settings, such as the one studied in this work, where players repeatedly interact with one another in a complex and dynamic environment. Our work presents an algorithmic *no-regret response* against a no-regret opponent in an auction environment.

2. Model

We consider a setting where a seller repeatedly interacts with a group of price-setting buyers, deciding at each time step which buyer to sell the next item to. In the context of display advertising, sellers and buyers correspond to publishers and ad exchanges, respectively; each time step represents an instance where a user visits the publisher's website and gives the publisher an advertising opportunity to sell at any of the advertising exchanges. In practice, an ad exchange is often an intermediary who runs an auction among advertisers to allocate the ad, and then determines how much to pay the publisher (typically based on the revenue generated from this auction). Nonetheless, by modeling the exchange as a buyer we implicitly assume it has full control over how much the publisher (seller) is paid. We argue that this assumption is practical for multiple reasons: First, the amount the exchange pays the publisher does not have to be tied to the amount it receives from the advertisers[1]. Second, even if two are closely related, e.g. if the exchange decides to pay the publishers a fixed percentage of the revenue, it only needs to satisfy this constraint in sum across all impressions[2]. Third, the exchange has full control over the reserve price, which in practice often directly affects the auction revenue.

Consider a seller selling one unit of an identical good at each time step to a group of price-setting buyers. Time is assumed to be discrete and indexed by positive integers. We study the pricing problem from the perspective of a buyer interested in this good. At each time step, the seller must select whether to sell the good to us, or to one of the *outside options*. If the seller does not select us, which outside option it chooses does not affect our revenue. Therefore, without loss of generality, we represent the outside option with a single buyer. Let us denote the price offered by us (A) and by the outside option (B) for the good at time t by p_t^A and p_t^B, respectively. We assume at each time step the seller must select between A and B *before* seeing these prices. Once it picks a buyer, the seller will observe the price offered by that buyer. Note that while this would be an odd assumption in standard marketplaces, as noted earlier in the case of the online advertising market, it is standard practice: First, the publisher cannot send the same item to multiple exchanges at the same time. Second, once a user requests a page on the publisher's website, they must quickly be served an ad, therefore the publisher simply does not have enough time to check prices at multiple exchanges.

Since the seller cannot see the prices before selecting which buyer to choose, it employs a low-regret strategy to select the buyer that over time gives her a higher price. The *regret* of the seller up to time T is defined as

$$R(T) = \max\{\sum_{t=1}^{T} p_t^A, \sum_{t=1}^{T} p_t^B\} - \sum_{t=1}^{T} p_t^{X_t},$$

where $X_t \in \{A, B\}$ is the buyer chosen by the seller in time step t. We assume the seller uses a (possibly randomized) low-regret[3] algorithm to pick X_t's. We need to be careful about the definition of *low regret* here: in our setting we need the regret to be bounded not just in expectation, but with high probability. We follow the definition in (Bubeck & Cesa-Bianchi, 2012), and assume the seller's strategy satisfies the following: for every $\delta > 0$, with probability at least $1 - \delta$, seller's regret up to time T is

$$R(T) < cT^{\gamma} \log(\delta^{-1}), \tag{1}$$

where c and $\gamma < 1$ are constants (independent of T and δ). The standard adversarial multi-armed bandits algorithms (Bubeck & Cesa-Bianchi, 2012) satisfy the above bound with any $\gamma > \frac{1}{2}$.[4]

The pricing problem can be defined as follows: at each time step t, we (as a buyer) would like to set a price p_t^A. All we can observe at the end of each round is the actions of the seller, i.e. whether we are selected or not. Note that in practice we cannot directly observe when the publisher chooses our opponent (exchange A does not get a call every time the publisher sends an impression to exchange B), nonetheless it is relatively easy for exchange A to know the approximate amount of traffic the seller sends to other exchanges. This can be done with either estimating the overall traffic the publisher receives, or by randomly monitoring the publisher's website and observing the fraction of times the ads on the page are served by exchange A.

We assume the price of the outside option p_t^B is drawn i.i.d. from an unknown distribution $\mathcal{D}$ with mean $\mu \in [0, 1]$. Note that in large market places it is a common practice (see for example the literature on mean field equilibrium) to assume each player treats other players' strategies as sampled from a fixed distribution. Also as we will see in Section 5 the assumption that p_t^B's are drawn stochastically is necessary for the existence of a low regret pricing algorithm. We don't get to observe our competitor's prices.

Let v be our value for each unit of the good (v can be thought of as the value we can get from the advertisers in our exchange for an advertising opportunity on this publisher). For simplicity, we treat v as a constant value, but our results generalize to the case that v is a random variable drawn i.i.d. from a distribution.[5] A clairvoyant algorithm that knows μ can simply offer a constant price slightly higher than μ. At this price, the seller almost always selects us. So, if we value the good at $v > \mu$, the total utility earned by the clairvoyant algorithm after T rounds

[1] Of course, the amount collected from the advertisers determines the "value" that the exchange has for receiving the ad slot, but this will be captured in our model by the parameter v, the buyer's value for the good.

[2] Specifically, the exchange can take on the arbitrage risk, by promising the publisher a minimum price, and recouping the cost later if needed.

[3] Or sublinear regret.

[4] More precisely, the EXP3.P algorithm satisfies the regret bound with $\gamma = \frac{1}{2}$ and an additional polylog term on the right hand side.

[5] Note that we are making the assumption that v is drawn each time independently of other draws of v or other random variables in the model. In particular, v has to be independent of the price of the outside option. This assumption is realistic when the set of goods that are offered for sale are homogeneous, e.g., ad slots on a single web page on the publisher's website.

is asymptotically $(v - \mu)T$. Our objective is to get a total utility close to this quantity without knowing μ.

The loss of any pricing algorithm can be decomposed into two components: number of times we are not selected by the seller when we employ that algorithm, and the "extra" payment (i.e., amount of payment over μ) we pay the seller during the rounds we are selected. More formally, let's define

$$\textbf{not-selected} = \sum_{t=1}^{T} \mathbf{1}[X_t = B],$$

$$\textbf{extra-payment} = \sum_{t=1}^{T} \mathbf{1}[X_t = A](p_t^A - \mu).$$

The expected *regret* of the algorithm can be written as:

$$\textbf{not-selected} \cdot (v - \mu) + \textbf{extra-payment}. \qquad (2)$$

Our objective is to set the prices p_t^A in such a way that both terms in the above expression are sublinear ($o(T)$). Our main result is an algorithm that achieves a bound of $\tilde{O}(T^{\frac{1+\gamma}{2}})$ for these regret terms, where $\gamma < 1$ is the exponent in the regret bound (1) of the seller.

3. Algorithm

The idea behind our algorithm is simple: note that if we offer a constant price, the lowest price at which the seller still chooses us over the outside option without incurring linear regret is μ. We run a binary search to estimate this value. The subtlety here is that since the seller does not see the prices and is allowed some regret, we need to repeat offering the same price a number of times to accurately decide whether the price is too high or too low. Furthermore, if the price we offer is too close to μ, the seller can essentially choose arbitrarily without violating the regret bound. Therefore, the binary search will need to allow for some margin of error.

For simplicity, we assume the total number of rounds T is known, and we prove that at the end of the T rounds, our regret is bounded. Our proposed algorithm is described in Algorithm 1. The algorithm uses the function $f(k)$ and constant θ that will be fixed during the analysis. Also the variable t in the algorithm is only for bookkeeping purposes.

4. Analysis

The main result of this section is the following:

Theorem 1 *Consider a run of Algorithm 1 for T steps, and assume the seller follows a strategy that satisfies the regret bound (1). Then, with probability at least $1 - O(\frac{\log T}{T})$, both the number of times we are not selected by the seller and the extra payment to the seller are bounded by*

$$O\left(T^{\frac{1+\gamma}{2}} \log T\right).$$

Algorithm 1 Binary Search Pricing Algorithm

```
 1: $l_0 \leftarrow 0, u_0 \leftarrow 1, k \leftarrow 0, t \leftarrow 0$
 2: while $u_k - l_k > T^{-\theta}$ do
 3:     $p_k \leftarrow (l_k + u_k)/2$
 4:     Offer the seller a price of $p_k$ for $f(k)$ rounds
 5:     $x \leftarrow$ # of times the seller accepts the price of $p_k$.
 6:     $l_{k+1} \leftarrow l_k, u_{k+1} \leftarrow u_k$
 7:     if $x > f(k)/2$ then
 8:         $l_{k+1} = (2l_k + u_k)/3$
 9:     else
10:         $u_{k+1} = (l_k + 2u_k)/3$
11:     end if
12:     $t \leftarrow t + f(k)$
13:     $k \leftarrow k + 1$
14: end while
15: Offer a price of $u_k + T^{-\theta}$ for the remaining rounds.
```

Proof We start with a few notations. We call the steps during the binary search while loop (lines 2–14 of Algorithm 1) the *exploration phase*, and the steps after this loop (line 1) the *exploitation phase*. The k'th iteration of the exploration while loop (with k starting from 0) is called the *k'th exploration phase*, or simply *phase k*.

Since the length of the interval $u_k - l_k$ decreases by a factor of $2/3$ in each phase, the number of phases of the algorithm is at most $O(\log T)$. Therefore, using the regret bound (1) with $\delta = 1/T$ and the union bound, we know that with probability at least $1 - O(\frac{\log T}{T})$, at the end of every phase (both exploration phases and the exploitation phase), we have

$$R(t) < ct^{\gamma} \log(T). \qquad (3)$$

Throughout the rest of the proof, we assume the above event happens, and prove that the desired bounds on the regret of our algorithm follow from this.

The argument is in two steps. First, we show that if the function $f(k)$ is properly chosen, with high probability, the algorithm maintains the invariant that the value of μ lies in the interval $[l_k, u_k]$. In particular, this means that at the end of the exploration phases, the value of μ is at most u_k and is at least $l_k \geq u_k - T^{-\theta}$. This implies that in each of the steps in the exploitation phase, either the seller gets an expected regret of at least $T^{-\theta}$ by not accepting the price of $u_k + T^{-\theta}$, or she accepts and we make an extra payment that is at most $2T^{-\theta}$. The second step is to use this fact to bound the total regret of the algorithm.

We prove the invariant $\mu \in [l_k, u_k]$ by induction. Consider a phase k, and assume $\mu \in [l_k, u_k]$. We show that the probability that this property does not hold in the subsequent phase is small. To do this, we bound the regret of the seller in this phase, and show that if the algorithm makes the wrong decision about l_{k+1} or u_{k+1} in this phase, seller's regret must be too high.

First, consider the case that $\mu > (l_k + 2u_k)/3$. We show that in this case, with high probability the seller accepts the price p_k less than $f(k)/2$ times. Let x denote the number of times that the seller accepts the price p_k during this phase. Note that x is a random variable and can depend on the draws of the price of the outside option as well as the internal random bits of the seller's algorithm. We compare the expected total price the seller pays during phases 0 through k with the expected total price she would have gotten had she always picked the outside option. The latter value is simply $\sum_{i=1}^{k} f(i)\mu$. The total price the seller gets during phase k can be computed as follows: In x steps during this phase, the seller gets a price of p_k. In each of the remaining $(f(k) - x)$ steps, the seller gets a price that is drawn from a distribution with mean μ. We define a martingale $0 = Y_0, Y_1, Y_2, \ldots, Y_{f(k)}$ based on this process as follows: For each i, if the seller selects us in step i of phase k, we let $Y_i = Y_{i-1}$. Otherwise, we let Y_i be Y_{i-1} plus the price of the outside option in step i minus μ. Note that this is in fact a martingale. The total price of the outside option during this phase is precisely $Y_{f(k)} + (f(k) - x)\mu$. Therefore, the total price that the seller receives during this phase is

$$xp_k + (f(k)-x)\mu + Y_{f(k)} \le f(k)\mu - x \cdot \frac{u_k - l_k}{6} + Y_{f(k)}.$$

For each step in phase i ($0 \le i \le k-1$), the expected price the seller gets is at most $\max(\mu, p_i)$. Therefore, the expected total price during these phases is at most

$$\begin{aligned}\sum_{i=0}^{k-1} f(i)\max(\mu, p_i) &= \sum_{i=0}^{k-1} f(i)\mu + \sum_{i=0}^{k-1} f(i)\max(0, \mu - p_i)\\ &\ge \sum_{i=0}^{k-1} f(i)\mu + \sum_{i=0}^{k-1} f(i)\cdot\frac{u_i - l_i}{2}\end{aligned}$$

Therefore, the difference between the total price the seller gets and the price she would have gotten had she always picked the outside option is at least

$$x \cdot \frac{u_k - l_k}{6} - \sum_{i=0}^{k-1} f(i) \cdot \frac{u_i - l_i}{2} - Y_{f(k)}$$

The value of $u_i - l_i$ decreases by a factor of $2/3$ in each phase. Therefore, if $x > f(k)/2$, the regret of the seller is at least:

$$\text{Regret} \ge \frac{1}{12}(2/3)^k f(k) - \frac{1}{2}\sum_{i=0}^{k-1}(2/3)^i f(i) - Y_{f(k)}. \tag{4}$$

This means that if we select $f(k)$ in such a way that the above value is more than the regret bound (1), the above event cannot happen, and therefore, the algorithm makes the right choice and maintains the property that $\mu \in [l_k, u_k]$.

First, we use martingale inequalities to bound the term $Y_{f(k)}$. Using Azuma's inequality and the fact that prices are bounded by 1, the probability that $Y_{f(k)} > \epsilon(2/3)^k f(k)$ is at most $2\exp(-O(\epsilon^2(2/3)^{2k} f(k)))$. In this case, the regret of the seller is at least $(\frac{1}{12} - \epsilon)(2/3)^k f(k) - \frac{1}{2}\sum_{i=0}^{k-1}(2/3)^i f(i)$. We need to set $f(k)$ in such a way that this value is larger than the regret bound of the seller.

Assume $f(k)$ is of the form $f(k) = \alpha\beta^k$ for values $\alpha > 0$ and $\beta > 1$ that will be fixed later. The lower bound (4) on the regret of the seller can be written as

$$\begin{aligned}\text{Regret} &\ge \frac{\alpha}{12}(1-\epsilon)(\frac{2\beta}{3})^k - \frac{\alpha}{2}\sum_{i=0}^{k-1}(\frac{2\beta}{3})^i\\ &= \frac{\alpha}{12}(1-\epsilon)(\frac{2\beta}{3})^k - \frac{\alpha}{2}\cdot\frac{(\frac{2\beta}{3})^k - 1}{\frac{2\beta}{3} - 1}. \end{aligned}\tag{5}$$

On the other hand, since the value of t at the end of the k'th phase is $\sum_{i=0}^{k} f(i)$, the upper bound (3) on the regret can be written as

$$\begin{aligned}\text{Regret} &< c\log(T)\left(\sum_{i=0}^{k} f(i)\right)^{\gamma}\\ &= c\alpha^{\gamma}\log(T)\left(\frac{\beta^k - 1}{\beta - 1}\right)^{\gamma}.\end{aligned}\tag{6}$$

If we pick $\alpha = \left(\frac{c\log(T)}{\lambda}\right)^{\frac{1}{1-\gamma}}$ for another constant λ that will be fixed later, we would have

$$c\alpha^{\gamma}\log(T) = \lambda\left(\frac{c\log(T)}{\lambda}\right)^{1+\frac{\gamma}{1-\gamma}} = \lambda\alpha.$$

Therefore, after combining lower and upper bounds (5) and (6), we can cancel α from both sides of the inequality and obtain:

$$\frac{1-\epsilon}{12}(\frac{2\beta}{3})^k - \frac{1}{2}\cdot\frac{(\frac{2\beta}{3})^k - 1}{\frac{2\beta}{3} - 1} < \lambda\left(\frac{\beta^k - 1}{\beta - 1}\right)^{\gamma}$$

Assuming $\beta > \frac{3}{2}$, the above inequality implies

$$\left(\frac{1-\epsilon}{12} - \frac{1}{2(\frac{2\beta}{3} - 1)}\right)(\frac{2\beta}{3})^k < \lambda\frac{\beta^{\gamma k}}{(\beta - 1)^{\gamma}} \tag{7}$$

We now fix the value of β to $\beta = (\frac{3}{2})^{\frac{1}{1-\gamma}}$. Note that this value satisfies the assumption $\beta > 3/2$. We have:

$$\frac{2\beta}{3} = (\frac{3}{2})^{\frac{1}{1-\gamma} - 1} = \beta^{\gamma}.$$

Therefore, inequality (7) reduces to

$$\lambda > \left(\frac{1-\epsilon}{12} - \frac{1}{2(\frac{2\beta}{3} - 1)}\right)(\beta - 1)^{\gamma}.$$

This means that if we pick the value of λ to be the expression on the right-hand side of the above inequality, inequality (7) leads to a contradiction. Thus, with probability at least $1-O(\frac{\log T}{T})-2\sum_k \exp(-O(\epsilon^2(2/3)^{2k}f(k)))$, the event "$\mu > (l_k+2u_k)/3$ but $x > f(k)/2$" does not happen in any phase k. An almost identical proof shows that the event "$\mu < (2l_k+u_k)/3$ but $x < f(k)/2$" does not happen in these cases either. If these events happen, the algorithm maintains the invariant that $\mu \in [l_k, u_k]$ throughout the exploration steps. The probability that this is violated is at most

$$O(\frac{\log T}{T}) + 2\sum_k \exp(-O(\epsilon^2 \alpha (\frac{4\beta}{9})^k)).$$

It is not hard to see that with the above choice of the values of α and β, the above expression tends to zero as T tends to infinity.

Given this invariant, in each of the steps in the exploitation phase (line 1), either the seller incurs a regret of at least $T^{-\theta}$ by not accepting the price of $u_k+T^{-\theta}$, or she accepts and we get a regret of at most $2T^{-\theta}$. Let y denote the number of times we are not selected by the seller during the exploitation phase. We bound the total regret of the seller compared to the strategy that always selects us using a method similar to the first part of the proof. Since $\mu \in [l_i, u_i]$ for every i, in each step during phase i, the price of the option selected by the seller is at most u_i, i.e., at most $\frac{u_i-l_i}{2} = \frac{1}{2}(2/3)^i$ higher than our price . In each of the y steps that the seller chooses the outside option during the exploitation phase, her regret is at least $T^{-\theta}$. Therefore, the total regret of the seller is at least

$$yT^{-\theta} - \frac{1}{2}\sum_{i=0}^{k^*-1}(2/3)^i f(i),$$

where k^* is the value of k at the end of the algorithm. Using the regret bound for the seller at the end of the T steps, we get the following inequality:

$$yT^{-\theta} - \frac{1}{2}\sum_{i=0}^{k^*-1}(2/3)^i f(i) < c\log(T)T^{\gamma}.$$

Replacing $f(i) = \alpha\beta^i$, we obtain:

$$yT^{-\theta} < \frac{\alpha}{2}\left(\frac{2\beta}{3}-1\right)^{-1}(\frac{2\beta}{3})^{k^*} + c\log(T)T^{\gamma}$$

Since $u_k - l_k = (2/3)^k$, we have $k^* = \log(T^{-\theta})/\log(2/3)$. Therefore,

$$(\frac{2\beta}{3})^{k^*} = (\frac{3}{2})^{\frac{\gamma k^*}{1-\gamma}} = T^{\frac{\theta\gamma}{1-\gamma}}$$

Therefore,

$$y < \frac{\alpha}{2}\left(\frac{2\beta}{3}-1\right)^{-1} T^{\theta+\frac{\theta\gamma}{1-\gamma}} + c\log(T)T^{\gamma+\theta}$$

Furthermore, the total length of the exploration phases is $\alpha\sum_{i=0}^{k^*-1}\beta^i < \frac{\alpha}{\beta-1}T^{\frac{\theta}{1-\gamma}}$. Therefore, even assuming that the seller never chooses us during the exploration phase, the total number of times the seller does not chose us can be written as

$$\frac{\alpha}{\beta-1}T^{\frac{\theta}{1-\gamma}} + \frac{\alpha}{2}\left(\frac{2\beta}{3}-1\right)^{-1}T^{\frac{\theta}{1-\gamma}} + c\log(T)T^{\gamma+\theta}.$$

Since β is a constant and $\alpha = O((\log T)^{1/(1-\gamma)})$, the above expression is at most

$$O(\log(T)T^{\max(\frac{\theta}{1-\gamma},\gamma+\theta)}). \tag{8}$$

Finally, we bound the amount of extra payment (i.e., payment beyond μ) made to the seller. By the invariant $\mu \in [l_i, u_i]$, we know that in each round in the i'th exploration phase, this extra payment is at most $\frac{1}{2}(u_i - l_i)$. Also, during the exploitation phase, the extra payment is at most $2T^{-\theta}$ per round. Therefore, the total extra payment made to the seller can be bounded by

$$\frac{1}{2}\sum_{i=0}^{k^*-1}(2/3)^i f(i) + 2T^{-\theta}\cdot T = O(\alpha T^{\frac{\theta\gamma}{1-\gamma}} + T^{1-\theta}). \tag{9}$$

Now, if we select $\theta = \frac{1-\gamma}{2}$, both expressions (8) and (9) will be at most $O(\log(T)T^{\frac{1+\gamma}{2}})$. ■

5. Extensions

Here, we discuss some of the assumptions we made in our model. In particular, we sketch how the assumptions that the number of rounds T is known and that μ should be in $[0,1]$ can be relaxed. We also show that the assumption that the outside option is stochastic is necessary.

Unknown number of rounds The assumption that the number of rounds T is known can be relaxed using a standard "doubling" trick. The main observation is that Theorem 1 holds even if the number of rounds turns out to be not precisely T but a constant multiple of T. Therefore, we can start running the algorithm with a small value of T as an estimate for the number of rounds, and each time we discover that the actual number of rounds is more than the current estimate, we multiply the estimate by a constant and restart the algorithm from scratch. It is not hard to show that this algorithm satisfies the same regret bounds (with larger constants hidden in the $O(\cdot)$ notation).

Range of μ The assumption that the mean μ of the outside is between 0 and 1 can be relaxed by adding an initial "doubling" stage to the binary search algorithm to find an upper bound M on μ. A term containing the value the upper bound M will be added to the regret of the algorithm.

Arbitrary buyer values If our value for the good offered by the seller is v, the expression (2) gives the value of our regret, *assuming* $v > \mu$. This assumption can be relaxed with a simple modification of Algorithm 1 that caps the offered price at v. The proof is straightforward and is omitted due to space constraints.

Non-stochastic outside option Since we offer prices based on the observed behavior of the seller, it is reasonable to ask why we assume that the prices offered by the outside option are drawn i.i.d. from a fixed distribution $\mathcal{D}$. Consider an alternate model where the outside option can offer arbitrary prices, and the goal is for our expected total utility to asymptotically approach $(v - \mu_T) \cdot T$, where $\mu_T = E\left[\frac{1}{T}\sum_{t=1}^{T} p_t^B\right]$. Unfortunately, allowing the outside option this much flexibility makes our goal impossible.

To see this, consider an outside option that simulates our algorithm and offers identical prices, so that the distribution of p_t^A and p_t^B are the same (note that the outside option can also observe the seller's behavior, so this simulation is feasible). Clearly one way for the seller to ensure that her regret $R(T) = 0$ is to select between us and the outside option via an independent coin toss in every round. However, in this case our expected total utility will be

$$\begin{aligned} E\left[\sum_{t=1}^{T} \mathbf{1}[X_t = A] \cdot (v - p_t^A)\right] &= \sum_{t=1}^{T} \frac{1}{2} \cdot E[(v - p_t^A)] \\ &= \sum_{t=1}^{T} \frac{1}{2} \cdot E[(v - p_t^B)] \\ &= \frac{1}{2}(v - \mu_T) \cdot T, \end{aligned}$$

and thus the difference between $(v - \mu_T) \cdot T$ and our total utility is linear in T, disallowing the possibility that any pricing algorithm can have low regret. One potential approach to get around this impossibility result is to assume more information about the particular no-regret algorithm the seller is using. We leave the analysis of this alternative model as an interesting direction for future work.

6. A Heuristic Algorithm

The idea behind Algorithm 1 was to zero in on the smallest price the seller is willing to sell her goods for. To do this, we maintained the invariant that the target price is always within a shrinking interval around the price we offered. Maintaining this invariant made it possible to theoretically analyze the regret of the algorithm: we could use a simple union bound to handle the highly-correlated error events, and get around the complexity arising from the sequential stochastic nature of the errors. This invariant, however, came at a cost: we needed to offer the same price many times to ensure that the average response of the seller gives us a reliable signal about the target price, and make the decision about the next step based on this reliable signal. An alternative approach is to forgo the invariant, and adjust the price based on signals that are unreliable on their own right, but stochastically lead us in the right direction. This is what Algorithm 2 does.

There are a few subtleties in the process of updating prices in Algorithm 2: To ensure that the prices eventually get closer to the target price we need to update them in a way that the changes become smaller and smaller as time goes on. To do this, we update the prices by multiplying or dividing the current price by a time-dependent factor. Note that to ensure our price remains above the target price significantly more often than below it, we need to use different factors for multiplication and division. So every time the price is rejected we multiply it by a factor of $(1 + t^{-\alpha})$ for some $0 < \alpha < 1$, and when it is accepted, we divide it by a *smaller* factor $(1 + t^{-\beta})$ (i.e., $\beta > \alpha$). Aside from this, we leave it to the simulation to determine the best values for the parameters α and β.

Algorithm 2 Heuristic Pricing Algorithm

```
$t \leftarrow 0, p_t \leftarrow \frac{1}{2}$
while true do
    Offer the seller a price of $p_t$
    if the seller rejects then
        $p_{t+1} = (1 + t^{-\alpha})p_t$
    else
        $p_{t+1} = (1 + t^{-\beta})^{-1}p_t$
    end if
    $t \leftarrow t + 1$
end while
```

While Algorithm 2 is simple and natural, and as we will see in Section 7 performs well in practice, the fact that the sequence of errors it generates is correlated makes it difficult to analyze its performance theoretically. In the next section we evaluate the performance of the algorithm via simulations, and leave its theoretical analysis for future work.

7. Simulations

In this section we empirically evaluate the performance of Algorithm 1 and 2, and compare them with a baseline.

Baseline We compare our algorithms with a naive baseline that works as follows: Given parameters $0 < \epsilon < 1$, it discretizes the price space (i.e. [0,1]) into $\frac{1}{\epsilon}$ equally spaced prices and treats each of these prices as an arm. When the algorithm offers the price p_i the seller, the reward from the corresponding arm is equal to p_i if the seller chooses our

Table 1. Regret values after $T = 10^6$ steps

ALGORITHM	NOT SELECTED	EXTRA PAYMENT	REGRET
ALGORITHM 1	61110	32040	74817
ALGORITHM 2	8585	9227	15236
BASELINE	840149	-908	587196

buyer, and is 0 otherwise. The baseline simply runs the algorithm EXP3.P (see (Bubeck & Cesa-Bianchi, 2012)). Note that from a theoretical stand-point we don't expect this algorithm to perform well for the following reason: Given that the seller is playing a no-regret algorithm, in order for us to observe her eventual reaction to a particular price, we need to offer the same price to them multiple times, i.e. long enough for their no-regret algorithm to realize the price change and respond to it. The baseline fails to do this, and as the result we expect its regret to be high.

Simulation setup The simulation setup is as follows: we assume the price p_t^B of the outside option comes from a uniform distribution on $[0, 2\mu]$ where $\mu = 0.3$. For this and other parameters, we experimented with other values as well and did not observe any significant difference in the outcome. For the seller, we use the algorithm EXP3.P (see (Bubeck & Cesa-Bianchi, 2012)). We take $T = 10^6$ and run both Algorithms 1, 2 with a range of values for their free parameters (i.e. the function f and the value θ for Algorithm 1, and the values α and β for Algorithm 2). We track the number of rounds our exchange is not selected by the seller, the extra payment to the seller, and the overall regret. For the baseline $\epsilon = 0.001$. The regret values reported here use a value of $v = 1$ in the regret expression (2). Each simulation is repeated 100 times, and the computed values are averaged over these runs. Confidence intervals are very small, hence omitted for better readability.

Optimal setting of the parameters For Algorithm 1, we use the functional form $f(k) = a \cdot \log(T)^2 \beta^k$ (see Section 4). A grid search over the ranges $a \in [0.5, 2.5], \beta \in [1, 2.5]$, and $\theta \in [0.1, 0.3]$ reveals that the values $a = 2$, $\beta = 1.5$, and $\theta = 0.2$ result in the lowest regret. Observe that the values of β and θ are close to the values derived in the analysis. For Algorithm 2, a grid search over the range $0 < \alpha < \beta \leq 1$ finds that the combination $\alpha = 0.1$ and $\beta = 0.5$ results in the lowest regret.

Comparison of the algorithms In Table 1 we present the following quantities for each algorithm: The number of times the price is not accepted by the seller, the extra payment to the seller, and the overall regret. Figure 7 illustrates the total regret of each algorithm as a function of time in the logarithmic scale. One can see that Algorithms 1 and 2 both significantly outperform the baseline in terms of the total regret. Furthermore, the regret of Algorithms 1 and 2 are sublinear, while that of the baseline is growing linearly with time. Also, interestingly algorithm 2 incurs less regret than Algorithm 1.

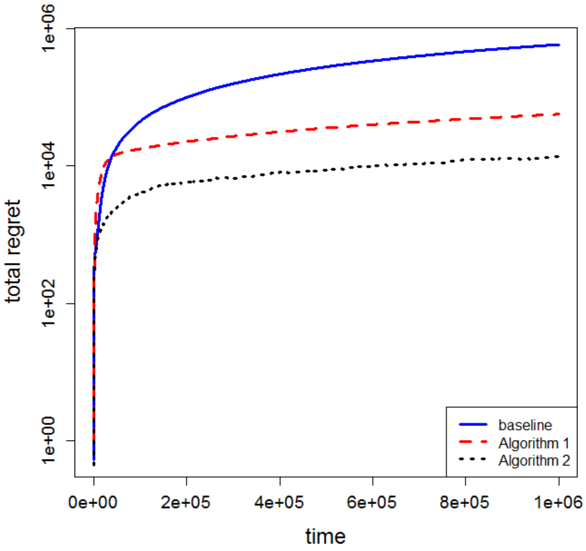

Figure 1. Regret of Algorithms 1, 2, and the baseline as a function of time.

8. Future Directions

We presented a binary search-style pricing algorithm for a buyer facing a no-regret seller. Our main contribution was the analysis of this algorithm and showing that it guarantees the buyer vanishing regret. It remains an open question whether the regret bound presented here is asymptotically tight. Furthermore, we focused on the *buyer* side of the market only and ignored the possibility of the seller responding strategically to our proposed algorithm. We leave the equilibrium analysis and the study of the seller-side implications of the algorithm for future work.

References

Acquisti, Alessandro and Varian, Hal R. Conditioning prices on purchase history. *Marketing Science*, 24(3):367–381, 2005.

Amin, Kareem, Rostamizadeh, Afshin, and Syed, Umar. Learning prices for repeated auctions with strategic buyers. In *NIPS*, 2013.

Amin, Kareem, Rostamizadeh, Afshin, and Syed, Umar. Repeated contextual auctions with strategic buyers. In *NIPS*, 2014.

Auer, Peter, Cesa-Bianchi, Nicolò, , and Fischer, Paul. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2–3):235–256, 2002.

Auer, Peter, Cesa-Bianchi, Nicolò, Freund, Yoav, and Schapire, Robert E. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2003.

Bar-Yossef, Ziv, Hildrum, Kirsten, and Wu, Felix. Incentive-compatible online auctions for digital goods. In *SODA*, pp. 964–970, 2002.

Bikhchandan, Sushil. Reputation in repeated second-price auctions. *Journal of Economic Theory*, 46(1):97–119, October 1988.

Blum, Avrim, Kumar, Vijay, Rudra, Atri, and Wu, Felix. Online learning in online auctions. In *SODA*, pp. 202–204, 2003.

Bubeck, Sébastien and Cesa-Bianchi, Nicolo. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1–22, 2012.

Cary, Matthew, Das, Aparna, Edelman, Ben, Giotis, Ioannis, Heimerl, Kurtis, Karlin, Anna R., Mathieu, Claire, and Schwarz, Michael. Greedy bidding strategies for keyword auctions. In *EC*, pp. 262–271, 2007.

Cesa-Bianchi, Nicolo, Gentile, Claudio, and Mansour, Yishay. Regret minimization for reserve prices in second-price auctions. In *SODA*, 2013.

Chapelle, Olivier and Li, Lihong. An empirical evaluation of thompson sampling. In *NIPS*, 2011.

Chouinard, Hayley H. Repeated auctions with the right of first refusal and asymmetric information. *Manuscript*, 2006.

Edelman, Benjamin and Ostrovsky, Michael. Strategic bidder behavior in sponsored search auctions. *Decision support systems*, 43(1):192–198, 2007.

Graepel, Thore, Candela, Joaquin Quinonero, Borchert, Thomas, and Herbrich, Ralf. Web-scale bayesian click-through rate prediction for sponsored search advertising in microsoft's bing search engine. In *ICML*, 2010.

Gummadi, Ramakrishna, Key, Peter, and Proutiere, Alexandre. Repeated auctions under budget constraints: Optimal bidding strategies and equilibria. In *the Eighth Ad Auction Workshop*, 2012.

Jofre-Bonet, Mireia and Pesendorfer, Martin. Bidding behavior in a repeated procurement auction: A summary. *European Economic Review*, 44:1006–1020, 2000.

Kanoria, Yash and Nazerzadeh, Hamid. Dynamic reserve prices for repeated auctions: Learning from bids. *Manuscript*, 2014.

Kitts, Brendan, Laxminarayan, Parameshvyas, LeBlanc, Benjamin, and Meech, Ryan. A formal analysis of search auctions including predictions on click fraud and bidding tactics. In *the Workshop on Sponsored Search Auctions*, 2005.

Kitts, Brenden and Leblanc, Benjamin. Optimal bidding on keyword auctions. *Electronic Markets*, 14(3):186–201, 2004.

Kleinberg, Robert and Leighton, Tom. The value of knowing a demand curve: Bounds on regret for online posted-price auctions. In *FOCS*, pp. 594–605, 2003.

Lai, T.L and Robbins, Herbert. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.

Lucier, Brendan. Beyond equilibria: Mechanisms for repeated combinatorial auctions. *Manuscript*, 2009.

Medina, Andres Munoz and Mohri, Mehryar. Learning theory and algorithms for revenue optimization in second-price auctions with reserve. In *ICML*, pp. 262–270, 2014.

Nekipelov, Denis, Syrgkanis, Vasilis, and Tardos, Eva. Econometrics for learning agents. In *EC*, 2015.

Roughgarden, Tim. The price of anarchy in games of incomplete information. In *EC*, 2012.

Syrgkanis, Vasilis and Tardos, Eva. Composable and efficient mechanisms. In *EC*, 2013.

Thomas, Charles J. Market structure and the flow of information in repeated auctions. *Working paper*, 1996.

Thompson, William R. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933.