5th International Conference on Advanced Research Methods and Analytics (CARMA2023)
Universidad de Sevilla, Sevilla, 2023
DOI: http://dx.doi.org/10.4995/CARMA2023.2023.16452

Optimizing floor price in Real Time Bidding

David Gávez[1], Víctor Dugo[2]
[1]Departamento de Estadística e Investigación Operativa , Universidad de Sevilla, Spain, [2]Departmento de Análisis Económico y Economía Política, Universidad de Sevilla, Spain

Abstract

Between 150 and 200 words briefly specifying the aims of the work, the main results obtained, and the conclusions drawn

AdNetwork companies are very much a part of today's new digital marketing methods. This paper aims to develop an algorithm that solves the problems of AdNetwork companies in setting optimal floor prices. Establishing the optimal starting price for the bid is equivalent to setting the price that maximises revenue, which is optimal for the publisher and the AdNetwork company. In this market, that price will balance two opposite scenarios: a high floor price could lead to some impressions unsold, while a low floor price could be insufficient to reach profit margins. The contribution is twofold. First, this paper extends the problem of optimal price flor in real time bidding auctions for advertising in current scenarios where a DSP (Demand Side Platform) acts as a filter and only one bid is received by the AdNetwork and thus, the price paid corresponds to the reserve price. and, moreover, it is implemented in reality with a pseudo-algorithm (not provided for commercial reasons). It allowed to be implemented in a real case scenario for three publishers, obtaining an average increase of revenue of 127%.

Keywords: *AdNetwork; DSP; optimal floor price; regret.*

1. Introduction

Digital marketing has become companies' primary advertising tool and has changed how firms communicate with customers (Chaffey & Ellis-Chadwick, 2019). Within this field, some companies operate through what is known as an 'AdNetwork'. An AdNetwork can essentially be defined as a network that acts as a link between advertisers and publishers (sites or websites that sell their advertising space), thus generating income for publishers and placing advertisers' resources in the most appropriate places for the advertising content (D'Annunzio & Russo, 2020; Tahaei & Vaniea, 2021). An AdNetwork has a dual revenue source: on the one hand, it generates income from the advertising campaigns it sells or manages for advertisers. On the other hand, it administers the publishers' advertising space and takes a share of the revenue generated by the advertisements shown (impressions) for offering the technology, posting and linking. The revenue share determines this part of the price.

The management of publisher inventory is done by purchasing advertising impressions in this inventory. The AdNetwork will then seek advertisers who will pay the price ("revenue" from now on) to advertise in that inventory by purchasing those impressions. It does this through blinded second-price real-time auctions (Myerson, 1981) in which the auction to acquire those inventory impressions is won by the advertiser who bids the most but pays the price of the second highest bid, which is the final "revenue". These auctions have a floor or reserve price built into them, which is the minimum price the publisher will receive for ad impressions in its inventory. Impressions are the number of times an ad appears in that inventory, and the price is usually per thousand impressions.

To this "revenue" is applied the share agreement so that the publisher will receive the agreed percentage of the revenue, and if this percentage results in a value below the floor price, the floor price is paid, and the company (AdNetwork) reduces its profit margin. Thus, if there is a revenue share commitment of 70/30 (70% corresponds to the publisher while the remaining 30% is the AdNetwork company's profit for its services), this means that Adnetwork auctions the impressions from the publisher's inventory at a price where at least 70% of that price corresponds to the floor price set by the publisher. Thus, for example, if a publisher sets a floor price of 7 euros, Adnetwork will put it up for auction at 10 euros in order to maintain its revenue share of 70/30 and will not accept bids below that price, which is the price that allows it to meet the publisher's floor price and maintain its profit.

Therefore, establishing the optimal starting price for the bid is equivalent to setting the one that maximizes revenue, which is optimal for the publisher and the "AdNetwork" company. In this market, that price will balance two opposite scenarios: a high floor price could lead to some impressions unsold, while a low floor price could be insufficient to reach profit margins. The development of the article is as follows. In Section 2, The proposed approach and different scenarios are presented. In Section 3 the problem is mathematically formalized and developed. Section 4 presents some particular cases due to different auction platform practices. Finally, Section 5 provides some actual results and conclusions.

2. Justification of the approach and scenarios.

2.1. Justification of the approach

The reliability of the algorithm is established according to the concept of regret. This concept answers the question: at the end of T iterations of the algorithm, where we have all the information of what happened, what would happen if it had simply applied the same decision rule **h** at each iteration? One could calculate the loss of this fixed hypothesis by adding the personal loss of the T iterations. If this value is less than the loss incurred by doing different actions, the decision maker is incurring *regret*, which is the difference between these two losses because we could have chosen a single action each iteration and obtained better results than we did. Typically, the regret is calculated with respect to the optimal strategy known after the period is over, i.e. ex-post, so, at each iteration, it is necessary to choose the action that is understood to minimize the regret (although this is not known until the end of the period).

Let us assume that each loss for each iteration of T is between 0 and 1, so the total loss at the end of the period will be between 0 and T. For a hypothesis and a loss function, if the algorithm guarantees that for all possible states, the regret is $\bar{O}(T)$, it means that as T tends to infinity, the average regret per iteration tends to zero (since $\bar{O}(\cdot)$ is the rate at which regret converges to zero in each iteration), and there are T iterations.

In other words, if we design an algorithm and implement it T times, it incurs a loss after that period. The goal is to avoid the situation where seen in retrospect (a posteriori), the algorithm has incurred a lower loss with a constant rule of decision. Thus, the regret of an algorithm is the difference between the loss of the algorithm and the loss from using the constant alternative.

Translating this to our problem, the algorithm sets a floor price. After the period, the revenue obtained is compared with the one that would have resulted from applying the optimal price calculated a posteriori based on the actual data and the loss is obtained for having set that price and not the optimal one in that period. This comparison is the regret. The algorithm has been set to have a regret $\bar{O}(T^{1/2})$, which is a good result and means that when T tends to infinity, the regret converges to zero in $T^{1/2}$, which implies that it needs a smaller number of steps for the regret to converge.

2.2. Scenarios

2.2.1. Initial scenario

In an initial scenario, AdNetwork companies faced repeated auctions as a seller, each with a different number of bidders. Each bidder will submit a bid, which will be an unknown and random value to other bidders' eyes. The "revenue" is the second highest bid price, and the aim is to maximize the profit by considering only the "revenues" from previously recorded bids. A fair starting price (reserve or floor price) must be set to do this. The algorithm established a mechanism to find the optimal starting price that maximizes "revenue" by accessing only two pieces of information: the final "revenues" of the previous auctions and the number of bidders that finally participate in each auction (both pieces of information are obtained as outputs given by the digital ad auction platforms). After the algorithm's development, the game's initial rules changed.

2.2.2. Current scenario

In the current scenario, the different bidders have been replaced, and bids are now placed through Google's DSP, and Google only sends the winning bid. Under these new conditions, there is only access to one bid above the floor price; therefore, whatever the bid's value is, it ends up paying the floor price with the consequent loss of profitability. This is why it is necessary to create a new algorithm to establish an optimal price within the new market rules.

3. Formal problem statement and development

3.1. Mathematical formulation

Without loss of generality, the algorithm has been designed for prices between 0 and 1, with 0 being the minimum price and 1 the maximum possible price. This only requires rescaling the actual prices according to these limits. The algorithm launches a reserve price, observes what happened and does the necessary calculations. Based on these calculations, it chooses a new price to maximize the expected revenue and launches it, repeating the process. The algorithm was first proposed in (Cesa-Bianchi et al., 2015) as follows.

The firm conducts an auction to sell an item. In the initial scenario, after the auction, it collects $m \geq 2$ bids: $B_1, B_2, \ldots, B_m$, which are observations of m independent and identically distributed (i.i.d.) random variables. As indicated, prices will be between zero and one, so $B_i \in [0,1]$, $i = 1, \ldots, m$. These random variables have a common distribution function (since they are i.i.d) F, arbitrary and unknown, which will show the probability that a bid is below a certain value. We will denote $B^{(1)}, B^{(2)}, \ldots, B^{(m)}$ by the statistical order of the bids such that $B^{(1)} \geq B^{(2)} \geq \cdots \geq B^{(m)}$.

The algorithm will set a starting price (reserve price hereafter) $p \in [0,1]$ for the auction, and after the auction is conducted, it observes the revenue $R(p)$, which will depend on the chosen reserve price, and the values of the bids, i.e. $R(p) = R(p; B_1, B_2, \dots, B_m)$ which is defined as:

$$R(p) = \begin{cases} B^{(2)} & if \quad p \leq B^{(2)} \\ p & if \quad B^{(2)} < p \leq B^{(1)} \\ 0 & if \quad p > B^{(1)} \end{cases}$$

That is, if bids are received below p (or no bids are received), the item is not sold, and the "revenue" is zero, and if bids are received above that price, the item is sold to the bidder who bids $B^{(1)}$ at the price of the second highest bid, i.e. $B^{(2)}$, this $B^{(2)}$ being the revenue. If the item is sold, the middle condition guarantees that the revenue will always have the reserve price as the minimum price. However, this information is only obtained a posteriori. When the algorithm launches a price p, it does so, expecting a revenue $\mu(p) = E[R(p)]$, which is the expected revenue when the algorithm uses the price p. With the appropriate mathematical manipulations, the expected revenue can be rewritten as:

$$\mu(p) = \int_p^1 x\, dF_2(x) + pP(B^{(2)} < p \leq B^{(1)}) = E[B^{(2)}] + \int_0^p F_2(t)\, dt - p(F(p))^m$$

where $F_2(x)$ denotes the probability that $B^{(2)}$, is less than or equal to x (distribution function of $B^{(2)}$ and $(F(p))^m$ is the joint distribution function of $B^{(1)}, B^{(2)}, \dots, B^{(m)}$ and the price that maximizes it: $p^* = \arg \underset{p \in [0,1]}{max} \mu(p)$

This expected revenue will depend on the bid distribution function, i.e. F. The algorithm allows for cases where more than one bid is received as long as this number constitutes a percentage less than the parameter α to be defined later.

However, in the current scenario where the platform registers only one bid (winning bid in a DSP where there are several bidders) and therefore the advertiser always pays the floor price, and there is only access to the value of the winning bids, which are always (or 1-α% of the total number of times) the only one above the floor price and to the number of bidders in the DSP, if P_F is the advertiser floor price agreed with the publisher, the revenue function is:

$$R(p) = \begin{cases} P_F & if \quad B^{(1)} \leq P_F \\ 0 & if \quad P_F > B^{(1)} \end{cases}$$

And the scenario differs from the one analyzed in (Cesa-Bianchi, Gentile and Mansour, 2015). It is different in that information is only available for the highest bid and not for the second bid, as the floor price is always paid when only one bid is received from the Demand Side Platform. That is, everything must be inferred from the distribution function of the highest bid F_1. Let F_1 denote the distribution function of $B^{(1)}$. This function will indicate the probability that the winning bid is less than or equal to a certain value. Thus, $F_1(x)$ indicates the probability that the value of the highest bid, i.e. $B^{(1)}$, is less than or equal to x, while $F_2(x)$, as indicated, indicates the probability that $B^{(2)}$, is less than or equal to x.

At this point, it is necessary to establish a lemma whose demonstration has been developed but can also be mathematically intuited. The algorithm is going to raise the floor price in order to receive it as revenue constantly. It will raise it according to the values of the first bids and the revenue it obtains from the net increase in impressions that it stops obtaining when advertisers are unwilling to pay these higher floor prices. In this circumstance, if the algorithm can only access the winning bids on the floor price and the number of bidders in the DSP, it is possible to establish the revenue function and optimal price as:

$$p^* = \arg \max_{p \in [0,1]} \mu(p) = \arg \max_{p \in [0,1]} E\left[B^{(2)}\right] + \int_0^p F_2(t)dt - p(F(p))^m$$

$$\equiv \arg \max_{p \in [0,1]} E\left[B^{(1)}\right] + \int_0^p F_1(t)dt - p(F(p))^m = \arg \max_{p \in [0,1]} \mu^{(1)}(p)$$

As the function $(F(p))^m$ is not available, an approximation is computed from what is available, which is the distribution $F(1)$.

$F_1(p) = \beta((F(p))^m) = m((F(p))^m)\frac{m-1}{m} - (m-1)((F(p))^m)$ for $m \geq 2$

So: $$\mu^{(1)}(p) = E[B^{(1)}] + \int_0^p F_1(t)\, dt - p\beta^{-1}(F_1(p))$$

If $m = 1$, then the joint distribution function $(F(p))^m$ corresponds to the observed distribution, i.e. $F_1(p)$.

From here, the designed algorithm works in each auction as follows. In auction t, it will set the price p_t, and will have a revenue after the auction of $R_t(p_t) = R(p_t; B_{t,1}, B_{t,2}, \dots, B_{t,m})$ which is a function of the random variables $B_{t,1}, B_{t,2}, \dots, B_{t,m}$ in the auction or time t. The price given by the algorithm will depend on the previously observed $B^{(1)}$ and the floor price, and therefore on the past bids, as it learns and updates from them.

Thus, given a sequence of reserve prices $p_1, p_2, \dots, p_T$ set by the algorithm, the cumulative regret up to T will be given by:

$$\Sigma_1^T (\mu(p^*) - \mu(p_t))$$

Therefore, the regret (the reliability or goodness of an algorithm) will be a random variable since it depends on p_t, which will depend on the previous revenues that will depend on $B_1, B_2, \dots, B_m$.

It is important to note that we do not have access to the actual distribution of $B^{(1)}$, which is unknown, but to its empirical distribution function, which allows us to calculate the equivalent revenue $\mu^{(1)}(p)$ that provides the same maximizer as the expected revenue $\mu(p)$.

The algorithm works in stages. In each stage, the algorithm is run a certain number of times with the same reserve price. This is necessary to obtain the empirical distribution function of $B^{(1)}$, i.e. F_1. In principle, it is assumed that the algorithm will run a total number of T times.

- Stage 1 will contemplate T_1 auctions (implementations of the algorithm), and therefore the price it will use will be $p_t = \hat{p}_1;\ t = 1, \dots, T_1$.
- Stage 2 will contemplate T_2 auctions (implementations of the algorithm), and therefore the price it will use will be $p_t = \hat{p}_2;\ t = T_1 + 1, \dots, T_1 + T_2$.
- And so on.

In this way, the algorithm will produce reserve prices of $0 = \hat{p}_1 \leq \hat{p}_2 \leq \dots \leq 1$. They are set from an interval built according to a signification level α, choosing the price from this interval that minimizes risk subject to constraints related to the distribution function. The total number of stages is denoted as S (stages). It is shown mathematically that for the algorithm to have the agreed regret, each stage must have a number of implementations or auctions $T_i = T^{1-2^{-i}}$. From here, the number of stages, or at least their upper limit, can be determined so that $S \leq \lceil 2\log_2 \log_2 T \rceil$, i.e. it shall be set to the smallest integer not less than $2\log_2 \log_2 T$. The total cumulative regret of the algorithm shall be:

$$\Sigma_1^S (\mu(p^*) - \mu(p_i)) T_i$$

4. Special cases

4.1 Treatment of the algorithm when the number of bids is not known

In this case, the number of bids is not known. However, a limited number of different floor prices can be set for different advertisers. In this way, each advertiser who wants to advertise sees a different price set, depending on the algorithm that predicts how much they are willing to pay. According to empiric approaches (Seljan et al., 2014; Ballesteros et al., 2015), it could be assumed that the number of bidders follows a discrete normal distribution:

$$H(m) = P(M = m) = \frac{e^{\frac{-1}{2\sigma^2}(m-\mu_m)^2}}{\Sigma_{m_i} e^{\frac{-1}{2\sigma^2}(m_i-\mu_m)^2}}, m_i = -\infty, \dots, -1, 0, +1, \dots, +\infty$$

Thus, the expected revenue will be: $\mu(p) = E_M E[R^M(p)] = E_M E\left[B_M^{(2)}\right] + \int_0^p E_M[F_2, M](t)dt - pE_M[F^M](p)$

Where $E_M[F^M](p)$ can be estimated from the support function $T(x) = \Sigma_{m=2}^{\infty} H(m)x^m$ and its auxiliary function $A(x) = T(x)(1-x)T'(x)$ with the appropriate mathematical steps (Cesa-Bianchi et al., 2015).

5. Conclusions

This paper extends the problem of optimal price flor in real time bidding auctions for advertising in current scenarios where a DSP acts as a filter and only one bid is received by the AdNetwork and thus, the price paid corresponds to the reserve price. It also materialized the case in which the number of bidders is unknown using the normal discrete distribution. After an evaluation period carried out by an Andalusian digital marketing agency (Creafi), the revenue after using the algorithm described in section 4 increased by 127% compared to the revenue obtained if the default price agreed with the publisher had been used.

References

Ballesteros-Pérez, P., González-Cruz, M. C., Fuentes-Bargues, J. L., & Skitmore, M. (2015). Analysis of the distribution of the number of bidders in construction contract auctions. Construction management and economics, 33(9), 752-770.

Cesa-Bianchi, N., Gentile, C. and Mansour, Y. (2015). Regret Minimization for Reserve Prices in Second-Price Auctions, IEEE Transactions on Information Theory, 61(1), pp. 549–564.

Chaffey, D., & Ellis-Chadwick, F. (2019). Digital marketing. Pearson UK.

D'Annunzio, A., & Russo, A. (2020). Ad networks and consumer tracking. Management Science, 66(11), 5040-5058.

Myerson, R. B. (1981). Optimal auction design. *Mathematics of operations research*, *6*(1), 58-73.

Tahaei, M., & Vaniea, K. (2021). "Developers Are Responsible": What Ad Networks Tell Developers About Privacy. In Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems (pp. 1-11).

Yuan, S., Wang, J., Chen, B., Mason, P., & Seljan, S. (2014, August). An empirical study of reserve price optimisation in real-time bidding. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 1897-1906).