Private Matchings and Allocations*

Justin Hsu[†]
University of Pennsylvania

Zhiyi Huang
Stanford University
University of Hong Kong

Aaron Roth[‡]
University of Pennsylvania

Tim Roughgarden[§]
Stanford University

Zhiwei Steven Wu[¶]
University of Pennsylvania

ABSTRACT

We consider a private variant of the classical *allocation problem*: given k goods and n agents with individual, private valuation functions over bundles of goods, how can we partition the goods amongst the agents to maximize social welfare? An important special case is when each agent desires at most one good, and specifies her (private) value for each good: in this case, the problem is exactly the maximum-weight matching problem in a bipartite graph.

Private matching and allocation problems have not been considered in the differential privacy literature, and for good reason: they are plainly impossible to solve under differential privacy. Informally, the allocation must match agents to their preferred goods in order to maximize social welfare, but this preference is exactly what agents wish to hide! Therefore, we consider the problem under the relaxed constraint of *joint differential privacy*: for any agent i, no coalition of agents excluding i should be able to learn about the valuation function of agent i. In this setting, the full allocation is no longer published—instead, each agent is told what good to get. We first show that with a small number of identical copies of each good, it is possible to efficiently and accurately solve the maximum weight matching problem while guaranteeing joint differential privacy. We then consider the more general allocation problem, when bidder valuations satisfy the *gross substitutes* condition. Finally, we prove that the allocation problem cannot be solved to non-trivial accuracy under joint differential privacy without requiring multiple copies of each type of good.

Categories and Subject Descriptors

F.2.0 [**Analysis of Algorithms**]: General

Keywords

Differential Privacy, Matching, Ascending Auction, Gross Substitutes

1. INTRODUCTION

The classic maximum-weight matching problem in bipartite graphs can be viewed as follows: there are k goods $j \in \{1, \ldots, k\}$ and n buyers $i \in \{1, \ldots, n\}$. Each buyer i has a value $v_{ij} \in [0, 1]$ for each good j, and the goal is to find a matching μ between goods and buyers which maximizes the social welfare: $\text{SW} = \sum_{i=1}^{n} v_{i,\mu(i)}$. When the goods are sensitive,[1] it is natural to ask for a matching that hides the reported values of each of the players.

It is not hard to see that this is impossible under the standard notion of differential privacy, which insists that the allocation must be insensitive to the reported valuations of each player. We formalize this in Section 5, but the intuition is simple: consider the case with two types of goods with n identical copies each, and suppose that each buyer has a private preference for one of the two types: value 1 for the good that he likes, and value 0 for the other good. There is no contention since the supply of each good is larger than the total number of buyers, so any allocation achieving social welfare $\text{OPT} - \alpha n$ can be used to reconstruct a $(1 - \alpha)$ fraction of the preferences; this is impossible for non-trivial values of α under differential privacy.

In light of this observation, is there any hope for privately solving maximum-weight matching problems? In this paper, we show that the answer is *yes*: it is possible to solve matching problems (and more general allocation problems) to high accuracy assuming at least a small number of identical copies of each good, while still satisfying an extremely strong variant of differential privacy. We observe that the matching problem has the following two features:

1. Both the input and solution are naturally partitioned amongst the same n people: in our case, each buyer i receives the item $\mu(i)$ she is matched to in the solution.

2. The problem is not solvable privately because the item given to a buyer must reflect her private data, but this need not (necessarily) be the case for items given to other buyers.

By utilizing these two features, we show that the matching problem can be accurately solved under the constraint of *joint differ-*

[1]For instance, the goods might be related to the treatment of disease, or might be indicative of a particular business strategy, or might be embarrassing in nature.

*A full version of this paper can be found at http://arxiv.org/abs/1311.2828

[†]Supported in part by NSF Grant CNS-1065060.

[‡]Supported in part by an NSF CAREER award, NSF Grants CCF-1101389 and CNS-1065060, and a Google Focused Research Award.

[§]This research was supported in part by NSF Awards CCF-1016885 and CCF-1215965 and an ONR PECASE Award.

[¶]Supported in part by NSF Grant CCF-1101389.

STOC '14, May 31–June 03 2014, New York, NY, USA

ACM 978-1-4503-2710-7/14/05 ...$15.00.
http://dx.doi.org/10.1145/2591796.2591826.

ential privacy [12]. Informally speaking, this requires that for every buyer i, the joint distribution on items $\mu(j)$ for $j \neq i$ must be differentially private in the reported valuation of buyer i. As a consequence, buyer i's privacy is protected even if *all* other buyers collude against him, potentially sharing the identities of the items they receive. As long as buyer i does not reveal her own item, her privacy is protected.

We then show that our techniques generalize well beyond the max-matching problem, to the more general *allocation* problem—in this setting, each buyer i has a valuation function defined over subsets of goods $v_i : 2^{[k]} \to [0, 1]$ from some class of valuations, and the goal is to find a partition of the goods $S_1, \ldots, S_n$ maximizing social welfare. (Note that the maximum-weight matching problem is the special case when agents are *unit demand*, i.e., only want bundles of size 1.) We generalize our algorithm to solve the allocation problem when bidders' valuations satisfy the *gross substitutes* condition. This is an economically meaningful class of valuation functions that is a strict subclass of submodular functions, and (as we will explain) are the most general class of valuation functions for which our techniques could possibly apply.

1.1 Our Techniques and Results

Our approach makes a novel connection between *market clearing prices* and differential privacy. Prices have long been considered as a low information way to coordinate markets; conceptually, our paper formalizes this intuition in the context of differentially private allocation. Our algorithm is a differentially private implementation of m simultaneous ascending price auctions, one for each type of good. Following the classic analysis of Kelso and Crawford [13], the prices in these auctions converge to *Walrasian equilibrium prices*: prices under which each buyer is simultaneously able to buy his most preferred bundle of goods. We show that although the allocation itself cannot be computed under standard differential privacy, the Walrasian equilibrium prices can be, and that the computation of these prices can be used to coordinate a high welfare allocation while satisfying joint differential privacy.

The classical ascending price auction works as follows. Each good begins with a price of 0, and each agent is initially unmatched to any good. Unmatched agents i take turns bidding on the good j^* that maximizes their utility at the current prices: i.e., $j^* \in \arg\max(v_{ij} - p_j)$. When a bidder bids on a good j^*, he becomes the new high bidder and the price of j^* is incremented. Bidders are tentatively matched to a good as long as they are the high bidder. The auction continues until there are no unmatched bidders who would prefer to be matched to any of the goods at the current prices. The algorithm necessarily converges because each bid increases the sum of the prices of the goods, and prices are bounded by some finite value.[2] Moreover, by construction, every bidder ends up matched to their most preferred good given the prices. Finally, by the "First Welfare Theorem" of Warlasian equilibria, any matching that corresponds to these equilibrium prices maximizes social welfare. We emphasize that it is this final implication that is the key: "prices" play no role in our problem description, nor do we ever actually charge "prices" to the agents—the prices are purely a device to coordinate the matching.

We give an approximate, private version of this algorithm based on several observations. First, in order to implement this algorithm, it is sufficient to maintain the sequence of prices of the goods privately: given a record of the price trajectory, each agent can figure out for himself what good he is matched to. Second, in order to privately maintain the prices, it suffices to maintain a private count of the number of bids each good has received over the course of the auction. Finally, it turns out that it is possible to halt the algorithm early without significantly harming the quality of the final matching. This guarantees that no bidder ever makes more than a small number (independent of both n and k) of total bids, which allows us to bound the sensitivity of the bid-counters. Together, these observations allow us to implement the auction privately using work by Dwork et al. [5] and Chan et al. [2], who introduce counters with the privacy properties we need. The result is an algorithm that converges to a matching together with prices that form an approximate Walrasian equilibrium. We complete our analysis by proving an approximate version of the first welfare theorem, which shows that the matching has high weight.

Our algorithm actually works in a stronger privacy model, which we call the *billboard model*. The algorithm posts the prices publicly on a *billboard* as a differentially private signal such that every player can deduce what object she should be matched to just from her own private information and the contents of the billboard. As we show, algorithms in the billboard model automatically satisfy joint differential privacy.

Forthermore, we view implementations in the billboard model as preferable to arbitrary jointly differentially private implementations. This is because algorithms in the billboard model only need the ability to publish sanitized messages to all players, and do not need a secure channel to communicate the mechanisms' output to each player (though of course, there still needs to be a secure channel from the player to the mechanism). The work of McSherry and Mironov [14] and some of the results of Gupta et al. [9] can be viewed as previous algorithms implemented in this mold.

The algorithm of Kelso and Crawford [13] extends to the general allocation problem when players have gross substitute preferences, and our private algorithm does as well. We note that this class of preferences is the natural limit of our approach, which makes crucial use of equilibrium prices as a coordinating device: in general, when agents have valuations over bundles of goods that do not satisfy the gross substitutes condition, Walrasian equilibrium prices may not exist.

Finally, we give lower bounds showing that our results are qualitatively tight: not only is the problem impossible to solve under the standard differential privacy, to get any non-trivial solution even under *joint* differential privacy, it is necessary to assume that there are multiple copies of each type of good. Our lower bounds are all fundamentally reductions to database reconstruction attacks. Our lower bound for joint-differentially private algorithms may be of general interest, as we believe it forms a good template for other lower bounds for joint differential privacy.

We first state our main result informally in the special case of max-matchings, which we prove in Section 3. We prove our more general theorem for allocation problems with gross substitutes preferences in Section 4. Here, privacy is protected with respect to a single agent i changing her valuations v_{ij} for possibly *all* goods j.

Theorem (Informal)**.** *There is a computationally efficient ε-joint differentially private algorithm which computes a matching of weight $\mathrm{OPT} - \alpha n$ in settings in which there are n agents and k types of goods, with s copies of each good when:*

$$s \geq O\left(\frac{1}{\alpha^3 \varepsilon} \cdot \mathrm{polylog}\left(n, k, \frac{1}{\alpha}\right)\right).$$

In certain settings, the welfare guarantee can be improved to $(1 - \alpha)\,\mathrm{OPT}$.

We complement this result with several lower bounds in Section 5. We show that no algorithm can solve the private max-

[2] Bidders do not bid on goods for which they have negative utility; in our case, $v_{ij} \in [0, 1]$

matchings problem to non-trivial accuracy under the standard constraint of differential privacy. We also show that even under joint differential privacy, it is necessary to assume that there are multiple copies of each item.

Theorem (Informal)**.** *No joint differentially private algorithm can compute matchings of weight greater than* $\mathrm{OPT} - \alpha n$ *on instances in which there are* n *agents and* s *copies of each good, when* $s \leq O\left(1/\sqrt{\alpha}\right)$.

In particular, no algorithm can compute matchings of weight $\mathrm{OPT} - o(n)$ on instances for which the supply $s = O(1)$. In addition, we show that when goods have supply only $s = O(1)$, it is not even possible to compute the equilibrium prices privately under standard differential privacy.

1.2 Related Work

Differential privacy, first defined by Dwork et al. [4], has become a standard "privacy solution concept" in the theoretical computer science literature. There is far too much work to survey comprehensively; for a textbook introduction, see Dwork and Roth [7].

The privacy of our algorithms relies on work by Dwork et al. [5] and Chan et al. [2], who show how to release a running count of a stream of bits under *continual observation*—i.e., report the count as the stream is revealed, provide high accuracy at every point in time, while keeping the transcript differentially private.

Beginning with Dinur and Nissim [3], much work in differential privacy has focused on answering numeric valued queries on a private dataset (e.g., Blum et al. [1], Dwork et al. [4], Hardt and Rothblum [10], among many others). In contrast, work on private combinatorial optimization problems has been sporadic (but not non-existant, e.g., Gupta et al. [9], Nissim et al. [15]). Part of the reason is that many combinatorial optimization problems are impossible to solve under differential privacy (including the allocation problems we consider in this paper). To sidestep this problem, we employ the solution concept of *joint differential privacy*. First formalized by Kearns et al. [12], similar ideas are present in the vertex and set-cover algorithms of Gupta et al. [9], the private recommendation system of McSherry and Mironov [14], and the analyst private data analysis algorithms of Dwork et al. [6], Hsu et al. [11].

The utility of our algorithm relies on analysis due to Kelso and Crawford [13], who study the problem of matching *firms* to *workers* when the firms have preferences that satisfy the *gross substitutes* condition. They give an algorithm based on simulating simultaneous ascending auctions that converge to *Walrasian equilibrium prices*, together with a corresponding matching. In this respect, our approach is complete: Gul and Stacchetti [8] show that gross substitutes preferences are precisely the set of preferences for which Walrasian equilibrium prices are guaranteed to exist.

While our approximate equilibrium achieves good approximation to the optimal welfare at the expense of certain incentive properties, our work is closely related to recent work on privately computing various kinds of equilibrium in games (e.g., correlated equilibrium [12], Nash equilibrium [17], and minmax equilibrium [11]). These works belong to a growing literature studying the interface of game theory and differential privacy; for a recent survey, see Pai and Roth [16].

2. PRELIMINARIES

2.1 The Allocation Problem

We consider allocation problems defined by a set of goods G, and a set of n agents $[n]$. Each agent $i \in [n]$ has a *valuation function* $v_i : 2^G \to [0,1]$ mapping bundles of goods to values. A *feasible allocation* is a collection of sets $S_1, \ldots, S_n \subseteq G$ such that $S_i \cap S_j = \emptyset$ for each $i \neq j$: i.e., a partition of goods among the agents. The *social welfare* of an allocation $S_1, \ldots, S_n$ is defined to be $\sum_{i=1}^n v_i(S_i)$, the sum of the agent's valuations for the allocation; we are interested in finding allocations which maximize this quantity. Given an instance of an allocation problem, we write $\mathrm{OPT} = \max_{S_1,\ldots,S_n} \sum_{i=1}^n v_i(S_i)$ to denote the social welfare of the optimal feasible allocation.

A particularly simple valuation function is a *unit demand valuation*, where bidders demand at most one item. Such valuation functions take the form $v_i(S) = \max_{j \in S} v_i(\{j\})$, and can be specified by numbers $v_{i,j} = v_i(\{j\}) \in [0,1]$, which represent the value that bidder i places on good j. When bidders have unit demand valuations, the allocation problem corresponds to computing a maximum weight matching in a bipartite graph.

Our results will also hold for *gross substitute valuations*, which include unit demand valuations as a special case. Informally, for gross substitute valuations, any set of goods S' that are in a most-demanded bundle at some set of prices p remain in a most-demanded bundle if the prices of *other* goods are raised, keeping the prices of goods in S' fixed. Gross substitute valuations are a standard class of valuation functions: they are a strict subclass of submodular functions, and they are precisely the valuation functions with Walrasian equilibria in markets with indivisible goods [8].

Before giving the formal definition, we first introduce some notation. Given a vector of prices $\{p_g\}_{g \in G}$, the (quasi-linear) *utility* that player i has for a bundle of goods S_i is defined to be $u_i(S_i, p) = v_i(S_i) - \sum_{j \in S_i} p_j$.[3] Given a vector of prices p, for each agent i, we can define his set of *most demanded bundles*: $\omega(p) = \arg\max_{S \subseteq G} u_i(S, p)$. Given two price vectors p, p', we write $p \preceq p'$ if $p_g \leq p'_g$ for all g.

Definition 1. *A valuation function* $v_i : 2^G \to [0,1]$ *satisfies the* gross substitutes condition *if for every pair of price vectors* $p \preceq p'$*, and for every set of goods* $S \in \omega(p)$*, if* $S' \subseteq S$ *satisfies* $p'_g = p_g$ *for every* $g \in S'$*, then there is a set* $S^* \in \omega(p')$ *with* $S' \subseteq S^*$.

Finally, we will always consider markets with multiple copies of each type of good. Two goods $g_1, g_2 \in G$ are *identical* if for every bidder i and for every bundle $S \subseteq G$, $v_i(S \cup \{g_1\}) = v_i(S \cup \{g_2\})$: i.e., the two goods are indistinguishable according to every valuation function. Formally, we say that a set of goods G consists of k *types* of goods with s *supply* if there are k representative goods $g_1, \ldots, g_k \in G$ such that every good $g' \in G$ is identical to one of $g_1, \ldots, g_k$, and for each representative good g_i, there are s goods identical to g_i in G. For simplicity of presentation we assume throughout the paper that the supply of each good is the same, but this is not necessary; all of our results continue to hold when the supply s denotes the *minimum* supply of any type of good.

2.2 Differential Privacy Preliminaries

Although it is impossible to solve the allocation problem under standard differential privacy (see Section 5), standard differential privacy plays an essential role in our analysis; let us begin here.

Suppose agents have valuation functions v_i from a class of functions C. A database $D \in C^n$ is a vector of valuation functions, one for each of the n bidders. Two databases D, D' are *i-neighbors* if they differ in only their i'th index: that is, if $D_j = D'_j$ for all $j \neq i$.

[3] This is a natural definition of utility if agents must pay for the bundles they buy at the given prices. In this paper we are concerned with the purely algorithmic allocation problem, so our algorithm will not actually charge prices. However, prices will be a convenient abstraction throughout our work.

If two databases D, D' are i-neighbors for some i, we say that they are *neighboring databases*. We will be interested in randomized algorithms that take a database as input, and output an element from some range $\mathcal{R}$. Our final mechanisms will output sets of n bundles (so $\mathcal{R} = (2^G)^n$), but intermediate components of our algorithms will have different ranges.

Definition 2 (Dwork et al. [4])**.** *An algorithm $\mathcal{M} : C^n \to \mathcal{R}$ is* (ε, δ)-differentially private *if for every pair of neighboring databases $D, D' \in C^n$ and for every set of subset of outputs $S \subseteq \mathcal{R}$,*

$$\Pr[\mathcal{M}(D) \in S] \leq \exp(\varepsilon) \Pr[\mathcal{M}(D') \in S] + \delta.$$

If $\delta = 0$, we say that $\mathcal{M}$ is ε-differentially private.

When the range of a mechanism is also a vector with n components (e.g., $\mathcal{R} = (2^G)^n$), we can define *joint differential privacy*: this requires that simultaneously for all i, the *joint* distribution on outputs given to players $j \neq i$ is differentially private in the input of agent i. Given a vector $x = (x_1, \ldots, x_n)$, we write $x_{-i} = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$ to denote the vector of length $n-1$ which contains all coordinates of x except the i'th coordinate.

Definition 3 (Kearns et al. [12])**.** *An algorithm $\mathcal{M} : C^n \to (2^G)^n$ is* (ε, δ)-joint differentially private *if for every i, for every pair of i-neighbors $D, D' \in C^n$, and for every subset of outputs $S \subseteq (2^G)^{n-1}$,*

$$\Pr[\mathcal{M}(D)_{-i} \in S] \leq \exp(\varepsilon) \Pr[\mathcal{M}(D')_{-i} \in S] + \delta.$$

If $\delta = 0$, we say that $\mathcal{M}$ is ε-joint differentially private.

Note that this is still an extremely strong definition that protects i from arbitrary coalitions of adversaries—it weakens the constraint of differential privacy only in that the output given specifically to agent i is allowed to be sensitive in the input of agent i.

2.3 Differentially Private Counters

The central tool in our matching algorithm is the private streaming counter proposed by Chan et al. [2] and Dwork et al. [5]. Given a bit stream $\sigma = (\sigma_1, \ldots, \sigma_T) \in \{0,1\}^T$, a streaming counter $\mathcal{M}(\sigma)$ releases an approximation to $c_\sigma(t) = \sum_{i=1}^{t} \sigma_i$ at every time step t. We can define what it means for a streaming counter to be accurate.

Definition 4. *A streaming counter $\mathcal{M}$ is* (α, β)-useful *if with probability at least $1 - \beta$, for each time $t \in [T]$,*

$$|\mathcal{M}(\sigma)(t) - c_\sigma(t)| \leq \alpha.$$

For the rest of this paper, let $\mathbf{Counter}(\varepsilon, T)$ denote the Binary Mechanism of Chan et al. [2], instantiated with parameters ε and T. The mechanism produces a monotonically increasing count, and satisfies the following accuracy guarantee. (Further details may be found in the full version.)

Theorem 1 (Chan et al. [2])**.** *For $\beta > 0$, $\mathbf{Counter}(\varepsilon, T)$ is ε-differentially private with respect to a single bit change in the stream, and (α, β)-useful for*

$$\alpha = \frac{2\sqrt{2}}{\varepsilon} \ln\left(\frac{2}{\beta}\right) \left(\sqrt{\log(T)}\right)^5.$$

3. PRIVATE MAX-WEIGHT MATCHING

In this section, we study the special case of unit demand valuations. Though our later algorithm for gross substitutes valuations generalizes this case, we first present our algorithm in this simpler setting to highlight the key features of our approach.

Consider a matching market with n bidders and k different types of goods, where each good has supply s and bidder i has valuation $v_{ij} \in [0,1]$ for good j. Some agents may not end up being matched to a good: to simplify notation, we will say that unmatched agents are matched to $\perp$, a special dummy good.

To reach a maximum weight matching, we first formulate an intermediate goal: we want to privately compute prices $p \in [0,1]^k$ and an allocation of the goods $\mu\colon [n] \to [k] \cup \{\perp\}$ such that *most* bidders are matched with their *approximately* favorite goods *given the prices* and each over-demanded good almost clears, where a good is *over-demanded* if its price is strictly positive.[4] We will show that if this intermediate goal is met, then in fact we have computed an approximate maximum weight matching.

Definition 5. *A price vector $p \in [0,1]^k$ and an assignment $\mu\colon [n] \to [k] \cup \{\perp\}$ of bidders to goods is an* (α, β, ρ)-approximate matching equilibrium *if*

1. *All but a ρ fraction of bidders i are matched to an α-approximate favorite good: i.e., $v_{i\mu(i)} - p_{\mu(i)} \geq v_{ij} - p_j - \alpha$ for every good j, for at least $(1-\rho)n$ bidders i (we call these bidders* satisfied*);*
2. *the number of bidders assigned to any type of good does not exceed its supply; and*
3. *each over-demanded good clears except for at most β supply.*

3.1 Overview of the Algorithm

Our algorithm takes in the valuations as input, and outputs a trajectory of prices that can be used by the agents to figure out what they are matched to. Throughout, we will sometimes talk of the bidders performing some action, but this actually means that our algorithm simulates the actions of the bidders internally—the actual agents do not interact with our algorithm.

Algorithm 1 (**PMatch**) is a variant of a *deferred acceptance* algorithm first proposed and analyzed by Kelso and Crawford [13], which runs k simultaneous ascending price auctions: one for each type of good. At any given moment, each type of good has a *proposal price* p_j. In rounds (passing through each bidder once in some fixed, publicly known order), unsatisfied bidders bid on a good that maximizes their utility at the current prices: that is, a good j that maximizes $v_{ij} - p_j$. (This is the $\mathbf{Propose}$ function.)

The s most recent bidders for a type of good are tentatively matched to that type of good (these are the current *high bidders*). A bidder tentatively matched to a good with supply s becomes unmatched to that good once the good he is matched to receives s subsequent bids (he has been *outbid*). Every s bids on a good increases its price by a fixed increment α. Bidders keep track of which good they are matched to (in the variable μ), if any, and can determine whether they are currently matched or unmatched by looking at a count of the number of bids received by the last good they bid on.

To implement this algorithm privately, we count the number of bids each good has received using private counters. Unsatisfied bidders can infer the prices of all goods based on the number of bids each has received, and from this information, they determine which good to bid on (their favorite good at the given prices). Their bid is recorded by sending a "1" to the appropriate counter. (This is the $\mathbf{Bid}$ function.) Matched bidders remember the reading of the bid counter on the good they are matched to at the time that they last

[4] This is the notion of approximate Walrasian equilibrium we will use.

bid (in the variable d_i); when the counter ticks s bids past this initial count, the bidder concludes that he has been outbid, and becomes unmatched. The final matching is communicated implicitly: the real agents observe the full published price trajectory, and simulate what good they would have been matched to had they bid according to the published prices.

Since the private counters are noisy, the more than s bidders may be matched to a good. To maintain feasibility, the auction is run with some supply m withheld: i.e., it is run as if the supply of each good were $s - m$, rather than s. The *reserved supply* m is used to satisfy the demand of all bidders who believe themselves to be matched to each type of good; the number of such bidders is at most s, with high probability.

Our algorithm stops as soon as fewer than ρn bidders place bids in a round. We show that this early stopping condition does not significantly harm the welfare guarantee of the matching, while it substantially reduces the *sensitivity* of the counters: no bidder ever bids more than $O(1/(\alpha\rho))$ times in total. Crucially, this is independent of both the number of types of goods k, and the number of bidders n. This greatly improves the accuracy of the prices: the degree to which we have to perturb the bid counts to protect privacy is proportional to the sensitivity of the counters.

To privately implement this stopping condition, we maintain a separate counter (counter(0)) which counts the number of unsatisfied bidders throughout the run of the algorithm. At the end of each proposal round, bidders who are unsatisfied will send "1" to this counter, and bidders who are matched will send "0". If this counter increases by less than roughly ρn in any round, we conclude the algorithm. (This is the $\mathbf{CountUnsatisfied}$ function.)

3.2 Privacy Analysis

In this section, we show that the allocation (implicitly) output by our algorithm satisfies joint differential privacy with respect to a single bidder changing *all* of her valuations. We first show a basic but useful lemma: to show joint differential privacy, it is sufficient to show that the output sent to each agent i is an arbitrary function only of some global signal that is computed under the standard constraint of differential privacy, together with agent i's private data. We call this the *billboard model*: some message is viewable by all agents, as if placed on a public billboard, and this message is differentially private. In our case, the price history over the course of the auction is the differentially private message posted on the billboard. Combined with their personal private valuation, each agent can compute their personal allocation.

Lemma 1 (Billboard Lemma)**.** *Suppose $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ is (ε, δ)-differentially private. Consider any set of functions $f_i : \mathcal{D}_i \times \mathcal{R} \to \mathcal{R}'$, where $\mathcal{D}_i$ is the portion of the database containing i's data. The composition $\{f_i(\Pi_i D, \mathcal{M}(D))\}$ is (ε, δ)-joint differentially private, where Π_i is the projection to i's data.*

PROOF. We need to show that for any agent i, the view of the other agents is (ε, δ)-differentially private when i's private data is changed. Suppose databases D, D' are i-neighbors, so $\Pi_j D = \Pi_j D'$ for $j \neq i$. Let $\mathcal{R}_{-i}$ be a set of views of the bidders besides i. Let $\mathcal{R}^* = \{r \in \mathcal{R} \mid \{f_j(\Pi_j D, r)\}_{-i} \in \mathcal{R}_{-i}\}$. Then, we need

$$\begin{aligned} &\Pr[\{f_j(\Pi_j D, \mathcal{M}(D))\}_{-i} \in \mathcal{R}_{-i}] \\ &\leq e^{\varepsilon} \Pr[\{f_j(\Pi_j D', \mathcal{M}(D'))\}_{-i} \in \mathcal{R}_{-i}] + \delta \\ &= e^{\varepsilon} \Pr[\{f_j(\Pi_j D, \mathcal{M}(D'))\}_{-i} \in \mathcal{R}_{-i}] + \delta \\ \text{and so } &\Pr[\mathcal{M}(D) \in \mathcal{R}^*] \leq e^{\varepsilon} \Pr[\mathcal{M}(D') \in \mathcal{R}^*] + \delta, \end{aligned}$$

but this is true since $\mathcal{M}$ is (ε, δ)-differentially private. □

Algorithm 1 $\mathbf{PMatch}(\alpha, \rho, \varepsilon)$

Input: Bidders' valuations $(\{v_{1j}\}_{j=1}^m, \ldots, \{v_{nj}\}_{j=1}^m)$
Initialize: for bidder i and good j,

$$T = \frac{8}{\alpha\rho}, \qquad \varepsilon' = \frac{\varepsilon}{2T},$$

$$E = \frac{2\sqrt{2}}{\varepsilon'}(\log nT)^{5/2} \log\left(\frac{4k}{\gamma}\right), \qquad m = 2E + 1$$

$$\text{counter}(j) = \textbf{Counter}(\varepsilon', nT) \qquad p_j = c_j = 0,$$

$$\mu(i) = \emptyset, \qquad d_i = 0, \qquad \text{counter}(0) = \textbf{Counter}(\varepsilon', nT)$$

$\mathbf{Propose}$ T times; **Output:** prices p and allocation μ.

Propose:
for all bidders i **do**
  if $\mu(i) = \emptyset$ **then**
    Let $\mu(i) \in \text{argmax}_j\, v_{ij} - p_j$, breaking ties arbitrarily
    if $v_{i\mu(i)} - p_{\mu(i)} \leq 0$ **then**
      Let $\mu(i) :=\perp$ and **Bid**$(\mathbf{0})$.
    else Save $d_i := c_{\mu(i)}$ and **Bid**$(\mathbf{e}_{\mu(i)})$.
  else Bid$(\mathbf{0})$
CountUnsatisfied
  Bid: On input bid vector $\mathbf{b}$
  for all goods j **do**
    Feed $\mathbf{b}_j$ to counter(j).
    Update count $c_j :=$ counter(j).
    if $c_j \geq (p_j/\alpha + 1)(s - m)$ **then**
      Update $p_j := p_j + \alpha$.

  CountUnsatisfied:
  for all bidders i **do**
    if $\mu(i) \neq \perp$ and $c_{\mu(i)} - d_i \geq s - m$ **then**
      Feed 1 to counter(0); Let $\mu(i) := \emptyset$
    else Feed 0 to counter(0).
  if counter(0) increases by less than $\rho n - 2E$ **then**
    Halt; For each i with $\mu(i) = \emptyset$, let $\mu(i) =\perp$

With this lemma, the privacy proof is largely routine. We defer the details to the full version.

Theorem 2. $\mathbf{PMatch}(\alpha, \rho, \varepsilon)$ *is ε-joint differentially private.*

PROOF SKETCH. Note that given the sequence of prices, counts of unsatisfied bidders, and the private valuation of any bidder i, the final allocation to that bidder can be computed by simulating the sequence of bids that bidder i would make: these are determined by the price when bidder i is slotted to bid, and by whether the halting condition has been met. Bidder i's final allocation is simply the final item that he bids on. The prices and halting condition are computed as a deterministic function of the noisy counts, which are ε-differentially private. So, Lemma 1 shows that **PMatch** is ε-joint differentially private. □

3.3 Utility Analysis

In this section, we compare the weight of the matching produced by **PMatch** with OPT. As an intermediate step, we first show that the resulting matching *paired with the prices* output by the algorithm forms an approximate matching equilibrium. We next show that any such matching must be an approximately max-weight matching.

The so-called "first welfare theorem" from general equilibrium theory guarantees that an exact (i.e., a $(0,0,0)$-) matching equilibrium gives an exact maximum weight matching. Compared to this ideal, **PMatch** loses welfare in three ways. First, a ρ fraction of bidders may end up unsatisfied. Second, the matched bidders are not necessarily matched to goods that maximize their utility given the prices, but only to goods that do so approximately (up to additive α). Finally, the auction sets aside part of the supply to handle over-allocation from the noisy counters, which may not end up being sold (say, if the counters are accurate or actually under-allocate). That is, we compute an equilibrium of a market with reduced supply, so our welfare guarantee requires that the supply s be significantly larger than the necessary reserved supply m.

The key performance metric is *how much* supply is needed to achieve a given welfare approximation in the final matching. On the one hand, we will show later that the problem is impossible to solve privately if $s = O(1)$ (Section 5). On the other hand, the problem is trivial if $s \geq n$: every agent can be simultaneously matched to her favorite good with no coordination; this is trivially both optimal and private. Our algorithm will achieve positive results when $s \geq \text{polylog}(n)$.

Theorem 3. *Let $\alpha > 0$, and μ be the matching computed by $\mathbf{PMatch}(\alpha/3, \alpha/3, \varepsilon)$. Let* OPT *denote the weight of the optimal (max weight) matching. Then, if the supply satisfies*

$$s \geq \frac{16E' + 4}{\alpha} = O\left(\frac{1}{\alpha^3\varepsilon} \cdot \text{polylog}\left(n, k, \frac{1}{\alpha}, \frac{1}{\gamma}\right)\right),$$

and $n > s$, the matching μ has social welfare at least $\text{OPT} - \alpha n$ *with probability $\geq 1 - \gamma$, where*

$$E' = \frac{288\sqrt{2}}{\alpha^2\varepsilon}\left(\log\left(\frac{72n}{\alpha^2}\right)\right)^{5/2} \log\left(\frac{4k}{\gamma}\right).$$

Remark 1. *Our approximation guarantee here is* additive*. In Section 4, we show that if we are in the* unweighted *case where $v_{ij} \in \{0, 1\}$, the above guarantee can be made* multiplicative*, unusual in the context of differential privacy. That is, we can find a matching μ with welfare at least $(1 - \alpha)$*OPT*. Also, the second assumption $n > s$ is minimal, as the problem is trivially solvable for $s \geq n$.*

The proof follows from the following lemmas. (We defer some proofs to the full version.)

Lemma 2. *We call a bidder who wants to continue bidding* unsatisfied*; otherwise bidder i is* satisfied*. At termination of $\mathbf{PMatch}(\alpha, \rho, \varepsilon)$, all satisfied bidders i are matched to a good $\mu(i)$ such that*

$$v_{i,\mu(i)} - p_{\mu(i)} \geq \max_j (v_{i,j} - p_j) - \alpha.$$

Lemma 3. *Assume all counters have error at most E throughout the run of $\mathbf{PMatch}(\alpha, \rho, \varepsilon)$. Then the number of bidders assigned to any good is at most s, and each over-demanded good clears except for at most β supply, where*

$$\beta = 4E + 1 = O\left(\frac{1}{\alpha\rho\varepsilon} \cdot \text{polylog}\left(\frac{1}{\alpha}, \frac{1}{\rho}, \frac{1}{\gamma}, k, n\right)\right).$$

PROOF. Since the counter for each under-demanded good never exceeds $s-m$, we know that each under-demanded good is matched to no more than $s - m + E < s$ bidders.

Consider any counter c for an over-demanded good. Let t be a time step in counter c such that $c(nT) - c(t+1) \leq s - m < c(nT) - c(t)$. Note that the bidders who bid after time t are the only bidders matched to this good at time nT. Let σ be the true bid stream for this good, so the total number of bidders allocated to this good at time nT is

$$\begin{aligned} c_\sigma(nT) - c_\sigma(t) &\leq c_\sigma(nT) - c_\sigma(t+1) + 1 \\ &\leq (c(nT) + E) - (c(t+1) - E) + 1 \\ &\leq s - m + 2E + 1 = s. \end{aligned}$$

Similarly, we can lower bound the number of bidders allocated to this good:

$$\begin{aligned} &c_\sigma(nT) - c_\sigma(t) \\ &= (c_\sigma(nT) - c(nT)) + (c(nT) - c(t)) + (c(t) - c_\sigma(t)) \\ &> s - m - 2E > s - 4E - 1. \end{aligned}$$

Therefore, every over-demanded good clears except for at most $\beta = 4E + 1$ supply, which gives the dependence

$$\begin{aligned} \beta &= \frac{16\sqrt{2}}{\alpha\rho\varepsilon}\left(\log\left(\frac{6n}{\alpha\rho}\right)\right)^{5/2} \log\left(\frac{4k}{\gamma}\right) + 1 \\ &= O\left(\frac{1}{\alpha\rho\varepsilon} \cdot \text{polylog}\left(\frac{1}{\alpha}, \frac{1}{\rho}, \frac{1}{\gamma}, k, n\right)\right). \end{aligned}$$

□

Lemma 4. *Assume all counters have error at most E throughout the run of $\mathbf{PMatch}(\alpha, \rho, \varepsilon)$. Then at termination, all but a ρ fraction of bidders are satisfied, so long as $s \geq 8E+1$ and $n \geq 8E/\rho$.*

PROOF. First, we claim that the total number of bids made over the course of the algorithm is bounded by $3n/\alpha$. We account separately for the under-demanded goods (those with price 0 at the end of the auction) and the over-demanded goods (those with positive price). For the under-demanded goods, since their prices remain 0 throughout the algorithm, their corresponding noisy counters never exceeded $(s - m)$. Since no bidder is ever unmatched after having been matched to an under-demanded good, the set of under-demanded goods can receive at most one bid from each agent; together the under-demanded goods can receive at most n bids.

Next, we account for the over-demanded goods. Note the bidders matched to these goods are precisely the bidders who bid within $s-m$ ticks of the final counter reading. Since the counter has error bounded by E at each time step, this means at least $s - m - 2E$ bidders end up matched to each over-demanded good. Since no agent can be matched to more than one good there can be at most $n/(s - m - 2E)$ over-demanded goods in total.

Likewise, we can account for the number of price increases per over-demanded good. Prices never rise above 1 (because any bidder would prefer to be unmatched than to be matched to a good with price larger than 1). Therefore, since prices are raised in increments of α, each over-demanded good can have its price incremented at most $1/\alpha$ times. Since there can be at most $(s - m + 2E)$ bids between each price update (again, corresponding to $s - m$ ticks of the counter), the total number of bids received by all of the over-demanded goods in total is at most

$$\frac{n}{s - m - 2E} \cdot \frac{1}{\alpha} \cdot (s - m + 2E).$$

Since each bid is either on an under or over-demanded good, we can upper bound the *total* number of bids B by

$$B \leq n + \frac{n}{\alpha}\left(\frac{s - m + 2E}{s - m - 2E}\right) = \frac{n}{\alpha}\left(\alpha + \frac{s - m + 2E}{s - m - 2E}\right).$$

We set the reserved supply to be $m = 2E + 1$ and by assumption, we have $s \geq 8E + 1$. Since we are only interested in cases where

$\alpha < 1$, we conclude

$$B \le n + \frac{n}{\alpha}\left(\frac{s-m+\alpha_2}{s-m-\alpha_2}\right) \le \frac{3n}{\alpha}. \quad (1)$$

Now, consider the halting condition. There are two cases: either the algorithm halts early, or it does not. We claim that at termination, at most ρn bidders are unsatisfied. The algorithm halts early if at any round of **CountUnsatisfied**, counter(0) (which counts the number of unsatisfied bidders) increases by less than $\rho n - 2E$. So if the algorithm halts early, there must be at most $\rho n - 2E + 2E = \rho n$ unsatisfied bidders.

Otherwise, suppose the algorithm does not halt early. At the start of each round there must be at least $\rho n - 4E$ unsatisfied bidders. Not all of these bidders must bid during the **Propose** round since price increases while they are waiting to bid might cause them to no longer demand any item, but this only happens if bidders prefer to be unmatched at the new prices. Since prices only increase, these bidders remain satisfied for the rest of the algorithm. If the algorithm runs for R rounds and there are B true bids, $B \ge R(\rho n - 4E) - n$. Combined with our upper bound on the number of bids (Equation (1)) and our assumption $\rho n \ge 8E$, we can upper bound the number of rounds R:

$$R \le \left(\frac{3n}{\alpha} + n\right) \cdot \left(\frac{1}{\rho n - 2E}\right) \le \left(\frac{4n}{\alpha}\right)\left(\frac{2}{\rho n}\right) = \frac{8}{\alpha\rho} := T$$

Thus, running the algorithm for T rounds leads to all but ρn bidders satisfied. □

Lemma 5. *With probability at least $1-\gamma$, $\mathbf{PMatch}(\alpha, \rho, \varepsilon)$ computes an (α, β, ρ)-matching equilibrium, where*

$$\beta = 4E + 1 = O\left(\frac{1}{\alpha\rho\varepsilon} \cdot \text{polylog}\left(\frac{1}{\alpha}, \frac{1}{\rho}, \frac{1}{\gamma}, k, n\right)\right)$$

so long as $s \ge 8E + 1$ and $n \ge 8E/\rho$.

With these lemmas im place, it is straightforward to prove the welfare theorem (Theorem 3).

PROOF OF THEOREM 3. By Lemma 5, $\mathbf{PMatch}(\alpha/3, \alpha/3, \varepsilon)$ calculates a matching μ that is an $(\alpha/3, \beta, \alpha/3)$-approximate matching equilibrium with probability at least $1-\gamma$, where $\beta = 4E' + 1$. Let p be the prices at the end of the algorithm, and S be the set of satisfied bidders. Let μ^* be the optimal matching achieving welfare $\sum_{i=1}^{n} v_{i,\mu^*(i)} = \text{OPT}$. We know that $|S| \ge (1 - \alpha/3)n$ and

$$\sum_{i \in S}(v_{i\mu(i)} - p_{\mu(i)}) \ge \sum_{i \in S}(v_{i\mu^*(i)} - p_{\mu^*(i)}) - \alpha|S|/3.$$

Let N_j^* and N_j be the number of goods of type j matched in matchings μ^* and μ respectively, and let G be the set of over-demanded goods at prices p.

Since each over-demanded good clears except for at most β supply, and since each of the n agents can be matched to at most 1 good, we know that $|G| \le n/(s - \beta)$. Since the true supply in OPT is at most s, we also know $N_j^* - N_j \le \beta$ for each over-demanded good j. Finally, by definition, under-demanded goods j have price $p_j = 0$. So,

$$\begin{aligned}\sum_{i \in S} v_{i\mu^*(i)} - \sum_{i \in S} v_{i\mu(i)} &\le \sum_{i \in S} p_{\mu^*(i)} - \sum_{i \in S} p_{\mu(i)} + \alpha|S|/3 \\ &= \sum_{j \in G} p_j(N_j^* - N_j) + \alpha|S|/3 \\ &\le \sum_{j \in G} \beta + \alpha|S|/3 \le \frac{n\beta}{s-\beta} + \alpha|S|/3.\end{aligned}$$

If $s \ge 4\beta/\alpha$, the first term is at most $\alpha n/3$. Finally, since all but $\alpha n/3$ of the bidders are matched with goods in S, and their valuations are upper bounded by 1, we can conclude:

$$\sum_i v_{i\mu(i)} - \sum_i v_{i\mu^*(i)} \le \alpha n/3 + \alpha|S|/3 + \alpha n/3 \le \alpha n.$$

Unpacking β from Lemma 5, we get the stated bound on supply. □

4. EXTENSIONS

In this section, we extend our algorithm in two ways. First, we show how to compute approximately max-welfare allocations under general gross substitutes valuations. We also show how to modify and analyze the algorithm for computing max-weight matchings in the *unweighted* case when $v_{ij} \in \{0, 1\}$ to get *multiplicative* rather than additive approximation, which can be substantially better in the case when OPT is small. (More generally, the approximation depends on the minimum nonzero valuation.)

4.1 Gross Substitute Valuations

Let us first introduce some notation. Let $\Omega = 2^G$ denote the space of bundles (i.e., subsets of goods). Like previous sections, let k be number of types of goods, and let s be the supply of each type of good. Let d denote the *market size*—the total number of goods, including identical goods, so $d = ks$. (We remark that we assume each good has the same supply s only for convenience. In general, goods may have different supplies, if s denotes the *minimum* supply of any good. Hence, d is not necessarily dependent on s.) We assume each bidder has a valuation function on bundles, $v_i : \Omega \to [0, 1]$, and that this valuation satisfies the gross substitutes condition (Definition 1).

Like before, we simulate k ascending price auctions in rounds. Bidders now maintain a bundle of goods that they are currently allocated to, and bid on one new good each round. For each good in a bidder's bundle, the bidder keeps track of the count of bids on that good when it was added to the bundle. When the current count ticks past the supply, the bidder knows that he has been outbid for that good.

The main subtlety is in how bidders decide which goods to bid on. Namely, each bidder considers goods in his bundle to be fixed in price (i.e., bidders ignore the price increment of at most α that might have occurred after winning the item). Goods outside of his bundle (even if identical to goods in his bundle) are evaluated at the true price. We call these prices the bidder's *effective* prices, so each bidder bids on an arbitrary good in his most-preferred bundle at the effective prices. The full algorithm is given in Algorithm 2.

Privacy is very similar to the case for matchings.

Theorem 4. $\mathbf{PAlloc}(\alpha, \rho, \varepsilon)$ *satisfies ε-joint differential privacy.*

Theorem 5. *Let $0 < \alpha < n/d$, and g be the allocation computed by $\mathbf{PAlloc}(\alpha/3, \alpha/3, \varepsilon)$, and let* OPT *be the optimum max welfare. Then, if $d \ge n$ and*

$$s \ge \frac{12E' + 3}{\alpha} = O\left(\frac{1}{\alpha^3\varepsilon} \cdot \text{polylog}\left(n, k, \frac{1}{\alpha}, \frac{1}{\gamma}\right)\right),$$

the allocation g has social welfare at least

$$\sum_{i=1}^{n} v_i(g(i)) \ge \text{OPT} - \alpha d,$$

with probability at least $1 - \gamma$, where

$$E' = \frac{360\sqrt{2}}{\alpha^2\varepsilon}\left(\log\left(\frac{90n}{\alpha^2}\right)\right)^{5/2} \log\left(\frac{4k}{\gamma}\right) + 1.$$

Algorithm 2 $\mathbf{PAlloc}(\alpha, \rho, \varepsilon)$ (with Gross Substitute Valuations)

Input: Bidders' gross substitute valuations on the bundles $\{v_i : \Omega \to [0,1]\}$

Initialize: for bidder i and good j,

$$T = \frac{10}{\alpha\rho}, \qquad \varepsilon' = \frac{\varepsilon}{2T},$$

$$E = \frac{2\sqrt{2}}{\varepsilon'}(\log nT)^{5/2}\log\left(\frac{4k}{\gamma}\right) + 1, \qquad m = 2E + 1,$$

$$\text{counter}(0) = \mathbf{Counter}(\varepsilon', nT),$$

$$\text{counter}(j) = \mathbf{Counter}(\varepsilon', nT), \qquad p_j = c_j = 0,$$

$$d_g = 0, \qquad g(i) = \{\emptyset\} \qquad \text{for every bidder } i$$

Propose T times; **Output:** prices p and allocation g.

Propose:
for all bidders i **do**
  for all goods $g \in g(i)$ **do**
    if $c_{type(g)} - d_g \geq s - m$ **then**
      Remove $g(i) := g(i) \setminus g$
  Let p_0 be the original cost of $g(i)$.
  Let $\omega^* \in \underset{\omega \supseteq g(i)}{\text{argmax}}\, v_i(\omega) - p(\omega \setminus g(i)) - p_0$ arbitrary.
  if $v_i(\omega^*) - p(\omega \setminus g(i)) - p_0 \geq v_i(g(i)) - p_0$ **then**
    Let $j \in \omega^* \setminus g(i)$ arbitrary.
    Save $d_j := c_{type(j)}$
    Add $g(i) := g(i) \cup j$ and **Bid**$(\mathbf{e_j})$
  else Bid$(\mathbf{0})$
CountUnsatisfied
  Bid: On input bid vector $\mathbf{b}$
  for all goods j **do**
    Feed $\mathbf{b}_j$ to counter(j).
    Update count $c_j :=$ counter(j).
    if c_j is a multiple of $s - m$ **then**
      Update $p_j := p_j + \alpha$.

  CountUnsatisfied:
  for all bidders i **do**
    if i wants continue bidding **then**
      Feed 1 to counter(0)
    else Feed 0 to counter(0)
  Halt if counter(0) increases by less than $\rho d - 2E$

Remark 2. *In comparison with Theorem 3, Theorem 5 requires a similar constraint on supply, but promises welfare only* $\text{OPT} - \alpha d$ *rather than* $\text{OPT} - \alpha n$. *Since* $\text{OPT} \leq n$, *this guarantee is only non-trivial for* $\alpha \leq n/d$, *and so the supply has a polynomial dependence on the total size of the market, d. In contrast, Theorem 3 guarantees good welfare when the supply has a logarithmic dependence on the total number of goods in the market.*

However, we note that if bidders demand bundles of size at most b, then we can improve the above welfare bound to $\text{OPT} - \alpha n b$. *Note that this is independent of the market size d, and strictly generalizes the matching case (where* $b = 1$*).*

Similar to Definition 5, we define an *approximate allocation equilibrium* as a prerequisite for showing our welfare guarantee.

Definition 6. *A price vector* $p \in [0,1]^k$ *and an assignment* $g\colon [n] \to \Omega$ *of bidders to goods is an* (α, β, ρ)-approximate allocation equilibrium *if*

1. *for all but* ρd *bidders,* $v_i(g(i)) - p(g(i)) \geq \max_{\omega \in \Omega} v_i(\omega) - p(\omega) - \alpha|g(i)|$;
2. *the number of bidders assigned to any good is at most s; and*
3. *each overdemanded good clears except for at most* β *supply.*

The following lemmas show that our algorithm finds an approximate allocation equilibrium. (We defer proofs to the full version.)

Lemma 6. *Assume all counters have error at most E throughout the run of* $\mathbf{PAlloc}(\alpha, \rho, \varepsilon)$. *Then, the number of bidders assigned to any good is at most s, and each overdemanded good clears except for at most* β *supply, where*

$$\beta = 4E + 1 = O\left(\frac{1}{\alpha\rho\varepsilon} \cdot \text{polylog}\left(n, k, \frac{1}{\alpha}, \frac{1}{\rho}, \frac{1}{\gamma}\right)\right).$$

Lemma 7. *We call a bidder who wants to bid more* unsatisfied*; otherwise, a bidder is* satisfied. *At termination of* $\mathbf{PAlloc}(\alpha, \rho, \varepsilon)$, *all satisfied bidders are matched to a bundle* $g(i)$ *that is an* $\alpha \cdot |g(i)|$*-most preferred bundle.*

Lemma 8. *Suppose all counters have error at most E throughout the run of* $\mathbf{PAlloc}(\alpha, \rho, \varepsilon)$. *Then at termination, all but* ρd *bidders are satisfied, so long as*

$$n \leq d \quad \text{and} \quad d \geq \frac{8E}{\rho} = \Omega\left(\frac{1}{\alpha\rho^2\varepsilon} \cdot \text{polylog}\left(n, k, \frac{1}{\alpha}, \frac{1}{\rho}, \frac{1}{\gamma}\right)\right).$$

Lemma 9. *With probability at least* $1 - \gamma$, $\mathbf{PAlloc}(\alpha, \rho, \varepsilon)$ *computes an* (α, β, ρ)*-approximate allocation equilibrium, where*

$$\beta = O\left(\frac{1}{\alpha\rho\varepsilon} \cdot \text{polylog}\left(n, k, \frac{1}{\alpha}, \frac{1}{\rho}, \frac{1}{\gamma}\right)\right),$$

so long as

$$d \geq \frac{8E}{\rho} = \Omega\left(\frac{1}{\alpha\rho^2\varepsilon} \cdot \text{polylog}\left(n, k, \frac{1}{\alpha}, \frac{1}{\rho}, \frac{1}{\gamma}\right)\right) \text{ and } n \leq d.$$

Now, it is straightforward to prove the welfare theorem (Theorem 5). The proof follows Theorem 3 quite closely; we defer the proof to the full version.

4.2 Multiplicative Approximation to Welfare

In certain situations, a close variant of **PMatch** (Algorithm 1) can give a multiplicative welfare guarantee. In this section, we will work with matchings and we will assume that the value of the maximum weight matching OPT is known. (It is possible to privately estimate this quantity to high accuracy.) Our algorithm is exactly the same as **PMatch**, except with a different halting condition: rather than count the number of unmatched bidders each round, count the number of bids per round. Once this count drops below a certain threshold, halt the algorithm.

More precisely, we use a function $\mathbf{CountBids}$ (Algorithm 3) in place of $\mathbf{CountUnsatisfied}$ in Algorithm 1.

Theorem 6. *Suppose bidders have valuations* $\{v_{ij}\}$ *over goods such that* $\min_{v_{ij} > 0} v_{ij} \geq \lambda$. *Then Algorithm 1, with* $T = 24/\alpha^2$ *rounds, using stopping condition* $\mathbf{CountBids}$ *(Algorithm 3) in place of* $\mathbf{CountUnsatisfied}$, *and stopped once the total bid counter increases by less than* $\alpha\,\text{OPT}/2\lambda - 2E$ *bids in a round, satisfies*

Algorithm 3 Modified Halting Condition **CountBids**

CountBids:
for all bidders i **do**
  if $\mu(i) \neq \perp$ and $c_{\mu(i)} - d_i \geq s - m$ **then**
    Let $\mu(i) := \emptyset$
  if i bid this round **then**
    Feed 1 to counter(0).
  else Feed 0 to counter(0).
if counter(0) increases by less than $\frac{\alpha OPT}{2\lambda} - 2E$ **then**
  Halt; For each i with $\mu(i) = \emptyset$, let $\mu(i) = \perp$

ε-joint differential privacy and outputs a matching that has welfare at least $O((1-\alpha/\lambda)\,\mathrm{OPT})$, so long as

$$s = \Omega\left(\frac{1}{\alpha^3\varepsilon} \cdot \mathrm{polylog}\left(n, k, \frac{1}{\alpha}, \frac{1}{\gamma}\right)\right)$$

and $$\mathrm{OPT} = \Omega\left(\frac{\lambda}{\alpha^3\varepsilon} \cdot \mathrm{polylog}\left(n, k, \frac{1}{\alpha}, \frac{1}{\gamma}\right)\right).$$

Privacy follows like Theorem 2. Utility follows a similar analysis as for the matching case, with one main twist: in the unweighted case, there can be at most OPT/λ bidders matched to a prefered good, since each matched bidder contributes weight λ. Thus, we can halt the algorithm sooner when OPT is small. Details can be found in the full version.

Remark 3. *For a comparison with Theorem 3 and* **PMatch***, consider the "unweighted" case where bidders have valuations in $\{0,1\}$ (i.e., $\lambda = 1$). Note that both* **PMatch** *and the multiplicative version require the same lower bound on supply. Ignoring log factors,* **PMatch** *requires $n = \widetilde{\Omega}(1/\alpha^3\varepsilon)$ for an additive αn approximation, while Theorem 6 shows $\mathrm{OPT} = \widetilde{\Omega}(1/\alpha^3\varepsilon)$ is necessary for a multiplicative α, hence additive $\alpha\,\mathrm{OPT}$, approximation. Hence, Theorem 6 gives a stronger guarantee if $\mathrm{OPT} = \widetilde{o}(n)$ in the unweighted case, ignoring log factors.*

5. LOWER BOUNDS

Our lower bounds all reduce to a basic database reconstruction lower bound for differential privacy.

Theorem 7. *Let mechanism $\mathcal{M}\colon \{0,1\}^n \to \{0,1\}^n$ be (ε,δ)-differentially private, and suppose that for all database D, with probability at least $1-\beta$, $\|\mathcal{M}(D) - D\|_1 \leq \alpha n$. Then,*

$$\alpha \geq 1 - \frac{e^\varepsilon + \delta}{(1+e^\varepsilon)(1-\beta)} := c(\varepsilon,\delta,\beta).$$

In other words, no (ε,δ)-private mechanism can reconstruct more than a fixed constant fraction of its input database. For ε,δ,β small, $c(\varepsilon,\delta,\beta) \sim 1/2$. Informally, this theorem states that a private reconstruction mechanism can't do much better than guessing a random database. Note that this holds even if the adversary doesn't know which fraction was correctly reconstructed. This theorem is folklore; a proof can be found in the full version.

Our lower bounds will all be proved using the following pattern:

- First, we describe how to convert a database $D \in \{0,1\}^n$ to a market, by specifying the bidders, the goods, and the valuations $v_{ij} \in [0,1]$ on goods.
- Next, we analyze how these valuations change when a single bit in D is changed. This will control how private the matching algorithm is with respect to the original database, when applied to this market.
- Finally, we show how to output a database guess $\widehat{D}$ from the matching produced by the private matching algorithm.

This composition of three steps will be a private function from $\{0,1\}^n \to \{0,1\}^n$, so we can apply Theorem 7 to lower bound the error. This will in turn imply a lower bound on the error of the matching algorithm.

5.1 Standard Differential Privacy

Note that Algorithm 1 produces market clearing prices under standard differential privacy. We will first show that this is not possible if each good has unit supply. Recall that prices correspond to an (α,β,ρ)-*approximate matching equilibrium* if all but ρ bidders can be allocated to a good such that their utility (valuation less price) is within α of their favorite good (Definition 5). We will ignore the β parameter, which controls how many goods are left unsold. (We defer the proof to the full version.)

Theorem 8. *Let n bidders have valuations $v_{ij} \in [0,1]$ for n goods. Suppose that mechanism $\mathcal{M}$ is (ε,δ)-differentially private, and calculates prices corresponding to an (α,β,ρ)-approximate matching equilibrium for $\alpha < 1/2$ and some β with probability $1-\gamma$. Then, $\rho \geq \frac{1}{2}c(2\varepsilon, \delta(1+e^\varepsilon), \gamma)$. Note that this is independent of α.*

5.2 Separation Between Standard and Joint Differential Privacy

While we can compute an approximate maximum-weight matching under joint privacy when the supply of each good is large (Lemma 5), this is not possible under standard differential privacy even with infinite supply. (In fact, it is not possible with finite supply either.)

Theorem 9. *Let n bidders have valuations $v_{ij} \in \{0,1\}$ for 2 goods with infinite supply. Suppose that mechanism $\mathcal{M}$ is (ε,δ)-differentially private, and computes a matching with weight at least $\mathrm{OPT} - \alpha n$ with probability $1-\gamma$. Then, $\alpha \geq c(\varepsilon,\delta,\gamma)$.*

PROOF. Let $D \in \{0,1\}^n$. We assume two goods, $\mathbf{0}$ and $\mathbf{1}$. We have one bidder b_i for each bit $i \in [n]$, who has valuation D_i for $\mathbf{1}$, and valuation $1 - D_i$ for $\mathbf{0}$. Since changing a bit changes a single bidder's valuation, applying $\mathcal{M}$ to this market is (ε,δ)-private with respect to D. To guess the database $\widehat{D}$, we let $\widehat{D}_i$ be 0 if b_i is matched to $\mathbf{0}$, 1 if b_i is matched to $\mathbf{1}$, and arbitrary otherwise.

Note that the maximum welfare matching assigns each b_i the good corresponding to D_i, and achieves social welfare $\mathrm{OPT} = n$. If $\mathcal{M}$ computes an matching with welfare $\mathrm{OPT} - \alpha n$, it must give all but an α fraction of bidders b_i the good corresponding to D_i. So, the reconstructed database will miss at most αn bits with probability $1-\gamma$, and by Theorem 7, $\alpha \geq c(\varepsilon,\delta,\gamma)$. □

Note that this gives a separation: under joint differential privacy, Algorithm 1 can release a matching with welfare $\mathrm{OPT} - \alpha n$ for any α, provided supply s is large enough (by Theorem 3). This is not possible under standard differential privacy, even with *infinite* supply.

5.3 Joint Differential Privacy

Finally, we show that a large supply assumption is necessary in order to compute an additive α maximum welfare matching under joint differential privacy.

Theorem 10. *Let n bidders have valuations $v_{ij} \in [0,1]$ for k types of goods with supply s each. Suppose mechanism $\mathcal{M}$ is (ε,δ)-joint differentially private for $\varepsilon,\delta < 0.1$, and calculates a matching with welfare at least $\mathrm{OPT} - \alpha n$ with probability $1-\gamma$ for $\gamma < 0.01$, and all n,k,s. Then, $s = \Omega(\sqrt{1/\alpha})$.*

We will only sketch the idea here, deferring the full proof to the full version. Given a database $D \in \{0,1\}^n$, we will have one real bidder, m "spy" bidders, and two goods for each bit. The real bidder will have valuation for one of the two goods determined by the private data D, while the spy bidders will all have the same preference for one of the two goods, set uniformly at random (independent of the private data). By arranging the valuations of the spy bidders appropriately, we can show that any algorithm that achieves good welfare must serve many of the spy bidders. When the spy bidder and the true bidder prefer the same good (which happens half of the time), the spy bidders can learn about the true bidder's preferences when they don't get their preferred good. By taking the joint view of spy bidders, we can reconstruct a large enough portion of the database to contradict Theorem 7: Under *joint* differential privacy, the view of the spy bidders should satisfy *standard* differential privacy with respect to the data from outside the coalition, i.e., the private data.

6. CONCLUSION AND OPEN PROBLEMS

In this paper we gave algorithms to accurately solve the private allocation problem when bidders have gross substitute valuations. Our results are qualitatively tight: it is not possible to strengthen our approach to standard differential privacy (from joint differential privacy), nor is it possible to solve even max-matching problems to non-trivial accuracy under joint differential privacy with constant supply. Moreover, our approach cannot be pushed any further: our algorithm fundamentally relies on computing Walrasian equilibrium prices for the underlying market, and such prices are not guaranteed to exist for valuation functions beyond the gross substitutes class. This does not mean that the allocation problem cannot be solved for more general valuation functions, only that fundamentally new ideas would be needed.

Along with Kearns et al. [12] and other works in the joint privacy model, our work adds compelling evidence that substantially more is possible under the relaxation of *joint* differential privacy, as compared to the standard notion of differential privacy. For both the allocation problem studied here and the equilibrium computation problem studied in Kearns et al. [12], non-trivial results are impossible under differential privacy, while strong results can be derived under joint-differential privacy. Characterizing the power of joint differential privacy, as compared to the standard differential privacy, continues to be a fascinating direction for future work.

More specifically, in this paper we achieved joint differential privacy via the *billboard lemma*: we showed that the allocation given to each player can be derived as a deterministic function only of 1) a differentially private message revealed to all players, and 2) their own private data. However, this isn't necessarily the only way to achieve joint differential privacy. How much further does the power of joint differential privacy extend beyond the billboard model?

Acknowledgments

The authors would like to thank Cynthia Dwork, Sudipto Guha, Moritz Hardt, Sanjeev Khanna, Scott Kominers, Mallesh Pai, David Parkes, Adam Smith, and Kunal Talwar for helpful discussions. In particular, we would like to thank Scott Kominers for suggesting the connection to Kelso and Crawford, and Adam Smith for discussions on the "billboard model" of privacy. Finally, we thank the anonymous reviewers.

References

[1] BLUM, A., LIGETT, K., AND ROTH, A. 2013. A learning theory approach to noninteractive database privacy. *Journal of the ACM 60,* 2, 12.

[2] CHAN, T.-H. H., SHI, E., AND SONG, D. 2011. Private and continual release of statistics. *ACM Transactions on Information and System Security 14,* 3, 26.

[3] DINUR, I. AND NISSIM, K. 2003. Revealing information while preserving privacy. In *ACM SIGACT–SIGMOD–SIGART Symposium on Principles of Database Systems (PODS), San Diego, California*. 202–210.

[4] DWORK, C., MCSHERRY, F., NISSIM, K., AND SMITH, A. 2006. Calibrating noise to sensitivity in private data analysis. In *IACR Theory of Cryptography Conference (TCC), New York, New York*.

[5] DWORK, C., NAOR, M., PITASSI, T., AND ROTHBLUM, G. N. 2010. Differential privacy under continual observation. In *ACM SIGACT Symposium on Theory of Computing (STOC), Cambridge, Massachusetts*. 715–724.

[6] DWORK, C., NAOR, M., AND VADHAN, S. 2012. The privacy of the analyst and the power of the state. In *IEEE Symposium on Foundations of Computer Science (FOCS), New Brunswick, New Jersey*. 400–409.

[7] DWORK, C. AND ROTH, A. 2013. The algorithmic foundations of differential privacy.

[8] GUL, F. AND STACCHETTI, E. 1999. Walrasian equilibrium with gross substitutes. *Journal of Economic Theory 87,* 1, 95–124.

[9] GUPTA, A., LIGETT, K., MCSHERRY, F., ROTH, A., AND TALWAR, K. 2010. Differentially private combinatorial optimization. In *ACM–SIAM Symposium on Discrete Algorithms (SODA), Austin, Texas*. 1106–1125.

[10] HARDT, M. AND ROTHBLUM, G. N. 2010. A multiplicative weights mechanism for privacy-preserving data analysis. In *IEEE Symposium on Foundations of Computer Science (FOCS), Las Vegas, Nevada*. 61–70.

[11] HSU, J., ROTH, A., AND ULLMAN, J. 2013. Differential privacy for the analyst via private equilibrium computation. In *ACM SIGACT Symposium on Theory of Computing (STOC), Palo Alto, California*. 341–350.

[12] KEARNS, M., PAI, M., ROTH, A., AND ULLMAN, J. 2014. Mechanism design in large games: Incentives and privacy. In *ACM SIGACT Innovations in Theoretical Computer Science (ITCS), Princeton, New Jersey*.

[13] KELSO, A. AND CRAWFORD, V. 1982. Job matching, coalition formation, and gross substitutes. *Econometrica 50,* 6, 22.

[14] MCSHERRY, F. AND MIRONOV, I. 2009. Differentially private recommender systems: Building privacy into the netflix prize contenders. In *KDD*. 627–636.

[15] NISSIM, K., RASKHODNIKOVA, S., AND SMITH, A. 2007. Smooth sensitivity and sampling in private data analysis. In *ACM SIGACT Symposium on Theory of Computing (STOC), San Diego, Illinois*. 75–84.

[16] PAI, M. AND ROTH, A. 2013. Privacy and mechanism design. *ACM SIGecom Exchanges*.

[17] ROGERS, R. AND ROTH, A. 2013. Asymptotically truthful equilibrium selection in large congestion games.