

Optimal Auctions via the Multiplicative Weight Method

ANAND BHALGAT, University of Pennsylvania
SREENIVAS GOLLAPUDI, Microsoft Research Search Labs
KAMESH MUNAGALA, Duke University

We show that the multiplicative weight update method provides a simple recipe for designing and analyzing optimal Bayesian Incentive Compatible (BIC) auctions, and reduces the time complexity of the problem to pseudo-polynomial in parameters that depend on single agent instead of depending on the size of the joint type space. We use this framework to design computationally efficient optimal auctions that satisfy ex-post Individual Rationality in the presence of constraints such as (hard, private) budgets and envy-freeness. We also design optimal auctions when buyers and a seller's utility functions are non-linear. Scenarios with such functions include (a) auctions with "quitting rights", (b) cost to borrow money beyond budget, (c) a seller's and buyers' risk aversion. Finally, we show how our framework also yields optimal auctions for variety of auction settings considered in [Daskalakis and Weinberg 2012; Alaei et al. 2012; Cai et al. 2012a; 2012b; 2013], albeit with pseudo-polynomial running times.

Categories and Subject Descriptors: F.2.2 [Theory of Computation]: Nonnumerical Algorithms and Problems

General Terms: Algorithms, Economics, Theory

Additional Key Words and Phrases: Bayesian Mechanism Design, Auctions

1. INTRODUCTION

In this paper, we present a simple framework for designing Bayesian Incentive Compatible (BIC) auctions. This model of auction design was pioneered by Myerson [Myerson 1981]. There is a common prior over the private valuations of all buyers, and the mechanism has to enforce incentive compatibility in expectation over the revealed types of other buyers (where these types are assumed to be drawn from the prior), and the randomness introduced by the mechanism. The optimal BIC auction design problem is purely computational - in most cases, it is possible to design an optimal mechanism in time polynomial in the support of the *joint type space* of the priors of all buyers. As the support can be exponential in the number of buyers, the question arises: *Can we design optimal auctions whose computational complexity is polynomial in the number of buyers, items, and parameters that only depend on individual buyers?* This was first posed as a computational problem in [Conitzer and Sandholm 2004].

Though this question was answered in the affirmative in economic literature [Myerson 1981; Border 2007] for simple settings such as multi-unit auctions, it remained largely open in more general settings. An impressive sequence of recent papers [Hartline and Lucier 2010;

Anand Bhalgat is supported in part by NSF grant IIS-0904314. Kamesh Munagala is supported in part by NSF grants CCF-0745761, CCF-1008065, and IIS-0964560, by an Alfred P. Sloan research fellowship, and an award from Cisco. Part of this work was done while Munagala was visiting Twitter, Inc.

Author's addresses: Anand Bhalgat, Department of Computer and Information Science, University of Pennsylvania; Sreenivas Gollapudi, Microsoft Research Search Labs; Kamesh Munagala, Department of Computer Science, Duke University.

Permission to make digital or hardcopies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credits permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.

EC'13, June 16–20, 2013, Philadelphia, USA. Copyright © 2013 ACM 978-1-4503-1962-1/13/06...\$15.00

Daskalakis and Weinberg 2012; Alaei et al. 2012; Cai et al. 2012a; 2012b; 2013] answer this question in the affirmative under a wide variety of allocation constraints, with both single and multi-dimensional types for each buyer. These algorithms are based on reducing the revenue maximization problem to a convex combination over analogous welfare maximization problems on the same allocation constraints via convex programming techniques, hence reducing an exponential time computation over the joint type space to polynomial time in the type space of a single buyer.

Our Contributions. In this paper, we show that the classical multiplicative weight update method [Young 1995; Plotkin et al. 1995; Arora et al. 2005] yields a simple framework for designing optimal auctions along with an elementary analysis. This framework, presented in Section 5, is based on writing an *explicit program* connecting ex-interim variables to ex-post variables (allocations, payments, utilities, etc). It then reduces the revenue maximization problem to single-bidder optimization problems using the dual (Lagrangian) variables in the explicit program, hence reducing the running time to depend only on single-buyer input parameters. Its key advantages stem from the nature of the problem formulation that *explicitly encodes the optimal mechanism’s action in each realization*:

- It is extremely simple to state and generalize.
- In contrast with previous techniques, we do not have to argue about feasibility of ex-interim solutions or explicitly convert them to ex-post allocations. The technique is “one-shot” yielding the mechanism directly from the dual variables.
- The ellipsoid based techniques developed in the previous work are sensitive to the geometry of the feasible ex-interim allocation region defined by the separation oracle; when this region is not convex, such as when the welfare maximization problem can only be solved approximately [Cai et al. 2013], the optimization becomes complex. In contrast, the multiplicative weight update based techniques do not face any such hurdle (as long as the oracle subproblem can be solved) and yield a relatively simple approach to design mechanisms in such conditions.

Our approach can be used to derive recent results (mentioned above) on optimal auction design with multi-dimensional type spaces and allocation constraints [Daskalakis and Weinberg 2012; Alaei et al. 2012; Cai et al. 2012a; 2012b; 2013] (details deferred to the full version [Bhalgat et al. 2012b]), albeit with running times that are pseudo-polynomial in single-bidder parameters (instead of fully polynomial). This is however *not* the main focus of the paper. We focus on designing optimal mechanisms for several new settings where the only known previous results were approximation algorithms. Though previous techniques can be adapted to solve these variants, we illustrate how our technique provides a direct approach to such auction design problems. The settings we consider involve designing mechanisms for the following scenarios:

- Auctions with *hard payment constraints* such as budgets;
- Making the BIC mechanism and utility model more robust to the *common prior assumption*, by enforcing individual rationality ex-post (in presence of hard payment constraints), and
- *Non-linear utility functions* for seller and buyers, which includes scenarios such as risk averse seller and/or buyers, buyers with quitting rights, and cost to borrow money beyond budget.

We discuss these aspects briefly next, before presenting our results.

Ex-post Rationality and Payment Constraints: A criticism of Bayesian Incentive Compatibility (BIC) is that it is not robust to the common prior assumption on all buyers; in particular, if a buyer is agnostic to this prior information, the allocations and payments may make no sense to him. See for instance [Bergemann and Morris 2005; Mookherjee and Reichelstein

1992] for a discussion of this aspect, as well as the discussion implicit in [Daskalakis and Weinberg 2012]. One simple constraint to make the mechanism more robust is to ensure that the voluntary participation (or individual rationality (IR)) constraint always holds: Given a buyer’s revealed type, its allocation and payment in *any* outcome of the mechanism yield it non-negative utility for that revealed type. This is a very intuitive constraint, since a buyer will participate even when he has no knowledge of the prior. In several settings, a BIC mechanism can indeed be converted into one where this constraint holds as well. A notable exception to this happens when there are *payment constraints* on the buyers.

We consider BIC mechanism design in settings where there are general ex-post constraints not only on allocations, but also on the *payments* buyers make. The most important example is constraints on payments made by individuals, *i.e.*, budget constraints; another example is fairness constraints on allocations and payments across buyers, such as the popular envy-freeness constraint. The presence of these ex-post payment constraints reduces the participation risk from an individual buyer’s perspective and encourages their participation.

Consider for instance the case where buyers have (possibly private) *hard budget constraints*. There is a fairly long line of work [Che and Gale 1996; 2000; Laffont and Robert 1996; Border 2007; Pai and Vohra 2008] in economics literature on optimal BIC auctions with either public or private budget constraints. It was realized early on that budgets make the problem significantly harder even when they are publicly known. One crucial difference is that in the presence of payment constraints, a BIC mechanism can be implemented as an all-pay mechanism where buyers submit payment with bids (regardless of final allocations); this is ex-interim IR, but cannot be converted to a mechanism that is ex-post IR. As far as we are aware, the design of optimal auctions with ex-post IR constraints has not been considered in literature before.

Non-linear Utilities: In many auction design problems, the buyer’s utility is not linear in his allotment or payment. Consider, for example, auctions with *quitting rights* for buyers [Compte and Jehiel 2007]. Unlike the traditional BIC model where the allocations and payments are binding ex-post (so that the utility is linear in each scenario and only quasi-linear in expectation), in the quitting rights model, a buyer can decide to not participate (or quit) ex-post, so that the utility is quasi-linear in each outcome of the mechanism, and hence non-linear. The presence of quitting rights is important for a mechanism, as in variety of settings, the outcome cannot be legally enforced on buyers and it remains a buyer’s choice to accept the outcome *ex-post*.

We also consider a setting where a buyer incurs an arbitrary monotone cost to arrange for payments beyond his budget [Dutting et al. 2011]; this includes scenarios where a buyer can take a loan for additional payment, but he incurs an extra cost for the loan set up and the interest. Lastly, we consider risk-aversion of buyers and the seller [Eso and Futo 1999].

Informal Problem Statements and Our Results. We present *efficient algorithms for optimal BIC auctions* that satisfy ex-post IR in the presence of constraints on both payment and allocation in fairly general settings. We also design the *optimal auctions* in many models where buyers have *non-linear utility over allocation and payment*.

We assume the buyer’s type is discrete, and there are n buyers and m items. The running time we obtain are polynomial in the $n, m, \frac{1}{\epsilon}$, and the size of the type space of each buyer. It also depends polynomially on the ratio of largest to smallest valuations, and the ratio largest to smallest marginal prior probabilities for a single buyer.

- *Multi-unit auction with arbitrary valuations* (Section 4): There are m homogeneous items. Each buyer is associated with an arbitrary valuation function over the number of items, parameterized by his type. Each buyer can also have a hard (possibly private) budget constraint, and we seek ex-post IR mechanisms.

- *Non-linear utility functions* (Sec. 6): We consider the multi-unit auction setting when the private utility of a buyer is an arbitrary function of allocation and payment, which is parametrized by a discrete type. As special cases, we consider the quitting rights model [Compte and Jehiel 2007] described above, as well as the setting where an agent can borrow money at a cost [Dutting et al. 2011]. Finally, we allow the seller to have utilities on payments received [Eso and Futo 1999; Bhalgat et al. 2012a]. Our setting allows positive transfers, and do not need individual rationality as a constraint for optimality (both of which can raise more utility for a risk-averse seller).
- *Multi-item auctions* (see the full version): There are m divisible items (for instance, ad categories in an adword auction). Each buyer’s type is multi-dimensional but additive across items. Given any vector of revealed types, the allocations and payments must satisfy an arbitrary set of polyhedral constraints. As before, each buyer can have (potentially private) hard budget constraint, and we require IR to hold ex-post. In the absence of other constraints on allocations and payments, we enforce ex-post envy-freeness constraints where a buyer should not derive more utility from the allocation and payment made to another buyer.
- *Procurement auctions* (see the full version): The agents have private costs on procuring items. The utility of the auctioneer is linear in the subset of items he procures. The auctioneer has a budget constraint on payments. This is a special case of *budget feasible mechanisms* [Singer 2010; Bei et al. 2012] with linear utility functions.

We note that all results in this work easily extend when the distributions are correlated, as long as there is black box access to samples from the joint type space. We defer these details to the full version of the paper.

Overview of the Technique. The general framework to derive auctions using the multiplicative weight method is summarized in Section 5. A key ingredient of this framework is to explicitly encode the *mechanisms’s action* (allocation and payment for each buyer) *in each realization*, and connect them to the set of *holistic variables* using the *linearity of expectation* constraints. The former are exponentially many in number, but are connected to the latter via polynomially many linearity of expectation constraints. Our main insight is to use the multiplicative weight update (or Lagrangean) method to decide feasibility of these linearity of expectation constraints, and show that the oracle sub-problem that this method requires has a very simple structure - it decouples into a linear program over holistic variables, and into a per sample (or type vector) optimization problem of a certain form over the mechanism’s *action variables*. Even though the latter is over exponentially many type vectors, it can be estimated by solving the per-sample optimization subproblem over polynomially many samples. The per sample optimization subproblem is problem-specific, yet it is an easy exercise to decide when this problem will admit to polynomial time solutions. Assuming it does, the framework yields a pseudo-polynomial time algorithm to computer a ϵ -BIC mechanism that is an additive ϵ approximation to the optimal revenue.

We note, the classical multiplicative weight update framework [Arora et al. 2005] looks at the per-constraint violation by the *average-of-all-rounds* solution; we require (and establish) a slight generalization to this framework (see Section 3):

- The framework can handle different linear constraints in each round.
- We consider the average violation of the i th constraint (for each i) by the solution to the oracle sub-problem for a uniformly chosen round. With this change, the framework no longer requires the underlying solution space to be convex.

In comparison with previous work, our optimization technique is based on the Lagrangean of an *explicitly written program* instead of trying to directly characterize the space of primal interim allocations rules. Our running times are pseudo-polynomial in the ratio of largest

to smallest marginal prior probabilities. We leave improving this dependence without sacrificing any BIC or optimality properties as future work.

Related Work. Though many auctions implemented in practice incorporate budget constraints (for instance, ad allocation auctions), most theoretical results in the absence of a Bayesian assumption show impossibility of designing auctions with specific properties. For instance, recent work [Dobzinski et al. 2008; Goel et al. 2012] shows the impossibility of designing auctions that are Pareto-optimal in terms of welfare even in multi-unit settings. Furthermore, Nisan [Nisan 2009] show that competitive equilibria need not exist even when agents have additive valuations for items. The lone exceptions are positive results for simple multi-unit settings [Dobzinski et al. 2008; Goel et al. 2012; Bhattacharya et al. 2010a; Abrams 2006] with unlimited supply of the item. In view of these impossibility results, it is more natural to consider the Bayesian setting; however all results so far [Che and Gale 1996; 2000; Laffont and Robert 1996; Border 2007; Pai and Vohra 2008; Daskalakis and Weinberg 2012; Cai et al. 2012b] only satisfy individual rationality in expectation over the priors. We present (what we believe are) the first efficient algorithms for designing optimal auctions that satisfy individual rationality *ex-post* in a wide variety of situations.

We finally note that there has been a sequence of recent work on designing approximation algorithms for auctions in the Bayesian setting. For instance, the results in [Chawla et al. 2010; Bhattacharya et al. 2010b; Bhattacharya et al. 2010a; Chawla et al. 2011] and subsequent papers show constant factor approximations to optimal auctions under a stronger model called dominant strategy incentive compatibility; the results in [Singer 2010; Bei et al. 2012] show constant factor approximations to the welfare of procurement auctions subject to a budget constraint (what they term *budget feasible mechanisms*); and the results in [Bhalgat et al. 2012a] considers the seller’s as well as buyers’ risk-aversion. In this paper, our focus is on designing optimal (or near-optimal) auctions and not on approximation algorithms.

2. PRELIMINARIES

We describe notations used in the paper: There are n buyers and a risk-neutral seller. The type space for buyer i is T_i , and we denote the individual type of buyer i by t_i . We use \mathbf{t} to indicate the vector of reported types of all buyers in a realization of the mechanism. We also abuse the notation slightly to use t_i to indicate buyer i ’s type in \mathbf{t} . The seller maintains a prior distribution over the types of the buyers. The probability distribution of buyer i over its type is given by \mathcal{D}_i and we use \mathcal{D} to indicate the joint distribution over types of all buyers. We use $\mu(\mathbf{t})$ to denote the density of type vector \mathbf{t} , and $f_i(t_i)$ to denote the probability of type t_i for buyer i . For exposition of our results, we assume independence across buyers; however our results easily extend when buyers’ distributions are correlated. Further, for each auctions setting studied in this work, we use OPT to denote the expected revenue (or the utility) of the optimal mechanism under the same set of constraints.

2.1. Bayesian Incentive Compatibility

A mechanism is a mapping from a vector of revealed types (or bids) to a (possibly randomized) set of allocations and payments for each buyer. For most of this paper, we assume the utility of a buyer is *linear*: For any allocation made and payment charged to the buyer, his utility is the difference between the welfare from the allocation and the payment.

Bayesian (ex-interim) Incentive Compatibility. In this model of incentive compatibility, we assume the prior is common knowledge, and buyers are interested in maximizing their expected utility, where the expectation is over the types of the other buyers drawn from their respective priors. More formally, let $U_i(t'_i, t_i)$ denote the expected utility for buyer i when he reveals type t'_i and his true type is t_i , where this expectation is over the

types of the other buyers (drawn from their prior distributions) and the randomness in the mechanism. A Bayesian Incentive Compatible (BIC) mechanism has to enforce that:

Incentive Compatibility (IC):. $U_i(t_i, t_i) \geq U_i(t'_i, t_i)$ for every pair of types $t_i, t'_i \in T_i$, and
Individual Rationality (IR):. $U_i(t_i, t_i) \geq 0$ for all types $t_i \in T_i$.

A mechanism is said to be ϵ -BIC if it enforces $U_i(t_i, t_i) \geq U_i(t'_i, t_i) - \epsilon$ for every pair of types $t_i, t'_i \in T_i$.

Robust (ex-post) Individual Rationality. The BIC constraint only requires the allocations and payments to satisfy IR ex-interim. In this paper, we consider a sub-class of mechanisms that satisfy a more robust (ex-post) notion of IR: If buyer i reveals type t_i , then its allocation and payment should yield non-negative utility for the reported type *in each outcome* of the mechanism. We term this constraint *ex-post IR*.

Budgets Constraints. Each buyer is associated with a budget constraint. The budgets can be either publicly known, or it is buyer's private information specified using his type. We denote buyer i 's budget by B_i , when budgets are public. For private budgets, we use $B_i(t)$ to denote buyer i 's budget when his type is t . Unless mentioned otherwise, we assume that a buyer's utility of paying more than his budget is $-\infty$.

2.2. Problem Statements

In this work, we primarily focus on designing BIC mechanisms that maximize expected revenue and satisfy IR and budget constraints *ex-post*.

- *Multi-unit Auctions (Section 4)*: The seller has m identical items, each buyer i is associated with a publicly known valuation function $v_i : \mathcal{N} \times T_i \rightarrow \mathcal{R}^+$; his valuation for receiving k items is $v_i(k, t_i)$ when his type is t_i . We assume that $v_i(k, t_i) \geq 0$ for each $k \geq 0, i, t_i \in T_i$, and $v_i(0, t_i) = 0$ for each $i, t_i \in T_i$. We note, these valuations easily capture scenarios where each buyer wants at most one item.
- *Non-linear Utility (Section 6)*: Various multi-unit auction settings that involves buyers as well as the seller with non-linear utility functions are illustrated in Section 6.

In the full version of the paper, we discuss how to extend our technique to *multi-item auctions with divisible items* and *budget feasible mechanisms*.

Running Times. We are given a parameter L , that for any single buyer, bounds the absolute value of (a) his payment, (b) his valuation for any set of items, and (c) his utility (for any allocation and payment). Further, we assume that $f_i(t) \geq 1/L$ for each $i, t \in T_i$. Our algorithms produce ϵ -BIC mechanisms that are additive ϵ approximation to revenue in time polynomial in n (#buyers), m (#items), $\max_i |T_i|$, and the parameters L and $1/\epsilon$.

3. THE MULTIPLICATIVE WEIGHTS UPDATE FRAMEWORK

In this section, we present a slight generalization to the multiplicative weights update framework to decide the feasibility of a set of linear constraints over arbitrary spaces. We note its difference compared to the classical framework:

- The framework can (and needs to) handle different linear constraints in each round.
- We consider the average violation of the i th constraint (for each i) by the solution to the oracle sub-problem for a uniformly chosen round (instead of the violation by the average solution of all rounds). With this change, the framework no longer requires the underlying solution space to be convex (Observation 1).

We next illustrate the framework using the exposition and bounds due to [Arora et al. 2005]. We first define the generic problem to decide the feasibility of linear constraints: Given an arbitrary set $P \in \mathbf{R}^s$, and an $r \times s$ matrix A ,

$$\text{LP}(A, b, P): \exists x \in P \text{ such that } Ax \geq b?$$

The running time of the algorithm is quantified in terms of the WIDTH defined as:

$$\rho = \max_i \max_{x \in P} |a_i x - b_i|$$

The algorithm assumes an efficient oracle of the form. Let $y \geq 0$ be a r dimensional dual vector.

$$\text{ORACLE}(A, y): \text{Solve } C(A, y) = \max\{y^T A z : z \in P\}.$$

We now describe a generalization of the Arora-Hazan-Kale (AHK) procedure that allows *the matrix A and vector b to be different in each step*, albeit with the same number of rows r , and same width bound ρ . Denote the matrix at step j to be A_j , and its i^{th} row by a_{ij} ; denote the vector b at time j to be b_j .

ALGORITHM 1: Generalized AHK Algorithm

```

Let  $K \leftarrow \frac{4\rho^2 \log r}{\epsilon^2}$ ,  $y_1 = \mathbf{1}$ .
for  $j = 1$  to  $K$  do
    Find  $x_j$  using ORACLE( $A_j, y_j$ ).
    if  $C(A_j, y_j) < y_j^T b_j$  then
        | Declare LP( $A_j, b_j, P$ ) infeasible and terminate;
    end
    for  $i = 1, 2, \dots, r$  do
        | Set  $M_{ij} = a_{ij}x_j - b_{ij}$ . Update  $y_{i(j+1)}$  for  $i = 1, 2, \dots, r$  as follows.
        |  $y_{i(j+1)} \leftarrow y_{ij}(1 - \epsilon)^{M_{ij}/\rho}$  if  $M_{ij} \geq 0$ .
        |  $y_{i(j+1)} \leftarrow y_{ij}(1 + \epsilon)^{-M_{ij}/\rho}$  if  $M_{ij} < 0$ .
        | Optional step: Normalize  $y_{j+1}$  to have unit  $l_1$ -norm.
    end
end
Return  $(x_1, x_2, \dots, x_K)$ .

```

The following theorem follows by simply re-working the analysis in [Arora et al. 2005], we omit its proof.

THEOREM 3.1. *If for all steps j , LP(A_j, b_j, P) is feasible, the procedure never declares infeasibility, and the sequence (x_1, x_2, \dots, x_K) satisfies:*

$$\frac{\sum_j (a_{ij}x_j - b_{ij})}{K} + \epsilon \geq 0 \quad \forall i$$

OBSERVATION 1. *If P is convex and (A_j, b_j) are the same for all steps j , then $x^* = (\sum_j x_j)/K$ is a feasible solution with $Ax^* \geq b - \epsilon \mathbf{1}$. Further, even if P is not convex, the procedure is well-defined and we can interpret x^* as choosing one of the x_j uniformly at random from $\{x_1, x_2, \dots, x_K\}$.*

4. MULTI-UNIT AUCTIONS

In this section, we describe a framework for designing mechanisms by the multiplicative weight update method using multi-unit auctions as an example. We generalize this example to a structural framework in Section 5, and present more instantiations of this framework in later sections. We begin by the case where valuations of different buyers are independent, the only feasibility constraint is on the supply of items, and i.e. at most m items can be allocated, each buyer is associated with a public budget constraint, and the objective is to

design an ϵ -BIC mechanism that is (a) feasible, (b) ex-post IR, and (c) satisfies the budget constraints in each realization. The technique to handle private budgets is explained in Section 4.3.

The following theorem states our result for multi-unit auctions:

THEOREM 4.1. *For multi-unit auctions with per buyer budget constraints, there exists a polynomial time algorithm to compute a mechanism that is ex-post IR, ϵ -BIC, and is an ϵ -approximation to the revenue of the optimal mechanism (under the same set of constraints).*

In the rest of the section, we prove Theorem 4.1. Consider a realization of the mechanism, let \mathbf{t} be the reported type vector of all buyers. In the realization, let $x_{iq}(\mathbf{t}) \in \{0, 1\}$ be an indicator variable indicating whether buyer i is allocated $q \in \{0, 1, \dots, m\}$ items when the reported type vector is \mathbf{t} . Let $p_i(\mathbf{t})$ be the payment made by buyer i . As vector $\mathbf{x}(\mathbf{t}), \mathbf{p}(\mathbf{t})$ must be feasible, it should satisfy the constraints $\mathcal{F}(\mathbf{t})$ shown below.

$$\begin{aligned} \sum_i q x_{i,q}(\mathbf{t}) &\leq m \\ \sum_q v_i(q, t_i) x_{iq}(\mathbf{t}) &\geq p_i(\mathbf{t}) \quad \forall i, q \\ x_{iq}(\mathbf{t}) &\in \{0, 1\} \quad \forall i, q \\ \sum_q x_{iq} &= 1 \quad \forall i \\ p_i(\mathbf{t}) &\in [0, B_i] \quad \forall i \end{aligned}$$

Let $X_{iq}(t_i)$ be the probability that buyer i is allocated q items when his reported type is t_i , and let $P_i(t_i)$ be his expected payment when his type is t_i . This probability is over the randomness due to the other buyers' valuations, and due to the randomness in the mechanism. We get

$$f_i(t_i) X_{iq}(t_i) = \sum_{\mathbf{t} | t_i \in \mathbf{t}} \mu(\mathbf{t}) x_{iq}(\mathbf{t}) \quad \forall i, q, t_i \in T_i \quad (1)$$

$$f_i(t_i) P_i(t_i) = \sum_{\mathbf{t} | t_i \in \mathbf{t}} \mu(\mathbf{t}) p_i(\mathbf{t}) \quad \forall i, t_i \in T_i \quad (2)$$

The BIC constraint is the following: $\forall i, t_i, t'_i \in T_i$,

$$\sum_q v_i(q, t_i) X_{iq}(t_i) - P_i(t_i) \geq \sum_q v_i(q, t_i) X_{iq}(t'_i) - P_i(t'_i) \quad (3)$$

Further, we add a constraint to check whether revenue OPT is feasible. This can be written as the constraint:

$$\sum_{i, t_i \in T_i} f_i(t_i) P_i(t_i) \geq \text{OPT} \quad (4)$$

Randomized Mechanisms. An astute reader will object that the set $\mathcal{F}(\mathbf{t})$ only captures the space of *deterministic* mechanisms and *an optimal mechanism can take a randomized action for type vector \mathbf{t}* . To capture randomized mechanisms, we have to replace $\mathcal{F}(\mathbf{t})$ with the *convex hull* of the set $\mathcal{F}(\mathbf{t})$. However, this will only be a technicality - the multiplicative weight update framework described below optimizes a linear function of $\{x(\mathbf{t}), p(\mathbf{t})\}$ over this set, and this optimization problem is clearly the same whether we use $\mathcal{F}(\mathbf{t})$ or its convex hull. Since the same observation can be made of all the problems considered in this paper, we will use $\mathcal{F}(\mathbf{t})$ for expositional simplicity.

4.1. Structural Characterization

We define the set of feasibility constraints \mathcal{C} as the set of constraints $\mathcal{F}(\mathbf{t})$ for all type vectors \mathbf{t} , combined with the BIC constraint (3) and the revenue constraint Eq (4). Consider

deciding feasibility of Equations 1 and 2 subject to the constraints \mathcal{C} . This is equivalent to deciding the feasibility of following constraints, referred to as EQ:

$$\begin{aligned} X_{iq}(t_i) &= \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t}) x_{iq}(\mathbf{t}) / f_i(t_i) \quad \forall i, q, t_i \in T_i \\ P_i(t_i) &= \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t}) p_i(\mathbf{t}) / f_i(t_i) \quad \forall i, t_i \in T_i \end{aligned}$$

Using the multiplicative weight method, the feasibility problem reduces to a sequence of ORACLE(α, β) problems, where $\alpha_{iq t_i}, \beta_{it_i}$ ($1 \leq i \leq n, q \in \{0, 1, \dots, m\}, t_i \in T_i$) are the dual multipliers which could be positive or negative (as the constraints are equality constraints). ORACLE(α, β) asks us to solve, for α, β :

$$\begin{aligned} \text{Max} \quad & - \sum_{i, t_i \in T_i} f_i(t_i) \left(\sum_q \alpha_{iq t_i} X_{iq}(t_i) + \beta_{it_i} P_i(t_i) \right) \\ & + \sum_{\mathbf{t}} \mu(\mathbf{t}) \left(\sum_i \left(\sum_q (\alpha_{iq t_i} x_{iq}(\mathbf{t})) + \beta_{it_i} p_i(\mathbf{t}) \right) \right) \end{aligned}$$

subject to the following constraints:

$$\begin{aligned} \sum_q v_i(q, t_i) X_{iq}(t_i) - P_i(t_i) &\geq \sum_q v_i(q, t_i) X_{iq}(t'_i) - P_i(t'_i) \quad \forall i, t_i, t'_i \in T_i \\ \sum_{i, t_i \in T_i} f_i(t_i) P_i(t_i) &\geq \text{OPT} \\ \{x_{iq}(\mathbf{t})\} &\in \mathcal{F}(\mathbf{t}) \quad \forall \mathbf{t} \end{aligned}$$

Decoupling of the Oracle. Note that ORACLE(α, β) separates into two optimization problems. The first is the poly-size linear program LP_{exp} :

$$\text{Minimize} \quad \sum_{i, t_i \in T_i} f_i(t_i) \left(\sum_q \alpha_{iq t_i} X_{iq}(t_i) + \beta_{it_i} P_i(t_i) \right)$$

subject to the following constraints:

$$\begin{aligned} \sum_q v_i(q, t_i) X_{iq}(t_i) - P_i(t_i) &\geq \sum_q v_i(q, t_i) X_{iq}(t'_i) - P_i(t'_i) \quad \forall i, t_i, t'_i \in T_i \\ \sum_{i, t_i \in T_i} f_i(t_i) P_i(t_i) &\geq \text{OPT} \end{aligned}$$

The second is the problem $\mathcal{A}(\alpha, \beta)$, which for every type vector \mathbf{t} , solves the following problem:

$$\text{Maximize} \quad \sum_i \left(\sum_q \alpha_{iq t_i} x_{iq}(\mathbf{t}) \right) + \beta_{it_i} p_i(\mathbf{t})$$

subject to the constraints $\mathcal{F}(\mathbf{t})$.

Solving the Problem $\mathcal{A}(\alpha, \beta)$. Now we give a poly-time algorithm to solve the problem $\mathcal{A}(\alpha, \beta)$. Observe that, in an optimal assignment, if buyer i is given q items, then (a) $\beta_{it_i} \leq 0 \Rightarrow p_i(\mathbf{t}) = 0$, and (b) $\beta_{it_i} > 0 \Rightarrow p_i(\mathbf{t}) = \min\{B_i, v_i(q, t_i)\}$.

The allocation problem can be solved using dynamic programming as follows: Process buyers in order n to 1. Let $A[i, k]$ be the value of $\mathcal{A}(\alpha, \beta)$ considering buyers i through n and k items. The values of $A[i, k]$ can be computed as follows:

$$A[i, k] = \min_{0 \leq j \leq k} (A[i+1, k-j] + j \times \alpha_{ijt_i} F + \max\{\beta_{it_i} \times \min\{B_i, v_i(j, t_i)\}, 0\})$$

As we shall see later, we will require the algorithm to break ties consistently; this can always be achieved by having a state-less implementation of the algorithm.

Using Multiplicative Weight Update Framework. Now we setup the parameters for the multiplicative weight update framework. Recall that L is the largest possible welfare a single buyer can receive from any allocation. Further, L also upper bounds individual buyer's budgets. Then, the width of each equality constraint (EQ) is bounded by $\max\{n, L\} = L$. As $\max_i |T_i| \leq L$, the number of constraints in EQ are bounded by $2nmL$. Further, the

minimum support for any type for a buyer is bounded by $1/L$. Given any $\epsilon > 0$, choose $\delta = \epsilon/(nmL)$ and $K = O\left(\frac{L^2 \log(nmL)}{\delta^2}\right)$; let $(\alpha^1, \beta^1), (\alpha^2, \beta^2), \dots, (\alpha^K, \beta^K)$ be the dual variables in rounds $1, \dots, K$ associated with the multiplicative weight update process. Our mechanism $\mathcal{M}(\alpha[1 \dots K], \beta[1 \dots K])$ is defined below; Lemma 4.3 establishes its important properties.

Definition 4.2. Let \mathbf{t} be the reported type vector. The mechanism $\mathcal{M}(\alpha[1 \dots K], \beta[1 \dots K])$ chooses an integer ℓ between 1 and K uniformly at random, and solves the problem $\mathcal{A}(\alpha^\ell, \beta^\ell)$ for type vector \mathbf{t} . It then executes allocations and payments according to the solution of $\mathcal{A}(\alpha^\ell, \beta^\ell)$.

LEMMA 4.3. (a) *The mechanism $\mathcal{M}(\alpha[1 \dots K], \beta[1 \dots K])$ is ϵ -BIC, ex-post IR, (b) a buyer never pays more than his budget, and (c) its expected revenue is at least $\text{OPT} - \epsilon$.*

PROOF. Using the properties of the multiplicative update framework, Equations 1, and 2 are violated by at most $\epsilon/(nL)$. Since $\max_{i,q,t_i \in T_i} v_i(q, t_i) \leq L$, the error in each BIC constraint is bounded by ϵ/n . Further, the error in the revenue equation is bounded by ϵ as $\max_{i,t_i \in T_i} P_i(t_i) \leq L$. This completes the proof of the lemma. \square

The only remaining issue is to compute the values of the dual variables efficiently; computing them exactly will require time $\Pi_i |T_i|$ which is exponential in n .

4.2. Efficient Computation via Sampling

The main hurdle in running multiplicative update framework is that each constraint in EQ is over exponentially many variables. We relax each constraint in EQ by $\delta = \frac{\epsilon}{nmL}$. The relaxed constraints are

$$X_{iq}(t_i) \in \frac{\sum_{\mathbf{t}|t \in \mathbf{t}} \mu(\mathbf{t}) x_{iq}(\mathbf{t})}{f_i(t_i)} \pm \delta \quad \forall i, q, t_i \in T_i \quad (5)$$

$$P_i(t_i) \in \frac{\sum_{\mathbf{t}|t \in \mathbf{t}} \mu(\mathbf{t}) p_i(\mathbf{t})}{f_i(t_i)} \pm \delta \quad \forall i, t_i \in T_i \quad (6)$$

If the relaxed constraints are feasible (with respect to feasibility constraints \mathcal{C}), then it is easy to check that this implies that the BIC constraints are satisfied to an additive ϵ , and the revenue generated is at least $\text{OPT} - \epsilon$. Further, note that the *width* of each of relaxed constraint is bounded by $\rho = mL$. We apply the multiplicative weight update method as follows: In each round ℓ , given dual weights $(\alpha^\ell, \beta^\ell)$, we do the following. Generate $C = O(nmL^3 \log(nmLK)/\epsilon)$ samples from \mathcal{D} ; let S_ℓ be this set. Let $S_{it_i\ell} \subset S_\ell$ be the set of samples in which buyer i has type $t_i \in T_i$. We consider relaxed and sampled version of constraints EQ:

$$X_{iq}(t_i) \in \frac{\sum_{\mathbf{t} \in S_{it_i\ell}} x_{iq}(\mathbf{t})}{|S_{it_i\ell}|} \pm \delta \quad \forall i, q, t_i \in T_i$$

$$P_i(t_i) \in \frac{\sum_{\mathbf{t} \in S_{it_i\ell}} p_i(\mathbf{t})}{|S_{it_i\ell}|} \pm \delta \quad \forall i, t_i \in T_i$$

We now apply the ORACLE to these relaxed constraints using the dual weight vector $(\alpha^\ell, \beta^\ell)$. We make a crucial observation:

OBSERVATION 2. *The function ORACLE reduces to solving $\mathcal{A}(\alpha^\ell, \beta^\ell)$ on these samples.*

We now establish the correctness of our procedure. Let $\text{LP}(\mathcal{D})$ be the feasibility problem (constraints \mathcal{C} and EQ) defined on distribution \mathcal{D} and let $\text{LP}(S_\ell)$ for $1 \leq \ell \leq K$ be the feasibility problem defined on S_ℓ , where constraints in EQ are relaxed by δ . We focus on the

allocation constraints; the analysis for payment constraints is similar. Let $\{x_{iq}^\ell(\mathbf{t})\}$ denote the solution to the ORACLE on $\text{LP}(S_\ell)$. Let $\{x_{iq}^*(\mathbf{t})\}$ denote the feasible solution for $\text{LP}(\mathcal{D})$ with no slack.

LEMMA 4.4. *With high probability ($\geq 1 - 1/\text{poly}(nmLK/\epsilon)$), for all $1 \leq l \leq K$, we have:*

$$\left| \frac{\sum_{\mathbf{t} \in S_{it_i\ell}} x_{iq}^\ell(\mathbf{t})}{|S_{it_i\ell}|} - \frac{\sum_{\mathbf{t} \in \mathcal{D}|t_i \in \mathbf{t}} \mu(\mathbf{t}) x_{iq}^\ell(\mathbf{t})}{f_i(t_i)} \right| \leq \delta$$

$$\left| \frac{\sum_{\mathbf{t} \in S_{it_i\ell}} x_{iq}^*(\mathbf{t})}{|S_{it_i\ell}|} - \frac{\sum_{\mathbf{t} \in \mathcal{D}|t_i \in \mathbf{t}} \mu(\mathbf{t}) x_{iq}^*(\mathbf{t})}{f_i(t_i)} \right| \leq \delta$$

PROOF. Since we assumed $f_i(t_i) \geq 1/L$, using Hoeffdings's bounds and union bound, the probability that, for each $i, t_i \in T_i, \ell$, $|S_{it_i\ell}| \in (1 \pm \epsilon)C \times f_i(t_i)$ is $1 - \text{poly}(nmLK/\epsilon)$. Thus $|S_{it_i\ell}| = \Omega(nmL^2K/\epsilon)$ for each $i, t \in T_i, K$.

We now prove the first part. The value of $x_{iq}^\ell(\mathbf{t})$ is determined by the values of dual variables α^ℓ, β^ℓ . From the previous observation, the ORACLE decomposes into solving $\mathcal{A}(\alpha^\ell, \beta^\ell)$ on individual samples. The values of $x_{iq}(\mathbf{t})$ for different values of \mathbf{t} can be seen as a distribution parameterized by α^ℓ, β^ℓ , and since the set S_ℓ is chosen independent of previous rounds and of α^ℓ, β^ℓ , the process of drawing type vectors for S_ℓ can be seen as drawing independent samples from this distribution. As each $x_{iq}(\mathbf{t}) \in \{0, 1\}$ (i.e. a bounded random variable), we can apply Hoeffding's inequality and union bound to prove the first part.

For the second part, note $x_{iq}^*(\mathbf{t}) \in \{0, 1\}$, and the process of sampling valuation vectors \mathbf{t} corresponds to sampling from distribution $\{x_{iq}^*\}$. The proof now follows using Hoeffding's inequality and union bound. \square

As a corollary, suppose $\text{LP}(\mathcal{D})$ is feasible without any slack, then $\text{LP}(S_\ell)$ is feasible for $1 \leq \ell \leq K$ with high probability ($\geq 1 - 1/\text{poly}(nmLK/\epsilon)$). This implies that the ORACLE will not return infeasibility w.h.p.

To establish the correctness, it remains to bound the error in each constraint when the samples are drawn from \mathcal{D} and $\mathcal{M}(\alpha[1, \dots, K], \beta[1, \dots, K])$ is used determine allocations and payments. We focus on one constraint:

$$\frac{\sum_{\mathbf{t} \in S_{it_i\ell}} x_{iq}(\mathbf{t})}{|S_{it_i\ell}|} - X_{iq}(t_i) \geq -\delta \quad \forall i, t_i \in T_i$$

Using Theorem 3.1, we get

$$\frac{1}{K} \times \sum_{1 \leq \ell \leq K} \left(\frac{\sum_{\mathbf{t} \in S_{it_i\ell}} x_{iq}^\ell(\mathbf{t})}{|S_{it_i\ell}|} - X_{iq}^\ell(t_i) \right) \geq -\delta - \delta$$

where $x_{iq}^\ell(\mathbf{t})$ denotes whether in round ℓ , buyer i is given q items when the type vector is \mathbf{t} . Applying Lemma 4.4 once again, we have:

$$\sum_{1 \leq \ell \leq K} \frac{\sum_{\mathbf{t} \in \mathcal{D}|t_i \in \mathbf{t}} \mu(\mathbf{t}) x_{iq}^\ell(\mathbf{t}) / f_i(t_i) - X_{iq}^\ell(t_i)}{K} \geq -3\delta$$

Rearranging, we obtain:

$$\sum_{\mathbf{t} \in \mathcal{D}|t_i \in \mathbf{t}} \frac{\mu(\mathbf{t})}{f_i(t_i)} \times \sum_{1 \leq \ell \leq K} \frac{x_{iq}^\ell(\mathbf{t})}{K} - \sum_{1 \leq \ell \leq K} \frac{X_{iq}^\ell(t_i)}{K} \geq -3\delta$$

This shows that if round $\ell \in \{1, 2, \dots, K\}$ is chosen uniformly at random and its allocation and payment according to mechanism $\mathcal{A}(\alpha^\ell, \beta^\ell)$ are used, this yields an ϵ -BIC mechanism with revenue $\text{OPT} - \epsilon$.

We note, the algorithm fails to compute a mechanism with required guarantees on incentive compatibility and revenue w.p. $< 1/\text{poly}(1/\epsilon, n, m, L)$, due to various errors mentioned in Lemma 4.4. This is easy to fix: we still perform the allocation and the payment according to the mechanism generated by the above process. As the maximum valuation of a buyer for any allocation is bounded above by L , the expected error (over the randomness introduced by the sampling process over various rounds) in the BIC constraint is less than ϵ . This approach does not work when there exists a type for a buyer without required support in the sampled set in any round. The probability of this event is $< 1/\text{poly}(1/\epsilon, n, m, K, L)$ and the algorithm can be restarted in this case, without affecting the expected running time.

4.3. Handling Private Budgets

Next, we illustrate how to extend our techniques when buyers' budgets are private and specified through the reported types. Recall that $B_i(t_i)$ denotes buyer i 's budget when his reported type is t_i . Under the assumption that, the utility of paying more than the budget is $-\infty$ for buyer, we offer all items to an arbitrarily chosen buyer with a vanishingly small probability; otherwise we run the mechanism $\mathcal{M}(\alpha[1 \dots T], \beta[1 \dots T])$. In such case, buyer i never reports type t'_i with $B_i(t'_i) > B_i(t_i)$ when his real type is t_i . Now the BIC constraint changes to

$$\sum_q v_i(q, t_i) X_{iq}(t_i) - P_i(t_i) \geq \sum_q v_i(q, t_i) X_{iq}(t'_i) - P_i(t'_i) \\ \forall i, t_i, t'_i \in T_i \text{ with } B_i(t_i) \geq B_i(t'_i)$$

Thus, in-spite of having utility of $-\infty$ for paying more than the budget, the width of the LP does not increase and we can use techniques developed in the earlier part of this section to develop an ϵ -BIC, ex-post IR mechanism.

5. OVERALL FRAMEWORK FOR OPTIMAL AUCTION DESIGN

We next summarize the generic framework for mechanism design using the multiplicative weight update method. There are two sets of variables:

- *Action Variables*: For each type vector \mathbf{t} , $x_i(\mathbf{t})$ and $p_i(\mathbf{t})$ indicates the mechanism's allocation and payment for buyer i . As discussed earlier, the optimal mechanism can take a randomized action for a type vector \mathbf{t} ; it does not affect the analysis of the framework.
- *Holistic Variables*: For buyer i , each $t_i, t'_i \in T_i$, $U_i(t_i, t'_i)$ indicates the average utility for reporting type t_i when the real type is t'_i . Further, $\mathcal{P}_i(t_i, t'_i)$ is his average payment for reported type t_i when the real type is t'_i .

There are three sets of constraints:

- $\mathcal{F}(\mathbf{t})$: the feasible set of actions for type vector \mathbf{t} . In each realization, $\langle \mathbf{x}(\mathbf{t}), \mathbf{p}(\mathbf{t}) \rangle \in \mathcal{F}(\mathbf{t})$.
- EQ: the set of constraints that connect action variables to holistic variables. *This connection is problem specific*, we illustrate its generic structure.

For buyer i and a reported type vector \mathbf{t} , where $t_i \in \mathbf{t}$ is his reported type, let $\mathbf{u}_i(\mathbf{t}_{t_i \in \mathbf{t}}, t'_i, x_i(\mathbf{t}), p_i(\mathbf{t}))$ and $h_i(\mathbf{t}_{t_i \in \mathbf{t}}, t'_i, x_i(\mathbf{t}), p_i(\mathbf{t}))$ be his utility and payment when t'_i is his real type and $\langle x_i(\mathbf{t}), p_i(\mathbf{t}) \rangle$ are the mechanism's action variables. Then the constraints

for buyer i are

$$U_i(t_i, t'_i) = \sum_{\mathbf{t}|t_i \in \mathbf{t}} \frac{\mu(\mathbf{t})}{f_i(t_i)} \mathbf{u}_i(\mathbf{t}_{t_i \in \mathbf{t}}, t'_i, x_i(\mathbf{t}), p_i(\mathbf{t}))$$

$$\mathcal{P}_i(t_i, t'_i) = \sum_{\mathbf{t}|t_i \in \mathbf{t}} \frac{\mu(\mathbf{t})}{f_i(t_i)} h_i(\mathbf{t}_{t_i \in \mathbf{t}}, t'_i, x_i(\mathbf{t}), p_i(\mathbf{t}))$$

— Linear constraints on holistic variables such as BIC, revenue, and other ex-interim constraints.

Each equation in EQ plays the role of an expert in the multiplicative weight update framework. The oracle problem $\text{ORACLE}(\alpha^\ell, \beta^\ell)$ in round ℓ asks us to solve, for dual multipliers α^ℓ, β^ℓ , the following:

$$\text{Max} \sum_{\mathbf{t}} \left(\mu(\mathbf{t}) \times \sum_{i, t'_i \in T_i} \left(\alpha_{it_i t'_i}^\ell \mathbf{u}_i(\mathbf{t}_{t_i \in \mathbf{t}}, t'_i, x_i(\mathbf{t}), p_i(\mathbf{t})) + \beta_{it_i t'_i}^\ell h_i(\mathbf{t}_{t_i \in \mathbf{t}}, t'_i, x_i(\mathbf{t}), p_i(\mathbf{t})) \right) \right)$$

subject to constraint $\mathcal{F}(\mathbf{t})$ for each \mathbf{t} . The optimization problem $\mathcal{A}(\alpha^\ell, \beta^\ell)$ for type vector \mathbf{t} maximizes, subject to constraint $\mathcal{F}(\mathbf{t})$, the quantity

$$\sum_{i, t'_i \in T_i} \left(\alpha_{it_i t'_i}^\ell \mathbf{u}_i(\mathbf{t}_{t_i \in \mathbf{t}}, t'_i, x_i(\mathbf{t}), p_i(\mathbf{t})) + \beta_{it_i t'_i}^\ell h_i(\mathbf{t}_{t_i \in \mathbf{t}}, t'_i, x_i(\mathbf{t}), p_i(\mathbf{t})) \right)$$

Let ALG be the algorithm that can solve this problem optimally. Then, as established in Section 4, it suffices to solve $\mathcal{A}(\alpha, \beta)$ for C samples of type vectors drawn from the joint distribution; $C = \text{poly}(n, m, \sum_i |T_i|, 1/\epsilon, L)$. The dual multipliers $(\alpha^{\ell+1}, \beta^{\ell+1})$ are computed based on the error in each constraint in round ℓ . The process is repeated $K = \text{poly}(n, \sum_i |T_i|, 1/\epsilon, L)$ times; this generates K sets of dual variables $((\alpha^1, \beta^1), (\alpha^2, \beta^2), \dots, (\alpha^K, \beta^K))$. The eventual mechanism is as follows: for reported type vector \mathbf{t} , pick a round at random, say ℓ , and solve the optimization problem $\mathcal{A}(\alpha^\ell, \beta^\ell)$ for type vector \mathbf{t} .

In general, a main hurdle to achieving tractability is that the Lagrange multipliers, α, β can be negative. We note, in many cases the *full generality of the framework is not needed*: We can eliminate several variables, and the constraints EQ as well as the holistic variables can take a simple form that is more tractable.

We present instantiations of this framework for multi-item auctions next. The key point is the following: *As long as the ORACLE subproblem constructed above is polynomial time solvable or admits to an approximation algorithm, the entire framework yields a polynomial time algorithm, or a similar approximation guarantee.*

Multi-item Auctions and Tractability. It is quite straightforward to extend the above framework to multi-dimensional auctions. We defer the discussion on the case with budgets and other ex-post IR constraints such as envy-freeness to the full version of the paper, and we only summarize it below. The action variables are $x_{ij}(\mathbf{t})$, which denotes whether item j allocated to buyer i when the buyers' type vector is \mathbf{t} ; and $p_i(\mathbf{t})$, which is the price paid by buyer i . For any \mathbf{t} , the allocation and payments must be feasible for a set of allocation and IR constraints, denoted by $\mathcal{F}(\mathbf{t})$. The holistic variables are X_{ijt_i} , denoting the expected allocation of item j to buyer i when his type is t_i , and P_{it_i} denoting the expected price paid. When a buyer's welfare is linear in the valuations for different items, the BIC constraints are linear in the holistic variables. The overall problem reduces to checking feasibility of constraints

$$f_i(t_i)X_{ijt_i} = \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t})x_{ij}(\mathbf{t}) \quad \forall i, j, t_i \in T_i$$

$$f_i(t_i)P_{it_i} = \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t})p_i(\mathbf{t}) \quad \forall i, t_i \in T_i$$

In the full version of the paper, we show that when the items are divisible, the buyers' utilities are linear, and the set of feasible allocations is convex, the ORACLE problem remains tractable even with private ex-post budgets and other IR constraints.

We note, if there are no ex-post constraints on payments (setting in [Daskalakis and Weinberg 2012; Cai et al. 2012b; 2013]), the only action variables capture allocations for each buyer given the joint type revealed, so that $\mathcal{F}(\mathbf{t})$ is simply the space of feasible allocations. The constraints EQ relate these per-scenario allocation variables to the expected allocations for each buyer given his type. Since there are no per-scenario payment constraints, the holistic payment variables are only present in the BIC and ex-interim IR constraints, and our approach can handle general allocation constraints (over indivisible items) as considered in [Daskalakis and Weinberg 2012; Cai et al. 2012b; 2013].

We illustrate this for the multi-item auction problem with additive valuations and indivisible items, where the allocation set $\mathcal{F}(\mathbf{t}) = \{\mathbf{x}(\mathbf{t}) \in \mathcal{P}\}$ is downward closed and only admits to c -approximately optimizing welfare [Cai et al. 2013]. The key hurdle with applying the above methodology as is, is that the dual multipliers can be negative. But this is easy to fix: Since we do not have payment constraints, we simply need to check the feasibility of the following *inequality* constraint:

$$f_i(t_i)X_{ijt_i} \leq \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t})x_{ij}(\mathbf{t}) \quad \forall i, j, t_i$$

subject to the constraints given by $\mathcal{F}(\mathbf{t})$ for all \mathbf{t} and the BIC, ex-interim IR, and revenue constraints on $\{\mathbf{X}, \mathbf{P}\}$.

Consider a relaxed feasibility problem for $c \geq 1$:

$$f_i(t_i)X_{ijt_i} \leq c \times \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t})x_{ij}(\mathbf{t}) \quad \forall i, j, t_i$$

The ORACLE(α) problem maximizes welfare, $\sum_{i,j} \alpha_{ijt_i} x_{ij}(\mathbf{t})$ subject to $\mathcal{F}(\mathbf{t})$. Since the constraint is inequality, we have $\alpha \geq 0$ so that the ORACLE sub-problem is precisely welfare maximization. Suppose this has a c approximation. Suppose there is a feasible solution to the original program of the following form

$$f_i(t_i)X_{ijt_i}^* \leq \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t})x_{ij}^*(\mathbf{t}) \quad \forall i, j, t_i$$

Then, we must have:

$$\sum_{i,j,t_i} \alpha_{ijt_i} f_i(t_i)X_{ijt_i}^* \leq \sum_{i,j,t_i} \alpha_{ijt_i} \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t})x_{ij}^*(\mathbf{t})$$

Since ORACLE(α) has a c -approximation, there exists a solution $\tilde{\mathbf{x}}$ that can be computed in polynomial time such that:

$$\sum_{i,j,t_i} \alpha_{ijt_i} f_i(t_i)X_{ijt_i}^* \leq c \times \sum_{i,j,t_i} \alpha_{ijt_i} \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t})\tilde{x}_{ij}(\mathbf{t})$$

This implies that the multiplicative weight update method using the c -approximate oracle will never declare infeasibility of the relaxed program. The rest of the argument is the same as before, and yields a randomized mechanism using the multiplicative weight method.

The only issue is in making the final mechanism obey the BIC constraints since the inequality constraint implies the probability of allocation need not be exactly X_{ijt_i} in expectation, and could be larger. The fix is however a standard trick that works for downward closed environments. Suppose the final slack in the constraint for (i, j, t_i) , that is, the ratio of LHS to RHS, is $\beta_{ijt_i} \leq 1$, then whenever the mechanism decides to allocate item j to buyer i for type t_i , we allocate with probability β_{ijt_i} . This ensures the expected allocation is *exactly* X_{ijt_i}/c . If the charge exactly P_{it}/c via an all-pay scheme, then this ensures that the mechanism is BIC with revenue OPT/ c .

We note, when an ellipsoid based algorithm is used for this problem [Cai et al. 2013], the feasible region defined by the separation oracle is not convex and additional steps are required to design the eventual mechanism. In comparison, the multiplicative weight update

based technique does not have any such dependence and we get a mechanism using a direct application of this technique.

6. NON-LINEAR UTILITY FUNCTIONS

In this section, we consider multi-unit auctions (with budgets) when agents have non-linear utility functions. We first consider two well studied special cases of buyers with non-linear utility function over allocation and payment, namely (a) auction with quitting rights and (b) the soft budget constraint with cost to borrow. We also address a general setting in which the seller has a non-linear utility function over revenue and buyers have arbitrary utility functions over allocations and payments.

6.1. Auctions with Quitting Rights

We first consider a model of auctions with *quitting rights* [Compte and Jehiel 2007]: The outcome of the mechanism is not binding on the buyer, and *in each realization* of the mechanism, the buyer has the option to quit if his payment is more than his valuation of the allocated set. In other words, in each outcome, his utility is the maximum of zero and the difference in his welfare and payment. We note, even though the ex-post IR constraint makes a buyer's utility non-negative for the correctly reported type, the non-binding nature of contract makes utility non-negative even for mis-reporting the type. This makes the BIC constraint non-linear in the expected allocations and payments, and a buyer's expected utility for each type needs to be computed explicitly.

As before, consider a realization of the mechanism, let \mathbf{t} be the reported type vector of all buyers. In the realization, let $x_{iq}(\mathbf{t}) \in \{0, 1\}$ be an indicator variable indicating whether buyer i is allocated $q \in \{0, 1, \dots, m\}$ items when the reported type vector is \mathbf{t} . Let $p_i(\mathbf{t})$ be the payment made by buyer i . The variables should satisfy the constraints $\mathcal{F}(\mathbf{t})$ below.

$$\begin{aligned} \sum_q v_i(q, t_i) x_{iq}(\mathbf{t}) - p_i(\mathbf{t}) &\geq 0 && \forall i \\ \sum_q q x_{iq}(\mathbf{t}) &\leq m && \forall i \\ x_{iq}(\mathbf{t}) &\in \{0, 1\} && \forall i, q \\ \sum_q x_{iq} &= 1 && \forall i \\ p_i(\mathbf{t}) &\in [0, B_i] && \forall i \end{aligned}$$

Let $U_i(t_i, t'_i)$ be buyer i 's expected utility when his reported type is t_i and the real type is t'_i . We get, $\forall i, t_i, t'_i \in T_i$

$$U_i(t_i, t'_i) = \sum_{\mathbf{t} | t_i \in \mathbf{t}} \frac{\mu(\mathbf{t})}{f_i(t_i)} \left(\max \left\{ \sum_q v_{iq}(t'_i) x_{iq}(\mathbf{t}) - p_i(\mathbf{t}), 0 \right\} \right)$$

(Recall, we denote buyer i 's reported type in \mathbf{t} by t_i .) The BIC constraint is the following:

$$U_i(t_i, t_i) \geq U_i(t'_i, t_i) \quad \forall i, t_i, t'_i \in T_i$$

We add the revenue constraints:

$$\begin{aligned} \sum_{i, t_i} f_i(t_i) P_i(t_i) &\geq \text{OPT} \\ f_i(t_i) P_i(t_i) &= \sum_{\mathbf{t} | t_i \in \mathbf{t}} \mu(\mathbf{t}) p_i(\mathbf{t}) \quad \forall i, t_i \in T_i \end{aligned}$$

We discuss the correctness of the revenue constraints: as the mechanism is always individual rational for the reported type, the revenue from buyer i for type vector \mathbf{t} is $p_i(\mathbf{t})$, when he reports his type truthfully. The ORACLE problem $\mathcal{A}(\alpha, \beta)$, which for every type vector \mathbf{t} , solves the following subject to constraints $\mathcal{F}(\mathbf{t})$:

$$\text{Maximize } \sum_{it'_i} \alpha_{it'_i} \left(\max \left\{ \sum_q v_i(q, t'_i) x_{iq}(\mathbf{t}) - p_i(\mathbf{t}), 0 \right\} \right) + \sum_i \beta_{it_i} p_i(\mathbf{t})$$

For any type vector \mathbf{t} , this can be solved using dynamic programming: if q items are allocated to buyer i then the payment is uniquely determined as follows:

- (a) If $\beta_{it_i} \leq 0$, then $p_i(\mathbf{t}) = 0$
- (b) If $\beta_{it_i} > 0$, then $p_i(\mathbf{t})$ is the price that maximizes

$\sum_{t'_i} \alpha_{it_it'_i} \left(\max\{\sum_q v_i(q, t'_i)x_{iq}(\mathbf{t}) - p_i(\mathbf{t}), 0\} \right)$. Its value lies in the set $\{0, B_i\} \cup \{v_i(q, t'_i) | t'_i \in T_i\}$ and can be computed in polynomial time. This implies a poly-time dynamic program analogous to that in Section 4. We get the following theorem using multiplicative weight update method.

THEOREM 6.1. *For multi-unit auctions with quitting rights and budget constraints, there exists a polynomial time algorithm to compute an ϵ -BIC mechanism with revenue $\text{OPT} - \epsilon$.*

6.2. Soft Budget Constraints

Until now, we have assumed that buyers have fixed budgets and cannot pay more than their budgets. Recently, models with *soft* budgets have been considered assuming a buyer's ability to borrow at an additional cost [Dutting et al. 2011]. The model is formally defined as follows: buyer i is associated with a piecewise linear function $c_i : \mathcal{R} \rightarrow \mathcal{R}$; $c_i(p)$ denotes the total cost for arranging payment p and we assume that $c_i(p) - p$ is monotone. For instance, if buyer i can arrange for interest-free B_i dollars, and has to pay an interest rate of r beyond that; then $c_i(p) = p + \max\{(1+r) \times (p - B_i), 0\}$. The function c_i easily accounts for situations where (a) there is an additional fixed cost to setup the loan, and (b) the total interest amount increases with the amount of the loan.

The buyer's utility in a realization is his valuation of the allocated items minus the cost for his payment. The details are similar to the previous setting, we note the differences. The function $U(t_i, t'_i)$ changes to

$$f_i(t_i)U_i(t_i, t'_i) = \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t}) \left(\sum_q v_i(q, t'_i)x_{iq}(\mathbf{t}) - c_i(p_i(\mathbf{t})) \right)$$

The additional constraint in $\mathcal{F}(\mathbf{t})$ is the IR constraint that the mechanism is rational for the reported type:

$$\sum_q v_i(q, t_i)x_{iq}(\mathbf{t}) - c_i(p(\mathbf{t})) \geq 0$$

As before, for each type vector \mathbf{t} , the ORACLE solves the following subject to constraints $\mathcal{F}(\mathbf{t})$:

$$\text{Maximize } \sum_{it'_i} \alpha_{it_it'_i} \left(\sum_q v_i(q, t'_i)x_{iq}(\mathbf{t}) - c_i(p_i(\mathbf{t})) \right)$$

The optimal solution has the following property: if q items are allocated to buyer i , then his payment within the set $\{0, v_i(q, t_i)\}$ and among the end points of the linear pieces of $c_i(p)$. Thus the optimal assignment can be found using dynamic programming and an ϵ -BIC mechanism can be designed using the multiplicative update framework.

6.3. Non Linear Utility for Seller

The settings considered until now assume that the seller's utility is linear in revenue and the objective is to maximize his expected revenue. In this section, we address the setting in which the seller is associated with an arbitrary monotone utility function $\mathbf{U} : \mathcal{R} \rightarrow \mathcal{R}$; *i.e.* the seller's utility for revenue Z is $\mathbf{U}(Z)$. The objective of the mechanism is to maximize the seller's expected utility $\mathbf{E}_Z[\mathbf{U}(Z)]$. Further, each buyer i is associated with an arbitrary utility function $\mathbf{u}_i : \mathcal{Z} \times \mathcal{N} \times T_i \rightarrow \mathcal{R}^+$; buyer i 's utility for receiving q items for a payment of p is $\mathbf{u}_i(p, q, t_i)$ where t_i is his type. This includes scenarios where the seller and/or the buyers are risk averse.

We make following assumptions on the nature of utility functions, as well as on the optimal mechanism: (a) a buyer's payment is allowed to be negative (positive transfer is

allowed) however the set of possible payments from a buyer is restricted to integers between $-L$ and L , (b) for each $i, t_i \in T_i, p, q, |\mathbf{u}_i(p, q, t_i)| \leq L$, and (c) $\mathbf{U}(0) = 0$.

The assumption about integral payments is for simplicity in exposition; if all buyers' and seller's utility functions are monotone and have bounded derivative, then any *BIC* mechanism can be converted into an ϵ -BIC mechanism whose payments are integral - we simply round down the payment to a suitably discretized value so that the utility does not change by more than ϵ . The reason we allow positive transfers in the above framework is that it can raise much more utility for the seller (see [Bhalgat et al. 2012a] for details).

In the optimal mechanism, let $x_{ipq}(\mathbf{t})$ be an indicator variable that denote whether buyer i is given q items for a payment of p when buyers' type vector is \mathbf{t} . For any \mathbf{t} , allocation and payments must be feasible for the following set of constraints $\mathcal{F}(\mathbf{t})$, where the final constraint imposes ex-post IR:

$$\begin{aligned} \sum_{i,p,q} q x_{ipq}(\mathbf{t}) &\leq m \\ \sum_{p,q} x_{ipq}(\mathbf{t}) &= 1 \quad \forall i \\ x_{ipq}(\mathbf{t}) &\in \{0, 1\} \quad \forall i, p, q \\ x_{ipq}(\mathbf{t}) &= 0 \quad \forall i, p, q \text{ s.t. } \mathbf{u}_i(p, q, t_i) < 0 \end{aligned}$$

Let $X_{ipq}(t_i)$ denote the probability that buyer i is allocated q items at price p when his revealed type is t_i . The problem reduces to checking the feasibility of the following set of constraints

$$\begin{aligned} f_i(t_i)X_{ipq}(t_i) - \sum_{\mathbf{t}|t_i \in \mathbf{t}} \mu(\mathbf{t})x_{ipq}(\mathbf{t}) &= 0 \quad \forall i, t_i \in T_i, p, q \\ \sum_{\mathbf{t}} \mu(\mathbf{t})\mathbf{U} \left(\sum_{i,p,q} p \times x_{ipq}(\mathbf{t}) \right) &\geq R \end{aligned}$$

subject to constraints given by $\mathcal{F}(\mathbf{t})$ (for all type vectors \mathbf{t}) and the BIC constraint given below:

$$\sum_{p,q} (X_{ipq}(t_i) - X_{ipq}(t'_i)) \times \mathbf{u}_i(p, q, t_i) \geq 0 \quad \forall i, t_i, t'_i \in T_i$$

As before, the ORACLE problem $\mathcal{A}(\alpha, \beta)$, for each type vector \mathbf{t} , solves the following

$$\text{Max} \left(\sum_{i,p,q,t_i \in \mathbf{t}} \alpha_{ipqt_i} x_{ipq}(\mathbf{t}) + \beta \times \mathbf{U} \left(\sum_{i,p,q} p \times x_{ipq}(\mathbf{t}) \right) \right)$$

subject to constraints $\mathcal{F}(\mathbf{t})$. We note that α_{ipqt} can be both positive or negative, however β is always positive.

Solving $\mathcal{A}(\alpha, \beta)$ for \mathbf{t} : The problem can be solved optimally using a dynamic program as follows: let $A(i, k, z)$ be the maximum value of the objective function when k items are allocated to first i buyers generating a total revenue of z from them. The recurrence relation for the dynamic program is

$$A(i, k, z) = \min_{q \leq m-k} \max_p \{ \beta \times (U(z+p) - U(z)) + A(i+1, k+q, z+p) + \alpha_{ipqt_i} \}$$

Note, our assumption that payments to buyers are always integral and bounded between $-L$ and $+L$. Thus, $A(*, *, z)$ needs to be defined only for $2nL$ different values of revenue, and the dynamic program can be solved in time polynomial in L . Using the multiplicative weight update framework, we get following theorem:

THEOREM 6.2. *Given a seller with arbitrary utility function over total revenue and buyers with arbitrary utility functions over payments and allocations, there exists a polynomial time algorithm to compute an ϵ -BIC mechanism with expected utility $\text{OPT} - \epsilon$ for the seller.*

REFERENCES

- ABRAMS, Z. 2006. Revenue maximization when bidders have budgets. In *SODA*. 1074–1082.
 ALAEI, S., FU, H., HAGHPANAH, N., HARTLINE, J. D., AND MALEKIAN, A. 2012. Bayesian optimal auctions via multi- to single-agent reduction. In *EC*.

- ARORA, S., HAZAN, E., AND KALE, S. 2005. The multiplicative weights update method: A meta algorithm and applications. *Theory of Computing* 8, 121–164.
- BEI, X., CHEN, N., GRAVIN, N., AND LU, P. 2012. Budget feasible mechanism design: from prior-free to bayesian. In *STOC*. 449–458.
- BERGEMANN, D. AND MORRIS, S. 2005. Robust mechanism design. *Econometrica* 73, 6, 1771–1813.
- BHALGAT, A., CHAKRABORTY, T., AND KHANNA, S. 2012a. Mechanism design for a risk averse seller. In *WINE*.
- BHALGAT, A., GOLLAPUDI, S., AND MUNAGALA, K. 2012b. Mechanism design with multiplicative weight method. In *Arxiv*.
- BHATTACHARYA, S., CONITZER, V., MUNAGALA, K., AND XIA, L. 2010a. Incentive compatible budget elicitation in multi-unit auctions. In *SODA*.
- BHATTACHARYA, S., GOEL, G., GOLLAPUDI, S., AND MUNAGALA, K. 2010b. Budget constrained auctions with heterogeneous items. In *STOC*.
- BORDER, K. C. 2007. Reduced form auctions revisited. *Economic Theory* 31, 1, 167–181.
- CAI, Y., DASKALAKIS, C., AND WEINBERG, S. M. 2012a. An algorithmic characterization of multi-dimensional mechanisms. In *STOC*.
- CAI, Y., DASKALAKIS, C., AND WEINBERG, S. M. 2012b. Optimal multi-dimensional mechanism design: Reducing revenue to welfare maximization. In *FOCS*.
- CAI, Y., DASKALAKIS, C., AND WEINBERG, S. M. 2013. Reducing revenue to welfare maximization : Approximation algorithms and other generalizations. In *SODA*.
- CHAWLA, S., HARTLINE, J. D., MALEC, D. L., AND SIVAN, B. 2010. Multi-parameter mechanism design and sequential posted pricing. In *STOC*. 311–320.
- CHAWLA, S., MALEC, D. L., AND MALEKIAN, A. 2011. Bayesian mechanism design for budget-constrained agents. In *EC*.
- CHE, Y. AND GALE, J. 1996. Expected revenue of the all-pay auctions and first-price sealed bid auctions with budget constraints. *Economic Letters* 50, 373–380.
- CHE, Y. AND GALE, J. 2000. The optimal mechanism for selling to a budget constrained buyer. *Journal of Economic Theory* 92, 2, 198–233.
- COMPTE, O. AND JEHIEL, P. 2007. On quitting rights in mechanism design. *American Economic Review* 97, 2, 137–141.
- CONITZER, V. AND SANDHOLM, T. 2004. Self-interested automated mechanism design and implications for optimal combinatorial auctions. In *ACM Conference on Electronic Commerce*. 132–141.
- DASKALAKIS, C. AND WEINBERG, S. M. 2012. Symmetries and optimal multi-dimensional mechanism design. In *EC*.
- DOBZINSKI, S., LAVI, R., AND NISAN, N. 2008. Multi-unit auctions with budget limits. In *FOCS*.
- DUTTING, P., HENZINGER, M., AND WEBER, I. 2011. An expressive mechanism for auctions on the web. In *WWW*.
- ESO, P. AND FUTO, G. 1999. Auction design with a risk averse seller. In *Economic Letters*.
- GOEL, G., MIRROKNI, V., AND PAES LEME, R. 2012. Polyhedral clinching auctions and the adwords polytope. In *STOC*.
- HARTLINE, J. D. AND LUCIER, B. 2010. Bayesian algorithmic mechanism design. In *STOC*. 301–310.
- LAFFONT, J. AND ROBERT, J. 1996. Optimal auction with financially constrained buyers. *Economic Letters* 52, 2, 181 – 186.
- MOOKHERJEE, D. AND REICHELSTEIN, S. 1992. Dominant strategy implementation of Bayesian incentive compatible allocation rules. *J. Econ. Theory* 56, 2, 378–399.
- MYERSON, R. B. 1981. Optimal auction design. *Mathematics of Operations Research* 6, 1, 58–73.
- NISAN, N. 2009. Google’s auction for tv ads. In *ESA*.
- PAI, M. AND VOHRA, R. 2008. Optimal auctions with financially constrained bidders. *Working Paper*.
- PLOTKIN, S. A., SHMOYS, D. B., AND TARDOS, E. 1995. Fast approximation algorithms for fractional packing and covering problems. *Math. Oper. Res.*
- SINGER, Y. 2010. Budget feasible mechanisms. In *FOCS*.
- YOUNG, N. E. 1995. Randomized rounding without solving the linear program. In *SODA*.