An Efficient Deep Distribution Network for Bid Shading in First-Price Auctions

Tian Zhou, Hao He, Shengjun Pan, Niklas Karlsson, Bharatbhushan Shetty, Brendan Kitts,
Djordje Gligorijevic, San Gultekin, Tingyu Mao, Junwei Pan, Jianlong Zhang and Aaron Flores

Yahoo Research, Verizon Media

Sunnyvale, CA, USA

{tian.zhou,hao.he,alanpan,niklas.karlsson,bharatbs,brendan.kitts,djordje,sgultekin,tingyu.mao}@verizonmedia.com,
jonaspan@tencent.com,jianlong.zhang@gmail.com,aaron.flores@verizonmedia.com

ABSTRACT

Since 2019, most ad exchanges and sell-side platforms (SSPs), in the online advertising industry, shifted from second to first price auctions. Due to the fundamental difference between these auctions, demand-side platforms (DSPs) have had to update their bidding strategies to avoid bidding unnecessarily high and hence overpaying. *Bid shading* was proposed to adjust the bid price intended for second-price auctions, in order to balance cost and winning probability in a first-price auction setup. In this study, we introduce a novel deep distribution network for optimal bidding in both *open (non-censored)* and *closed (censored)* online first-price auctions. Offline and online A/B testing results show that our algorithm outperforms previous state-of-art algorithms in terms of both surplus and effective cost per action (*eCPX*) metrics. Furthermore, the algorithm is optimized in run-time and has been deployed into VerizonMedia DSP as production algorithm, serving hundreds of billions of bid requests per day. Online A/B test shows that advertiser's ROI are improved by +2.4%, +2.4%, and +8.6% for impression based (CPM), click based (CPC), and conversion based (CPA) campaigns respectively.

CCS CONCEPTS

• **Applied computing** → **Online auctions**; • **Information systems** → **Display advertising**; • **Computing methodologies** → **Machine learning algorithms**.

KEYWORDS

Online Auction, Real-Time Bidding, Display Advertising, Bid Shading, Distribution Learning

1 INTRODUCTION

In online advertising, inventories that are not directly sold are primarily auctioned programmatically in real-time bidding (RTB). Before 2018, second-price was the dominant form of auction for RTB, where the winner only needs to pay the second highest bid price. However, starting in 2017, all the major exchanges/SSPs including AppNexus, Index Exchange, OpenX, Rubicon Project, and Pubmatic, with the exception of Google AdX, were rolling out or testing first-price auctions, where the winner must pay whatever bid it submitted, in varying degrees [35]. Google transitioned to first-price auctions in 2019 [4]. Several motivations were behind the transition away from second-price auctions. Firstly, first price auctions provided greater transparency and accountability for bidders, since the bidder was always charged exactly what they offered [9, 14, 32, 36]. Secondly, the unmodified second price auctions proved to be incompatible with the widespread and popular practice of Header Bidding [21].

For demand-side platforms (DSPs), which are the bidders in the auctions, transitioning from second-price auctions to first-price auctions meant that bidding strategies would need to be dramatically adjusted. In second-price auctions, auction theory states that it is a dominant strategy for a bidder to bid *truthfully* [12], namely it is the optimal strategy for a DSP to compute the value of the inventory being auctioned and submit this value as the bid price, regardless of the other bidders' behavior. However, for first-price auctions such a strategy would cause DSPs to overbid and thus lose money. This comes from the fundamental difference in the payment between second-price and first-price auctions.

Unlike in second-price auctions, in first-price auctions a DSP must incorporate other bidders' behavior, more precisely its estimates of other bidders' bid prices, into its own bidding strategy. If the competing bidders' prices were known in advance, which is impossible in practice, the optimal bidding strategy would be to submit a bid price that is slightly higher than the highest competing bid price so as to win the auction with the lowest price possible. In reality, a DSP has to estimate the minimum winning price as best as it can, and lower its original bid price intended for second-price auction, i.e., *shade* the truthful value of the inventory, accordingly. This process is known as *bid shading*. Bid shading is relatively new to online advertising, but it has been used in auctions from other industries [7, 8, 11, 20].

The most important aspect of bid shading is a trade-off between the winning rate and (Return On Investment) ROI. The more the bid price is shaded, namely the lower the final bid price, the better the ROI if the bid is won. However, lower bid price also lead to lower

probability of wining a bid. The buyer or bidder *surplus* [20, 21] is the shaded amount, namely the difference between the bid price before shading and the final bid price if the bid is won. The surplus is 0 if the bid is lost. The quantitative objective of bid shading is thus to maximize the surplus, either directly or implicitly.

In the first price auction, an important piece of information is the *minimum winning price*, which is the highest competing bid, inclusive of the floor price, if provided. However, it's up to an SSP to decide whether to provide the minimum winning price to the participating DSPs after the auction. *Open (non-censored)* first price auctions refer to the auctions where feedback including *minimum winning price* are shared to all participants regardless of auction outcome while in *closed (censored)* auction, only the win or loss feedback is available.

In this paper, we propose a deep distribution network to learn the distribution of *minimum winning price* for both censored and non-censored first price auctions, as well as for efficient search of the optimal bid price to maximize the profit in the real-time serving. The model has the flexibility of explicitly choosing the distributions of the minimum winning price and the structures of the network. Comprehensive experiments have been conducted and the model has been successfully deployed in one of the biggest DSPs in the world. We demonstrate the effectiveness of the algorithm by showing its performance lift compared to the existing models in offline and online setting.

To summarize, the main contributions of this study are the following:

- We propose a novel framework for bid shading that, to the best of our knowledge, is the first unified distribution-based bid shading method can be applied for both *censored* and *non-censored* first-price auctions.
- We mathematically prove that an efficient search algorithm can be used to find the optimal bid price that maximizes surplus given the distributions. This allows the model to be deployed online under strict latency constraints.
- We implement and test the proposed algorithm in online and offline settings. The key metrics results show **+9.7%** and **+14.3%** lift for offline and online A/B tests respectively, compared to the existing models. Moreover, advertiser's ROI are improved by **+2.4%**, **+2.4%**, **+8.6%** for impression based (CPM), click based (CPC), and conversion based (CPA) campaigns respectively.

The rest of the paper is organized as follows: In Section 2, we discuss related work in bid shading and online auctions. In Section 3, we describe the formulation of our bid shading algorithms, including details on how we train models to estimate the distributions of *minimum winning price* and how to search for the optimal bid price efficiently on both *censored* and *non-censored* first-price auctions. In Section 4, we show the comprehensive experiments of different choices of distributions and network structures compared to the current state-of-art bid shading algorithms for both *censored* and *non-censored* first price auctions. In Section 5, we briefly introduce the online deployment of deep distribution network in the serving system of our DSP and show the online ROI improvements compared to productions. Finally we conclude in Section 6.

2 RELATED WORK

Bid optimization is one of the most fundamental problems in online advertising [6, 18, 37, 40, 41]. Recently, bid shading [11, 20, 42] attracts much attention since most ad exchanges and SSPs are shifting from second to first price auctions. Bid Shading shares characteristics with the Seller's (Reserve) Price Optimization Problem [5, 23, 25, 26, 33, 34], where sellers have a good with a manufacturing cost, and their task is to set their price above this cost to maximize their profit. However, a variety of constraints exist in the bid shading buyer problem that are unique: (i) Buyers need to predict the bidding behavior of *competing buyers* on each auction, leading to strategic considerations. (ii) Buyer Feedback is constrained in systematic ways, with censored and uncensored information provided. (iii) Buyers need to find a bid price for billions of auctions, each of which has combinatorial aspects; whereas sellers generally have a set of fixed inventory [1, 3]. For these reasons, most authors talk about the Bidding problem as a Buyer specific activity, distinct from Seller reserve pricing, and we take the same approach in this paper.

There have been two general approaches for bid shading, depending on whether the minimum winning price is provided or censored. The first assumes that the minimum winning price is provided, and builds a machine learning algorithm to predict the *optimal shading factor* - the ratio of the minimum winning price to the bid price before shading. For instance, Logistic Regression (*LogReg*) and Factorization Machines (*qFwFM*) [15] have been used previously to predict the optimal shading factor, using an asymmetric loss function which penalizes losses. The drawback of these approaches is that they learn from the winnable bids only [15], ignoring a large portion of available data. This paper proposes an algorithm for modeling distributions over the entire bidding landscape which allows for learning from both won and lost auctions.

The other general approach tries to estimate the distribution of the minimum winning price at segment level, and then finds the optimal bid price by maximizing the *expected* surplus with respect to the estimated distribution. The *NonLinear* algorithm in [22] attempts to estimate the distribution using a non-linear approximation on a predefined segment. The main drawback of this method is that the distribution is estimated separately, and thus cross-segment information is not utilized. Furthermore, the segments must be explicitly defined and small segments must be manually grouped together. The recent WinRate model from [29] takes a similar approach but approximates the distributions, implicitly with *log-logistic*, for all segments simultaneously, while its drawback is that it doesn't utilize the minimum winning price information when available.

Outside of the problem of bid shading, there has been some related research trying to characterize winning prices on auctions. For example, in [38, 39] the authors first estimate the winning price distribution, and then use it to do point-wise prediction of the winning price via a mixture model. The distribution implicitly assumes that the winning price follows logistic distributions, while a log-logistic distribution is used in [29]. Additionally, there has been prior work estimating the bidding landscape in second-price auctions [38, 39], in which the minimum winning price feedback is only available while winning the auction. These approaches would

need to be extended to work under first-price auctions, and the problem of surplus maximization.

Sell-Side Platforms are motivated to provide bid shading services to the bidders, especially during the transition period from second-price to first-price auctions. Such services include *Bid Translation Service* from Google AdX [16], *Estimated Market Rate* from Rubicon Project [31], and *Bid Price Optimization* system from AppNexus [2]. However, these services are rather a transition tool in helping DSP's transition to first-price auctions, and many of them are being deprecated. For example, AdX deprecated its Bid Translation Service in May 2020 [19].

3 ALGORITHMS

For the reader's convenience, we list some notations that will be used throughout the paper.

V	bid price before shading (true value of the ad opportunity)
S	surplus
s	expected surplus: $\mathbb{E}[S]$
b	bid price
$\hat{b}$	minimum winning price
b^*	optimal bid price that maximizes the expected surplus s
D	distribution of the minimum winning price
$F(b)$	Cumulative Distribution Function (CDF) of distribution D, i.e., winning probability at bid price b
$f(b)$	Probability Density Function (PDF) of distribution D: $F'(b)$
x	input feature vector that defines a segment
α	parameters of a distribution

First of all, we define *surplus* mathematically. Upon receiving a bid request for first-price auction, its value V is estimated based on its event rate, such as click-through rate (CTR) or conversion rate (CVR), and a campaign level price control signal that helps to pacing campaign budget smoothly across the flight. Let b be the bid price to be submitted, and $\hat{b}$ be the minimum winning price. $\mathbf{I}(b > \hat{b})$ equals 1 if $b > \hat{b}$ and 0 otherwise, which indicates with bid price b whether we win the auction. Then the *surplus* is defined as

$$S(b; V, \hat{b}) = (V - b)\mathbf{I}(b > \hat{b}) = \begin{cases} V - b, & \text{if } b > \hat{b}, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Let $x = (x_1, x_2, \ldots, x_k)$ be the input feature vector derived from the current publisher and user attributes, such as top level domain, sub-domain, layout, day of week, etc. We calculate the optimal bid price b^* given the current input feature vector in two steps:

Distribution Estimation First we build a machine learning model to approximate the conditional distribution $D_{\hat{b}|x}$ of the highest competing bid price $\hat{b}$ by modeling its PDF or CDF.

Surplus Maximization Then we find the bid price $b = b^*$ that maximizes the expected surplus $\mathbb{E}_{\hat{b}|x}[S]$:

$$
\begin{aligned}
b^* &= \arg\max_{b \in (0,V)} \mathbb{E}_{\hat{b}|x}[S(b; V, \hat{b})] \\
&= \arg\max_{b \in (0,V)} \mathbb{E}_{\hat{b}|x}\left[(V - b)\,\mathbf{I}(b > \hat{b})\right] \\
&= \arg\max_{b \in (0,V)} (V - b)\Pr(\hat{b} < b \mid x).
\end{aligned} \tag{2}
$$

In the following sections, we describe in more details how the distribution estimation and surplus maximization are conducted.

3.1 Inference of Distribution

Conditioning on input feature vector x of a bid request, to find the optimal bid price b* that maximizes the expected surplus, modelling of winning probability $\Pr(\hat{b} < b; \hat{b} \mid x)$ for any intended bid price b is the key. We assume that for given x, the minimum winning price follows a conditional distribution $D_{\hat{b}|x}$. We further assume that each $\hat{b}$ is drawn independently from a probability distribution $D_{\hat{b}; \alpha(x)}$ that belongs to a family of known distributions, where $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m)$ is its m-parameters vector. Note that the parameter vector α is a function of x, namely, bid samples with the same input feature vector follow the same distribution. We will discuss details about different distributions at greater length in Section 3.3

3.1.1 Minimum winning price is provided by SSPs. In non-censored price auctions, the minimum winning price $\hat{b}$ is provided by the SSP after the auction regardless of the outcome. We build a machine learning model to estimate the distribution of $D_{\hat{b}|x}$. Let $\{(x^{(i)}, \hat{b}_i)\}_{i=1}^{n}$ be the training data set of bid samples, where $x^{(i)}$ is the input vector, $\hat{b}_i$ the minimum winning price provided by the SSP, and n is the total number of training samples. Let $f(\hat{b}; \alpha(x^{(i)}))$ be the corresponding PDF. Then we estimate α using maximum likelihood estimation (MLE):

$$\max_{\substack{\alpha \\ \hat{b}|x \in \mathcal{D}}} \sum_{i=1}^{n} \log f\left(\hat{b}_i; \alpha(x^{(i)})\right), \tag{3}$$

where $\mathcal{D}$ is a predetermined distribution family. The model structure for maximizing the log-likelihood is illustrated in Figure 1. Note that compared to x, α is of much lower dimension, namely $k \gg m$.
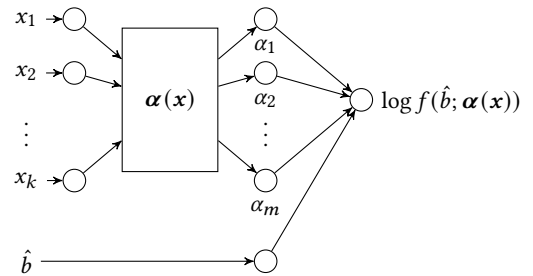

Figure 1: Model structure for PDF estimation when the minimum winning price is provided by SSPs

3.1.2 Minimum winning price is censored by SSPs. If the SSP does not provide the minimum winning price after the auction, the previous MLE approach will not work. In this case, we use an approach similar to that in [29] to estimate the CDF of $D_{\hat{b}|x}$, adding the flexibility of choosing the distribution family. Let $\{(x^{(i)}, b_i, \ell_i)\}_{i=1}^n$ be the training data set of bid samples, where $x^{(i)}$ is the input feature vector, b_i is the submitted bid price, and $\ell_i \in \{0, 1\}$ indicates if the bid was won. Let $F(b; \alpha(x^{(i)})) = \Pr(\hat{b}_i < b; \alpha(x^{(i)}))$ be the CDF of $D_{\hat{b}|x}$. Then the likelihood of winning the bid with submitted price b_i is $F(b_i; \alpha(x^{(i)}))$. Since we know the result of the auction, we formulate it as a prediction problem. More precisely, we estimate α by minimizing the loss between the likelihood of winning and the actual result:

$$\min_{\substack{\alpha \\ \hat{b}|x \in \mathcal{D}}} \sum_{i=1}^n L\left(F(b_i; \alpha(x^{(i)})), \ell_i\right), \qquad (4)$$

where $\mathcal{D}$ is a predetermined distribution family, and L is a loss function. In this paper, we use the well-known log loss:

$$L(F(b; \alpha(x)), \ell) = \ell \log F(b; \alpha(x)) + (1 - \ell) \log(1 - F(b; \alpha(x))).$$

The model structure for minimizing the log loss is illustrated in Figure 2. As can be seen, the win probability is first evaluated at bid price b for given CDF and the loss is then calculated with the binary feedback l.
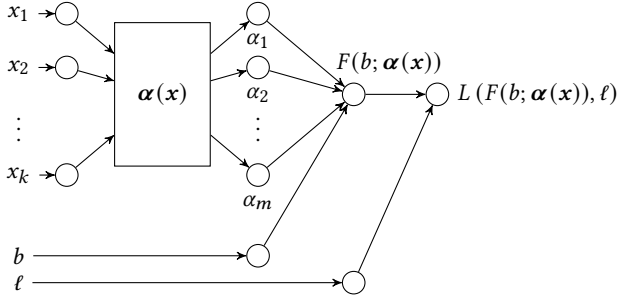

Figure 2: Model structure for CDF estimation when the minimum winning price is censored by SSPs

In Section 4 we will compare models built on different distribution families $\mathcal{D}$, as well as different structures of the block $\alpha(x)$.

3.2 Surplus Maximization

Regardless of whether the SSP provides the minimum winning price, let

$$s(b; V, x) \stackrel{\text{def}}{=} (V - b)\Pr(\hat{b} < b; \hat{b} \mid x)$$

be the expected surplus function with respect to $D_{\hat{b}|x}$, the distribution of the minimum winning price for the current input feature vector x. Given x and the bid price before shading V, the objective is to solve the following maximization problem:

$$\max_{b \in (0, V)} s(b; V, x).$$

If the underlying distribution $D_{\hat{b}|x}$ results in a surplus function that has a unique local extremum (maximum or minimum), we can adopt

the golden section search algorithm [24] shown in Algorithm 3.1 which converges in logarithmic time. It is a more numerically stable approach for most distributions, especially for gamma distribution and log-normal distribution, since no calculation of gradient is needed. The golden section search is more versatile and robust than other algorithms that require the calculation of gradients, which makes it more suitable for online implementation.

Algorithm 3.1 Golden Section Search for Surplus Maximization

Require:
1: • V: estimated value of the current ad opportunity
   • $s(b)$: expected surplus function;
   • $\epsilon > 0$: minimum valid interval length
   • N: maximum number of search steps
Ensure: $\beta > 0, V > 0$, and $s(b)$ has exactly one local maximum and no local minimum in $(0, V)$.
2: $b_{\min}, b_{\max} \leftarrow 0, V$
3: $gr \leftarrow (\sqrt{5} + 1)/2$
4: $x_1 \leftarrow b_{\max} - (b_{\max} - b_{\min})/gr$
5: $x_2 \leftarrow b_{\min} + (b_{\max} - b_{\min})/gr$
6: **for** $i = 1, 2, \ldots, N$ **do**
7:      **if** $s(x_1) > s(x_2)$ **then** $b_{\max} \leftarrow x_2$ **else** $b_{\min} \leftarrow x_1$
8:      **if** $b_{\max} - b_{\min} < \epsilon$ **then** break
9:      $x_1 \leftarrow b_{\max} - (b_{\max} - b_{\min})/gr$
10:      $x_2 \leftarrow b_{\min} + (b_{\max} - b_{\min})/gr$
11: **end for**
     return $(b_{\min} + b_{\max})/2$

3.3 Distribution Families

There are some constraints in choosing the distribution of minimum winning price to make economic sense. For example, its PDF should have a support between 0 and positive infinity. We mainly focus on truncated-normal, exponential, gamma, and log-normal distribution families and show that the surplus functions for them have a unique local extrema, so that Algorithm 3.1 is applicable to them all. We first introduce a few notations. Assuming that the minimum winning price $\hat{b}$ follows a probability distribution D with CDF $F(b)$ and PDF $f(b)$, then expected surplus and its first and second derivatives can be calculated as

$$s(b) = (V - b)\Pr(\hat{b} < b) = (V - b)F(b), \qquad (5)$$
$$s'(b) = (V - b)f(b) - F(b), \qquad (6)$$
$$s''(b) = (V - b)f'(b) - 2f(b). \qquad (7)$$

For simplicity, we don't explicitly write the above as functions of input feature vector x and bid price before shading V here.

We first show that for all truncated-normal, exponential, gamma, and log-normal distributions, $s''(b)$ has at most one root in $(0, V)$.

Let

$$f_{tn}(b) = \frac{\exp\left(-\frac{(b-\mu)^2}{2\sigma^2}\right)}{\sqrt{2\pi}\sigma\left[\Phi\left(\frac{B-\mu}{\sigma}\right) - \Phi\left(\frac{A-\mu}{\sigma}\right)\right]}, \text{ where } \sigma > 0 \text{ and}$$

$$-\infty \leq A < B \leq \infty, \qquad \text{(truncated-normal)}$$

$$f_e(b) = \lambda e^{-\lambda b}, \text{ where } \lambda > 0, \qquad \text{(exponential)}$$

$$f_g(b) = \frac{\beta^\alpha}{\Gamma(\alpha)} b^{\alpha-1} e^{-\beta b}, \text{ where } \alpha > 0, \beta > 0, \text{ and} \quad \text{(gamma)}$$

$$f_{ln}(b) = \frac{1}{\sqrt{2\pi}\sigma b} \exp\left(-\frac{(\ln b - \mu)^2}{2\sigma^2}\right), \text{ where } \sigma > 0.$$

$$\text{(log-normal)}$$

Throughout this paper, we use the truncated-normal distribution with $A = 0$ and $B = \infty$. Thus, for all these four distributions, the support of b are $[0, \infty]$.

LEMMA 1. *For any truncated-normal, exponential, gamma, or log-normal distribution of the minimum winning price, $s''(b)$ has at most one root in $(0, V)$. Furthermore, $s''(V) < 0$.*

PROOF. Using Equation (7), we can calculate $s''(b)$ for each of the listed distributions.

- For truncated-normal distribution we can show that

$$s''(b) = f_{tn}(b)\left[(V-b)(\mu-b)\sigma^{-2} - 2\right].$$

Let $g(b) = (V-b)(\mu-b)\sigma^{-2} - 2$. Note that $g(b)$ is quadratic and hence convex, and $g(V) = -2 < 0$. Thus $g(b)$ has at most one root in $(0, V)$, and so does $s''(b)$.

- For gamma distribution we can show that

$$s''(b) = \frac{f_g(b)}{b}\left[(\alpha - 1 - \beta b)(V - b) - 2b\right].$$

Let $g(b) = (\alpha - 1 - \beta b)(V - b) - 2b$. Again, $g(b)$ is quadratic and hence convex, and $g(V) = -2V < 0$. Thus $g(b)$ has at most one root in $(0, V)$, and so does $s''(b)$.
Since exponential distribution is a special case of gamma with $\alpha = 1$ and $\beta = \lambda$, the same proof holds for exponential distribution.

- For log-normal distribution we can show that

$$s''(b) = \frac{f_{ln}(b)}{b\sigma^2}\left[(\mu - \sigma^2 - \ln b)(V - b) - 2\sigma^2 b\right].$$

Let $g(b) = (\mu - \sigma^2 - \ln b)(V - b) - 2\sigma^2 b$. It's easy to verify that $g''(V) = (b+V)/b^2 > 0$, so $g(b)$ is convex. Since $g(V) = -2\sigma^2 V < 0$, $g(b)$ has at most one root in $(0, V)$, and so does $s''(b)$.

In summary, for all the listed distributions the corresponding $s''(b)$ has at most one root in $(0, V)$, and $s''(V) < 0$. □

Finally we show that for all the listed distributions the corresponding surplus function has one global maximum.

THEOREM 1. *For any truncated-normal, exponential, gamma, or log-normal distribution of the minimum winning price, the surplus function $s(b)$ has one global maximum and no local minimum in $(0, V)$.*

PROOF. From Lemma 1, for any truncated-normal, exponential, gamma, or log-normal distribution, $s''(b)$ has at most one root in $(0, V)$ and $s''(V) < 0$. Then only one of the following two cases could happen:

- $s''(b) < 0$ for all $b \in (0, V)$, in which case $s(b)$ is concave and hence has at most one local maximum and no local minimum.
- $s''(b)$ has a unique root $b_0 \in (0, V)$ such that $s''(b) > 0$ for all $b \in (0, b_0)$ and $s''(b) < 0$ for all $b \in (b_0, V)$, in which case $s(b)$ is convex in $(0, b_0)$ but concave in (b_0, V). Note that $s(0) = s(V) = 0$, then $s(b)$ must have at most one local maximum in (b_0, V) and no local minimum.

In both cases, $s(b)$ has at most one local maximum and no local minimum. Further, since $s(0) = S(V) = 0$ and $s(b) > 0$ for all $b \in (0, V)$, $s(b)$ must have one global maximum. □

4 OFFLINE EXPERIMENTS

In this section, we present comprehensive offline experiments on our DSP private bidding dataset. The following questions would be answered in the following sub-sections:

- **Q1:** Does lower log-loss or better distribution fit results in a higher surplus?
- **Q2:** How much is the surplus lift when minimum winning price is available in training the deep distribution network compared with when minimum winning price is not available?
- **Q3:** Will performance be improved by using more powerful network structures (deepFM etc.) that capture high-order feature interactions compared to logistic regression or Factorization Machines (FM)?

4.1 Dataset

The dataset we use for offline experiments is VerizonMedia DSP private bidding dataset on Adx exchanges. We extracted 12 fields through feature engineering, including exchange id, top level domain of the ad opportunity, sub domain, layout of the ad, position of the ad, device type, name of app, publisher id of the ad request, country, user local hour of the day, user local day of the week, if the user is new. There are billions of records, with millions of active features and minimum winning price available. We use 7 day's data to train the model and use 1 day's data to test. Notice that we train two types of models separately to simulate censored and non-censored first-price auctions. For the pdf estimation model which introduced in Fig. 1, minimum winning price are used as labels, while for the cdf estimation model in Fig. 2, the binary win or lose information (1/0) is used as label. We used the bid requests in the past 7 days for training to mitigate impact of the day of week pattern. Log-loss and surplus are the main two metrics we use in the offline experiment.

4.2 Minimum winning price provided by SSPs

The production bid shading algorithms for non-censored first-price auctions is Factorization Machine based point estimation algorithm. It uses Field-weighted Factorization Machine (FwFM) as the model structure and learns the optimal shading factor for each bid request

with an asymmetric loss function (penalize more when losing the bid) [15]. To have a fair comparison with the baseline, we uses same model structure: FwFM as the model structure in Figure 1, and conduct experiments of different distributions we introduced in Section 3.3. For surplus metric, we only show the percentage of lift compared to the production model for privacy reason.

Table 1 summarizes the results and we do observe the correlation between log-loss and surplus, in the sense that lower log-loss results in higher surplus. Among all the distributions, log-normal has the best performance with 9.7% surplus lift compared to the current production algorithm possibly due to its capability to better model the long-tail distribution of the minimum winning price.

Model	Log loss	Surplus Lift Percentage
truncated-normal	0.87	1.45%
exponential	0.68	5.12%
gamma	0.58	6.85%
log-normal	0.56	9.65%

Table 1: Performance of different distributions compared to production FwFM algorithm where highest competing bid price is not censored by the SSP

4.3 Minimum winning price not provided by SSPs

To simulate the censored first-price auctions where minimum winning price feedback is not available. We train the deep distribution network on the same dataset while not using minimum winning price information during training. The loss function is defined in Section 3.1.2. The production algorithm on censored SSPs is win-rate distribution algorithm, in which winning probability function is estimated as a sigmoid function [29].

Model	Log loss	Surplus Lift Percentage
truncated-normal	1.14	0.10%
exponential	0.89	3.29%
gamma	0.72	3.58%
log-normal	0.68	5.32%

Table 2: Performance of different distributions compared to production win-rate distribution algorithm

The results are presented in Table 2, and we observe the same correlation between log-loss and surplus again, which answers **Q1**. Notice that the log-loss is not applicable to the production algorithm since its definition of loss function is different from our model. Additionally, in comparison with non-censored training results in Table 1, there is a slightly drop in surplus performance lift for all distributions. This shows the importance of minimum winning price feedback in deep distribution network training which answers **Q2**.

As can be seen from the offline results corresponding to the cases with minimum winning price in Table 1 and the cases without minimum winning price in Table 2, log-normal distribution results

in the lowest log-loss and the highest surplus. It indicates that among all the above distributions that we considered so far, log-normal fits the minimum winning price better. Therefore, we will choose log-normal as the output layer distribution in the online A/B test in Section 5.

4.4 Network structure comparison

In this subsection, we present the results on comprehensive experiments on applying well-known click-through rate prediction models like FM, FwFM, DeepFM, Wide & Deep [10, 13, 17, 27, 28, 30], to deep distribution network, as the deep distribution network structure, in learning the distribution of minimum winning price . The results are shown in Table 3 which answers **Q3**. It can be seen that deep models, like deepFM, wide & deep, which capture high-order feature interaction are able to learn a better distribution with relatively low log-loss, and thus have a better surplus compared to linear model LR, and shallow models FM, FwFM.

network structure	Log loss	Surplus Lift Percentage
LogReg	0.718	/
FM	0.569	4.52%
FwFM	0.558	4.70%
wide & deep	0.522	6.36%
deepFM	0.521	7.10%

Table 3: Performance of different network structures with lognormal as pre-defined distribution

Based on the offline experiments results and online latency constraint, we eventually decide to use FwFM as the network structure and log-normal distribution for online experiments. However, we show the potential of more complex deep network structure, if the online latency requirements can be satisfied.

5 ONLINE EXPERIMENTS

In this section, we will briefly introduce the real-time bid shading serving module we implemented, which serves billions of bid requests per day in one of the world's largest DSP. The online experimental results, including Return on Investment (ROI) for advertisers, are also presented, which further prove that deep distribution network outperforms other existing bid shading algorithms in literature.

5.1 Bidding System Overview

Our DSP is a single platform that brings programmatic, premium, and its native marketplace inventory, formats, targeting and measurement together. We provide an overview of the bid shading aspects of the system, as illustrated in Figure 3.

When a user visits a web page with an ad opportunity, an ad request is sent to the SSP responsible for selling it. The SSP then packages available user and page information into bid requests and send them to multiple DSPs for auction. Upon receiving a bid request, within hundreds of milliseconds our bidding system goes through various stages such as fetching user profiles, ads targeting,
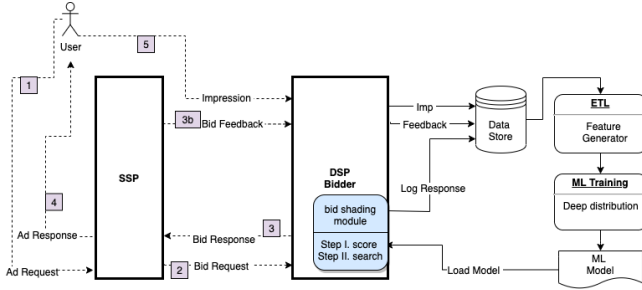
Figure 3: An Overview of VZDSP Bid Shading System

making click or conversion predictions, etc., and eventually estimates the *true value* of the underlying ad opportunity for a selected candidate ad, or multiple candidates.

For second-price auctions, this *true value* would be the bid price. For first-price auctions, we have an additional *bid shading module*, as shown on the right-hand side of the flow chart. In this bid shading module, historical bid information, including impressions and bid feedback from SSPs, are used to generate features and train bid shading models. The model is periodically updated and loaded into the real-time bidding system, and used to shade the *true value* to produce the final bid price, which, as part of the bid response, is sent back to the SSP.

The SSP collects bid responses from all participating DSPs, determines the winner, and sends an ad response to the web page. If our DSP wins the auction, an impression with the selected ad would be shown on the user's web page, and relevant user information would be sent back to us. The SSP also sends bid feedback to its bidders, but it may or may not include the minimum winning price.

5.2 Online Evaluation Metrics

The most important key performance metrics for campaign delivery that the DSP system seeks to optimizes towards are CPM (Cost per thousand impressions), CPC (Cost per click) and CPA (Cost per action). For our online A/B test experiments of the bid shading algorithms described in this paper, we mainly focus on campaigns with one of those 3 optimization goal types, which cover more than 80% of total DSP revenue. For a given campaign spend budget, the DSP optimization system seeks to minimize the campaign's CPM, CPC or CPA as indicated by its optimization goal. For convenience, we use the term effective cost per event (eCPX) to denote either CPM, CPC or CPA.

Since optimization goals are defined at the campaign level, there is a need to define metrics to measure impact of algorithm improvements (in our case bid shading) across multiple campaigns. Simple aggregations of events (e.g. actions) and cost across campaigns to define a simple aggregated eCPX metric are not good metrics, since a few campaigns can dominate such metrics if it turns out they have orders of magnitude more events than the rest of the campaigns, even though their cost (campaign spend) is as high as any other campaign's spend. Moreover, it's not fair to compare two algorithms when their spends are different since the one with more spend tends to have higher eCPX. Thus, we are going to introduce two novel online DSP business-related metrics: campaign

level eCPX statistics, and Bidder Performance Index (BPI), which can be used to measure improvements in campaign performance and ROI for our advertisers.

5.2.1 *Campaign Level eCPX statistics*.
A natural way to put together eCPX metrics across campaigns is to generate statistics on improvements measured at the campaign level, i.e. based on the eCPX of control and test buckets for each campaign. To avoid the asymmetry implied by the usual percentage difference in eCPX between control and test, we define the log of eCPX ratio for each campaign as the following,

$$r_{\text{ecpx}} = \log\left(\frac{\text{eCPX}_{\text{Test}}}{\text{eCPX}_{\text{Control}}}\right).$$

For a campaign that has similar spend in both control and test bucket, if $r_{\text{ecpx}} < 0$, then the test algorithm outperforms the production model. This provides a metric of improvement (or deterioration) in eCPX that is symmetric in the sense that the absolute value of the metric remains unchanged if control and test roles are interchanged. A histogram depicting the distribution of r_{ecpx} is shown in Fig 4 for CPA goal type campaigns. Based on the r_{ecpx} distribution, we can measure that 71.5% of campaigns have better eCPX, and the median and mean of r_{ecpx} are, respectively, -0.035 and -0.038. Since campaigns can have very different budget and spend, the weighted histogram of r_{ecpx} is shown in Fig 5 where the weights are the spend of each campaign.

5.2.2 *Bidder Performance Index (BPI)*.
Minimizing eCPX for a given spend amount is equivalent to maximizing value or Return to advertisers under the same amount of spend (Cost), where Return to advertisers is the monetary value of the events (impressions, clicks or actions) driven by the campaign. This motivates the so-called BPI metric, defined as:

$$\frac{(\text{Return}_{\text{test}} - \text{Cost}_{\text{test}}) - (\text{Return}_{\text{control}} - \text{Cost}_{\text{control}})}{\text{Cost}_{\text{control}}}$$

where Return and Cost can be aggregated across campaigns. The numerator shows the extra Return the new algorithm brings to the advertiser compensated by the extra cost it may also incur. Notice that it can be negative if the test algorithm is no better than the control one. The BPI metric is an aggregation-based type of metric that is less prone to be dominated by campaigns with very large number of events as explained before.

5.3 Online A/B test results

After rolling out the deep distribution network algorithm with FwFM network structure and log-normal distribution on Adx, one of the largest SSPs, we were able to monitor its online performance by maintaining a percentage of traffic that was randomly allocated to each algorithm. There are thousands of campaigns running everyday in our DSP, the summary of these campaigns under our A/B test environment during 3 consecutive days is shown in Table 4.

It can be seen that for all goal types, the median of r_{ecpx} are negative values, showing a overall better performance of our proposed algorithm. The weighted median, where the weights are the spend of each campaign, are also negative across all goal types. A majority of campaigns have a better eCPX performance on test bucket as can be seen from the third row of better campaign percentages. And
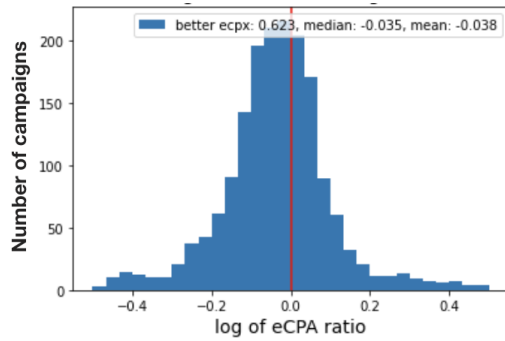
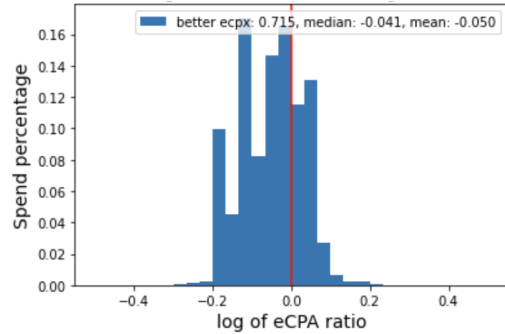**Figure 4: Histogram of log eCPX ratio: CPA campaign as an example**

Figure 5: Histogram of log eCPX ratio weighted by campaign spend: CPA campaign as an example

These campaigns take up to 70% to 80% of the total spend. In all, from the online A/B test results, it can be shown that our proposed bid shading algorithm generates a significantly better performance than the current model in production.

Metric	campaign goal type		
	CPA	CPC	CPM
median of r_{ecpx}	-3.5%	-0.9%	-6.2%
median of spend weighted r_{ecpx}	-4.1%	-5.4%	-5.7%
better campaigns	62.3%	54.5%	81.6%
better spend	71.5%	80.6%	71.8%
BPI	+8.57%	+2.35%	+2.35%

Table 4: Online A/B test performance by goal types

6 CONCLUSIONS

In this work, we propose the deep distribution network for bid shading, which can be applied for both censored and non-censored first-price auctions. The parametric conditional distribution of minimum winning price is learnt through a FwFM network based on selected features of the bid request. For several well-known distributions, we proved that the resulting surplus function has a unique

local maximum. Based on such property, an efficient golden-section search algorithm was applied at the real-time in finding the optimal bid price that maximizes expected surplus. In offline experiments, the proposed model out-performed existing state of the art bid shading algorithm on both censored and non-censored scenarios with respectively 9.7% and 5.3% surplus lift. Online A/B test showed that the proposed algorithm increases surplus by +14.3% and brings +2.4%, +2.4%, +8.6% ROI lift for impression based (CPM), click based (CPC), and conversion based (CPA) campaigns respectively. The deep distribution network has been successfully deployed in one of the biggest DSPs in the world, serving billions of bid requests every day. Another major advantage of our framework is that, the overall structure with a deep neural network and a parametric distribution output layer, can be easily generalized. In the future, we will continue exploring more powerful and run-time efficient neural network structures, combining with other single or multi mode distributions for further improved performance.

REFERENCES

[1] 2021. Differential pricing. *Wikipedia* (January 2021). https://en.wikipedia.org/wiki/Price_discrimination

[2] AppNexus. 2018. Demystifying Auction Dynamics for Digital Buyers and Sellers. https://www.appnexus.com/sites/default/files/whitepapers/49344-CM-Auction-Type-Whitepaper-V9.pdf

[3] Paul Belleflamme and Martin Peitz. 2015. Personalized pricing, Group pricing, Menu pricing, Uniform pricing. *Industrial Organization: Markets and Strategies, Slides accompanying the book* (2015). http://web.uvic.ca/~pcourty/IO%20Pricing.pdf

[4] Jason Bigler. 2019. Rolling out first price auctions to Google Ad Manager partners. https://www.blog.google/products/admanager/rolling-out-first-price-auctions-google-ad-manager-partners/

[5] A. Blum and Hartline J. D. 2005. Near-optimal online auctions. *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms* (January 2005), 1156–1163.

[6] Han Cai, Kan Ren, Weinan Zhang, Kleanthis Malialis, Jun Wang, Yong Yu, and Defeng Guo. 2017. Real-time bidding by reinforcement learning in display advertising. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. 661–670.

[7] E.C. Capen, R.V. Clapp, and W.M. Campbell. 1971. Auctions and Bidding. *Journal of Petroleum Technology* 23, 6 (1971), 641–643.

[8] Bhaskar Chakravorti, William W Sharkey, Yossef Spiegel, and Simon Wilkie. 1995. Auctioning the airwaves: the contest for broadband PCS spectrum. *Journal of Economics & Management Strategy* 4, 2 (1995), 345–373.

[9] V Chari and Robert Weber. 1992. How the US Treasury should auction its debt. *Federal Reserve Bank of Minneapolis Quarterly Review* 16 (1992), 4. http://kylewoodward.com/blog-data/pdfs/references/chari+weber-quarterly-review-1992A.pdf

[10] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide & Deep Learning for Recommender Systems. *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, 7–10.

[11] John M Crespi and Richard J Sexton. 2005. A Multinomial logit framework to estimate bid shading in procurement auctions: Application to cattle sales in the Texas Panhandle. *Review of industrial organization* 27, 3 (2005), 253–278.

[12] Easley David and Kleinberg Jon. 2010. Networks, Crowds, and Markets: Reasoning About a Highly Connected World. , 43–44 pages. https://www.cs.cornell.edu/home/kleinber/networks-book/networks-book-ch09.pdf

[13] Wei Deng, Junwei Pan, Tian Zhou, Deguang Kong, Aaron Flores, and Guang Lin. 2021. DeepLight: Deep Lightweight Feature Interactions for Accelerating CTR Predictions in Ad Serving. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. WSDM'21, 922–930.

[14] Getintent. 2017. RTB Auctions: Fair Play? https://blog.getintent.com/rtb-auctions-fair-play-3b372d505089

[15] Djordje Gligorijevic, Tian Zhou, Bharatbhushan Shetty, Brendan Kitts, Shengjun Pan, Junwei Pan, and Aaron Flores. 2020. Bid Shading in The Brave New World of First-Price Auctions. In *The 29th ACM International Conference on Information and Knowledge Management*. CIKM'20.

[16] Google. 2019. Real Time Bidding Protocol, Release Notes. https://developers.google.com/authorized-buyers/rtb/relnotes#updates-2019-03-13

[17] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, 1725–1731.

[18] Yue He, Xiujun Chen, Di Wu, Junwei Pan, Qing Tan, Chuan Yu, Jian Xu, and Xiaoqiang Zhu. 2021. A Unified Solution to Constrained Bidding in Online Display Advertising. In *Proceedings of the 27th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*.

[19] Authorized Buyers Help. 2020. Transition of Ad Exchange inventory to a unified first-price auction complete. https://support.google.com/authorizedbuyers/answer/6077702?hl=en

[20] Ali Hortaçsu, Jakub Kastl, and Allen Zhang. 2018. Bid shading and bidder surplus in the us treasury auction system. *American Economic Review* 108, 1 (2018), 147–69.

[21] B. Hovaness. 2018. Sold for more than you should have paid. https://www.hearts-science.com/sold-for-more-than-you-should-have-paid/

[22] Niklas Karlsson and Qian Sang. 2021. Adaptive bid shading optimization of first price ad inventory. In *The 2021 American Control Conference*. ACC'21.

[23] N. B. Keskin and Zeevi A. 2014. Dynamic pricing with an unknown demand model: Asymptotically optimal semi-myopic policies. *Operations Research* 62, 5 (2014), 1142–1167.

[24] Jack Kiefer. 1953. Sequential minimax search for a maximum. *Proceedings of the American mathematical society* 4, 3, 502–506.

[25] B. Kitts and K. Hetherington-Young. 2005. Price Optimization in Grocery Stores with Cannibalistic Product Interactions. *Proceedings of the First Workshop on Data Mining Case Studies, Fifth IEEE International Conference on Data Mining (ICDM 2005)* (November 2005), 74–91. http://dataminingcasestudies.com/DMCS_WorkshopProceedings25.pdf

[26] R. Kleinberg and T. Leighton. 2003. The value of knowing a demand curve: Bounds on regret for online posted-price auctions. *44th Annual IEEE Symposium on Foundations of Computer Science* (2003), 594–605.

[27] Junwei Pan, Yizhi Mao, Alfonso Lobos Ruiz, Yu Sun, and Aaron Flores. 2019. Predicting different types of conversions with multi-task learning in online advertising. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2689–2697.

[28] Junwei Pan, Jian Xu, Alfonso LobosRuiz, Wenliang Zhao, Shengjun Pan, Yu Sun, and Quan Lu. 2018. Field-weighted factorization machines for click-through rate prediction in display advertising. *In Proceedings of the 2018 World Wide Web Conference*, 1349–1357.

[29] Shengjun Pan, Brendan Kitts, Tian Zhou, Hao He, Bharatbhushan Shetty, Aaron Flores, DjordjeGligorijevic, Junwei Pan, Tingyu Mao, San Gultekin, and Jianlong Zhang. 2020. Bid Shading by Win-Rate Estimation and Surplus Maximization. In *The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. AdKDD'20, San Diego, California, USA.

[30] Steffen Rendle. 2010. Factorization Machines. *Proceedings of the 2010 IEEE International Conference on Data Mining*, 995–1000.

[31] Rubicon. 2018. Bridging the Gap to First-Price Auctions: A Buyer's Guide. http://go.rubiconproject.com/rs/958-XBX-033/images/Buyers_Guide_to_First_Price_Rubicon_Project.pdf

[32] Rubicon. 2018. Principles for a Better Programmatic Marketplace, Open Letter from Rubicon, SpotX, OpenX, Pubmatic, Telaris, and Sovr. https://rubiconproject.com/insights/thought-leadership/principles-better-programmatic-marketplace-open-letter-advertisers-publishers/

[33] Ilya Segal. 2003. Optimal Pricing Mechanisms with Unknown Demand. *American Economic Review* 93, 3 (June 2003), 509–529. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=297674

[34] Hermann Simon. 1989. Price Management. (August 1989).

[35] Sarah Sluis. 2017. Big Changes Coming To Auctions, As Exchanges Roll The Dice On First-Price. https://www.adexchanger.com/platforms/big-changes-coming-auctions-exchanges-roll-dice-first-price/

[36] S. Sluis. 2017. Explainer: More On The Widespread Fee Practice Behind The Guardian's Lawsuit Vs. Rubicon Project. https://adexchanger.com/ad-exchange-news/explainer-widespread-fee-practice-behind-guardians-lawsuit-vs-rubicon-project/

[37] Di Wu, Xiujun Chen, Xun Yang, Hao Wang, Qing Tan, Xiaoxun Zhang, Jian Xu, and Kun Gai. 2018. Budget constrained bidding by model-free reinforcement learning in display advertising. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 1443–1451.

[38] Wush Chi-Hsuan Wu, Mi-Yen Yeh, and Ming-Syan Chen. 2015. Predicting Winning Price in Real Time Bidding with Censored Data. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Longbing Cao, Chengqi Zhang, Thorsten Joachims, Geoffrey I. Webb, Dragos D. Margineantu, and Graham Williams (Eds.). ACM, Sydney, NSW, Australia, 1305–1314. https://doi.org/10.1145/2783258.2783276

[39] Wush Chi-Hsuan Wu, Mi-Yen Yeh, and Ming-Syan Chen. 2018. Deep Censored Learning of the Winning Price in the Real Time Bidding. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Yike Guo and Faisal Farooq (Eds.). ACM, London, UK, 2526–2535. https://doi.org/10.1145/3219819.3220066

[40] Xun Yang, Yasong Li, Hao Wang, Di Wu, Qing Tan, Jian Xu, and Kun Gai. 2019. Bid optimization by multivariable control in display advertising. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1966–1974.

[41] Yunhong Zhou, Deeparnab Chakrabarty, and Rajan Lukose. 2008. Budget constrained bidding in keyword auctions and online knapsack problems. In *International Workshop on Internet and Network Economics*. Springer, 566–576.

[42] Christine Zulehner. 2009. Bidding behavior in sequential cattle auctions. *International Journal of Industrial Organization* 27, 1 (2009), 33–42.