Bidding on Configurations in Internet Ad Auctions

S. Muthukrishnan

Google Inc., 76 9th Av, 4th Fl., New York, NY, 10011. muthu@google.com

Abstract. In Internet advertising, a configuration of ads is determined by the seller, and advertisers buy spaces in the configuration. In this paper, motivated by sponsored search ads, we propose an auction where advertisers directly bid and determine the eventual configuration.

1 Motivation

In Internet advertising, a *configuration* of ads is determined by the seller, and advertisers buy spaces within the configuration. For example, a publisher like nytimes.com may set aside space at the top right corner for display of image or video ads, and space on the lower right corner for text ads. Advertisers then pay to show their ads in these spaces. In sponsored search, search engines set aside space on the right side of the page (and in some cases, at the top or even the bottom) for text ads. Advertisers take part in an auction that determines the arrangement of ads shown in these spaces.

The configuration chosen by the seller is designed to make the ads effective. Publishers have insights from marketing and sociological studies on effective placements and viewer satisfaction. Similarly, advertisers have their own insights and can influence the publishers directly by collaborating on the design of configurations or influence indirectly by buying spaces at different rates.

We study how the choice of the configuration can be determined directly by letting parties bid on the configuration. A configuration depends on not only where the space is allocated, but also, how much space is allocated, what type of ads are allowed, how many ads fit into a space, how the ads are formatted in font, color and size, etc. Thus the bidding language may become complex. In addition, effectiveness of configurations also depends on viewer behavior and response which has to be modeled. Thus, determining optimal or efficient auctions become sophisticated optimization problems. Finally, the game theory of bidding on configurations becomes nuanced. Therefore, study of ad configuration problems and associated mechanism design has the potential to become a rich research area in the future.

In this context, we formulate a very simple problem motivated by sponsored search. In particular, we let advertisers directly impact the *number* of ads shown. We present an auction that generalizes the current auctions for sponsored search, and show that it can be implemented in near-linear time.

2 Our Problem

In sponsored search, we have a set of advertisers who bid on keywords. When a user poses a search query, an auction is run among all advertisers who bid on keywords relevant to the query. The advertisers are sorted say based on their bids and the top ℓ are shown in the decreasing order of their bids, charging each advertiser only the bid of the ad below. This is called the *Generalized Second Price* auction. Search engines use this auction in different forms, for example, by sorting based on other criteria, adding reserve prices, etc. See [4, 3, 6, 9] for more details.

In this application, we refer to the ℓ positions arranged from top to bottom as the *configuration.* This focuses our attention on the simple aspect of the configuration, namely, the number ℓ of ad positions in it. In the problem we study, we make this biddable.

Precisely, our problem is as follows. We have n advertisers; ith advertiser has bid b_i, the maximum they are willing to pay for an ad, private value v_i, as well as p_i, that represents the condition that the advertiser will not like to appear in any auction that results in more than p_i positions. Thus, this model lets advertisers directly bid on the number of positions in the resulting configuration of ads shown. For example, if an advertiser i sets $p_i = 1$, they wish to be shown only if they are the exclusive (or solitary) ad. If every advertiser sets $p_i \geq \ell$ where ℓ is the maximum number of positions in the configuration, then this scenario becomes the standard ad auction described above with ℓ positions.

The problem now is to design a suitable auction, that is, the output will comprise

- the number $k^* \leq n$ of ads that will be shown,
- the arrangement of k^* ads to show, and
- a pricing for each ad.

3 Our Solution

We propose a mechanism that is in many ways similar to current ad auctions. This is described in [3, 5, 6] and we use insights and concepts from there.

Our algorithm works in two steps as follows.

- For each k, where k denotes the number of ads shown, we determine the maximum value assignment, assuming of course that $b_i = v_i$. This will gives us the choice of k^*.
- Then, we charge each advertiser the minimum bid they should have made (fixing all others) in order to get the assignment they obtained.

There are details to both steps.

Finding optimal value configuration. For each k, define $V(k)$ to be maximum value assignment of ads to at most k positions. Then, we define $k^* =$

$argmax(V(k))$. Renumber the advertisers so they are numbered in the increasing order of their bids. Thus, b_i is the ith largest bid, and corresponds now to advertiser renumbered to i.

Lemma 1. *Define i^*_k to be the smallest i such that $|\{j|j \leq i, p_j \geq k\}| = k$, if it exists. Then,*

$$V(k) = \sum_{j \leq i^*_k, p_j \geq k} b_j.$$

Proof. The maximum value assignment is simply the maximum bipartite matching of eligible ads to positions. ■

Our algorithm is simple. We represent each advertiser i by a two dimensional point (i, p_i) with weight b_i. Then, we can determine i^*_k by binary searching on i with range query that counts the number of points in $[0, i] \times [k, \infty]$. And we can determine $V(k)$ by a single range sum query on $[0, i^*_k] \times [k, \infty]$. Both of these can be answered in $O(\log n)$ time using range searching data structures [7]. Thus the total running time of the algorithm is $O(n \log n)$ (to sort the bids) + $O(n \log^2 n)$ (for each $k = 1, \ldots, n$, we do $O(\log n)$ binary search queries each of which takes $O(\log n)$ time); hence, the total time complexity is $O(n \log^2 n)$.

Theorem 1. *There is an $O(n \log^2 n)$ time algorithm to compute k^* and produce an arrangement of ads in k^* slots with maximum value $V(k^*)$.*

A natural question is if $V(k)$ is a single mode function, increasing as k increases from 1 and decreasing beyond the maximum at some k^*. If that were the case, the search for k^* would be much more efficient. Unfortunately, that is not true. Consider a sequence of (b, p) pairs as follows:

$$(1000, 1), (400, 3), (400, 3), (400, 3), (300, 5), (300, 5), (300, 5), (290, 5), (100, 5), ...$$

which produces a see-saw teeth of $V(k)$'s: $V(1) = 1000$, $V(2) = 800$, $V(3) = 1200$, $V(4) = 1190$, etc. We can construct examples with linear number of teeth of maxima if needed.

Another natural question is if $V(k^*)$ can be computed by simply auctioning one position after another, say going top down, as in [8]. This is unfortunately incorrect since without fixing a k *a priori*, it is impossible to determine the advertisers who compete for any position.

Pricing. We now design an algorithm to calculate a price for each advertiser; it will have the *minimum pay property* (mpp), that is, each advertiser would pay no more than what he would have bid, if he knew all others' bids, to get the exact assignment he got. More formally, say the solution above returns a configuration of k^* ads and an advertiser i has position j in it. We want to charge advertiser i, the minimum bid c_i he would make, if all other bids were fixed, to get that position j in a configuration with $k^* - 1$ other ads. In other words, c_i should not only be sufficient to get position j in the current configuration, but also should ensure that the outcome of the previous step, ie., the optimal configuration, has k^* positions altogether.

Determining prices that satisfy mpp is trivial in the absence of p_i constraints: advertiser i in position j would simply pay the bid of the ad in position $j+1$ plus ε, giving the *generalized second price* (GSP) auction that is currently popular [3, 5, 6].[1] In our case, determining mpp price is not trivial.

We first show a number of natural properties of mpp pricing from GSP auctions that no longer hold for our case.

- There are cases when mpp price for advertiser i is not the bid of the advertiser below.
  Consider optimal allocation with $k = 2$ of $(100, 50)$ and that with $k = 3$ of $(100, 30, 22)$ because the advertiser with bid 50 does not want to appear with 3 or more positions. Clearly $k = 3$ has the optimal value. Then, if 30 bid $22 + \varepsilon$, he would still get his position in the $k = 3$ configuration, but the $k = 2$ configuration is now the optimal one.
- The mpp price for advertiser i may not be the maximum of the mpp bids in each of the auction outcomes in which i appears.
  The example above works, since we can fix it so that 30 appears only on $k = 3$ case.
- The mpp price for advertiser i may not be anyone's bid (or $\pm\varepsilon$ of it).
  In the example above, the mpp price for advertiser 30 is $28 + \varepsilon$.

In what follows, we present an efficient algorithm to compute the mpp prices.

Definition 1. *Let C_i^* define the optimal configuration of advertisements with i positions, and let C_{-i}^* define the optimal configuration for i positions, but include the $i+1$ qualifying bidder as well.*[2] *Define b to be the mpp price for i and let $b_{\downarrow i}^j$ be the maximum bid below b_i in C_j^* for any j. Define $V(C)$ to be the value in C, that is $V(C) = \sum_{i \in C} b_i$.*

We make a few observations.

Observation 1 *For each advertiser i, there exist α_i, β_i, $1 \le \alpha_i \le \beta_i \le n$ such that i appears in every one of configurations $C_{\alpha_i}^*, \ldots, C_{\beta_i}^*$, and none other.*

Say $C_{k^*}^*$ is the optimal configuration and i appears in it.

Observation 2 *We have $b > b_{\downarrow i}^{k^*}$.*

We divide $C_1^*, \ldots, C_n^*$ into three categories and use structural properties in each category.

Consider $\beta_i \ge j > k^*$, if any. We have,

Lemma 2. *$b_{\downarrow i}^j < b_{\downarrow i}^{k^*} < b$.*

[1] The ε is a little extra, like say one cent, that advertisers are charged to "beat" the bid below.

[2] We assume such a bidder exists. The remaining arguments can be easily modified otherwise.

Consider any j such that C^*_j does not contain i. Then,

Lemma 3. $b \geq \max_j\{V(C^*_j) - V(C^*_{k^*}) + b_i\}$.

The lower bound above on b can be computed from $\max_{j<\alpha_i} V(C^*_j)$ and $\max_{j>\beta_i} V(C^*_j)$.

Consider any $\alpha_i \leq j < k^*$. If decreasing i's bid from b_i to b' does not change the set of advertisers in C^*_j, then $V(C^*_j) - b_i + b' < V(C^*_{k^*}) - b_i + b'$. Otherwise,

Lemma 4. $b \geq \max_j\{V(C^*_{-j} - b_i\}$.

The lower bound above on b can be computed from $\max_{\alpha_i \leq j < k^*} V(C^*_{-j})$.

Thus, we can now compute b to be the max of the two quantities above, plus ε; we can also easily see that the resulting term will be no larger than b_i, and therefore a legal price. This gives,

Theorem 2. *The mpp price for any advertiser i can be computed in $O(\log n)$ time. Over all advertisers, price computations take $O(n \log n)$ time.*

For comparison, one might wish to consider Vickrey-Clarke-Groves (VCG) pricing [5, 1, 2]. Consider any i in $C^*_{k^*}$. It's VCG price can be seen to be

$$V(C^*_{k^*}) - b_i - \max\{\max_{\ell<\alpha_i} V(C^*_\ell), \max_{\alpha_i \leq \ell \leq \beta_i} V(C^*_{-\ell}), \max_{\beta_i<\ell} V(C^*_\ell)\},$$

which can again be computed in $O(\log n)$ time.

Remark. Say there is an upper bound ℓ on the number of slots that the bidders are allowed to specify, as determined by the auctioneer (the search company). Then our auction can be implemented in $O(n\ell)$ time easily by considering each potential k of total slots from 1 to ℓ. However the implementation we have described above is more efficient for large ℓ (note that in theory, ℓ may be as high as n).

4 Concluding Remarks

We formulated a very simple ad configuration auction problem where advertisers directly bid on the number of positions in the configuration. We proposed an auction and showed a near-linear time algorithm to compute the minimum pay property (mpp) price as well as the VCG price. Of immediate interest is the equilibrium properties of our auction. When $p_i \geq 1$ for all i, our auction becomes the standard second price (VCG) auction and hence it is truthful. When $p_i \geq n$ for all i, our auction becomes the standard generalized second price (GSP) auction which is not truthful, but is known to have an equilibrium identical to the VCG auction [3, 5, 6]. Does our auction have such good equilibriums and are there natural dynamics that will let advertisers converge to the good equilibrium? Adopting the arguments in [3, 6] might help address this question.

More broadly, we hope mechanism design and game theory of ad configuration problems get studied in more detail. In particular, what are suitable bidding languages for configurations, what are the relative powers of different bidding languages, what are suitable auctions that are easy to understand, implement and have desirable efficiency, revenue properties, what are achievable equilibrium properties and associated dynamics, etc.

5 Acknowledgements

We thank ad colleagues at Google who provide a stimulating environment.

References

1. E. Clarke. Multipart pricing of public goods. *Public Choice*, 11:17–33, 1971.
2. T. Groves. Incentives in teams. Econometrica, 41(4):617–631, 1973.
3. B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords. In Second workshop on sponsored search auctions, 2006.
4. H. Varian. Position auctions. *International Journal of Industrial Organization*,25(6): 1163–1178, 2007.
5. W. Vickrey. Counterspeculation, auctions and competitive-sealed tenders. *Finance*, 16(1):8–37, 1961.
6. G. Aggarwal, A. Goel, and R. Motwani. Truthful auctions for pricing search keywords. In *ACM Conference on Electronic Commerce (EC)*, 2006.
7. P. Agarwal, and J. Erickson. Geometric range searching and its relatives. Advances in Discrete and Computational Geometry 1999, 1–56. American Mathematical Society.
8. G. Aggarwal, J. Feldman and S. Muthukrishnan. Bidding to the top: VCG and equilibria of position-based auctions. Proc. Workshop on Approximation and Online Algorithms (WAOA), 2006.
9. S. Muthukrishnan. Internet Ad Auctions: Insights and Directions. Proc. *ICALP* (1) 2008: 14-23.