

# Advancing Ad Auction Realism: Practical Insights & Modeling Implications

Ming Chen\*  
Outreach Inc.  
Seattle, USA

Sareh Nabi†  
Amazon Ads  
Seattle, USA

Marciano Siniscalchi‡  
Amazon Ads & Northwestern Univ.  
Evanston, USA

## ABSTRACT

This paper proposes a learning model of online ad auctions that allows for the following four key realistic characteristics of contemporary online auctions: (1) ad slots can have different values and click-through rates depending on users’ search queries, (2) the number and identity of competing advertisers are unobserved and change with each auction, (3) advertisers only receive partial, aggregated feedback, and (4) payment rules are only partially specified. We model advertisers as agents governed by an adversarial bandit algorithm, independent of auction mechanism intricacies. Our objective is to simulate the behavior of advertisers for counterfactual analysis, prediction, and inference purposes. Our findings reveal that, in such richer environments, “soft floors” can enhance key performance metrics even when bidders are drawn from the same population. We further demonstrate how to infer advertiser value distributions from observed bids, thereby affirming the practical efficacy of our approach even in a more realistic auction setting.

### ACM Reference Format:

Ming Chen, Sareh Nabi, and Marciano Siniscalchi. 2023. Advancing Ad Auction Realism: Practical Insights & Modeling Implications. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, AdKDD Workshop 2023, Long Beach, CA, USA (AdKDD’23)*. ACM, New York, NY, USA, 6 pages.

## 1 INTRODUCTION

Online ad auctions are an integral part of contemporary e-commerce. The prevalence of such auctions has led to the development of a large literature in both Economics and Computer Science. The vast majority of these contributions focus on relatively simple auction formats, in which one or more ad slots are sold using so-called generalized first- or second-price auctions [4, 26]. These mechanisms adapt the first- and second-price mechanisms from “textbook” auction theory [18, 21] to account for the fact that online advertising services often charge prices *per click* rather than *per impression*.<sup>1</sup>

However, real-world online ad auctions are highly complex.<sup>2</sup> This makes it virtually impossible to analyze equilibrium bidding behavior and compare the performance (e.g., revenues) of different auction formats. As a result, businesses selling online ad space lack principled guidance as to which auction format to adopt.

\*work done while at Amazon Ads; email: ming.chen198906@gmail.com

†can be reached at sareh@amazon.com

‡can be reached at siniscam@amazon.com or marciano@northwestern.edu

<sup>1</sup>That is, the winner is charged if / when the user clicks on the winning ad, rather than when the winning ad is displayed. The key issue is that the “click-through rates” (probabilities that the user clicks on displayed ads) typically vary across advertisers; auction mechanisms must then be adjusted to provide incentives for advertisers to bid according to their true value for showing an ad.

<sup>2</sup>For standard auctions, equilibrium behavior can be characterized, and conditions leading to revenue equivalence or revenue ranking can be established [20, 21].

We propose that simulating advertisers’ behavior using well-understood online-learning algorithms can provide such guidance. To demonstrate this, we develop an approach that allows for the following key departure from standard auction formats and their generalized variants:

- (1) The value of an impression or click for a given ad (and slot), as well as its click through rate, may vary depending on the shopper’s query, identity, and history. Advertisers can choose “targeting clauses” to only bid on certain sets of queries, but cannot condition their bid on individual queries.
- (2) The number and identity of advertisers changes from one auction to the next. However, advertisers do not observe who they are facing in each individual auction.<sup>3</sup>
- (3) Advertisers receive only incomplete feedback about the outcome of each individual auction. In particular, they do not observe other advertisers’ bids, and they only learn the market-clearing price if they win the auction.
- (4) Pricing mechanisms are often incompletely specified. For instance, ad services may indicate that reserve prices are used, but not what they are, or how they are computed.<sup>4</sup>

To reflect the scant information bidders possess when they participate in real-world ad auctions, we choose learning algorithms that only require feedback about *individual* rewards, and are *not tailored* to any specific auction format. We focus on two algorithms: Hedge [1] and EXP3-IX [19, Chap. 12]. These algorithms, with known regret bounds, are applied in auction learning literature and exemplify full- and partial-information algorithms respectively.

We provide two sets of results. In the first, we assign artificial valuations to bidders and simulate their bidding behavior. We consider both small and large number of bidders. We also allow a dense bid grid, especially compared with earlier studies, and complex auction environments with multi-query targeting and soft-floor reserve prices. These results demonstrate how our approach enables performance comparisons across different auction formats.

One specific finding is that, with multi-query targeting, soft floors can lift revenues, even if bidders are ex-ante symmetric. The second set of results instead demonstrates how to use our approach to solve the inverse problem, inferring bidders’ valuations through an iterative parameter search matching bid data<sup>5</sup>. To do so, we employ aggregate bid data from an actual production environment (a major e-commerce website).

<sup>3</sup>That is, we model auctions as games of *incomplete information* with a *random* number of participants.

<sup>4</sup>See for instance the documentation for Ad Thresholds in Google Ads.

<sup>5</sup>This is similar to structural estimation in empirical industrial organization. The key difference is that the structural equations are not derived from Bayesian Nash equilibrium behavior, but from the predictions of our learning model.

## 1.1 Related Literature

There is a vast literature in Economics that studies games from the perspective of learning agents; Fudenberg and Levine [9] provides an authoritative account.

In computer science, the literature on online learning in auctions can be roughly divided into two branches. The first takes the perspective of the seller and studies the design of revenue-maximizing auctions; recent studies include Roughgarden and Wang [25] and Guo et al. [11]. This branch of the literature studies sellers who employ learning algorithms, but bidders are not explicitly modelled.

The second branch instead takes the perspective of a single bidder who uses learning algorithms to guide her bidding process. Weed et al. [27] focus on second-price auctions for a single good, and assume that the valuation can vary either stochastically or adversarially in each auction. In a similar environment, Balseiro et al. [2] and Han et al. [12] study contextual learning in first-price auctions, where the context is provided by the bidder's value. For auctions in which the bidder must learn her own value (as is often the case in the settings we consider), Feng et al. [6] proposes an improved version of the EXP3 algorithm that attains a tighter regret bound. There is also a considerable literature that studies optimal bidding with a budget constraint using reinforcement-learning: e.g., Wu et al. [28], Ghosh et al. [10], and references therein.

Our paper differs from the above references since it models the *interaction* among bidders who adopt online learning algorithms. In this sense, it is closer in spirit to the Economics literature on learning in games. In addition, unlike the present paper, the cited references assume that the winning bidder observes her valuations and payment in each period; some of these papers leverage insights that depend on specific auction format; and all are concerned with auctions for a single object or ad slot.

To the best of our knowledge, the closest papers to our own are Kanmaz and Surer [16], Elzayn et al. [5], and Jeunen et al. [15]. The first reports on experiments using a multi-agent reinforcement-learning model in simple sequential (English) auctions for a single object, with a restricted bid space. Our analysis focuses on simultaneous bidding in scenarios that are representative of actual online ad auctions. The second focuses on position (multi-slot) auctions and, among other results, reports on experiments using no-regret learning (specifically, the Hedge algorithm we also use) under standard generalized second-price and Vickrey-Groves-Clarke pricing rules. Our analysis is complementary in that we allow for different targeting clauses and pricing rules. The third describes a simulation environment similar to ours that is mainly intended to help train sophisticated bidding algorithms for advertisers. We differ in that we allow for bids broadly targeting multiple queries, and focus on learning algorithms that allow us to model auctions with a large number of bidders; in addition, we demonstrate how to infer values from observed bids.

Feng et al. [7] establish the convergence to equilibrium of learning algorithms in first- and second-price auctions, as well as multi-slot VCG mechanisms. Hartline et al. [14] establish the convergence of no-regret learning to coarse Bayes correlated equilibrium in general games with incomplete information.

Nekipelov et al. [22] proposes techniques for estimating agents' valuations in generalized second-price auctions, which stands in

contrast to our method that directly utilizes agents' learning algorithms and is independent of the specific auction format. Rahme et al. [24] study revenue maximization in auctions as a mechanism design problem. Peysakhovich et al. [23] predict player behavior, based on an  $\epsilon$ -Bayesian Nash equilibrium instead of no-regret learning. Bichler et al. [3] suggests a method to calculate  $\epsilon$ -Bayesian Nash equilibria in sealed-bid auctions, focusing on approximating equilibria, as opposed to directly modeling bidder learning behavior.

## 2 AUCTION MODEL

This section introduces the auction model we analyze. For any finite set  $\mathcal{X}$ , we denote by  $\Delta(\mathcal{X})$  the set of probability distributions on  $\mathcal{X}$ .

We fix a single ad slot, or position, within the ad service's or publisher's web page. Time is discrete and indexed by  $t = 1, 2, \dots, T$ . In each time period  $t$ , a user visits the web page and submits a query, represented by a point  $q$  in a finite set  $Q$ . The probability that query  $q$  is submitted at time  $t$  is given by the probability distribution  $F_Q \in \Delta(Q)$ . The ad service then runs an auction to determine which ad to show in the given slot.

There are  $N$  (potential) bidders, whom we also refer to as advertisers, indexed by  $i = 1, \dots, N$ . Each bidder  $i$  is characterized by a *type*  $\tau_i$  drawn at the beginning of each period from a finite set  $\mathcal{T}_i$  according to a distribution  $F_i \in \Delta(\mathcal{T}_i)$ . We also fix functions  $V_i : \mathcal{T}_i \times Q \rightarrow \mathbb{R}$  and  $CTR_i : \mathcal{T}_i \times Q \rightarrow [0, 1]$  such that, whenever bidder  $i$ 's type is  $\tau_i \in \mathcal{T}_i$  and the shopper query is  $q \in Q$ ,  $i$ 's *value per click* is  $V_i(\tau_i, q)$  and the ad *click-through rate* is  $CTR_i(\tau_i, q)$ ; these functions are in practice represented by matrices.

In every period, each bidder  $i$  simultaneously submits a *bid*  $b_i \in \mathbb{R}$  and a *targeting clause*  $c_i \subset Q$ ; the interpretation is that the bidder intends to participate in the auction only if the user's query is an element of  $c_i$ . The timing is as follows: first, bidder  $i$  observes her type  $\tau_i$ ; then, she places a bid  $b_i$  and a targeting clause  $c_i$ ; finally, the user query  $q \in Q$  is realized. Thus, bidders *cannot* tailor their bid to the specific user query in a given period. On the other hand, their bids can and will in practice depend upon their realized type. The winner of the auction is the bidder with the highest *score*, i.e., the product of their bid and their click-through rate, provided they targeted the realized query. Formally, if the realized user query is  $q$  and each bidder  $i$  has type  $\tau_i$ , bids  $b_i$ , and targets  $c_i$ , then bidder  $i$ 's score is  $s_i = b_i \cdot CTR_i(\tau_i, q)$  if  $q \in c_i$  and 0 otherwise; the winner is any  $i \in \arg \max_{j \in N: q \in c_j} s_j$ . Ties are broken randomly.

If the realized user query is  $q$ , bidder  $i$  wins the auction, and the charged *price per click* is  $p$ , then the average reward (payoff) to bidder  $i$  with type  $\tau_i$  is  $CTR_i(\tau_i, q) \cdot [V_i(\tau_i, q) - p]$ . Finally, if bidder  $i$ 's targeting clause  $c_i$  does not include the realized  $q$ , or if it does but  $b_i$  is not the winning bid, her payoff is 0.

The price charged to the winner depends upon the auction rules; we consider different cases in §4, so we defer the specifics until then. Similarly, we specify what bidders observe at the end of each period in §3, as that is a function of the learning algorithm under consideration. However, we maintain throughout that bidders *only* observe their own reward, and not others' rewards or bids.

The concept of "type", as introduced by Harsanyi [13], is key to model games with incomplete information: in every period, each bidder  $i$  knows the distribution  $F_j$  of her opponents  $j$ , but not their realized type. In one common interpretation, there is a population

of potential bidders for every bidder role  $i$ , each with distinct value per click and click-through rate; in each period  $t$ , and for every bidder role  $i$ , a specific element of the corresponding population is drawn according to  $F_i$ . A symmetric environment is one where the set of bidder types and their distributions are identical; we use this in §4 for aggregate-level bidding behavior analysis. One alternative interpretation is that there is a single player (a firm) for every bidder role  $i$ ; her type can represent the fact that the firm may be selling a range of goods, with per-period variations in values and click-through rates reflecting product-specific margins and conversion rates, perhaps due to cost variability, promotions, etc.<sup>6</sup>

Types can also be used to model (indirectly) a *random number of bidders*. Suppose that, for each bidder (or for a subset of bidders), we define a type  $\tau_i^0$  who has zero value and click-through rate for every query. Then, in our learning algorithms, this bidder type will eventually stop bidding. Hence, out of  $N$  potential bidders, only a subset may be active, the ones whose types are different from  $\tau_i^0$ . We assume type distributions are independent across bidders, although this could be relaxed as our learning algorithms do not rely on independence.

*Standard auctions* are a special case of our model in which there is a single query: this can be captured by assuming that the set  $Q$  of queries is a singleton, say  $\{q^*\}$ . Thus, a bidder's type  $\tau_i$  fully characterizes her value for the object, and one may as well take  $V_i$  to be the identity:  $V_i(\tau_i, q^*) = \tau_i$ . Traditional "textbook" auctions do not consider click-through rates; this can be captured by assuming that each function  $CTR_i(\cdot)$  is identically equal to 1. However, pay-per-click auctions are common in advertising, so we consider them "standard" as well.

### 3 LEARNING MODELS

The two main learning models we analyze are Hedge (also referred to as Exponential Weights or Multiplicative Weights) algorithm [1, 8] and EXP3-IX [17]. We fix a grid  $B$  of possible bid values. To streamline notation, for each bidder  $i$  and time period  $t$ , we let  $a_{i,\tau_i,t}$  denote the pair  $(b_i, c_i) \in B \times 2^Q \equiv A$  of bid and targeting clause chosen by bidder  $i$  of type  $\tau_i$  in period  $t$ . Relative to standard expositions of these algorithms [19, e.g.], we add incomplete information and incorporate different queries drawn in each iteration.

Each bidder type learns from their own observations, not those of other types of the same bidder, to accommodate incomplete information. Learning is not conditioned on the unobserved realized query, reflecting the reality of online ad auctions.

#### 3.1 Model Setup for Auction Environment

We use the following additional notation to describe the auction environment and learning algorithms. Suppose each bidder  $i$  bids  $b_i$  on clause  $c_i$ , and prices per query are  $p = (p_q)$ . Denote by  $c_{iq}$  the indicator function that equals 1 if and only if  $q \in c_i$ . Then bidder  $i$ 's expected utility if she wins the auction for the (non-empty) set of queries  $Q \subseteq \{0, \dots, Q-1\}$ , given her value and CTR for each query  $q \in Q$  as  $v_{iq}$  and  $CTR_{iq}$  respectively, is

$$EU_i(b_i, c_i, p) = \sum_{q \in Q} F_Q(q) \cdot CTR_{iq} \cdot c_{iq} \cdot [v_{iq} - p_q], \quad (1)$$

<sup>6</sup>In either interpretation, formally, types are indices for the functions  $V_i$  and  $CTR_i$  with no intrinsic meaning.

where the expectation is due to the random arrival of queries ( $F_Q(q)$ ) and clicks ( $CTR_{iq}$ ). In case of ties, for simplicity of implementation, we just divide the winner's surplus  $v_{iq} - p_q$  by  $N$ , rather than (as we should) the number of tying bidders. Losing bidders get 0 expected utility. Since the EXP3-IX algorithm requires rewards in  $[0, 1]$ , we define the **normalized reward** of bidder  $i$  as

$$r_i = \frac{EU_i(b_i, c_i, p) - (0 - b_{\max})}{(v_{\max} - 0) - (0 - b_{\max})} = \frac{EU_i(b_i, c_i, p) + b_{\max}}{v_{\max} + b_{\max}}. \quad (2)$$

This is because the EU of a bidder cannot exceed the maximum value of the slot for any query if she were to get it for free, i.e.,  $v_{\max} - 0$ ; and it is always at least as large as getting a worthless slot and paying the maximum bid for it, i.e.,  $0 - b_{\max}$ .

In our auction setting, we apply the Hedge and EXP3-IX algorithms, described in the Appendix. Bidders independently use these to update their actions based on observed rewards. Hedge assumes bidders observe rewards for all possible actions, leading to faster learning but less realistic scenarios. EXP3-IX assumes observation of the chosen action's reward, providing a slower but more realistic learning process. Although Hedge is more computationally efficient and less dispersed, a deeper investigation of these algorithms' distributional characteristics is suggested. We chose EXP3-IX over EXP3 due to its tighter bounds on realized regret, despite EXP3's good expected regret guarantees.

## 4 EMPIRICAL RESULTS

### 4.1 Standard auctions

We first consider a standard, symmetric auction environment with a single query, so  $Q = \{q^*\}$ , and pay-per-impression pricing, so all CTRs are equal to 1. We consider both a small auction with  $N = 2$  bidders, and a more realistic one with  $N = 10$  bidders. Values are uniformly distributed on a grid  $B$  in the interval  $[0, 1]$ . Formally, given the grid  $B$ , for all  $i$ , we let  $\mathcal{T}_i = B$ ,  $V_i(b, q^*) = b$ ,  $CTR_i(b, q^*) = 1$ , and  $F_i(b) = \frac{1}{|B|}$  for all  $b \in B$ . Furthermore, we allow all bids  $b_i$  on the grid  $B$ .

In a second-price auction, it is a dominant strategy for every bidder to bid their value. In a first-price auction, with  $N$  bidders, a continuum of values (uniformly distributed on  $[0, 1]$ ) and bids, the unique symmetric Bayesian Nash equilibrium is for each bidder to bid a fraction  $\frac{N-1}{N}$  of their value. Furthermore, Myerson's revenue equivalence applies, so the expected revenue for the advertising service is the same under both formats, namely  $\frac{N-1}{N+1}$ . We want to compare these theoretical predictions with the output of our learning algorithms with a fine enough grid  $B$ , and a long enough horizon  $T$ .

We choose  $B = \{\frac{i}{20} : i = 0, \dots, 20\}$  and  $T = 1,000,000$  for EXP3-IX. We also choose the tuning parameters optimally, as described in Lattimore and Szepesvári [19] (Theorem 12.1, Chapter 12). Table 1 displays revenues for the advertising service in first- and second-price auctions under EXP3-IX, averaged over the last 10% of the learning period, i.e., 100,000 iterations, and 5 different runs of the algorithm, as well as the standard deviation of revenues across different runs.<sup>7</sup>

<sup>7</sup>Throughout the paper, we draw a fixed sequence of type realizations for all bidders. This ensures that the only randomness is from the algorithms' choice of action.

**Table 1: Standard single-item auction, EXP3-IX**

Auction format	Mean Revenue	Std. Dev.
First price, $N = 2$	0.3474	0.0205
Second price, $N = 2$	0.3346	0.0016
First price, $N = 10$	0.780	0.0095
Second price, $N = 10$	0.7723	0.0171

Table 2 reports the results for Hedge. We choose  $T = 400,000$  and  $\eta = 0.02$ . We averaged over 5 different runs of the algorithm and over the last 40,000 time periods.

**Table 2: Standard single-item auction, Hedge.**

Auction format	Mean Revenue	Std. Dev.
First price, $N = 2$	0.3201	0.0018
Second price, $N = 2$	0.3256	0.0018
First price, $N = 10$	0.8305	0.0011
Second price, $N = 10$	0.8334	0.0008

The key take-away is that expected revenues are close to the theoretical value  $\frac{1}{3}$  for  $N = 2$  under both Hedge and Exp3-IX. With  $N = 10$ , Hedge again approximates the theoretical value  $0.8\bar{1}$ , while Exp3-IX is not as close. Revenues do not vary much across different runs. Even when  $N = 2$ , EXP3-IX required a considerably larger number of periods to achieve similar results as Hedge.

## 4.2 Soft floors

Zeithammer [29] shows that, in a symmetric auction for a single object, bid functions are monotonic. As a consequence, the revenue equivalence theorem [21] applies,<sup>8</sup> and introducing soft floors in second-price auctions do not affect either the final allocation or the advertising service’s revenues. He then demonstrates by way of examples that, with asymmetric bid distributions, revenues in a second-price auction with a soft floor can be either higher or lower than in a standard second-price auction. In this section, we show that, in a more realistic environment in which types are multi-dimensional and bidders can choose which queries to target, different auction formats can yield different revenues even when bidder types are drawn from the same distribution.

Given a soft floor equal to  $s$ , the price is determined as follows; we describe the case of equal click-through rates for simplicity. Let  $b_{(1)}$  and  $b_{(2)}$  be the first- and, respectively, second-highest bids. If  $b_{(2)} \geq s$ , then  $p = b_{(2)}$ , as in a standard second-price auction. If  $b_{(1)} \geq s > b_{(2)}$ , then  $p = s$ , as if  $s$  was a standard reserve price (“hard floor”). Crucially, if  $s > b_{(1)}$ , then the high bidder still wins the auction (on the contrary, with a standard reserve price, the seller would keep the object) and  $p = b_{(1)}$ .

For completeness, we first consider the symmetric, single-query environment of §4.1 and simulate an auction with a soft floor equal to  $s = 0.5$ . Consistently with Zeithammer [29], we find that soft floors have virtually no impact on revenues.<sup>9</sup>

<sup>8</sup>Intuitively, under different pricing rules, bidding behavior is also different, in a way that exactly offsets differences in the way prices are computed.

<sup>9</sup>Results are as follows (compare with Tables 1 and 2): Hedge with  $N = 2$  bidders yields average revenues equal to 0.324 (standard deviation 0.0024); Hedge with  $N = 10$

Next, we consider a multi-query environment, which is beyond the scope of Zeithammer [29]. We assume  $N = 3$  bidders, all with the same set of possible types  $\mathcal{T}_1 = \mathcal{T}_2 = \mathcal{T}_3 = \{1, 2, 3\}$ . There are two queries, so  $Q = \{1, 2\}$ . Values and click-through rates for all bidders are as described in Table 3. Thus, for example,  $V_i(2, 1) = 0.25$  and  $CTR_i(1, 2) = 0.1$ . Both queries are equally likely, and all types are also equally likely. To clarify, these parameters are artificial and purposely chosen to illustrate the point. We let  $B = \{\frac{i}{20} : i = 0, \dots, 20\}$  and  $T = 1,000,000$ .<sup>10</sup>

**Table 3: Values and Click-Through Rates for all bidders**

$\tau_i$	$V_i(\tau_i, 1)$	$CTR_i(\tau_i, 1)$	$V_i(\tau_i, 2)$	$CTR_i(\tau_i, 2)$
1	0.5	0.3	0.25	0.1
2	0.25	0.1	1	0.1
3	0.25	0.1	1	0.2

Table 4 reports expected revenues per impression (where expectations are taken over search queries, bidder types, and click-through rates) for EXP3-IX, averaged over 5 runs. Table 5 reproduces the results for Hedge, with  $T = 400,000$ .

**Table 4: Multi-query auctions and soft floors, EXP3-IX output for  $T = 1M$ , averaged over 5 runs.**

Auction format	Mean Revenue	Std. Dev.
First price	0.0830	0.0007
Second price	0.0509	0.0008
Soft floor $s = 0.65$	0.0813	0.0007

**Table 5: Multi-query auctions and soft floors, Hedge output for  $T = 400K$ , averaged over 5 runs.**

Auction format	Mean Revenue	Std. Dev.
First price	0.0691	0.0016
Second price	0.0857	0.0001
Soft floor $s = 0.65$	0.0741	0.0061

Our first key finding is that, as anticipated, the three auction formats yield different expected revenues. Soft-floor reserve prices can impact revenues; thus, our simulations provide some support to this common industry practice. Under Hedge, revenues are higher in soft-floor reserve price auctions than in first-price auctions, but second-price auctions perform best. With EXP3-IX, second-price auctions do not fare as well; the highest revenues come from first-price auctions, with soft-floor pricing behind.

A second key finding is that EXP3-IX results do not align with Hedge even when run for longer periods. Each panel in Figure 1 shows the frequency of bids in the final 40,000 periods of the learning algorithm, summed over 5 runs, divided by type and targeting clause (i.e., queries actually targeted). The main take-away point is that every type eventually learns to choose a specific targeting clause *and* for the most part also places fairly concentrated bids.

yields 0.834 ( $7e - 6$ ); EXP3-IX with  $N = 2$  yields 0.377 (0.032); EXP3-IX with  $N = 10$  yields 0.774 (0.0176).

<sup>10</sup>We also ran these simulations with higher values of  $T$ , and similar patterns emerge.

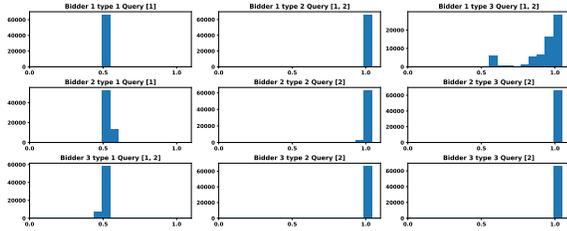


Figure 1: Bids in second-price auction under Hedge

Now compare with Figure 2, which summarizes predicted bids for bidder 1 under EXP3-IX, with Figure 1 above (again we sum over 5 runs and only look at the last 100,000 periods). Type 1 occasionally

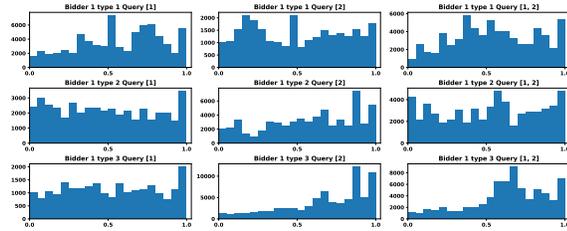


Figure 2: Bidder 1's bids in second-price auction, EXP3-IX

bids on one or both queries, showing little convergence even after one million runs. EXP3-IX requires more experimentation, as it only learns from played actions, while Hedge converges faster but assumes learning about unplayed actions. This supports the idea that leveraging knowledge of the pricing mechanism enhances performance. Hedge outperforms due to its reliance on full reward information. We suggest that incorporating heuristics into EXP3-IX may partially address this imbalance; we leave this to future work.

### 4.3 Inferring values and bid shading

Next, we use our approach to infer the distribution of bidders' values from the observed distribution of bids. We utilize Hedge for this inference procedure, due to the bid dispersion observed with EXP3-IX (§4.2).

The analysis is based on aggregated bid data for two specific shopper queries in an e-commerce setting, one characterized by low traffic and the other by high traffic. The data aggregation process converts all bids into a bid per impression, so we set all click-through rates to 1. Thus, we apply our analysis to a symmetric environment with a single query, unit click-through rates, and two different scenarios, one with a low number of bidders (low traffic) and one with a high number of bidders (high traffic).

We normalize all bids to lie in a grid  $B$  in the interval  $[0, 4]$ . Specifically, we take  $B = \{\frac{i}{10} : i = 0, \dots, 40\}$ . We then define the set of types to be quantiles of the given bid distribution, with a step size of 0.1. We initialize the procedure by setting the *inferred* value distribution to the *observed bid* distribution. We then apply multiple iterations of the following procedure. First, we run the learning algorithm and derive a *predicted* bid distribution.

Then, we adjust inferred values for each quantile. To do so, we use the following heuristic. Suppose that, for a given quantile, the

currently inferred value is  $v$ , the observed bid is  $b^o$ , and the predicted bid is  $b^p$ . This means that the predicted extent of bid shading (reducing one's bid below one's value) is  $\sigma = \frac{b^p}{v}$ . We then update  $v$  according to  $v \leftarrow v + \alpha \left( \frac{b^o}{\sigma} - v \right)$ . Finally, since the inferred values are associated with increasing quantiles, we apply a "flattening" step to ensure that they are indeed increasing. This completes one inference iteration. Intuitively, a bidder with value  $\frac{b^o}{\sigma}$  who bids by applying a shading factor of  $\sigma$  will bid exactly  $b^o$ , the actually observed bid for this quantile. We then adjust the inferred  $v$  in the direction of  $\frac{b^o}{\sigma}$ , applying a learning rate adjustment  $\alpha = 0.2$ .

Figure (3) represents the inferred values of bidders for the low-traffic search query, under the three pricing rules we analyzed in §4.2: first price, second price, and soft floor with a reserve price of \$0.65. We chose these pricing rules arbitrarily, to demonstrate their impact on the inference process. We used 8 iterations of the inference procedure and 3 runs of the learning algorithm per iteration, each with  $T = 500,000$ , averaging bids over the last 50,000 periods.

As anticipated, we observe bid shading in the first price auction, as well as within the lower to middle quantiles in the case of the soft floor. The second price auction also displays bid shading at lower quantiles. We hypothesize that this deviation from theoretical prediction is due to a lack of learning amongst low-valuation types: players with low valuation win rarely, so the feedback they receive is coarse on most periods and hence insufficient to converge to bidding one's value.<sup>11</sup>

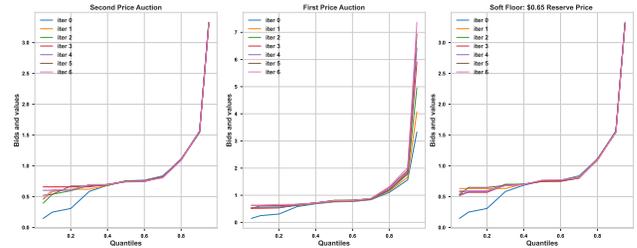


Figure 3: Value inference in three auction formats for the low traffic search query.

Figure 4 analyzes the high-traffic query. We increased the simulation length to  $T = 800,000$  periods. As one can see, even with a large number of bidders, inferred values converge in a few iterations<sup>12</sup>.

<sup>11</sup>We observe the same pattern of deviations from truthful bidding for low types in the second-price auction studied in §4.1.

<sup>12</sup>We used a realistic pricing function, but details are withheld due to confidentiality.

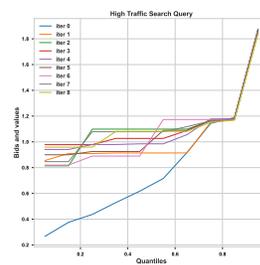


Figure 4: Inferring values for the high traffic search query. In this experiment, we set  $T = 800,000$  and applied 8 inference iterations. The results show that inferred values converge in a few iterations even in a more realistic auction environment.

## 5 CONCLUSIONS

Online learning algorithms can serve as effective tools in modeling bidders in complex ad auctions. We showed that soft floors can have a positive effect on the ad service's revenue, even in an ex-ante symmetric environment. We also demonstrated that the algorithm choice can significantly affect the learning rate in a more realistic auction environments, especially with a large number of bidders. Utilizing our simulation approach, we further inferred bidders' valuations in the presence of more realistic auction rules. This has been experimented with aggregated bid data from an e-commerce website for both low- and high-density auctions.

## REFERENCES

- [1] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pages 322–331. IEEE, 1995.
- [2] S. Balseiro, N. Golrezaei, M. Mahdian, V. Mirrokni, and J. Schneider. Contextual bandits with cross-learning. *arXiv preprint arXiv:1809.09582*, 2018.
- [3] M. Bichler, M. Fichtl, S. Heidekrüger, N. Kohring, and P. Sutterer. Learning equilibria in symmetric auction games using artificial neural networks. *Nature machine intelligence*, 3(8):687–695, 2021.
- [4] B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American economic review*, 97(1):242–259, 2007.
- [5] H. Elzayn, R. Colini Baldeschi, B. Lan, and O. Schrijvers. Equilibria in auctions with ad types. In *Proceedings of the ACM Web Conference 2022*, pages 68–78, 2022.
- [6] Z. Feng, C. Podimata, and V. Syrgkanis. Learning to bid without knowing your value. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 505–522, 2018.
- [7] Z. Feng, G. Guruganesh, C. Liaw, A. Mehta, and A. Sethi. Convergence analysis of no-regret bidding algorithms in repeated auctions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5399–5406, 2021.
- [8] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.
- [9] D. Fudenberg and D. K. Levine. *The theory of learning in games*. MIT press, 1998.
- [10] A. Ghosh, S. Mitra, S. Sarkhel, and V. Swaminathan. Optimal bidding strategy without exploration in real-time bidding. In *Proceedings of the 2020 SIAM International Conference on Data Mining*, pages 298–306. SIAM, 2020.
- [11] W. Guo, M. Jordan, and E. Zampetakis. Robust learning of optimal auctions. *Advances in Neural Information Processing Systems*, 34, 2021.
- [12] Y. Han, Z. Zhou, and T. Weissman. Optimal no-regret learning in repeated first-price auctions. *arXiv preprint arXiv:2003.09795*, 2020.
- [13] J. C. Harsanyi. Games with incomplete information played by “bayesian” players, i–iii part i. the basic model. *Management science*, 14(3):159–182, 1967.
- [14] J. Hartline, V. Syrgkanis, and E. Tardos. No-regret learning in bayesian games. *Advances in Neural Information Processing Systems*, 28, 2015.
- [15] O. Jeunen, S. Murphy, and B. Allison. Learning to bid with auctiongym. *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining AdKDD Workshop (AdKDD'22)*, 2022.
- [16] M. Kanmaz and E. Surer. Using multi-agent reinforcement learning in auction simulations. *arXiv preprint arXiv:2004.02764*, 2020.
- [17] T. Kocák, G. Neu, M. Valko, and R. Munos. Efficient learning by implicit exploration in bandit problems with side observations. In *Advances in Neural Information Processing Systems*, volume 27, 2014.
- [18] V. Krishna. *Auction theory*. Academic press, 2009.
- [19] T. Lattimore and C. Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.
- [20] P. R. Milgrom and R. J. Weber. A theory of auctions and competitive bidding. *Econometrica: Journal of the Econometric Society*, pages 1089–1122, 1982.
- [21] R. B. Myerson. Optimal auction design. *Mathematics of operations research*, 6(1): 58–73, 1981.
- [22] D. Nekipelov, V. Syrgkanis, and E. Tardos. Econometrics for learning agents. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pages 1–18, 2015.
- [23] A. Peysakhovich, C. Kroer, and A. Lerer. Robust multi-agent counterfactual prediction. *Advances in Neural Information Processing Systems*, 32, 2019.
- [24] J. Rahme, S. Jelassi, and S. M. Weinberg. Auction learning as a two-player game. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=YHdeAO6116T>.
- [25] T. Roughgarden and J. R. Wang. Minimizing regret with multiple reserves. *ACM Transactions on Economics and Computation (TEAC)*, 7(3):1–18, 2019.
- [26] H. R. Varian. Online ad auctions. *American Economic Review*, 99(2):430–34, 2009.
- [27] J. Weed, V. Perchet, and P. Rigollet. Online learning in repeated auctions. In *Conference on Learning Theory*, pages 1562–1583. PMLR, 2016.
- [28] D. Wu, X. Chen, X. Yang, H. Wang, Q. Tan, X. Zhang, J. Xu, and K. Gai. Budget constrained bidding by model-free reinforcement learning in display advertising. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1443–1451, 2018.
- [29] R. Zeithammer. Soft floors in auctions. *Management Science*, 65(9):4204–4221, 2019.

## 6 APPENDIX

### 6.1 Algorithms

Details of the Hedge and EXP3-IX algorithms are as follows.

---

#### Algorithm 1 EXP3-IX for Auctions

---

**Require:** parameters  $\eta > 0, \gamma > 0, N, T$   
**Require:** action sets  $A$  (the same for all bidders), query distribution  $F_Q$   
**Require:** type space  $\mathcal{T}_i$  and distribution  $F_i$ , and functions  $V_i, CTR_i$  for each  $i = 1, \dots, N$   
for all  $i$  and  $a \in A$ , set  $l_{i,\tau_i,0}(a) = 0$   
**for**  $t = 1, \dots, T$  **do**  
  draw  $q \sim F_Q$   
  **for**  $i = 1, \dots, N$  **do** {Play}  
    draw  $\tau_i \sim F_i$   
    **for**  $a_i \in A$  **do** {Softmax}  
      set  $p_{i,\tau_i,t}(a_i) = \frac{\exp(-\eta l_{i,\tau_i,t-1}(a_i))}{\sum_{a'_i \in A} \exp(-\eta l_{i,\tau_i,t-1}(a'_i))}$   
    **end for**  
    draw  $a_{i,\tau_i,t} \sim p_{i,\tau_i,t}$   
  **end for**  
  **for**  $i = 1, \dots, N$  **do** {Learning}  
    compute normalized reward  $r_{i,t} \in [0, 1]$  given  $q$  and  $\tau_i$  under the actual bids  $a_{i,\tau_i,t}$  and set  
     $l_{i,\tau_i,t}(a_{i,\tau_i,t}) = l_{i,\tau_i,t-1}(a_{i,\tau_i,t}) + \frac{1-r_{i,t}}{p_{i,\tau_i,t}(a_{i,\tau_i,t})+\gamma}$   
    **for**  $a'_i \in A \setminus \{a_{i,\tau_i,t}\}$  and  $\tau'_i \in \mathcal{T}_i \setminus \{\tau_i\}$  **do**  
      set  $l_{i,\tau'_i,t}(a'_i) = l_{i,\tau'_i,t-1}(a'_i)$   
    **end for**  
  **end for**  
**end for**

---



---

#### Algorithm 2 Hedge for Auctions

---

**Require:** parameters  $\eta > 0, N, T$   
**Require:** action sets  $A$  (the same for all bidders), query distribution  $F_Q$   
**Require:** type space  $\mathcal{T}_i$  and distribution  $F_i$ , and functions  $V_i, CTR_i$  for each  $i = 1, \dots, N$   
for all  $i, \tau_i \in \mathcal{T}_i$ , and  $a \in A$ , set  $w_{i,\tau_i,0}(a) = 0$   
**for**  $t = 1, \dots, T$  **do**  
  draw  $q \sim F_Q$   
  **for**  $i = 1, \dots, N$  **do** {Actual play}  
    draw  $\tau_i \sim F_i$   
    **for**  $b_i \in A$  **do** {Softmax}  
      set  $p_{i,\tau_i,t-1}(b_i) = \frac{\exp(w_{i,\tau_i,t-1}(b_i)/\eta)}{\sum_{b'_i \in A} \exp(w_{i,\tau_i,t-1}(b'_i)/\eta)}$   
    **end for**  
    draw  $a_{i,\tau_i,t} \sim p_{i,\tau_i,t-1}$   
  **end for**  
  **for**  $i = 1, \dots, N$  **do** {Learning}  
    **for**  $b_i \in A$  **do**  
      compute reward  $r_i$  given  $q$  and  $\tau_i$ , assuming  $i$  bids  $b_i$  and all  $j \neq i$  bid  $a_{j,t}$   
      set  $w_{i,\tau_i,t}(b_i) = w_{i,\tau_i,t-1}(b_i) + r_i$   
      **for**  $\tau'_i \in \mathcal{T}_i \setminus \{\tau_i\}$  **do**  
        set  $w_{i,\tau'_i,t}(b_i) = w_{i,\tau'_i,t-1}(b_i)$   
      **end for**  
    **end for**  
  **end for**  
**end for**

---