Revenue optimization in AdExchange against strategic advertisers

Abstract

In recent years, AdExchanges have become a popular tool for selling online advertisement space. This platform, has been beneficial not only to publishers, who have drastically increased their revenue by adopting it, but also to advertisers, who can design better marketing campaigns; and, ultimately to the user, who obtains a broader selection of ads relevant to his interests. Recently, some revenue optimization algorithms for AdExchanges have been proposed. However, none of these algorithms considers the possibility of facing a strategic advertiser. In this paper, the complications arising from these interactions are studied and a new revenue optimization algorithm is proposed and analyzed. In particular, we present tight bounds on the regret of an online learner attempting to optimize his revenue.

1. Introduction

Over the past decade, online advertisement has become one of the fastest growing industries in the world. In fact, in 2013, this industry experienced a growth of 32% over the previous year (Nielsen, 2013). More recently, a new ad space selling platform has gained momentum amidst advertising companies: AdExchanges, similar to a financial exchange, sell displaying rights to advertisers by conducting real-time auctions. Several advantages over traditional online advertising are offered by AdExchanges; first of all, due to the absence of contracts, publishers are not obliged to display a minimal amount of impressions from an advertiser. Consequently, a more diverse collection of ads is seen by the user. In addition, more control on the time and location of ads is given to the advertiser, resulting in a better targeted marketing campaign. Finally, because the publishers' inventory is sold using a second-price auction with reserve, fairness in the pricing scheme is guaranteed (Vickrey, 2012; Milgrom & Weber, 1982).

The main objective of a publisher selling advertisement space is to optimize his revenue. Due to the nature of AdExchanges, this may only be done by setting an appropriate reserve price. If set too low, the optimal revenue may not be attained by the publisher. However, when set too high, the bidder will not have any incentive to bid, resulting in no revenue for the publisher. Leveraging the transaction history collected by AdExchanges, some revenue optimization algorithms have been proposed in the literature (Cesa-Bianchi et al., 2013; Ostrovsky & Schwarz, 2011). More recently, a learning theory motivated algorithm was proposed by Mohri & Medina (2014) for optimizing revenue in second-price auctions with reserve.

The aforementioned algorithms heavily rely on the assumption that the outcomes of repeated auctions are i.i.d. realizations of a random variable. However, one cannot expect the outcome of repeated auctions not to depend on the past. Indeed, with the knowledge of the fact that the publisher is using a revenue optimization algorithm, an advertiser might try to mislead the publisher by under-bidding. In fact, empirical evidence of strategic behavior by advertisers has been found by Edelman and Ostrovsky Edelman & Ostrovsky (2007). Our objective will thus be to analyze the interactions between strategic advertisers and publishers.

In practice, several auctions in AdExchanges consist of a single bidder. In this setting, second-price auctions with reserve are equivalent to posted-price auctions: a seller sets a price for a good and the buyer decides whether or not to accept it (bid higher than the reserve price). In view of this fact and in order to capture the strategic behavior of a buyer, we will analyze online-algorithms for repeated posted-price auctions. At every time t, a price p_t is offered by the seller and the buyer must decide to accept it or leave it. This scenario may be modeled as a two player repeated non-zero sum game with incomplete information where the seller's objective is to maximize his revenue while the advertiser seeks to maximize her surplus (see Section 2 for a complete explanation of the setup).

The literature on non-zero sum games is very rich, however, much of the work on this subject has focused on characterizing different types of equilibria. Moreover, our problem has a particular structure that can be exploited to design efficient revenue optimization

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

algorithms.

From the seller's perspective, this game can also be seen as a bandit problem since only the revenue (reward) of the prices offered is accessible to the seller. Kleinberg & Leighton (2003) precisely study this continuous bandit setting under the assumption of an oblivious buyer (more precisely, the authors assume that at each round the seller interacts with a different buyer). The authors present a tight $\Theta(\log\log T)$ regret bound for the scenario of a buyer having a fixed valuation and a regret bound of $O(T^{2/3})$ when facing an adversarial buyer by using an elegant reduction to a discrete bandit problem. However, when dealing with a *strategic buyer*, the usual definition of regret is no longer meaningful. Indeed, consider the following example: let the valuation of the buyer be given by $v \in [0,1]$ and assume an algorithm with sub-linear regret is used for T by the seller. A possible strategy for the buyer, knowing the seller's algorithm, would be to accept prices only if they are smaller than some small value ϵ, certain that the seller will eventually learn to offer only prices less than ϵ. If $\epsilon \ll v$, the buyer would have considerably boosted its surplus while in theory the buyer would have not incurred a large regret since in hindsight, the best fixed strategy would have been to offer price ϵ for all rounds. This, however does not seem optimal for the seller.

In view of this, we use the definition of *strategic-regret* introduced by Amin et al. (Amin et al., 2013) to study precisely this problem. The authors give upper and lower bounds for the regret of a seller facing a strategic buyer and show that the surplus of the buyer must be discounted over time in order to be able to achieve sub-linear regret (see Section 2). However, the gap between the authors' upper and lower bound is in $O(\sqrt{T})$.

Our main contribution is to give an upper bound on the strategic regret that is only a factor of $O(\log T)$ away from the lower bound given in (Amin et al., 2013). In order to do this, we present a reduction of this problem to a truthful scenario. The proposed algorithm not only outperforms that of Amin et al. but also admits a much simpler analysis than the one given by the authors.

2. Setup

The following game is to be played by a buyer and a seller. A good, for instance advertisement space, is repeatedly sold to the buyer who has a private valuation for it of $v \in [0,1]$. At each round $t \in 1,\ldots,T$, a price p_t is offered by the seller and a decision $a_t \in \{0,1\}$ is made by the buyer. a_t takes value 1 when the buyer accepts to buy at that price and 0 otherwise. At the beginning of the game, the algorithm $\mathcal{A}$ used by the seller to set prices, is announced to the buyer and the buyer strategically plays against this algorithm. The knowledge of $\mathcal{A}$ is a standard assumption in mechanism design and also matches the practice in AdExchanges.

For a value of $\gamma < 1$, define the discounted surplus of the buyer to be

$$\mathrm{Sur}(\mathcal{A}, v) = \sum_{t=1}^{T} \gamma^{t-1} a_t (v - p_t).$$

The discounting factor γ represents the preference of the buyer for acquiring the good now and not in the future. The revenue of the seller is given by

$$\mathrm{Rev}(\mathcal{A}, v) = \sum_{t=1}^{T} a_t p_t.$$

Finally, the performance of a seller's algorithm will be measured by the notion of strategic regret (Amin et al., 2013)

$$\mathrm{Reg}(\mathcal{A}, v) = Tv - \mathrm{Rev}(\mathcal{A}, v).$$

The buyer's objective will be to maximize his discounted surplus while the seller seeks to minimize his regret. Notice that because of the discounting factor γ, the buyer cannot be fully adversarial. We will be interested in designing algorithms that can attain a sub-linear regret $o(T)$.

Before presenting the main results of this paper, let us study the complications that arise from dealing with a strategic buyer. Suppose the seller attempts to learn v by performing a binary search. This would be a natural algorithm when the buyer is truthful. However, since the algorithm is known to the buyer, for $\gamma \gg 0$, it is in the best interest of the buyer to lie on the initial rounds, thereby quickly, in fact exponentially, decreasing the price offered by the seller. The seller would attain an $\Omega(T)$ regret.

The main issue with the previous approach is that a binary search algorithm is "too aggressive". Indeed, an untruthful buyer can manipulate the seller into offering prices less than $v/2$ by lying about her value only once!

The previous discussion suggests following a more conservative approach, In the next section, we discuss a natural family of conservative algorithms for this problem.

3. Monotone algorithms

In the recent work of (Amin et al., 2013) the following conservative pricing strategy is introduced: let $p_1 = 1$

Algorithm 1 Family of monotone algorithms.

Let $p_1 = 1$ and $p_t \leq p_{t-1}$ for $t = 2, \dots T$.
$t = 1$
$p = p_t$
Offer price p
while (Buyer rejects p) **and** ($t < T$) **do**
  $t = t + 1$
  $p = p_t$
  Offer price p
end while
while ($t < T$) **do**
  $t = t + 1$
  Offer price p
end while

and $\beta < 1$. If price p_t is rejected at round t, the lower price $p_{t+1} = \beta p_t$ is offered at the next round. If at any time price p_t is accepted, then this price is offered for all the remaining rounds. We will denote this algorithm by `monotone`. The motivation behind this algorithm is clear: for an appropriate choice of β, the seller can slowly decrease the prices offered, thereby pressing the buyer to reject many prices (which is not convenient for her) before obtaining a favorable price. The authors present an $O(T_\gamma \sqrt{T})$ regret bound for this algorithm, with $T_\gamma = 1/(1-\gamma)$. A more careful analysis shows that this bound can be further tightened to $O(\sqrt{T_\gamma T} + \sqrt{T})$ when the discounting factor γ is known to the seller.

Despite its sublinear regret, the `monotone` algorithm remains suboptimal for certain choices of γ. Indeed, consider a scenario with $\gamma \ll 1$. For this setting, the buyer would no longer have an incentive to lie, thus, an algorithm such as binary search would achieve logarithmic regret, while the regret achieved by the `monotone` algorithm is only guaranteed to be in $O(\sqrt{T})$.

One may argue that the `monotone` algorithm is too specific since it admits a single parameter β and that perhaps a more complex algorithm with the same monotonic idea could achieve a more favorable regret. Let us therefore analyze a generic monotone algorithm $\mathcal{A}_m$ defined by Algorithm 1.

It can be shown that any monotone algorithm achieves regret $\Omega(\sqrt{T})$ against truthful buyers and under a mild condition on the prices offered it is also true that against a strategic buyer all monotone algorithms have regret in $\Omega(\sqrt{T_\gamma T} + \sqrt{T})$. Therefore, `monotone` is in fact optimal for this class of algorithms.

4. An optimal algorithm

We will now define an algorithm that forces a buyer to be almost truthful, while keeping an aggressive search

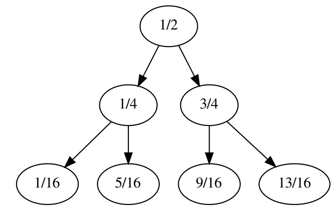

Figure 1. Tree $\mathscr{T}(3)$ associated to the algorithm proposed in (Kleinberg & Leighton, 2003).

strategy. The main idea of our algorithm is to press the buyer to stop lying by continuing to offer the price she rejects multiple times.

Let $\widetilde{\mathcal{A}}$ be an algorithm for revenue optimization used against a truthful buyer. Denote by $\mathscr{T}(T)$ the tree associated to $\widetilde{\mathcal{A}}$ after T rounds. That is, $\mathscr{T}(T)$ is a complete tree of height T with nodes $\mathfrak{n} \in \mathscr{T}(T)$ labeled with prices $p_\mathfrak{n}$ offered by $\widetilde{A}$. The right and left children of $\mathfrak{n}$ are denoted by $r(\mathfrak{n})$ and $l(\mathfrak{n})$ respectively. The price offered when $p_\mathfrak{n}$ is accepted by the buyer is the label of $r(\mathfrak{n})$ while the price offered by $\widetilde{\mathcal{A}}$ if $p_\mathfrak{n}$ is rejected is the label of $l(\mathfrak{n})$. Figure 1 depicts the tree generated by the algorithm proposed in (Kleinberg & Leighton, 2003), which is described later on.

Let $S \in \mathbb{N}$ and let $\mathcal{A}$ be defined by Algorithm 2. This algorithm mimics the behavior of $\widetilde{\mathcal{A}}$, however, when a price is rejected by the buyer, she must actually reject it for S rounds. In view of the discounting factor γ, it is not in the best interest of the buyer to lie for several rounds. Thus, if S is large enough, the buyer has a strong incentive to be truthful.

Let us properly analyze Algorithm $\mathcal{A}$. The prices offered by this algorithm define a path in $\mathscr{T}(T)$. For each node in this path, we may define time $t(\mathfrak{n})$ to be the number of rounds it takes for $\mathcal{A}$ to reach this node. Notice that since S may be greater than 1, the path chosen by $\mathcal{A}$ might not necessarily reach the leaves of $\mathscr{T}(T)$. We can also define the function $U : \mathscr{T}(T) \to \mathbb{R}_+$, whose value is the utility the buyer can get by playing an optimal strategy against $\mathcal{A}$ after node $\mathfrak{n}$ is first reached.

Lemma 1. *The function $U(\mathfrak{n})$ satisfies the following recursive relation*

$$U(\mathfrak{n}) = \max(\gamma^{t(\mathfrak{n})-1}(v - p_\mathfrak{n}) + U(r(\mathfrak{n})), U(l(\mathfrak{n}))). \quad (1)$$

The previous lemma immediately gives us conditions under which a buyer will reject a price.

Proposition 1. *For any reachable node $\mathfrak{n}$, if price $p_\mathfrak{n}$*

Algorithm 2 Definition of algorithm $\mathcal{A}$

```
n = the root of T(T)
while Number of offered prices less than T do
  Offer price p_n
  if Accepted then
    n = r(n)
  else
    Offer price p_n for S times
    n = l(n)
  end if
end while
```

is rejected by the buyer, then $v - p_{\mathfrak{n}} < \frac{\gamma^S}{(1-\gamma)(1-\gamma^S)}$.

Let us consider the following algorithm $\widetilde{\mathcal{A}}$ introduced in (Kleinberg & Leighton, 2003). The algorithm keeps track of a *feasible interval* [a,b] initialized to [0,1] and an increment parameter ϵ initialized to 1/2. The algorithm works in phases. In each phase, it offers prices $a+\epsilon, a+2\epsilon, \ldots$ until a price is rejected. If price $a+k\epsilon$ is rejected, then a new phase starts with the feasible interval set to $[a+(k-1)\epsilon, a+k\epsilon]$ and the increment parameter set to ϵ^2. This process continues until $b - a < 1/T$ at which point the price a is offered for the remaining rounds. It is evident that the number of phases needed by the algorithm is less than $\lceil \log_2 \log_2 T \rceil$, a more surprising fact is that this algorithm has been shown to achieve regret $O(\log\log T)$ when the seller faces a truthful buyer. Let us analyze the regret of our modified algorithm $\mathcal{A}$ when using the tree induced by $\widetilde{\mathcal{A}}$.

Proposition 2. *For any value of* $v \in [0,1]$ *and any* $\gamma \in (0,1)$*, the regret of algorithm* $\mathcal{A}$ *admits the following upper bound:*

$$\mathrm{Reg}(\mathcal{A}, v) \leq (vS+1)(\lceil \log_2 \log_2 T \rceil + 1) + \frac{(1+\gamma)\gamma^S T}{2(1-\gamma)(1-\gamma^S)}. \quad (2)$$

It is worth noting that for $S = 1$ and $\gamma \to 0$ the upper bound coincides with that of (Kleinberg & Leighton, 2003). When an upper bound on the discounting factor γ is known to the seller, he can leverage this information and optimize upper bound (2) with respect to the parameter S.

Theorem 1. *Let* $1/2 < \gamma < \gamma_0 < 1$ *and* $S^* = \left\lceil \operatorname{argmin}_{S\geq 1} + \frac{\gamma_0^S T}{(1-\gamma_0)(1-\gamma_0^S)} \right\rceil$. *For any* $v \in [0,1]$*, if* $T > 4$*, the regret of* $\mathcal{A}$ *satisfies*

$$\mathrm{Reg}(\mathcal{A}, v) \leq (2v\gamma_0 T_{\gamma_0} \log cT + 1 + v)(\log_2 \log_2 T + 1) + 4T_{\gamma_0},$$

where $c = 4\log 2$.

The theorem helps us define conditions under which logarithmic regret can be achieved. Indeed, if $\gamma_0 = e^{-1/\log T} = O(1 - \frac{1}{\log T})$, using the inequality $e^{-x} \leq 1 - x + x^2/2$ valid for all $x > 0$ we obtain

$$\frac{1}{1-\gamma_0} \leq \frac{\log^2 T}{2\log T - 1} \leq \log T.$$

It then follows from Theorem 1 that

$$\mathrm{Reg}(\mathcal{A}, v) \leq (2v \log T \log cT + 1 + v)(\log_2 \log_2 T + 1) + 4\log T. \quad (3)$$

Let us compare the regret bound given by Theorem 1 with the one given in (Amin et al., 2013). The above discussion shows that for certain values of γ, an exponentially better regret can be achieved by our algorithm. It can be argued that the knowledge of an upper bound on γ is required, whereas this is not needed for the `monotone` algorithm. However, if $\gamma > 1 - \frac{1}{\sqrt{T}}$ the regret bound on `monotone` is super-linear, and therefore uninformative. Thus, in order to properly compare both algorithms, we may assume that $\gamma < 1 - \frac{1}{\sqrt{T}}$ in which case, by Theorem 1 the regret of our algorithm is $O(\sqrt{T}\log T)$ whereas only linear regret can be guaranteed by the `monotone` algorithm. Even under the more favorable bound of $O(\sqrt{T_\gamma T} + \sqrt{T})$, for any $\alpha < 1$ and $\gamma < 1 - 1/T^\alpha$, the `monotone` algorithm will achieve regret $O(T^{\frac{\alpha+1}{2}})$ while a strictly better regret $O(T^\alpha \log T \log\log T)$ is attained by ours.

5. Lower bound

The following lower bounds have been given independently by Amin et. al (Amin et al., 2013) and Kleinberg and Leighton (Kleinberg & Leighton, 2003).

Theorem 2. *(Amin et al., 2013) Let* $\gamma > 0$ *be fixed. For any algorithm* $\mathcal{A}$*, there exists a valuation* v *for the buyer such that* $\mathrm{Reg}(\mathcal{A}, v) \geq \frac{1}{12} T_\gamma$.

The above theorem is in fact given for the stochastic setting where the buyer's valuation is a random variable taken from some fixed distribution $\mathcal{D}$. Nevertheless, the proof of this theorem selects $\mathcal{D}$ to be a point mass, therefore reducing the scenario to a fixed priced setting.

Theorem 3. *(Kleinberg & Leighton, 2003) Given any algorithm* $\mathcal{A}$ *to be played against a truthful buyer, there exists a value* $v \in [0,1]$ *such that* $\mathrm{Reg}(\mathcal{A}, v) \geq C \log\log T$ *for some universal constant* C.

Combining these results lead immediately to the following.

Corollary 1. *Given any algorithm $\mathcal{A}$, there exists a buyer's valuation $v \in [0,1]$ such that $Reg(\mathcal{A}, v) \geq \max\left(\frac{1}{12}T_\gamma, C \log\log T\right)$, for a universal constant C.*

We now compare the upper bounds given in the previous section with the bound of Corollary 1. For $\gamma > 1/2$, we have $\mathrm{Reg}(\mathcal{A}, v) = O(T_\gamma \log T \log\log T)$. On the other hand, for $\gamma \leq 1/2$, we may choose $r = 1$, in which case, by Proposition 2, $\mathrm{Reg}(\mathcal{A}, v) = O(\log\log T)$. Therefore, our upper and lower bounds match up to an $O(\log T)$ term.

6. Conclusion and further research

We have presented an analysis of revenue optimization algorithms against strategic buyers. In doing so, we have closed the gap between upper and lower bounds on strategic regret to a logarithmic factor. Furthermore, the algorithm presented here is simple to analyze and reduces to the truthful scenario when $\gamma \to 0$. This is an important property that previous algorithms did not admit. We believe that our analysis can be extended to the case of random valuations and, in fact, to general bandit problems with strategic opponents.

References

Amin, Kareem, Rostamizadeh, Afshin, and Syed, Umar. Learning prices for repeated auctions with strategic buyers. In *NIPS*, pp. 1169–1177, 2013.

Cesa-Bianchi, Nicolò, Gentile, Claudio, and Mansour, Yishay. Regret minimization for reserve prices in second-price auctions. In *SODA*, pp. 1190–1204, 2013.

Edelman, Benjamin and Ostrovsky, Michael. Strategic bidder behavior in sponsored search auctions. *Decision Support Systems*, 43(1), 2007.

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

Milgrom, P.R. and Weber, R.J. A theory of auctions and competitive bidding. *Econometrica: Journal of the Econometric Society*, pp. 1089–1122, 1982.

Mohri, Mehryar and Medina, Andres Muñoz. Learning theory and algorithms for revenue optimization in second-price auctions with reserve. In *Proceedings of The 31st International Conference on Machine Learning*. JMLR Workshop and Conference Proceedings, 2014.

Nielsen. Global adview pulse, quarter 3. Technical report, The Nielsen Company, 2013.

Ostrovsky, Michael and Schwarz, Michael. Reserve prices in internet advertising auctions: A field experiment. In *Proceedings of the 12th ACM conference on Electronic commerce*, pp. 59–60. ACM, 2011.

Vickrey, William. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, 16(1):8–37, 2012.