

General Auction Mechanism for Search Advertising

Gagan Aggarwal
 Google, Inc.
 1600 Amphitheatre Pkwy
 Mountain View, CA, 94043.
 gagana@google.com

Dávid Pál^{*}
 David R. Cheriton School of
 Computer Science
 University of Waterloo
 Waterloo, ON, Canada
 dpal@cs.uwaterloo.ca.

S. Muthukrishnan
 Google, Inc.
 76 Ninth Avenue, 4th Floor
 New York, NY, 10011
 muthu@google.com

Martin Pál
 Google, Inc.
 76 Ninth Avenue, 4th Floor
 New York, NY, 10011
 mpal@google.com

ABSTRACT

In sponsored search, a number of advertising slots is available on a search results page, and have to be allocated among a set of advertisers competing to display an ad on the page. This gives rise to a bipartite matching market that is typically cleared by the way of an automated auction. Several auction mechanisms have been proposed, with variants of the Generalized Second Price (GSP) being widely used in practice.

There is a rich body of work on bipartite matching markets that builds upon the stable marriage model of Gale and Shapley and the assignment model of Shapley and Shubik. This line of research offers deep insights into the structure of stable outcomes in such markets and their incentive properties.

In this paper, we model advertising auctions in terms of an assignment model with linear utilities, extended with bidder and item specific maximum and minimum prices. Auction mechanisms like the commonly used GSP or the well-known Vickrey-Clarke-Groves (VCG) can be interpreted as simply computing a *bidder-optimal stable matching* in this model, for a suitably defined set of bidder preferences, but our model includes much richer bidders and preferences. We prove that in our model the existence of a stable matching is guaranteed, and under a non-degeneracy assumption a bidder-optimal stable matching exists as well. We give an algorithm to find such matching in polynomial time, and use it to design truthful mechanism that generalizes GSP, is truthful for profit-maximizing bidders, correctly implements features like bidder-specific minimum prices and position-specific bids, and works for rich mixtures of bidders and preferences. Our main technical contributions are the existence of bidder-optimal matchings and strategyproofness of the resulting mechanism, and are proved by induction on the progress of the matching algorithm.

Categories and Subject Descriptors

F.2.2 [Theory of Computation]: Analysis of Algorithms and Problem Complexity—*Nonnumerical Algorithms and Problems*; G.2.2 [Mathematics of Computing]: Discrete Mathematics—*Graph Theory*

^{*}Work done during summer 2007 internship at Google New York.

Copyright is held by the International World Wide Web Conference Committee (IW3C2). Distribution of these papers is limited to classroom use, and personal use by others.

WWW 2009, April 20–24, 2009, Madrid, Spain.
 ACM 978-1-60558-487-4/09/04.

General Terms

Algorithms, Economics, Theory

Keywords

Game Theory, Sponsored Search Auctions, Stable Matchings

1. INTRODUCTION

Internet advertising is a prime example of a matching market: a number n of advertisers (bidders) are competing for a set of k advertising slots (items) offered for sale by a content publisher or a search engine. Internet advertising and sponsored search auctions have attracted wide attention in the academic literature, and there are several papers discussing various aspects of pricing ad slots and allocating them to interested advertisers.

Classical matching market models include the *stable marriage* model of Gale and Shapley [15] and the *assignment model* of Shapley and Shubik [25]. For these models and many of their extensions, we have a good understanding of the structure of their stable outcomes (“equilibria”) and their incentive properties. We take advantage of existing body of work on stable matchings and apply it to sponsored search.

We observe that existing auction mechanisms for sponsored search, most notably, variants of Generalized Second Price (GSP) and Vickrey-Clarke-Groves (VCG), merely compute a stable matching in a suitably defined model. We make this model explicit, and propose a new auction mechanism that includes the existing mechanisms as special cases. The model is flexible enough to allow for bidder and position specific minimum and maximum prices, as well as different values for different slots. Much of the existing literature does not address these features (like minimum prices) that are important in practice. Beyond that, our model of bidder preferences allows for a wider range of bidder behaviors than just profit maximization (i.e. we do not assume that the bidder’s payoff is quasi-linear in payment). As an example, a bidder who desires to win the highest slot possible subject to the constraint that his price be at most m (for some parameter m) is clearly not maximizing profit, but can be expressed in our model. It is important for us to include such bidders in order to correctly model the variants of GSP auction which have not been previously analyzed; it also happens that the basic GSP mechanism is truthful for such class of bidders.

Our proposed auction mechanism solicits bidder preferences from each bidder and then simply computes a *bidder-optimal stable match-*

ing given those preferences. The mechanism is truthful (and even group strategyproof if money transfers among players are not permitted).

On the algorithmic side, we show how to compute the allocation and prices corresponding to a bidder-optimal stable matching in time $O(nk^3)$, where n is the number of bidders and k is the number of slots to sell. Our algorithm is an extension of the Hungarian algorithm for finding maximum-weight matchings in bipartite graphs. The idea of the algorithm is simple, although some attention to detail is required to ensure correctness, and the algorithm has to be made fast enough for search advertising. Our proofs of existence of bidder-optimal matchings as well as proof of a key lemma establishing truthfulness of our auction mechanism follow by induction on the execution of the matching algorithm.

2. RELATED WORK

Matching Markets. The marriage model of Gale and Shapley [15] and the assignment model of Shapley and Shubik [25] are two standard models in the theory of matching markets.

In the marriage model, a set I of men and a set J of women is given, where each man and woman is endowed with a ranked list of members of the opposite sex. Men and women are to be matched in a one to one fashion. A matching is considered stable if there is no man and a woman who would simultaneously prefer each other to their respective assigned partners. A stable matching is guaranteed to exist, and the *deferred acceptance* algorithm can be used to find it. The stable matching found by this algorithm is *man-optimal*, in that every man prefers it to any other stable matching. Moreover when using the deferred acceptance algorithm, no man has an incentive to misreport his true preference order [23].

The assignment model [25], (see also [22, 10]) differs in that each player derives a certain value from being matched to each person of the opposite sex, and side payments between partners are allowed. The goal of each player is to maximize his or her payoff which is the sum of partner's value and monetary payment (positive or negative) from the partner. The set of stable outcomes is non-empty by a linear programming argument. In fact, each stable outcome corresponds to a maximum-weight matching, and player payoffs correspond to dual variables of the maximum matching LP. A man-optimal outcome is guaranteed to exist, and its allocation and prices are identical to the VCG mechanism for maximum weight matchings [20, 7].

Many variations and extensions of each model have been studied; see the monograph [24] for a nice overview. Payoff functions that are not necessarily linear in the payment were considered by [9, 10, 5, 6]. Even in such generality, there exists a man-optimal stable matching [9], and in a man-optimal auction mechanism, it is weakly dominant for each bidder to reveal his true utility (payoff) function. These results require the utility functions to be continuous, strictly monotone and defined on the whole range $(-\infty, +\infty)$, and therefore are not directly applicable in our setting.

Kelso and Crawford [18] and others have proposed a many to one variant in which firms may hire multiple workers. Recently, Fujishige and Tamura [14] proposed a very general many to many model with linear utility functions in which each worker can engage multiple firms, and allow lower and upper bounds to be placed on the range of payments allowed between any pair of players. Under an assumption on the payoff functions called M^b concavity, they give a proof of existence of a stable outcome and give an algorithm to find it.

The model considered in this paper is an assignment model with linear payoffs. It is a special case of the model of Fujishige and Tamura [14], in that we assume one to one matching of bidders

to items. In addition to non-emptiness, we show that the set of stable matchings in our model has a bidder-optimal element, and prove that an auction mechanism based on bidder-optimal matchings is truthful and present an efficient algorithm. Fujishige and Tamura [14] show existence of a stable matching in their very general model by running an algorithm somewhat similar to ours, but do not give any results on bidder-optimality or truthfulness.

Sponsored Search Auctions. Flavors of the Generalized Second Price (GSP) auction are the dominant vehicles for selling ads on the internet. In its basic form, GSP solicits a numeric bid from each advertiser, orders them in decreasing order of bids, and assigns slots to the first up to k bidders in this order. Each bidder is required to pay a price equal to the bid of the next bidder in the ordering (or a minimum price if this is the last bidder). In a per-click GSP, each bidder pays only in the event that his ad is clicked on. In a per-impression GSP, the advertiser pays each time her ad is displayed.

It has been observed that although it is not truthful for "profit maximizing" bidders, the per-click GSP mechanism does have a Nash equilibrium (under some assumptions on the structure of click probabilities across different positions) that is efficient and its resulting prices are equal to VCG prices; see [12, 2]. A variant of GSP in which the bidder can specify the lowest (maximum) acceptable position has been proposed in [3], which also has a Nash equilibrium equivalent to a suitably defined VCG auction. Even-Dar et al. [13] show that a Nash equilibrium of GSP exists even if minimum prices are bidder-specific, but that equilibrium is no longer related to a naturally defined VCG outcome.

One reason GSP works well in practice is that in most situations, bidders universally agree that higher slots are preferable to lower slots. With increasingly complex web page layouts and increasingly sophisticated advertisers this assumption may become less valid over time. Features like Google's Position Preference aim to rectify this by allowing advertisers to only bid for a specified subset (range) of positions.

The general class of VCG mechanisms follows from works of Vickrey [27], Clarke [8] and Groves [17]. For an overview of the VCG mechanism applied to sponsored search, see e.g. [1, 2]. VCG is a very natural mechanism and is truthful for profit maximizing bidders, but it is sufficiently different from GSP and bidders may find it difficult to interpret the prices they are charged.

In section 3 we describe the assignment model with minimum and maximum prices and state the main results. Section 5 gives a description of an algorithm to find a bidder-optimal stable matching. Sections 6.1 and 7 give high level overview of the proofs, with the details delegated to Appendix A and B. Appendix 4.1 discusses how current auction mechanisms for sponsored search fit in our model.

3. ASSIGNMENT MODEL WITH MAXIMUM AND MINIMUM PRICES

Our model that we call the *max-value* model, consists of the set $I = \{1, 2, \dots, n\}$ of bidders and the set $J = \{1, 2, \dots, k\}$ of items. We use letter i to denote a bidder and letter j to denote an item. Each bidder i has a *value* $v_{i,j}$ for each slot j how much is that slot worth to her, and a *maximum price* $m_{i,j}$ she is able and willing to pay for the slot.¹ In addition to bidder preferences, the

¹To motivate why $v_{i,j}$ and $m_{i,j}$ might be different, consider buying a house whose value to you is higher than the amount of money your bank is willing to lend you. Allowing the bidder to specify both a value and a maximum is also needed to model the GSP auction.

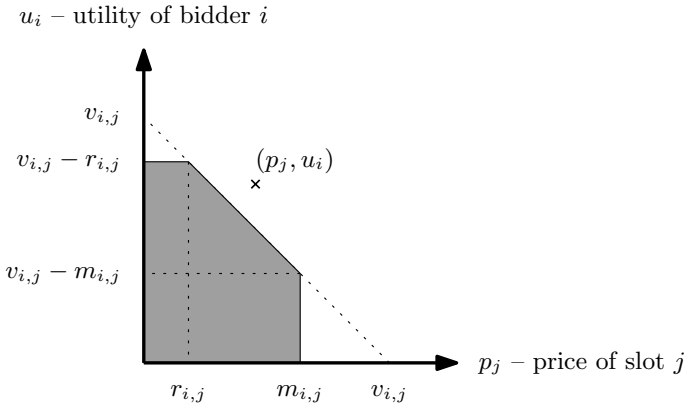


Figure 1: Matching is stable whenever for each bidder $i \in I$ and each slot $j \in J$ the point with coordinates (p_j, u_i) lies outside the gray region.

seller may specify for each item j a *reserve* or *minimum price* $r_{i,j}$.

For simplicity we assume that the minimum prices are known to the bidders in advance. For each i and each j we assume that $r_{i,j} \geq 0$, $v_{i,j} \geq 0$, $m_{i,j} \leq v_{i,j}$. If bidder i is interested in the slot j he specifies $m_{i,j} \geq r_{i,j}$. Otherwise, if bidder i has no interest in slot j he specifies negative $m_{i,j}$. We denote by v, m, r the $n \times k$ matrices with entries $v_{i,j}, m_{i,j}, r_{i,j}$ respectively. We refer to the triple (v, m, r) as an *auction instance* or simply *auction*.

Stable Matching. We formalize the notion of a matching in the following definitions.

DEFINITION 1 (MATCHING). A matching is a triple (u, p, μ) , where $u = (u_1, u_2, \dots, u_n)$ is a non-negative utility vector, $p = (p_1, p_2, \dots, p_k)$ is a non-negative price vector, and $\mu \subseteq I \times J$ is a set of bidder-slot pairs such that no slot and no bidder occurs in more than one pair.

If a pair $(i, j) \in \mu$, we say that bidder i is *matched* to slot j . We use $\mu(i)$ to denote the slot matched to a bidder i , and $\mu(j)$ to denote the bidder matched to a slot j . Bidders i and slots j that do not belong to any pair in μ are said to be *unmatched*.

DEFINITION 2 (FEASIBLE MATCHING). A matching (u, p, μ) is said to be *feasible* for an auction (v, m, r) , whenever for every $(i, j) \in \mu$,

$$p_j \in [r_{i,j}, m_{i,j}], \quad (1)$$

$$u_i + p_j = v_{i,j}, \quad (2)$$

and for each unmatched bidder i is $u_i = 0$ and for each unmatched slot j is $p_j = 0$.

DEFINITION 3 (STABLE MATCHING). A matching (u, p, μ) is *stable* for an auction (v, m, r) whenever for each $(i, j) \in I \times J$ at least one of the following inequalities holds:

$$u_i + p_j \geq v_{i,j}, \quad (3)$$

$$p_j \geq m_{i,j}, \quad (4)$$

$$u_i + r_{i,j} \geq v_{i,j}. \quad (5)$$

A pair $(i, j) \in I \times J$ which does not satisfy any of the three inequalities is called *blocking*.

Geometric interpretation of inequalities (3), (4), (5) is explained in Figure 1. Note that if a bidder i is not interested in a slot j , then (4) is trivially satisfied.

A feasible matching does not have to be stable, and a stable matching does not have to be feasible. However, we will be interested in matchings that are both stable and feasible, and in addition bidder-optimal.

DEFINITION 4 (BIDDER OPTIMALITY). A stable, feasible matching (u^*, p^*, μ^*) is *bidder-optimal* if for every stable feasible matching (u, p, μ) and every bidder $i \in I$ we have $u_i^* \geq u_i$.

Bidder Preferences. To study strategic behavior of bidders in an auction, we need to model bidder's preferences. We assume that each bidder is indifferent among various outcomes as long as her assigned slot (if any) and payment is the same. Let us define the utility (payoff) of a bidder i who is offered a slot j at price p as follows. If $p \leq m_{i,j}$, we set $u = v_{i,j} - p$. If $p > m_{i,j}$, we set $u = -1$. This payoff, interpreted as a function of the price, is not continuous at $p = m_{i,j}$. If the bidder is unmatched (at zero price), her payoff is 0. Given a choice between slot j_1 at price $q_1 \leq m_{i,j_1}$ and slot j_2 at price $q_2 \leq m_{i,j_2}$, the bidder prefers the offer with higher payoff, and is indifferent among offers that have the same payoff. In particular, the bidder prefers to be not matched to being matched to a slot j at price that exceeds her maximum price $m_{i,j}$. The bidder is indifferent between being matched with payoff 0 and not being matched.

We call a bidder whose preferences can be described by a vector of maximum prices and values a *max-value bidder*. We point out two classes of bidders that are of interest. For a more detailed discussion of bidder types and auction modeling issues, refer to Section 4.

A *profit maximizing bidder* i only cares about the values $v_{i,j}$ he can gain from each position, and seeks to maximize value of the item received minus payment. For such bidder we can render the maximum price $m_{i,j}$ ineffective by setting it to $v_{i,j}$.

A *maximum price bidder* is parametrized by a maximum price m_i he is willing to pay. He seeks to get the lowest-index position whose price is less than or equal to m_i .

3.1 Our Results

Every auction instance in our model has a stable matching by the result of [14]. We show that it also has a bidder-optimal matching, and to give an algorithm to find it.

THEOREM 5. *If the auction (v, m, r) is in a "general position", it has a unique bidder-optimal stable matching. This matching can be found in time $O(nk^3)$.*

We defer the precise definition of general position to Definition 14. In essence, any auction (v, m, r) can be brought into general position by arbitrarily small (symbolic) perturbations. In practice this assumption is easily removed by using a consistent tie-breaking rule.

Consider the following mechanism for auctioning off k items to n bidders. The auctioneer (seller) sets an arbitrary minimum price r_{ij} for each bidder-item pair. It then solicits vectors of maximum prices $m_i = (m_{i1}, m_{i2}, \dots, m_{ik})$ and values $v_i = (v_{i1}, v_{i2}, \dots, v_{ik})$ from each bidder i . Finally, the auctioneer computes a bidder-optimal stable matching (u^*, p^*, μ^*) for the auction instance (m, v, r) . It assigns each bidder i the item (if any) $j = \mu^*(i)$ and charges him price p_j^* (or 0 if $\mu^*(i) = \emptyset$). Let us call this mechanism the *Stable Matching Mechanism*. Our second technical contribution is to show that the Stable matching Mechanism is truthful for max-value bidders.

THEOREM 6 (TRUTHFULNESS). *In the Stable Matching Mechanism, it is a (weakly) dominant strategy for each bidder i to submit*

her true vectors v_i and m_i , as long as i 's preferences can be expressed in the max-value model.

4. MODELING ADVERTISING AUCTIONS

In this section, we present examples of auction mechanisms commonly used in sponsored search. We show how to model these mechanisms in our max-value model. In the next section we give examples of novel combined mechanisms that can be implemented in our model.

Translating between impressions and clicks. Typically, an auction is run to determine the placement of ads every time a results page is rendered; however, the advertiser only pays when a user actually clicks on the ad. It is straightforward to translate between the pay-per-impression and the pay-per-click model, provided that we know the probability ctr that a user will click on the ad: paying p_c per click is the same in expectation as paying $p_i = \text{ctr} \cdot p_c$ per impression. In the following, let $\text{ctr}_{i,j}$ be the probability that a user clicks on ad i if it is displayed in position j (and that this probability does not vary depending on the set of competing ads shown on the page). The *click separability assumption* says that $\text{ctr}_{i,j} = q_i \cdot \alpha_j$ is the product of a quality score q_i of the advertiser and a *position normalizer* α_j specific to the position j . Typically the position normalizers are assumed to be decreasing, i.e. $\alpha_1 \geq \alpha_2 \geq \dots \geq \alpha_k$.

GSP pay-per-impression. In a Generalized Second Price auction, each advertiser i submits a single number b_i as her bid, which is the maximum amount she is willing to pay for displaying her ad. The auctioneer orders bidders in decreasing order of their bids, and assigns the first k advertisers to the k available slots in this order. The i -th allocated advertiser pays amount equal to the $(i+1)$ -st bid for each impression.

GSP pay-per-click. An alternative is to charge the advertiser only in the event of a click on her ad. The bid b_i is interpreted as a maximum the advertiser is willing to pay for a click. Again, the advertisers are ordered by their per-click bid, and each allocated advertiser pays the next highest bid in the event of a click. In a quality-weighted variant, the ads are ordered by the product of their quality score q_i and bid b_i ; the i -th advertiser pays $b_{i+1} \frac{q_{i+1}}{q_i}$ in the event of a click. Note that the expected cost per impression $b_{i+1} \frac{q_{i+1}}{q_i} \text{ctr}_{i,i}$ depends not only on the next highest bid but also on the position, as long as the probability $\text{ctr}_{i,j}$ of clicking on the ad i in position j depends on the position. Thus, there is no direct way to translate a per-click bid to a per-impression bid, without looking at the competitor's bids.

The VCG mechanism for profit-maximizing bidders. In a variant of the VCG mechanism considered e.g. in [2], each bidder i states her value V_i for a click. The auctioneer derives the expected value of each slot $v_{i,j} = V_i \cdot \text{ctr}_{i,j}$ for that bidder by using an estimate $\text{ctr}_{i,j}$ of the probability that the ad i would be clicked on if placed in position j . The auctioneer computes a maximum-weight matching in the bipartite graph on bidders and positions with $v_{i,j}$ as edge weights. The maximum weight matching μ^* gives the final allocation. For pricing, the VCG formula sets the price per impression of slot $j = \mu^*(i)$ to be $p_j = \sum_{k \in I \setminus \{i\}} v_{k, \mu^*(k)} - v_{k, \mu^*(k)}$ where μ^* is a maximum-weight matching with the set of bidders $I \setminus \{i\}$. Note that the per-impression price p_j can be translated to a per-click price by charging bidder i price $p_j / \text{ctr}_{i,j}$ for each click. (Similar translation can be done for a generally defined user action other than a click, as long as the probability of the action can be estimated.)

For each of the above mechanisms, we define a corresponding type of bidder in the max-value model.

Max-per-impression bidder has a target cost per impression b_i .

She prefers paying b_i or less per impression to any outcome where she pays more than b_i . Given that her cost per impression is at most b_i , she prefers higher (with lower index) position to lower position. Given a fixed position, she prefers paying lower price to higher price. A max-per-impression bidder i can be translated into the max-value model by setting her $m_{i,j} = b_i$ for all positions $j \in J$, and setting her value $v_{i,j} = M(k+1-i)$ where M is a sufficiently large number ($M > b_i$ is enough).

Max-per-click bidder differs from a max-per-impression bidder in that she is not willing to pay more than b_i per click. We translate her per-click bid into our framework using predicted click probabilities: set $m_{i,j} = b_i \cdot \text{ctr}_{i,j}$ for $i \in I$ and $v_{i,j} = M(k+1-i)$ where $M > b_i \max_j \text{ctr}_{i,j}$.

Profit-maximizing bidder seeks the position and payment that maximizes her expected profit (value from clicks minus payment). If we assume that her value per click is V_i , such bidder is modeled by setting $v_{i,j} = m_{i,j} = V_i \cdot \text{ctr}_{i,j}$.

We formalize the correspondence between the mechanisms and corresponding bidder types in the following theorem.

THEOREM 7. *The outcome (allocation and payments) of a (1) per-impression GSP, (2) per-click GSP, (3) VCG auction, respectively is a bidder-optimal stable matching for a set of (1) max-per-impression bidders, (2) max-per-click bidders, (3) profit-maximizing bidders, respectively.*

PROOF. Part (3) of the theorem has first been shown by [20]. Chapter 7 of [24] as well as [7] discuss the relationship of the VCG mechanism for assignments and stable matchings.

We give a proof for part (1), per-impression GSP. The proof of part (2) for per-click GSP is very similar and is omitted. For simplicity, we assume that $n > k$ and all reserve prices are zero. Let $b_1 > b_2 > \dots > b_n$ be the per-impression bids of the bidders. Without loss of generality, the bidders are ordered by decreasing order of their bids. (By the general position assumption, assume bids are distinct.)

Recall that we encode a max-per-impression bidder by setting $v_{i,j} = M(k-j+1)$ and $m_{i,j} = b_i$. The matching produced by the GSP auction is as follows: the matched pairs are $\mu = \{(1, 1), (2, 2), \dots, (k, k)\}$, bidder's utilities $u_i = M(k-i+1) - b_{i+1}$ for $1 \leq i \leq k$, $u_i = 0$ for $i > k$, and prices $p_i = b_{i+1}$ for $i = 1, 2, \dots, k$. It is easy to verify this matching is feasible and stable according to Definitions 2 and 3.

First we show that any feasible matching in which the assignment is different from μ is not stable. Indeed, such a matching (u', p', μ') must have a bidder $i \leq k$ such that i was not allocated a slot among the first i slots, and a slot $j \leq i$ that is either unmatched or matched to some bidder $i' > i$.

From feasibility we have that $p_j = 0$ if slot j is unmatched and $p_j \leq b_{i'}$ in case it is matched. In either case, $p_j < b_i$. Also, since bidder i is matched to some slot $j' > i$ (or unmatched), we know that $u'_i \leq v_{i,j'} = M(k-j'+1)$. We now claim that (i, j) is a blocking pair. Since $v_{i,j} - u'_i \geq M[(k-j+1) - (k-j'+1)] \geq M$, inequalities (3) and (5) are violated, and since $p'_j < b_i$, inequality (4) is violated as well.

Consider any matching with the assignment $\mu = \{(1, 1), \dots, (k, k)\}$. It is easy to verify that in order to be stable, it must be that $p_i \geq b_{i+1}$, otherwise the pair $(i+1, i)$ would be a blocking pair. Hence the matching with prices $p_i = b_{i+1}$ has the lowest possible prices and hence is bidder-optimal. \square

Minimum prices. Some search engines impose a minimum price r_i for each ad (for example, based on perceived quality of the ad). In GSP, only bidders whose bid is above the reserve price can participate. The allocation is in decreasing order of bids, and each

bidder pays the maximum of her reserve price and the next bid. Minimum GSP prices are easily translated to the max-value model by setting $r_{i,j} = r_i$ (if paying per impression) or $r_{i,j} = r_i \cdot \text{ctr}_{i,j}$ (if paying per click). Our model allows for separate reserve prices for different slots (e.g. higher reserve price for certain premium slots) that are not easily implemented in the GSP world.

4.1 New Auction mechanisms

Let us give a few examples of new auction mechanisms that are special cases of the max-value model.

GSP with arbitrary position preferences. Consider an advertiser i who wishes for her ad to appear only in certain slots. For example, [3] propose a GSP variant in which each bidder has the option to specify a prefix of positions $\{1, 2, \dots, \beta_i\}$ for some β_i she is interested in and exclude the remaining slots. Also, tools like Google's Position Preference allow the advertiser to specify arbitrary position intervals $[\alpha_i, \beta_i]$. We are however not aware of any published work that discusses more sophisticated position preferences. One would imagine that in the world of content advertising where there may be multiple areas designed for ads on a single page, having a richer language in which to express the preferences over slots would be beneficial to the advertiser. Such preferences are readily expressible in the max-value model.

Combining click and impression bidders in GSP. Since both pay per click and pay per impression models are widely used in practice, it is useful to have a way of combining these two bidding modes. This can be easily done by computing a stable matching for a mixed pool of bidders. The following simpler approach is not appropriate, as it does not have the proper incentive structure.

Suppose we allow each bidder i to specify both a maximum price b_i , as well as a payment type $\tau_i \in \{\mathcal{I}, \mathcal{C}\}$. A naive combined auction orders bidders by decreasing b_i . Each advertiser with $\tau_i = \mathcal{I}$ is charged the next highest bid b_{i+1} for showing the ad. Each advertiser with $\tau_i = \mathcal{C}$ is charged b_{i+1} in the event that the user clicks on the ad. Note, this scheme gives advertisers a strong incentive to report $\tau_i = \mathcal{C}$ regardless of their true type (as long as the probability of user clicking is less than 1).

To offset this incentive, the auctioneer may introduce multipliers $0 < q_C < 1$ and $q_I = 1$ and set the effective bid of each bidder to be $b_i^{\text{eff}} = b_i q_{\tau_i}$. In the modified GSP auction, bidders are sorted by their effective bid. Each bidder i who reports type $\tau_i = \mathcal{I}$ is charged b_{i+1}^{eff} for each impression, while each bidder reporting $\tau_i = \mathcal{C}$ is charged b_{i+1}^{eff}/q_C in the event of a click.

For any value of $0 < q_C < 1$, there is a simple instance in which some bidder can gain by misreporting her type. Let ctr_1 and ctr_2 be the probability that an user will click on an ad in position 1 and 2 respectively. Assume this probability is the same for all ads, and that $\text{ctr}_1 > \text{ctr}_2$. Suppose that the first slot is won by a bidder of type \mathcal{I} , the second slot is won by a bidder of type \mathcal{C} , and that there is at least one more bidder with positive bid. If $q_C > \text{ctr}_2$, the bidder in the second position can lower her overall cost while keeping the same position by reporting type \mathcal{C} and keeping the same effective bid. On the other hand, if $q_C < \text{ctr}_1$, bidder in the first position can lower her cost by reporting type \mathcal{I} , and adjusting her bid so that her effective bid stays the same.

Diverse bidders. There are many types of bidders with different goals. Some like to think in terms of a maximum price per click or impression. Some prefer to target only certain positions (e.g. top of the page) for consistency or branding reasons. Others try to maximize their profit and are able to estimate the value of a specific user action. Each bidder may specify her goal in a language familiar to her. We are not aware of any prior research on auction mechanisms for such diverse set of bidders.

5. ALGORITHM TO COMPUTE A BIDDER-OPTIMAL MATCHING

We now describe algorithm STABLEMATCH that computes a feasible and stable matching for a given auction instance (v, m, r) . Later in Section 6 we show that the matching is also bidder-optimal, as long as the auction instance is in a general position (Definition 14).

The STABLEMATCH algorithm is an extension of the well known Hungarian Method [28, 19] for computing a maximum-weight matching in a bipartite graph. The Hungarian Method is a primal-dual algorithm that starts with an empty matching and repeatedly increases the size of the matching using a maximum-weight augmenting path. STABLEMATCH works the same way, except that it is designed to handle events corresponding to reaching minimum and maximum prices.

STABLEMATCH starts with an empty matching $(u^{(0)}, p^{(0)}, \mu^{(0)})$ which is defined as follows. Utility of each bidder i is $u_i^{(0)} = B$, where B is a large enough number, such that $B > \max\{v_{i,j} \mid (i, j) \in I \times J\}$. Price of each slot j is $p_j^{(0)} = 0$. There are no matched pairs, i.e. $\mu^{(0)} = \emptyset$.

In each iteration, STABLEMATCH finds an augmenting path, and updates the current matching $(u^{(t)}, p^{(t)}, \mu^{(t)})$ to the next matching $(u^{(t+1)}, p^{(t+1)}, \mu^{(t+1)})$. The algorithm stops when no more updates can be made, and outputs the current matching $(u^{(T)}, p^{(T)}, \mu^{(T)})$ at the end of the last iteration. We now describe an iteration in more detail. To do so, we introduce the concept of an update graph.

DEFINITION 8 (UPDATE GRAPH). *Given an auction (v, m, r) , the update graph for a matching (u, p, μ) is a directed weighted bipartite multigraph with partite sets I and $J \cup \{j_0\}$, where j_0 is the dummy slot. The update graph consists of five types of edges. For each bidder i and each slot $j \in J$ there is*

- a forward edge from i to j with weight $u_i + p_j - v_{i,j}$, if $p_j \in [r_{i,j}, m_{i,j}]$;
- a backward edge from j to i with weight $v_{i,j} - u_i - p_j$, if $(i, j) \in \mu$,
- a reserve-price edge from i to j with weight $u_i + r_{i,j} - v_{i,j}$, if $u_i + r_{i,j} > v_{i,j}$ and $m_{i,j} > r_{i,j}$,
- a maximum-price edge from i to j with weight $u_i + m_{i,j} - v_{i,j}$, if $u_i + m_{i,j} > v_{i,j}$ and $m_{i,j} > r_{i,j}$,
- a terminal edge from i to j_0 with weight u_i if $u_i > 0$.

An alternating path in the update graph starts with an unmatched bidder vertex i_0 with $u_{i_0} > 0$, follows a sequence of forward and backward edges, and ends with a reserve-price, maximum-price or terminal edge. We place the restriction that all vertices of the alternating path must be distinct, with the possible exception that the last vertex is allowed to appear once again along the path. The weight $w(P)$ of an alternating path P is the sum of weights of its edges.

Let $(u^{(t)}, p^{(t)}, \mu^{(t)})$ be a matching and $G^{(t)}$ be the corresponding update graph. A single iteration of the STABLEMATCH algorithm consists of the following steps.

1. If there is no alternating path, stop and output the current matching. Otherwise, let P be an alternating path in $G^{(t)}$ of minimum weight. Let $w^{(t)}(P)$ denote its weight, and let

$$P = (i_0, j_1, i_1, j_2, i_2, \dots, j_\ell, i_\ell, j_{\ell+1}) \quad \text{for some } \ell \geq 0.$$

2. Let $d^{(t)}(i_0, y)$ be the length of the shortest path in $G^{(t)}$ from i_0 to any vertex y , using only forward and backward edges. If a vertex y is not reachable from i_0 , $d^{(t)}(i_0, y) = \infty$.

3. Compute utility updates for each bidder $i \in I$. The vector $u^{(t+1)}$ gives the final utilities for the iteration.

$$u_i^{(t+1)} = u_i^{(t)} - \max\left(w^{(t)}(P) - d^{(t)}(i_0, i), 0\right) \quad (6)$$

4. Compute price updates for each slot $j \in J$.

$$p_j^{(t+1)} = p_j^{(t)} + \max\left(w^{(t)}(P) - d^{(t)}(i_0, j), 0\right) \quad (7)$$

The final prices $p_j^{(t+1)}$ are equal to $p_j^{(t)}$ with one exception. In case the last edge of P is a reserve-price edge, we set the price of slot $j_{\ell+1}$, the last vertex of P to be $p^{(t+1)} = \max(p^{(t)}, r_{i_\ell, j_{\ell+1}})$.

5. Update the assignment $\mu^{(t)}$ along the alternating path P to obtain the new assignment $\mu^{(t+1)}$.

We have not specified how should the set of assignment edges be updated. Before we do that, let us state two invariants maintained by STABLEMATCH.

(A1) The matching $(u^{(t)}, p^{(t)}, \mu^{(t)})$ is stable for the auction (v, m, r) .

(A2) For every matched pair $(i, j) \in \mu^{(t)}$, $u_i^{(t)}$ and $p_j^{(t)}$ satisfy (1) and (2).

An important consequence of invariant (A1) is that forward edges have non-negative weight. Indeed, it can be easily checked that a forward edge with a negative weight would be blocking pair. Invariant (A2) guarantees that backward edges have zero weight. Similarly, invariant (A2) implies that the weight of every backward edge must be zero. Finally, each reserve-price, maximum-price and terminal edges has non-negative weight by definition.

LEMMA 9. *All edge weights in each update graph $G^{(t)}$ are non-negative.*

With non-negative edge weights, single-source shortest paths can be computed using Dijkstra's algorithm in time proportional to the square of the number of vertices reachable from the source. Since no unmatched vertex is reachable from any other vertex, there are at most $2k$ reachable vertices at any time, thus the shortest alternating path P and distances $d^{(t)}(i_0, y)$ can be computed in time $O(k^2)$.

Finally, let us deal with updating the assignment μ . Since the alternating path alternates between using forward (i.e. non-matching) and backward (i.e. matching) edges, a natural move is to remove all the matching edges of P and replace them by non-matching edges of P . Care must be taken however to take into account the special nature of the last edge of P as well as the fact that the last vertex of P may be visited twice. We consider three cases:

Case 1: P ends with a terminal edge, i.e. $j_{\ell+1}$ is the dummy slot. Flip matching and non-matching edges along the whole length of P . Bidder i_ℓ ends up being unmatched, and for $x = 0, 1, \dots, \ell - 1$, bidder i_x will be matched to slot j_{x+1} .

Case 2: P ends with a maximum-price edge. Consider two sub-cases:

- (a) $j_{\ell+1} = j_\ell$. This means that the price bidder i_ℓ was matched to reached his maximum price. Flip matching and non-matching edges along P . This leaves bidder i_ℓ unmatched, and for $x = 0, 1, \dots, \ell - 1$ bidder i_x is matched with slot j_{x+1} .
- (b) Otherwise, the maximum price was reached on a non-matching edge. Keep the matching unchanged. That is, $\mu^{(t+1)} = \mu^{(t)}$.

Case 3: P ends with a reserve-price edge. This is the most complex case. Consider three subcases:

- (a) Item $j_{\ell+1}$ is unmatched in $\mu^{(t)}$. This case increases the size of the matching. For $x = 0, 1, \dots, \ell$, match bidder i_x with slot j_{x+1} .
- (b) Item $j_{\ell+1}$ is matched in $\mu^{(t)}$ and the reserve price $r_{i_\ell, j_{\ell+1}}$ offered by bidder i_ℓ does not exceed the current price $p_{j_{\ell+1}}^{(t+1)}$ of the slots. Keep the matching unchanged, that is, $\mu^{(t+1)} = \mu^{(t)}$.
- (c) Item $j_{\ell+1}$ is matched in $\mu^{(t)}$ to some bidder $i_{\ell+1}$ and $r_{i_\ell, j_{\ell+1}} > p_{j_{\ell+1}}^{(t+1)}$. If P is a path, that is, if P does not visit slots j_{i_ℓ} twice, we simply unmatched bidder $i_{\ell+1}$, and flip matching and non-matching edges of P . (This keeps the size of the matching the same, as bidder i_0 gets matched and bidder $i_{\ell+1}$ unmatched.)

If P visits $j_{\ell+1}$ twice, it must be that $j_{\ell+1} = j_d$ for some d . Note that it is not the case that $d = \ell$, since this would mean that i_ℓ was matched to $j_{\ell+1}$. This is impossible because the reserve price on this edge has been reached just now. This way, the end of P forms a cycle with at least 2 bidders and 2 slots. We flip the matching and non-matching edges along the cycle, but leave the rest of P untouched. This leaves bidder i_x matched to slot j_{x+1} , for $x = d, d + 1, \dots, \ell$.

6. ANALYSIS

We sketch a proof that the STABLEMATCH algorithm from Section 5 computes a bidder-optimal stable matching for any auction instance (v, m, r) in general position. We do this by establishing several invariants that hold throughout the execution of the STABLEMATCH algorithm.

Besides invariants (A1) and (A2) introduced in Section 5, we claim three more invariants.

(A3) Each unmatched slot has zero price.

(B1) if a bidder i is interested in slot j and $u_i^{(t)} + m_{i,j} = v_{i,j}$, then $(i, j) \notin \mu^{(t)}$.

(B2) If a bidder i is interested in a slot j and $u_i^{(t)} + r_{i,j} = v_{i,j}$, then $(i, j) \in \mu^{(t)}$ or $p_j^{(t)} \geq r_{i,j}$.

To prove that all five invariants hold from the beginning to the end of an execution of STABLEMATCH, we proceed by induction on t . The proof itself is technical and tedious and is available in the full version [4]. Invariants (B1) and (B2) are needed in the induction step to prove the first three invariants. They rely on the general position assumption.

Invariants (A1) and (A2) are enough to show that the resulting matching is feasible and stable.

LEMMA 10. *The matching $(u^{(T)}, p^{(T)}, \mu^{(T)})$ computed by the STABLEMATCH algorithm is feasible and stable.*

PROOF. Stability follows directly from invariant (A1). Feasibility follows from invariant (A2) and the fact that since there are no alternating paths, it must be that $u_i^{(T)} = 0$ for every unmatched bidder i . \square

Running Time. The number of iterations is bounded by $O(nk)$ in Lemma 11 below (see proof in Appendix A.1). Since each iteration can be implemented in time $O(k^2)$, this gives us overall running time $O(nk^3)$.

LEMMA 11. STABLEMATCH finishes after at most $n(2k + 1)$ iterations.

6.1 Bidder Optimality

While the matching returned by STABLEMATCH is always stable and feasible, it may not be bidder-optimal. As the following example shows, a bidder-optimal matching does not always exist.

EXAMPLE 12. Consider the case of a single slot and two bidders with identical maximum bids. There are two stable matchings. In each matching, the slot is allocated to one of the bidders at maximum price. Each matching is preferred by one bidder over the other; hence there is no matching preferred by both of them.

This example is degenerate in that the maximum bids of both bidders are the same. However it turns out that except for such degenerate cases, a bidder-optimal matching always exists and STABLEMATCH will find it. We make this precise in the following two definitions.

DEFINITION 13 (AUCTION GRAPH). The auction graph of an auction (v, m, r) is a directed weighted bipartite multigraph with partite sets I and $J \cup \{j_0\}$, where j_0 is the dummy slot. The auction graph contains five types of edges. For each bidder i and each slot $j \in J$ there exist

- a forward edge from i to j with weight $-v_{i,j}$,
- a backward edge from j to i with weight $v_{i,j}$,
- a reserve-price edge from i to j with weight $r_{i,j} - v_{i,j}$,
- a maximum-price edge from i to j with weight $m_{i,j} - v_{i,j}$,
- a terminal edge from i to j_0 with weight 0.

DEFINITION 14 (GENERAL POSITION). An auction (v, m, r) is in general position if for every bidder i , no two alternating walks in the auction graph that start at bidder i , follow alternating forward and backward edges and end with a distinct edge that is either a reserve-price, maximum-price or terminal edge, have the same weight.

Any auction (v, m, r) can be brought into general position by a symbolic perturbation. In the algorithm implementation, this can be achieved by breaking ties lexicographically by the identity of the final edge of the walk.

All we need now to prove Theorem 5 is the following lemma, proof of which appears in Appendix A.2.

LEMMA 15. Let (v, m, r) be an auction in general position, and let (u', p', μ') be any feasible stable matching. Then in any iteration t of STABLEMATCH, we have that $u'_i \leq u_i^{(t)}$ for all $i \in I$ and $p'_j \geq p_j^{(t)}$ for all $j \in J$.

PROOF OF THEOREM 5. Consider an auction instance (m, v, r) in general position. The STABLEMATCH algorithm on this instance outputs a matching u^*, p^*, μ^* that is stable and feasible by Lemma 10. Applying Lemma 15 to the current matching after the last iteration of the algorithm implies that u^*, p^*, μ^* is weakly preferred to any stable matching by every bidder and hence is bidder-optimal. Running time of the algorithm follows from Lemma 11. \square

7. INCENTIVE COMPATIBILITY

In this section we will prove Theorem 6. A mechanism based on computing men-optimal stable matching has been shown to be truth-revealing in several contexts. For the basic stable matching problem without payments, a concise proof can be found in [21].

For the case of continuous utilities, a proof was given in [9]. Our proof for the max-value model mimics the overall structure of its predecessors. First, we show that there is no feasible matching in which every single bidder would be better off than in the bidder-optimal matching. (Note that if an agent or set of agents were to successfully lie about their preferences, the mechanism would still output a matching that is feasible with respect to the true preferences.) This property is known as weak Pareto optimality of the bidder-optimal matching.

LEMMA 16 (PARETO OPTIMALITY). Let (v, m, r) be an auction in general position and let (u^*, p^*, μ^*) be the bidder-optimal matching. Then for any matching (u, p, μ) that is feasible for (v, m, r) , there is at least one bidder $i \in I$ such that $u_i \leq u_i^*$.

Second, we show that every feasible matching is either stable, or has a blocking bidder-slot pair that involves a bidder who is not better off in this matching than in the bidder-optimal matching. Versions of the following lemma appear in [16, 11, 24]. The original statement in a model without money is attributed to J. S. Hwang.

LEMMA 17 (HWANG'S LEMMA). Let (u, p, μ) be a matching that is feasible for an auction (v, m, r) in general position and let (u^*, p^*, μ^*) be the bidder-optimal matching for that auction. Let

$$I^+ = \{i \in I \mid u_i > u_i^*\}.$$

If I^+ is non-empty, then there exists a blocking pair $(i, j) \in (I - I^+) \times J$.

Proofs of Lemmas 16 and 17 appear in Appendix B. Theorem 6 directly follows from Lemma 17. In fact, the lemma implies the following stronger statement.

THEOREM 18. There is no way for a bidder or a coalition of bidders to manipulate their bids in a way such that every bidder in the coalition would strictly benefit from the manipulation.

PROOF. Suppose there is a coalition I^+ of bidders that can benefit from submitting false bids. Let (v, m, r) be an auction that reflects the true preferences of all bidders, and let (v', m', r) be an auction that reflects the falsified bids. Note that $v'_i = v_i$ and $m'_i = m_i$ except for bidders $i \in I^+$.

Let (u, p, μ) be the bidder-optimal stable matching for the auction (v', m', r) . First observe that the matching (u, p, μ) must be feasible for the true auction (v, m, r) . This is because for each bidder $i \in I - I^+$, the feasibility constraints are the same in both auctions. For bidders $i \in I^+$, we need to verify that $p_j \leq m_{i,j}$ whenever $(i, j) \in \mu$. This follows because the true bidder-optimal matching (u^*, p^*, μ^*) respects maximum prices, and any outcome that respects maximum prices is preferred over an outcome that doesn't.

Since (u, p, μ) is feasible, we can apply Lemma 17 and conclude that there is a pair (i, j) with $i \in I - I^+$ that is blocking for the auction (v, m, r) . \square

8. CONCLUSIONS

We have successfully applied the theory of stable matchings to sponsored search auctions. Several open questions remain.

Fujishige and Tamura [14] propose a general model in which a worker can engage several firms and vice versa, of which ours is a special case. It would be interesting to see if (and under what conditions) worker and firm-optimal equilibria exist, and whether our strategyproofness result carries through to this very general model.

Our max-value model assumes a constant “exchange rate” in that each dollar paid by the bidder is perceived as a dollar received by the seller, independent of the identity of the bidder and the item. Suppose the payment is conditioned on some event (such as a user clicking or making a purchase), as is common practice. At a mutually agreed (say) cost per click, the total revenue estimated by the seller may not be equal to the total cost estimated by the buyer, if they have different estimates of the probability of a click occurring. This discrepancy suggests that we introduce an exchange rate into equations (2) and (5). In such a model with exchange rates, we do not know if a stable (let alone bidder optimal) matching exists, or how to find such matching efficiently.

Existence of bidder-optimal matchings in our model has clear implications on the existence of Nash equilibria in (say) GSP auctions under various assumptions on bidder valuations / preferences. (For example, can the result of [13] be re-derived and extended by using guaranteed existence of bidder-optimal matchings?)
Acknowledgments. We would like to thank Hal Varian, Adam Juda and anonymous referees for helpful comments and pointers to literature.

9. REFERENCES

- [1] Gagan Aggarwal. *Privacy Protection and Advertising in a Networked World*. PhD thesis, Stanford University, 2005.
- [2] Gagan Aggarwal, Ashish Goel, and Rameesh Motwani. Truthful auctions for pricing search keywords. In *ACM Conf on Electronic commerce*, pages 1–7, 2006.
- [3] Gagan Aggarwal, S. Muthukrishnan, and Jon Feldman. Bidding to the top: VCG and equilibria of position-based auctions. In *WAOA*, 2006.
- [4] Gagan Aggarwal, S. Muthukrishnan, Dávid Pál, and Martin Pál. General auction mechanism for search advertising. Available from <http://arxiv.org/abs/0807.1297>.
- [5] Ahmet Alkan. Existence and computation of matching equilibria. *European Journal of Political Economy*, 5(2-3):285–296, 1989.
- [6] Ahmet Alkan and David Gale. The core of the matching game. *Games and Economic Behavior*, 2(3):203–212, 1990.
- [7] Sushil Bikhchandani and Joseph M. Ostroy. From the assignment model to combinatorial auctions. In *Combinatorial Auctions*. MIT Press, 2006.
- [8] Edward H. Clarke. Multipart pricing of public goods. *Public Choice*, 11(1):17–33, 1971.
- [9] G. Demange and D. Gale. The strategy structure of two-sided matching markets. *Econometrica*, 53(4):873–888, 1985.
- [10] Gabrielle Demange, David Gale, and Marilda Sotomayor. Multi-item auctions. *Journal of Political Economy*, 94(4):863–872, 1986.
- [11] Gabrielle Demange, David Gale, and Marilda Sotomayor. A further note on the stable matching problem. *Discrete Applied Mathematics*, 16:217–222, 1987.
- [12] Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords. *American Economic Review*, 97(1):242–259, March 2007.
- [13] Eyal Even-Dar, Jon Feldman, Yishay Mansour, and S. Muthukrishnan. On the effect of minimum prices on position auctions. In *WINE*, 2008.
- [14] Satoru Fujishige and Akihisa Tamura. A two-sided discrete-concave market with possibly bounded side payments: An approach by discrete convex analysis. *Mathematics of Operations Research*, 32(1):136–155, 2007.
- [15] David Gale and Lloyd S. Shapley. College admissions and the stability of marriage. *Am Math Monthly*, 69(1):9–15, 1962.
- [16] David Gale and Marilda Sotomayor. Some remarks on the stable marriage problem. *Discrete Applied Mathematics*, 11:223–232, 1985.
- [17] Theodore Groves. Incentives in teams. *Econometrica*, 41(4):617–631, 1973.
- [18] J. A. S. Kelso and V. P. Crawford. Job matching, coalition formation, and gross substitutes. *Econometrica*, 50:1483–1504, 1982.
- [19] Harold W. Kuhn. The Hungarian method for the assignment problem. *Naval Logistics Quarterly*, 2:83–97, 1955.
- [20] Herman B. Leonard. Elicitation of honest preferences for the assignment of individuals to positions. *Journal of Political Economy*, 91:461–479, 1983.
- [21] Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani, editors. *Algorithmic Game Theory*. Cambridge University Press, September 2007.
- [22] M. Quinzii. Core and competitive equilibria with indivisibilities. *International Journal of Game Theory*, 13(1):41–60, 1984.
- [23] Alvin E. Roth. The economics of matching: Stability and incentives. *Mathematics of Operations Research*, 7(4):617–628, 1982.
- [24] Alvin E. Roth and Marilda A. Oliveira Sotomayor. *Two-sided matching: A study in game-theoretic modeling and analysis*. Cambridge University Press, 1990.
- [25] Lloyd S. Shapley and Martin Shubik. The assignment game i: The core. *Intl J. of Game Theory*, 1(1):111–130, 1971.
- [26] Hal R. Varian. Position auctions. *International Journal of Industrial Organization*, 2006.
- [27] William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *J. of Finance*, 16(1):8–37, 1961.
- [28] Wikipedia. http://en.wikipedia.org/wiki/Hungarian_algorithm.

APPENDIX

A. ANALYSIS OF STABLEMATCH

Proofs of statements from Section 6.1.

A.1 Proof of Lemma 11

PROOF OF LEMMA 11. Consider the number of edges in the update graph. Initially, the graph $G^{(0)}$ has at most nk reserve-price, nk maximum-price and n terminal edges. We claim that in each iteration, the number of edges in the update graph is reduced by one. Since STABLEMATCH must stop when there are no more edges left, this bounds the total number of iterations.

Consider an iteration t of STABLEMATCH. We claim that in the alternating path $P = (i_0, j_1, i_1, \dots, j_\ell, i_\ell, j_{\ell+1})$, the last edge $(i, j) = (i_\ell, j_{\ell+1})$ will not appear in the update graph $G^{(t+1)}$. This is easily verified by considering three cases:

Case 1: If (i, j) is a terminal edge, then $w^{(t)}(P) = d^{(t)}(i_0, i) + u_i^{(t)}$ and hence $u_i^{(t+1)} = u_i^{(t)} - (w^{(t)}(P) - d^{(t)}(i_0, i)) = 0$.

Case 2: If (i, j) is a maximum-price edge, then $w^{(t)}(P) = d^{(t)}(i_0, i) + (u_i^{(t)} + m_{i,j} - v_{i,j})$ and hence $u_i^{(t+1)} + m_{i,j} = u_i^{(t)} - (w^{(t)}(P) - d^{(t)}(i_0, i)) + m_{i,j} = v_{i,j}$.

Case 3: If (i, j) is a reserve-price edge, then $w^{(t)}(P) = d^{(t)}(i_0, i) + (u_i^{(t)} + r_{i,j} - v_{i,j})$ and hence $u_i^{(t+1)} + r_{i,j} = u_i^{(t)} - (w^{(t)}(P) - d^{(t)}(i_0, i)) + r_{i,j} = v_{i,j}$.

The utilities never increase and the prices never decrease throughout the algorithm, thus the edge $(i_\ell, j_{\ell+1})$ does not appear in any update graph $G^{(t')}$ for any $t' > t$. \square

A.2 Proof of Lemma 15

Without loss of generality assume that (u, p, μ) is such that there does not exist a pair $(i, j) \in \mu$ such that $p_j = m_{i,j}$. If there was such a pair, then we can decrease prices of some of the items and increase utilities of some of the bidders such that $p_j < m_{i,j}$. This is possible because of the general position assumption. See full version of the paper.

We prove Lemma 15 by induction on t . The base case, $t = 0$, trivially holds true, since by feasibility of (u', p', μ') , $p'_j \geq 0$ for all $j \in J$ and $u'_i \leq B$ for all $i \in I$. In the inductive case, assume that $u^{(t)} \geq u'$ and $p^{(t)} \leq p'$. We first prove that

PROPOSITION 19. $u^{(t+1)} \geq u'$ and $p^{(t+1)} \leq p'$.

We look ‘‘continuously’’ at updates (6) and (7). For that purpose we define for each $i \in I$ a continuous non-increasing function $u_i(x)$,

$$u_i(x) = u_i^{(t)} - \max(x - d^{(t)}(i_0, i), 0),$$

and for each $j \in J$ a continuous non-decreasing function $p_j(x)$,

$$p_j(x) = p_j^{(t)} + \max(x - d^{(t)}(i_0, j), 0).$$

Clearly, $u^{(t+1)} = u(w^{(t)}(P))$ and $p^{(t+1)} = p(w^{(t)}(P))$. To prove that $u^{(t+1)} \geq u'$ and $p^{(t+1)} \leq p'$, suppose by contraction that there exists $y \in [0, w^{(t)}(P)]$ such that either $u_i(y) < u'_i$ for some $i \in I$ or $p_j(y) > p'_j$ for some $j \in J$. We choose infimal such y . Clearly, $u(y) \geq u'$, $p(y) \leq p'$ and $y < w^{(t)}(P)$. Consider the sets

$$I' = \{i \in I \mid u_i(y) = u'_i \text{ and } d^{(t)}(i_0, i) \leq y\},$$

$$J' = \{j \in J \mid p_j(y) = p'_j \text{ and } d^{(t)}(i_0, j) \leq y\}.$$

CLAIM 20. Each slot $j \in J'$ is matched in $\mu^{(t)}$ to some $i \in I'$.

PROOF. Let $j \in J'$. If j was unmatched, then either $d^{(t)}(i_0, j) = w^{(t)}(P)$ or $d^{(t)}(i_0, j) = \infty$; however both options contradict the choice of y and that $j \in J'$. Thus j is matched to some $i \in I$, hence in $G^{(t)}$ there is a backward edge from j to i and thus $d^{(t)}(i_0, i) = d^{(t)}(i_0, j)$ and therefore $u_i(y) + p_j(y) = v_{i,j}$. Further, invariants (A2) and (B1) imply that $p_j^{(t)} \in [r_{i,j}, m_{i,j}]$. Consequently, there is a maximum-price edge from i to j , $w^{(t)}(P) \leq d^{(t)}(i_0, i) + (u_i^{(t)} + m_{i,j} - v_{i,j})$, and hence $p'_j = p_j(y) < p_j^{(t+1)} = p^{(t)} + (w^{(t)}(P) - d^{(t)}(i_0, j)) \leq m_{i,j}$. Therefore $p'_j \in [r_{i,j}, m_{i,j}]$, and since (u', p', μ') is stable, $u'_i + p'_j \geq v_{i,j}$ and hence $u_i(y) = v_{i,j} - p_j(y) = v_{i,j} - p'_j \leq u'_i$. On the other hand, by infimality of y , $u_i(y) \geq u'_i$. Thus $i \in I'$. \square

CLAIM 21. Each bidder $i \in I'$ is matched in μ' to some $j \in J'$.

PROOF. Since in $G^{(t)}$ there is a terminal edge from i to the dummy slot, $w^{(t)}(P) \leq d^{(t)}(i_0, i) + u_i^{(t)}$. Hence

$$u'_i = u_i(y) = u_i^{(t)} - (y - d^{(t)}(i_0, i))$$

$$> u_i^{(t)} - (w^{(t)}(P) - d^{(t)}(i_0, i)) \geq 0,$$

and thus bidder i is matched in μ' to some slot $j \in J$.

By feasibility of (u', p', μ') , $p'_j \in [r_{i,j}, m_{i,j}]$. By the assumption made at the beginning $p_j \neq m_{i,j}$. Therefore in $G^{(t)}$ there is a forward edge from i to j and thus

$$d^{(t)}(i_0, j) \leq d^{(t)}(i_0, i) + (u_i^{(t)} + p_j^{(t)} - v_{i,j}). \quad (8)$$

Clearly, since $i \in I'$,

$$u_i(y) = u_i^{(t)} - (y - d^{(t)}(i_0, i)). \quad (9)$$

By the price update rule

$$p_j(y) \geq p_j^{(t)} + (y - d^{(t)}(i_0, j)). \quad (10)$$

We add (9) to (10) and subtract from that (8) and we obtain

$$p_j(y) \geq v_{i,j} - u_i(y).$$

Hence, since by feasibility of (u', p', μ') , $u'_i + p'_j = v_{i,j}$, we have

$$p_j(y) \geq v_{i,j} - u_i(y) = v_{i,j} - u'_i = p'_j.$$

Recalling that $p(y) \leq p'$ we see that $p_j(y) = p'_j$.

Subtracting (9) from (8) and cancelling common terms we have

$$d^{(t)}(i_0, j) \leq y + (u_i(y) + p_j^{(t)} - v_{i,j}).$$

We upper-bound the right side of the inequality using that $u_i(y) = u'_i$, $p_j^{(t)} \leq p_j(y)$ and $u'_i + p'_j = v_{i,j}$ and we have

$$d^{(t)}(i_0, j) \leq y + (u'_i + p'_j - v_{i,j}) = y.$$

Thus $j \in J'$. \square

From the two claims it follows that $|I'| = |J'|$ and that $\mu^{(t)}$ bijectively matches I' with J' . In particular $i_0 \notin I'$. Choose $j \in J'$ with smallest $d^{(t)}(i_0, j)$. Consider the minimum-weight path in $G^{(t)}$ from i_0 to j which uses only forward and backward edges. The vertex on the path just before j is a bidder $i \notin I'$. Clearly, $y \geq d^{(t)}(i_0, j) > d^{(t)}(i_0, i)$ and hence $u_i(y) < u'_i$. There is a forward edge from i to j , thus $p_j^{(t)} \in [r_{i,j}, m_{i,j}]$ and also $u_i(y) + p_j(y) = v_{i,j}$, and hence (*) $u'_i + p'_j < v_{i,j}$. Since in $G^{(t)}$ there is a maximum-price edge from i to j , $p'_j = p_j(y) < m_{i,j}$, which together with (*) contradicts stability of (u', p', μ') . This proves Proposition 19.

To prove Lemma 15 it remains to show that $p^{(t+1)} \leq p'$. This amounts to show that if $(u^{(t+1)}, p^{(t+1)}, \mu^{(t+1)})$ was obtained from $(u^{(t)}, p^{(t)}, \mu^{(t)})$ by updating along an alternating path P of which the last edge, $(i, j) = (i_\ell, j_{\ell+1})$, was a reserve-price edge and $p_j^{(t+1)} < r_{i,j}$, then

$$r_{i,j} \leq p'_j. \quad (11)$$

Since (u', p', μ') is stable, either $u'_i + p'_j \geq v_{i,j}$ or $p'_j \geq m_{i,j}$. In former case, (11) follows from that $u_i^{(t+1)} = v_{i,j} - r_{i,j}$, Proposition 19 and that (u', p', μ') is stable. In latter case, (11) follows since the presence of the reserve-price edge from i to j guarantees that $m_{i,j} > r_{i,j}$.

B. INCENTIVE COMPATIBILITY PROOFS

PROOF OF LEMMA 16. For the sake of contradiction, suppose that there is a feasible matching (u, p, μ) such that $u_i > u_i^*$ for all $i \in I$. Note that every bidder must be matched in μ , since $u_i > u_i^* \geq 0$.

For each bidder $i \in I$, consider the slot $j = \mu(i)$ matched to bidder i in the matching μ . Since the pair (i, j) is not blocking for

the bidder-optimal matching (u^*, p^*, μ^*) , it must be that $p_j^* > p_j$. In particular, the existence of μ implies that there must be n slots with positive prices in the bidder-optimal matching μ^* , and that these slots are matched in μ as well.

If a slot ever becomes matched to a bidder in the STABLEMATCH algorithm, it will never become unmatched. Thus before the last iteration, at most $n - 1$ slots have positive prices. Suppose the last iteration, iteration $T - 1$, increases the size of the matching to n , and let j be the last slot to be matched. Let $i' = \mu(j)$ be the bidder matched to j in the hypothetical matching μ .

Let P be the shortest alternating path found in Step 1 of the last iteration of STABLEMATCH. Recall that the first vertex of the path is denoted by i_0 and $w^{(T-1)}(P)$ denotes its length. If P ends with the reserve-price edge (i, j) , it must be that i and j are matched in both μ and μ^* at the same reserve price, contradicting our assumption that $u_i > u_i^*$.

On the other hand, if P does not end with the reserve-price edge (i, j) , we show that there is a shorter alternating path P' that does include this edge, which again leads to a contradiction. From Step 3 of the last iteration we have $u_i^{(T-1)} - u_i^* = w^{(T-1)}(P) - d^{(T-1)}(i_0, i)$. Let s be the length of the reserve price edge (i, j) ; recall from Definition 8 that $s = u_i^{(T-1)} + r_{i,j} - v_{i,j}$. Now consider the alternating path P' that consists of the shortest path from i_0 to i followed by the reserve price (i, j) edge. We have

$$w^{(T-1)}(P) - w^{(T-1)}(P') = u_i^{(T-1)} - u_i^* - s = v_{i,j} - r_{i,j} - u_i^*.$$

Since $u_i^* < u_i \leq v_{i,j} - r_{i,j}$, this difference is positive and hence P' must be a shorter alternating path than P . \square

PROOF OF LEMMA 17. Without loss of generality assume that (u, p, μ) is such that there does *not* exist a pair $(i, j) \notin \mu$ such that $u_i + r_{i,j} = v_{i,j}$. If there was such a pair, then we can decrease prices of some of the items and increase utilities of some of the bidders such that $u_i + r_{i,j} > v_{i,j}$. (This is possible because of the general position assumption. See full version of the paper.) The set I^+ would only grow by such operation.

Let us denote by $\mu(I^+)$, $\mu^*(I^+)$ the set of slots matched to bidders in I^+ in matching respectively μ , μ^* . We consider two cases:

Case 1: $\mu(I^+) \neq \mu^*(I^+)$. For any $i \in I^+$ we have $u_i > u_i^* \geq 0$ and hence each bidder in I^+ is matched in μ to some slot. There exists a slot $j \in \mu(I^+)$, $j \notin \mu^*(I^+)$. Let $i = \mu(j)$. Since $i \in I^+$, $u_i > u_i^*$.

We argue that $p_j < p_j^*$: By the general position assumption $p_j^* \neq m_{i,j}$, and hence by feasibility of (u, p, μ) , $p_j \in [r_{i,j}, m_{i,j})$ and $u_i + p_j = v_{i,j}$. Hence $u_i^* + p_j^* \geq v_{i,j}$. Therefore $p_j^* \geq v_{i,j} - u_i^* > v_{i,j} - u_i = p_j$.

In particular, j is matched in μ^* to some i' , and by the choice of j , $i' \notin I^+$. Thus $u_{i'} \leq u_{i'}^*$. By feasibility of (u^*, p^*, μ^*) , $p_j^* \in [r_{i',j}, m_{i',j})$ and $u_{i'}^* + p_j^* = v_{i',j}$. By the assumption on (u, p, μ) that we made at the beginning of the proof, $u_{i'} \neq v_{i',j} - r_{i',j}$.

Now, it is not hard to see that (i', j) is blocking pair for μ . This is because

$$\begin{aligned} p_j &< p_j^* \leq m_{i,j}, \\ u_{i'} &\leq u_{i'}^* = v_{i',j} - p_j^* \leq v_{i',j} - r_{i',j} \quad \text{and} \\ u_{i'} &\neq v_{i',j} - r_{i',j}, \\ u_{i'} + p_j &< u_{i'}^* + p_j^* = v_{i',j}. \end{aligned}$$

Case 2: $\mu(I^+) = \mu^*(I^+) = J^+$. Since $u_i > u_i^*$ for $i \in I^+$, by stability of (u^*, p^*, μ^*) it follows that $p_j < p_j^*$ for $j \in J^+$.

Consider a reduced auction (v', m', r') on the set of bidders I^+ and set of slots J^+ . We set the reserve prices to reflect the influence of bidders in $I \setminus I^+$. More specifically, let $I' = \{i \in I \setminus I^+ \mid u_{i'}^* \geq$

$v_{i',j} - r_{i',j}\}$. For every $i \in I^+$ and $j \in J^+$, we set

$$r'_{i,j} = \max(r_{i,j}, \max_{i' \in I'} \min(m_{i',j}, v_{i',j} - u_i^*)).$$

We also set $v'_{i,j} = v_{i,j}$ and $m'_{i,j} = m_{i,j}$ except that if $m_{i,j} \leq r'_{i,j}$ we set $m'_{i,j} = -1$. It is not hard to show that if v, m, r is in general position, then so is (v', m', r') , using the fact that each utility u_i^* was at some point set to be equal to the length of some alternating walk in the auction graph.

Now consider the matchings μ and μ^* restricted to the sets I^+ , J^+ . If the restricted μ is not feasible for (v', m', r') , it must be because $p_j < r_{i,j}$ for some position $j = \mu(i)$. This can only happen if $r'_{i,j} > r_{i,j}$ and hence $r'_{i,j} = \max(m_{i',j}, v_{i',j} - u_i^*)$ for some bidder $i' \in I \setminus I^+$.

On the other hand, it is easy to check that the restricted matching μ^* is feasible, stable and bidder-optimal for the auction (v', m', r') . If the restricted μ is feasible for this auction, by Lemma 16, there is a bidder $i \in I^*$ such that $u_i \leq u_i^*$. This however contradicts the definition of the set I^+ . \square

C. LATTICE PROPERTY

The set of feasible and stable outcomes in both the stable marriage and the assignment model has the algebraic structure of a lattice (see e.g. Chapter 3 in [24]). This result can be carried over to our assignment model with minimum and maximum prices as well. The following lemma can be proved using ideas and techniques from Section A. The proof is relatively long and tedious and is omitted.

LEMMA 22 (LATTICE PROPERTY). *Let (v, m, r) be an auction in general position. If (u^A, p^A, μ^A) and (u^B, p^B, μ^B) are two feasible stable matchings for (v, m, r) , then there exists a feasible stable matching (u^C, p^C, μ^C) for (v, m, r) such that*

$$\begin{aligned} u_i^C &= \max\{u_i^A, u_i^B\} && \text{for each } i \in I, \\ p_j^C &= \min\{p_j^A, p_j^B\} && \text{for each } j \in J, \end{aligned}$$

and there exists a feasible stable matching (u^D, p^D, μ^D) for (v, m, r) such that

$$\begin{aligned} u_i^D &= \min\{u_i^A, u_i^B\} && \text{for each } i \in I, \\ p_j^D &= \max\{p_j^A, p_j^B\} && \text{for each } j \in J. \end{aligned}$$

The set \mathcal{M} of feasible and stable matchings for an auction (v, m, r) is non-empty by 10. If the auction instance is in general position we know that \mathcal{M} is also a lattice by Lemma 22. It is not hard to see that \mathcal{M} is closed and bounded, and hence must have a minimum and maximum element. This gives us an alternate way of proving that a bidder-optimal stable matching exists.