Preventing Strategic Manipulation in Iterative Auctions: Proxy Agents and Price-Adjustment

David C. Parkes and Lyle H. Ungar
Computer and Information Science Department
University of Pennsylvania
200 South 33rd Street, Philadelphia, PA 19104
dparkes@unagi.cis.upenn.edu; ungar@central.upenn.edu

Abstract

Iterative auctions have many computational advantages over sealed-bid auctions, but can present new possibilities for strategic manipulation. We propose a two-stage technique to make iterative auctions that compute optimal allocations with myopic best-response bidding strategies more robust to manipulation. First, introduce proxy bidding agents to constrain bidding strategies to (possibly untruthful) myopic best-response. Second, after the auction terminates adjust the prices towards those given in the Vickrey auction, a sealed-bid auction in which truth-revelation is optimal. We present an application of this methodology to *i*Bundle, an iterative combinatorial auction which gives optimal allocations for myopic best-response agents.

Introduction

Many interesting problems involving distributed agents, e.g. task assignment, distributed scheduling, etc. can be formulated as resource allocation problems, with a set of discrete items to allocate to agents (Clearwater 1996). A common goal is to maximize the total value of the allocation over all agents, while respecting information decentralization, autonomy, and the self-interest of individual agents within a system. Auctions provide simple and robust mechanisms, and can compute optimal or near-optimal solutions in interesting problems (Wellman *et al.* 1999).

Iterative auctions, in which agents can bid continuously during an auction as prices are adjusted, have a number of computational advantages over sealed-bid auctions, in which agents must submit bids simultaneously in a single round. Agents can perform incremental computation about their values for different allocations as prices change (Parkes 1999a), and make new bids in response to bids from other agents. This is important in problems with hard valuation problems, consider for example a task allocation problem with agents that solve local optimization problems to compute the cost of performing additional task given existing commitments (Sandholm 1993).

Iterative auctions have been designed to solve non-trivial resource allocation problems, for example for auctions in multiple identical items (Ausubel 1997), and *i*Bundle (Parkes 1999b) for the combinatorial resource allocation problem.

However, iterative auctions present possibilities for strategic manipulation because information is exchanged between agents via bids and prices during an auction. A rational agent with look-ahead can try to manipulate the bids of other agents and the outcome of an auction, for example with *jump bids* at prices above the current ask price, or by delaying bids until the auction is about to close. Manipulation is undesirable because it reduces the economic efficiency of outcomes, and because it is inherently complex.

We propose a new method, "proxy agents and price-adjustment", to prevent strategic manipulation in iterative auctions. The method applies to iterative auctions that compute optimal resource allocations in *competitive equilibrium*.[1] We adjust prices retrospectively *after* an auction terminates towards prices that provide incentives for agents to bid truthfully.

The goal is to compute the prices that agents would pay in the Generalized Vickrey Auction (GVA) (Varian & MacKie-Mason 1995), a sealed-bid auction for combinatorial resource allocation problems. The prices in the GVA provide strong truth-revelation properties; truth-revelation is a *dominant strategy*, optimal for a self-interested agent for all strategies of other agents. When successful, in combination with *proxy bidding agents*, the iterative auction retains its computational advantages and inherits the strategy-proofness of the GVA. The proxy agents bid on behalf of agents, and constrain bidding strategies to best-response to prices based on (possibly untruthful) information received from agents about their values for items.

Our insight is that an interpretation of iterative auctions within *primal-dual* optimization theory presents a method, `Adjust`, to compute *minimal* competitive equilibrium prices after an auction terminates, based on bids placed by agents during the auction, i.e. prices that minimize the auctioneer's revenue in equilibrium. Extending recent results in Bikchandani & Ostroy (1998), we prove that GVA prices can always be computed from "enough" minimal CE prices. A variation, `Adjust*`, on `Adjust` closes the gap between minimal CE prices and GVA prices. We characterize necessary and sufficient conditions on agents' bids and prices for `Adjust*` to compute GVA prices, and propose a dynamic test allows an auctioneer to detect when the auction

[1] In competitive equilibrium all agents maximize utility with the final allocation given the final prices, and the auctioneer maximizes revenue.

has terminated with GVA prices.

We also suggest approximate procedures, `Adj-Pivot` and `Adj-Pivot*`, for `Adjust` and `Adjust*` with negligible computation that work well in practice. The methods leverage computation already performed by the auctioneer during the auction, in solving a sequence of winner-determination problems.

As an application of our framework, we consider *i*Bundle, an ascending-price combinatorial auction which gives optimal allocations for myopically-rational agents. *i*Bundle and `Adjust` compute minimal CE prices in all problems. We characterize sufficient conditions on agents' valuation functions for `Adjust*` to compute GVA prices. Experimental results verify that *i*Bundle with price-adjustment computes minimal CE prices across a suite of hard problems, and often compute prices which are within 2% of GVA prices.

Incentive Compatible Auctions

In this section, we explain why truth-revelation is optimal for an agent in the Generalized Vickrey Auction (GVA), and discuss the consequences of achieving Vickrey prices in an iterative auction.

The GVA computes optimal resource allocations even with strategic self-interested agents.[2] It is an incentive compatible auction: an agent's optimal bidding strategy is truth-revelation, i.e. *bid the exact amount that it values an item, or bundle of items.* The GVA extends Vickrey's (1961) seminal second-price sealed-bid auction, which sells a single item to the highest bidder for the second-highest price, to auctions for *bundles* of items.

Let G denote the set of items to be auctioned, I denote the set of agents, and $v_i(S)$ denote agent i's value for bundle $S \subseteq G$ of items. We assume risk-neutral agents with quasi-linear utilities in money, $u_i(S,p) = v_i(S) - p$, for price p, and equate optimal strategies with utility-maximization.

The GVA is a *direct-revelation mechanism*, in which agents report (possibly untruthful) values for bundles of items. Let $\hat{v}_i$ denote agent i's reported value, not necessarily equal to its true value. The auctioneer computes the allocation $\mathbf{S}^* = (S_1^*, \ldots, S_{|I|}^*)$ that maximizes the total reported value, where agent i receives bundle $S_i^* \subseteq G$.

Agent i pays $p_{\mathrm{gva}}(i) = \sum_{j \neq i} \hat{v}_j(S_j^{-i}) - \sum_{j \neq i} \hat{v}_j(S_j^*)$, where $\mathbf{S}^{-i}$ is the revenue-maximizing allocation with the bids from all agents except agent i. The GVA prices the marginal negative effect that an agent's presence has on the reported value of the outcome to the other agents.

Definition 1. *Dominant strategy. A bidding strategy is dominant if it is optimal for all bidding strategies of other agents.*

Truth-revelation, i.e. a bid $\hat{v}_i = v_i$, is a dominant strategy in the GVA. The proof is straightforward: agent i's utility, $u_i(S_i^*, p_{\mathrm{gva}}(i))$, given allocation S_i^* and price $p_{\mathrm{gva}}(i)$, is $u_i(S_i^*, p_{\mathrm{gva}}(i)) = v_i(S_i^*) - p_{\mathrm{gva}}(i) = v_i(S_i^*) + \sum_{j \neq i} \hat{v}_j(S_j^*) - \sum_{j \neq i} \hat{v}_j(S_j^{-i})$. Agent i can maximize the sum of the first two terms by reporting $\hat{v}_i = v_i$ because this is precisely the objective function that the auctioneer maximizes to select allocation $\mathbf{S}^*$. The final term is independent of agent i's bid.

We will refer to this outcome, i.e. allocation, $\mathbf{S}^*$ and payments $p_{\mathrm{gva}}(i)$, as the *Vickrey outcome.*

Vickrey Prices in an Iterative Auction

One might think that if an iterative auction implements the Vickrey outcome with agents that follow myopic best-response bidding strategies, then myopic best-response would be a dominant strategy for self-interested agents. In fact, manipulation remains possible with a non best-response strategy.

Definition 2. *Myopic best-response bidding strategy. Bid to maximize utility in the current round, taking prices as fixed.*

Definition 3. *Auction $\mathcal{A}$ myopically implements the Vickrey outcome if the auction terminates with the Vickrey outcome for agents that follow myopic best-response bidding strategies.*

Let $BR(v_i, p)$ denote the best-response bid for agent i with valuation function v_i, where p is the current prices in the auction. Call this a *truthful myopic strategy*. Also, let $BR(\hat{v}_i, p)$ denote an *untruthful myopic bidding strategy* for agent i, for some valuation function $\hat{v}_i \neq v_i$.

We derive the following result, for agents that are constrained to (possibly untruthful) myopic best-response bidding strategies. It is immediate from the incentive properties of the GVA:

Theorem 1. *Truthful myopic bidding is a dominant strategy in an iterative auction $\mathcal{A}$ that myopically-implements the Vickrey outcome, if all agents are constrained to following a (possibly untruthful) myopic best-response bidding strategy.*

That is, assume agent i must place bids in every round of the auction that are consistent with a myopic best-response bidding strategy, $BR(\hat{v}_i, p)$, for some valuation function $\hat{v}_i$, that does not need to equal the agent's actual valuation v_i. Given this, truth-revelation, i.e. following a best-response strategy for $\hat{v}_i = v_i$, is optimal.

This is weaker than the strategy-proofness of the GVA, where truthful bidding is dominant in a system with *unrestricted* bidding strategies. Gul & Stacchetti (1997) prove the following more general result:

Theorem 2. *Truthful myopic bidding is a sequentially rational best-response to truthful myopic bidding by other agents in an iterative auction $\mathcal{A}$ that myopically-implements the Vickrey outcome.*

We use proxy bidding agents to force agents to follow best-response bidding strategies, and leverage Theorem 1. With this, an iterative auction that myopically implements the Vickrey outcome inherits the incentive compatibility of the GVA.

[2] The GVA is not robust to manipulation by colluding agents (Sandholm 1996). Similarly, the methods that we present in this paper do not prevent collusive manipulation of iterative auctions.

Proxy Bidding Agents

We introduce semi-autonomous proxy bidding agents at the auctioneer, through which agents must interact with the auction. The proxy agents constrain agents' bidding strategies, following a best-response bidding strategy based on reported information about an agent's valuation function.

Let us first suggest (and reject) a couple of undesirable approaches to constrain agent strategies:

1. *Detect and penalize deviations from a myopic best-response strategy.* This is computationally expensive because the class of bidding strategies implemented by $BR(v, p)$ is large, and to detect an invalid strategy we must prove that no best-response strategy from this class can implement an agent's bids.
2. *Autonomous proxy bidding agents.* Proxy agents that must receive valuation functions in an initial stage, before bidding *autonomously* in the auction convert the iterative auction into a sealed-bid auction. This destroys many of the computational advantages that we outlined in the introduction.

Agent i provides incomplete information, $\hat{v}_{\mathrm{app},i}$ about reported value, $\hat{v}_i$, to its proxy agent. The reported value can be different from an agent's true value. Agent i can update the information $\hat{v}_{\mathrm{app},i}$ during the auction, but all new information must be consistent with previous information. The proxy agents must always have enough information to place best-response bids to the current prices in the auction.

With proxy bidding agents we have the following result, from Theorem 1:

Theorem 3. *Introducing myopic best-response proxy agents to auction $\mathcal{A}$ that myopically-implements the Vickrey outcome creates auction $Proxy(\mathcal{A})$, where truth-revelation is a dominant strategy.*

This solution retains the computational advantages of iterative auctions because agents do not need to provide complete value information up-front. If valuation functions are large and complex the proxy agents can be implemented at the client in a secure "wrapper".

Example: Single item auction. As an example, here is proxy bidding-agent variation on the English auction, in which the item is sold to the highest bidder for its bid. The new derivative auction is a "staged Vickrey auction". It is strategically equivalent to the standard Vickrey auction, but preferable because the optimal outcome is determined without complete information about all agents' values.

Agent i has a proxy agent that maintains a lower and upper bound, $\overline{\hat{v}}_i$ and $\underline{\hat{v}}_i$, on agent i's (possibly untruthful) value $\hat{v}_i$ for the item. When the ask price is below the lower bound the proxy agent will bid. When the ask price is above the upper bound the proxy agent will leave the auction. When the price is between the bounds the proxy places no bid, and asks the agent for new bounds that must be consistent with previous bounds, i.e. tighter. The English auction terminates with the Vickrey price if agents follow truthful best-response bidding strategies. Hence, by Theorem 3, it is a dominant strategy for agents to provide the proxy agents with true lower and upper bounds on value.

Adjusting Towards Vickrey Prices

Now onto the second step of our design paradigm, "price-adjustment". We present a method to adjust the final prices in an iterative auction towards the Vickrey prices after an auction terminates. The method is applicable to auctions that terminate in competitive equilibrium (CE), such that the allocation maximizes the utility of all agents at the final prices and the auctioneer maximizes its revenue. The *i*Bundle and English auctions terminate in CE. Indeed, a fundamental connection between primal-dual optimization theory and competitive equilibrium prices allows optimal auctions to be designed and analyzed (Bertsekas 1990; Parkes & Ungar 2000).

We introduce `Adjust`, a procedure to compute *minimal* CE prices from agents' bids after an auction terminates. Minimal CE prices are equilibrium prices that minimize the auctioneer's total revenue from all agents in the optimal allocation. The price paid by each agent with minimal CE prices is always an upper-bound on GVA prices, and equal to GVA prices when certain conditions hold on agents' values for bundles (Bikchandani & Ostroy 1998).

In fact, it is always possible to compute GVA prices with "enough" minimal CE prices (they are typically not unique), as the minimum price for each agent over all CE prices. We propose a slight variation on `Adjust`, `Adjust*`, and prove necessary and sufficient conditions on agents' bids and prices for `Adjust*` to compute GVA prices. Finally, we introduce approximate procedures `Adj-Pivot` and `Adj-Pivot*` to adjust prices.

For the rest of the paper we assume that agents follow myopic best-response bidding strategies.

Minimal Competitive Equilibrium Prices

We can interpret equilibrium conditions within primal-dual optimization theory (Papadimitriou & Steiglitz 1982). This provides the key to compute minimal CE prices from agents' bids and prices after an auction terminates. Complementary slackness conditions for appropriate primal and dual formulations of the global resource allocation problem are equivalent to equilibrium conditions between an allocation and prices (Bertsekas 1990; Parkes & Ungar 2000).

Consider an auction $\mathcal{A}$ that terminates in equilibrium, let $p_i(S)$ denote the price for bundle S to agent i, and let S_i^* denote the bundle allocated to agent i. In defining a competitive equilibrium we allow price discrimination, with different prices for agents, e.g. $p_i(S) \neq p_j(S)$ for some $i \neq j$ and some bundle S. This is the most general case. In competitive equilibrium the prices and allocation must satisfy the following CS conditions:

(CS-1) Given prices $p_i(S)$, allocation S_i^* maximizes agent i's utility, $u_i(S_i^*, p_i(S_i^*)) = v_i(S_i^*) - p_i(S_i^*) = \max_S \{v_i(S) - p_i(S)\}$.

(CS-2) Given prices $p_i(S)$, allocation $\mathbf{S}^* = (S_1^*, \ldots, S_{|I|}^*)$ maximizes the auctioneer's revenue over all *feasible* allocations.

A feasible allocation sells each item to at most one agent, and allocates at most one bundle to each agent.

The following result follows immediately from strong duality and the *complementary slackness theorem* (Papadimitriou & Steiglitz 1982) of linear programming:

Theorem 4. *In an auction that terminates in competitive equilibrium, minimal prices that satisfy complementary slackness with the final allocation are minimal competitive equilibrium prices.*

This allows the computation of minimal CE prices after an auction terminates, based on bids placed by agents. Reduce prices while: (CS-1) agents continue to maximize utility with allocation S_i^*; (CS-2) allocation $\mathbf{S}^*$ continues to maximize revenue.

Adjust. Procedure `Adjust` computes minimal CE prices from agents' bids when an auction terminates in competitive equilibrium. Assume that agents place *exclusive-or* (XOR) bids, such that they demand at most one bundle.[3] Let I^* denote the set of agents in the optimal allocation, $\hat{P}$ denote agents' prices (initialized to $p_i(S)$), and V^* denote the revenue of the final allocation. We will compute the values of **second-best allocations**. An allocation S^{-i} is a *second-best allocation* if it maximizes revenue for the auctioneer without allocating a bundle to agent i, i.e. it is the second-best allocation without agent i. Let $V^{-i}(\hat{P})$ denote the revenue from this allocation, computed at prices $\hat{P}$.

`Adjust` computes a price discount Δ_i to each agent i in the final allocation, such that agent i receives final price $\underline{p}_i(S_i^*) = p_i(S_i^*) - \Delta_i$.

`Adjust:`
`for each` $i \in I^*$ {
  $\Delta_i = \min\{V^* - V^{-i}(\hat{P}),\ \ p_i(S_i^*)\}$;
  $V^* = V^* - \Delta_i$;
  $\hat{P}_i = \max\{\hat{P}_i - \Delta_i,\ \ 0\}$;[4] }

`Adjust` selects each agent in the final allocation in turn, reducing its price for every bundle by the amount that the value of the optimal allocation exceeds the value of the best allocation without that agent.[5] The maximization problem, to solve $V^{-i}(\hat{P})$ in each iteration, is $\mathcal{NP}$-hard (Rothkopf, Pekeč, & Harstad 1998) in bundle auctions.[6] Later we introduce an efficient approximate procedure `Adj-Pivot`.

Note that price reductions to each agent in the allocation are considered incrementally and not independently, prices $\hat{P}$ are adjusted according to Δ_i before reducing prices to agent j.

[3] This is without loss of generality because XOR is a completely expressive bid language. The procedure can be extended to other bid languages, e.g. additive-or bids through the introduction of a dummy agent for each price bid.

[4] Operation $\hat{P}_i = \max\{\hat{P}_i - \Delta_i, 0\}$ indicates that price $p_i(S)$ to agent i is reduced to $\max\{p_i(S) - \Delta_i, 0\}$.

[5] A simple optimization is possible. If $\Delta_i \leq p_i(S_i^*)$ and agent $j > i$ is not in the revenue-maximizing allocation without agent i then $\Delta_j = 0$.

[6] It can be solved in average-case polynomial time in some hard problems with efficient search algorithms; see Sandholm (1999) for example.

Proposition. *Procedure* `Adjust` *maintains* CE *prices.*

Proof. `Adjust` maintains (CS-1). Prices to agents not in the allocation are left unchanged. Agent i in allocation $\mathbf{S}^*$ continues to maximize utility with bundle S_i^* at new prices $p_i(S) - \Delta_i$; its price is reduced by Δ_i on bundle S_i^*, and by Δ_i or less on all other bundles. By the lemma, `Adjust` maintains (CS-2) because it explicitly computes the maximum value of all allocations without agent i, and reduces agent i's prices by no more than the difference between V^* and this value. □

Lemma. *An allocation with more revenue to the auctioneer than S^* as prices are reduced to agent i must exclude agent i, since all prices to agent i are reduced by the same amount (or until they are zero).*

We derive a sufficient condition on agents' bids and prices for `Adjust` to compute minimal CE prices.

Assumption A. *(i) Every agent j in allocation S^* bids at price $p_j(S_j^{-i})$ for bundles allocated in all second-best allocations S^{-i};* ***and*** *(ii) Every agent j not in allocation S^* bids at price $p_j(S_j^{-i}) = v_j(S_j^{-i})$ for bundles allocated in all second-best allocations S^{-i}.*

Intuitively, when Assumption A holds, no bundles in second-best allocations are priced too high. If agent j receives bundle S_j^{-i} in a second-best allocation, it had better have bid the price of that bundle, else the price can be reduced (maintaining (CS-1)). In turn, this can allow agent i to pay a lower price but still maximize revenue with the final allocation S_i^*.

Theorem 5. *Procedure* `Adjust` *computes* ***minimal*** CE *prices if agents' bids and prices satisfy Assumption A.*

Proof. By contradiction. Assume that prices $\underline{p}_i(S_i^*)$ computed in `Adjust` are not minimal and Assumption A holds. If the prices are not minimal, then it must be possible to reduce the price $\underline{p}_j(S_j^*)$ to some agent, j, and still maintain (CS-1) and (CS-2). Therefore, there are some prices to agents $i \neq j$ that reduce the value $V^{-j}(\hat{P})$ of the second-best allocation without agent j, so that the price $\underline{p}_j(S_j^*)$ can be reduced without violating (CS-2).

However, Assumption A (i), any decrease in the price of bundle S_k^{-j} to some agent k in the optimal allocation and second-best allocation S^{-j} must be mirrored in a decrease in the price of S_k^* to maintain (CS-1); and (ii), any decrease in the price of bundle S_k^{-j} to some agent k not in the optimal allocation but in the second-best allocation S^{-j} violates (CS-1) because the agent has positive utility for that bundle but receives $S_k^* = \emptyset$. □

Computing GVA Prices

In fact, it is always possible to compute GVA prices from "enough" minimal CE prices. Minimal CE prices are often not unique, the same *total* revenue to the auctioneer can be achieved with different distributions of revenue across agents. We use this result to derive procedure `Adjust*` and to prove necessary and sufficient conditions for computing GVA prices in an auction. Let $\underline{p}_i(S_i^*)$ denote a minimal CE price to agent i for bundle S_i^*.

Theorem 6. *For agent j in the optimal allocation, the minimal price $\min \underline{p}_j(S_j^*)$ over all minimal CE prices $\underline{p}_j(S_j^*)$ equals the GVA price.*

Proof. The proof is constructive, using `Adjust` with alternative orders for selecting agents $i \in I^*$. First, observe that $p_i(S) = v_i(S)$ trivially satisfy (CS-1), and also Assumption A with best-response agents. Hence, `Adjust` will compute minimal CE prices by Theorem 5. Now, let j denote the first agent selected in `Adjust`. $\Delta_j = \min\{V^* - V_{-j}(\hat{P}), p_j(S_j^*)\} = \min\{\sum_i v_i(S_i^*) - \sum_{i \neq j} v_i(S_i^{-j}), v_j(S_j^*)\} = \sum_i v_i(S_i^*) - \sum_{i \neq j} v_i(S_i^{-j})$. Hence, $\underline{p}_j(S_j^*) = v_j(S_j^*) - \Delta_j = \sum_{i \neq j} v_i(S_i^{-j}) - \sum_{i \neq j} v_i(S_i^*) = p_{\text{gva}}(j)$. Therefore, the price $\underline{p}_j(S_j^*)$ for bundle S_j^* to agent j equals its GVA price in at least when agent $j \in I^*$ is selected first in `Adjust`. Finally, $p_{\text{gva}}(j) = p_{\min,i}(S_j^*) = \min \underline{p}_j(S_j^*)$, over all minimal CE prices. □

Adjust*. This leads to procedure `Adjust*`, a slight variation on `Adjust` that computes price discounts for each agent independently:

`Adjust*:`
`for each` $i \in I^*$
$\quad \Delta_i = \min\{V^* - V^{-i}(\hat{P}),\ \ p_i(S_i^*)\}$;

Although adjusted prices $\hat{p}_i(S_i^*) = p_i(S_i^*) - \Delta_i$ may not be CE prices, the prices are strictly closer to GVA prices. Assumption B characterizes conditions on agents' bids and prices that, together with Assumption A, are necessary and sufficient for `Adjust*` to compute GVA prices after an auction terminates.

Assumption B. *When there is more than one agent in the optimal allocation, an agent j in the optimal allocation but not in a second-best allocation S^{-i} for some agent $i \neq j$ bids $p_j(S_j^*) = v_j(S_j^*)$ for the bundle S_j^* it receives in the optimal allocation.*

In other words, every agent in the optimal allocation must bid its value for the bundle that it receives, unless it remains in the revenue-maximizing allocations as bids from the other agents in the optimal allocation are ignored in turn.

Here is some intuition for the rule. Consider two agents, 1 and 2, that receive a bundle in the final allocation, and suppose that agent 2 bids less than its value for its bundle S_2^* in the allocation. Suppose, in addition, that bids from agents 3 and 4 maximize revenue in the second-best allocation as agent 1's prices are reduced. Agent 1's prices can be reduced further and still achieve more revenue than the bids from agents 3 and 4 if agent 2 bids more for bundle S_2^*. In procedure `Adjust` to compute minimal CE prices this effect is neutral because the price decrease is received in only a single agent, but in `Adjust*` the price decrease is received by all agents in the optimal allocation.

Theorem 7. *Assumptions A and B are necessary and sufficient conditions on agents' bids and prices for `Adjust*` to compute GVA prices.*

Proof. [Sufficient.] The proof follows from Theorem 6, show that Assumptions A and B imply that `Adjust*` computes the same price to each agent in the optimal allocation as when the agents bid at prices $p_i(S) = v_i(S)$.

[Necessary.] By contradiction. (Case 1) Assume GVA prices and *not* Assumption A. Consider agent j in allocation S^* that does not bid at price $p_j(S_j^{-i})$ for a bundle S_j^{-i} that it receives in second-best allocation without an agent $i \neq j$. Now, agent i can receive a larger discount by reducing the price $p_j(S_j^{-i})$ to agent j, still maintaining (CS-1) for agent j. Similarly for an agent j not in allocation S^* that does not bid at price $p_j(S_j^{-i}) = v_j(S_j^{-i})$ for a bundle S_j^{-i} that it receives in second-best allocation without agent $i \neq j$. The proof of (Case 2), assuming GVA and *not* Assumption B is similar, consider an agent j in the optimal allocation that is not in some second-best allocation and does not bid $p_j(S_j^*) = v_j(S_j^*)$ for its optimal allocation. □

This leads to a test that allows an auctioneer to determine whether `Adjust*` computes GVA prices.The Vickrey-Test is sufficient but not necessary for GVA prices. [7]

Vickrey-Test. *Procedure `Adjust*` computes GVA prices if agents' bids and prices satisfy: (1) all second-best allocations can be computed from agents' bids; (2) every agent in the optimal allocation is in every second-best allocation if there is more than one agent in the optimal allocation.*

Property (1) implies Assumption A, and Property (2) implies Assumption B. Assumption B also holds if agents in the optimal allocation bid $p_j(S_j^*) = v_j(S_j^*)$, but there is no easy way for the auctioneer to detect this.

Example: Computing GVA Prices Consider a problem with three agents, $I = \{1, 2, 3\}$ and two items, $G = \{A, B\}$. The agents have the following values for bundles: $v_1 = \{30, 0, 30\}$, $v_2 = \{0, 40, 40\}$ and $v_3 = \{0, 20, 40\}$, for bundles A, B, and AB. The optimal allocation is $\mathbf{S}^* = (A, B, \emptyset)$, i.e. with items are allocated to agents 1 and 2. The Vickrey prices are $p_{\text{gva},1} = 40 - 40 = 0$ and $p_{\text{gva},2} = 50 - 30 = 20$. We consider adjusting prices in two scenarios. In both cases initial prices are competitive equilibrium prices, and best-response bids satisfy Assumption A with the prices. `Adjust` computes minimal CE prices in both scenarios, while `Adjust*` computes GVA prices in Scenario 2.

(Scenario 1) Prices are $p_1 = \{25, 0, 25\}$, $p_2 = \{0, 25, 25\}$ and $p_3 = \{0, 20, 40\}$. `Adjust` computes minimal CE prices: $\underline{p}_1(A) = 25 - (50 - 40) = 15$ and $\underline{p}_2 = 25 - (40 - 40) = 25$; or $\underline{p}_2(B) = 25 - (50 - 45) = 20$ and $\underline{p}_1(A) = 25 - (45 - 40) = 20$. The result depends on which agent is selected first. `Adjust*` computes $\hat{p}_1(A) = 15$ and $\hat{p}_2(B) = 20$. Agent 2 pays its GVA price because agent 1 is

[7] Furthermore, GVA prices are approximately computed when an agent in the optimal allocation "almost" bids for a bundle in a second-best allocation, or is "almost" in every second-best allocation.

in the second-best allocation without bids from agent 2, but agent 1 pays above its GVA price.

(Scenario 2) Now, assume prices to agent 2 are $p_2 = \{0, 40, 40\}$. The prices and agents' best-response bids now satisfy Assumption B, because agent 2 bids its value $p_2(B) = v_2(B)$ for item 2. In this case `Adjust` computes: $\underline{p}_1(A) = 0$ and $\underline{p}_2(B) = 40$, or $\underline{p}_2(B) = 20$ and $\underline{p}_1(A) = 20$. `Adjust*` computes $\hat{p}_1(A) = 0$ and $\hat{p}_2(B) = 20$, equal to GVA prices.

A Fast and Approximate Method

Procedure `Adj-Pivot` is a fast approximation to `Adjust`, that leverages computation already performed by the auctioneer to solve the winner-determination problem in each round of the auction. Experimental results show that it works well in practice.

`Adj-Pivot` uses an approximate formulation of `Adjust` as a linear program, where the value of $V^{-i}(\hat{P})$ is computed as the maximum value over all provisional allocations during the auction. These are *pivotal* allocations, likely to represent allocations with high value. `Adj-Pivot` computes $\max \sum_i \Delta_i$ such that $\hat{p}_i(S) = \max\{0, p_i(S) - \Delta_i\}$ for all agents, $\Delta_i = 0$ for agents not in the optimal allocation, and the revenue from the optimal allocation maximizes revenue over the set of pivotal allocations at prices $\hat{p}_i(S)$.

Similarly, `Adj-Pivot*` approximates `Adjust`. The price discount Δ_i is computed for each agent independently: compute $\max \Delta_j$ such that $\hat{p}_j(S) = \max\{0, p_j(S) - \Delta_j\}$, and the revenue from the optimal allocation maximizes revenue over all pivotal allocations.

Preprocessing.

As described, the price adjust procedures compute adjusted prices from individual prices to each agent. In an auction without price discrimination, in which each bundles are priced the same to all agents, the first step is to construct prices for each agent. Simply replicate the prices, i.e. $p_i(S) = p(S)$. Preprocessing can then be optionally applied, to adjust prices towards prices that satisfy Assumption A, such that agent i would bid for all bundles with a positive price. To give a simple example, we can reduce prices to an agent not in the final allocation to the prices in the last round in which the agent placed bids.

Application: *i*Bundle

*i*Bundle (Parkes 1999b; Parkes & Ungar 2000) is an ascending-price combinatorial auction in which agents can bid directly for bundles of items. It generalizes the English auction to the combinatorial resource allocation problem. Bundles are priced explicitly, and prices are increased whenever agents' bids are unsuccessful at the current prices. The auctioneer selects a provisional allocation in each round of the auction to maximize revenue, given the bids received.

*i*Bundle computes optimal resource allocations, and terminates in competitive equilibrium, with agents that follow myopic best-response bidding strategies, i.e. bid for all bundles that maximize utility in each round given the prices.

We present an application of the price-adjustment technique to variation *i*Bundle(*3*) that maintains price discrimination throughout the auction.[8] It is trivial to prove that *i*Bundle(*3*) terminates with bids and prices that satisfy Assumption A, because agents bid for all priced bundles. By Theorem 5, *i*Bundle with `Adjust` computes minimal CE prices.

Theorem 8. i*Bundle(*3*) with* `Adjust` *and myopic best-response proxy agents computes the minimal* CE *prices in combinatorial resource allocation problems.*

We have the following key result, that follows from Theorems 3 and 7.

Theorem 9. i*Bundle(*3*) with* `Adjust*` *and myopic best-response proxy agents is incentive-compatible and allocatively-efficient in combinatorial resource allocation problems in which Assumption B holds when the auction terminates.*

The *Vickrey-test* allows an auctioneer to be sure that *i*Bundle computes GVA prices. In addition, we can characterize properties on agents' valuation functions $v_i(S)$ in which Assumption B will hold. As an example, Assumption B holds in these problems: in the assignment problem with unit-demands; with multiple identical items and subadditive valuation functions (i.e. decreasing returns); and in problems with linear-additive valuation functions in items. In all of these problems agents in the optimal allocation will remain in all second-best allocations.

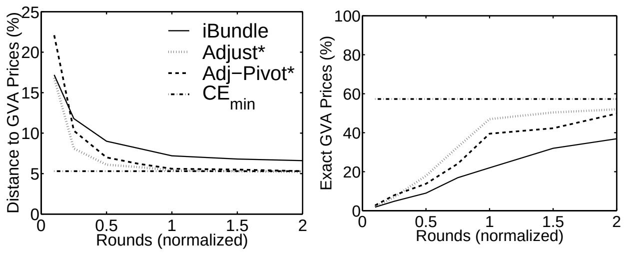
Experimental Results

We present experimental results for *i*Bundle(*3*) with `Adjust*` and `Adj-Pivot*`, comparing its performance with the GVA in a number of hard problems. The problems are PS 1–12 from (Parkes 1999b), and also problems Decay, Weighted-random (WR), Random and Uniform in Sandholm (1999). Each problem set defines a distribution over agents' values for bundles of items. Implementation details for *i*Bundle, e.g. the algorithm for winner-determination in each round, are as described in (Parkes & Ungar 2000). A standard Simplex algorithm computes adjusted prices with `Adj-Pivot*`.

The distance $\mathcal{D}(p_i(S_i^*), p_{\text{gva}}(i))$ between prices $p_i(S_i^*)$ and GVA prices is measured with an L_1 norm, as $\sum_i |p_i(S_i) - p_{\text{gva}}(i)| / \sum_i v_i(S_i)$, i.e. the sum absolute difference between the price charged to each agent and its GVA price normalized by the total value of the allocation over all agents.[9] We compute the average distance over problem instances in which *i*Bundle computes the optimal allocation, which approaches 100% of problems as the bid increment gets small. It is not clear how to measure distance to GVA

[8] *i*Bundle auction has three variations, that differ in their price-update rules. In this paper we use *i*Bundle both to refer to the family of auctions in general, and also to variation *i*Bundle(*3*) in particular.

[9] An L_1 norm is appropriate because minimal CE prices is computed with a linear additive measure over the auctioneer's price to each agent in the allocation.



(a) Distance. (b) Fraction Correct.

Figure 1: Average performance of *i*Bundle with price-adjustment `Adjust*` and `Pri-Adjust*` in problems PS 1–12. The number of rounds to termination is varied by adjusting the minimal bid increment.

prices in problems in which the auction's allocation is suboptimal and different from the GVA.

Figure 1 plots the distance to the GVA prices in *i*Bundle, before and after price-adjustment using `Adjust*` and `Adj-Pivot*`, averaged over 25 trials each of problems PS 1–12. We ran *i*Bundle with different bid increments to vary the number of rounds to termination, and average performance across problem sets by normalizing the number of rounds to termination according to the minimal number of rounds in which *i*Bundle achieves 100% allocative efficiency. For comparison, we also plot the performance of minimal CE prices.

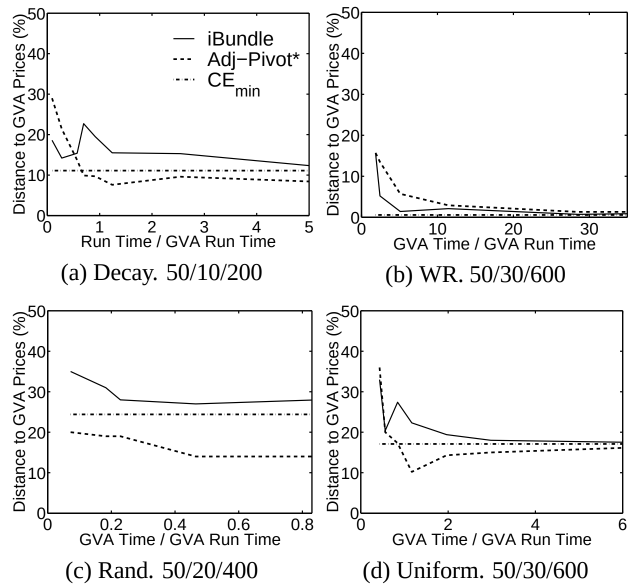
The average distance between minimal CE prices and GVA prices across these problems is 5.3%. For small bid increments *i*Bundle computes prices to within 6.5% of the GVA prices, with `Adjust` to within 5.5% (not plotted), and with `Adjust*` and `Adj-Pivot*` to within 5.2%. Notice that the prices continue to adjust towards the min CE prices for bid increments smaller than those required for 100% allocative efficiency, corresponding to normalized rounds to termination > 1.

We also compute the fraction of all problems in which $\mathcal{D}(p_i, p_{\text{gva}}(i)) < 2\%$, to test the proportion of problems in which prices are approximately Vickrey. CE prices are equal to GVA prices in approximately 57% of problem instances. *i*Bundle computes GVA prices in around 38% of problem instances, compared to approaching 57% with `Adjust*` and `Adj-Pivot*`. Clearly, the results verify that `Adjust*` computes minimal CE prices when Assumption A holds, as it will in *i*Bundle.

The minimal CE prices are close to GVA prices (average distance $< 2.5\%$) in problems 4–8, in which the agents in the optimal allocation also tend to be in the second-best allocations. In contrast, the minimal CE prices differ from the GVA payments by more than 5% in problems PS 1, 3, 9, 11 and 12, which are characterized by optimal allocations that are very different from second-best allocations, and agents with complementary demands for bundles.

As expected, an application of the Vickrey-Test over all problems confirmed no false positives, a specificity of 100%, but some false negatives, a sensitivity of 56%. The outcome was always approximately Vickrey when indicated by the Vickrey-Test, but Vickrey-outcomes went undetected in some problems.

It is noteworthy that the approximate method `Adj-Pivot*` is as effective as `Adjust*` for small bid increments. We use `Adj-Pivot*` in the harder problems plotted in Figure 2.



(a) Decay. 50/10/200 (b) WR. 50/30/600

(c) Rand. 50/20/400 (d) Uniform. 50/30/600

Figure 2: Performance of *i*Bundle with price-adjustment `Adj-Pivot*` problem sets from Sandholm (1999). The bid increment in *i*Bundle is adjusted to give different run times.

Figure 2 illustrates the performance of *i*Bundle with `Adj-Pivot*` in Decay, WR, Random, and Uniform, with problem sizes selected to give reasonable winner-determination computation times. In Decay we set Sandholm's α parameter to 0.85. We plot the distance to GVA prices against the relative run time of *i*Bundle with `Adj-Pivot*` to the time to compute winner-determination and agent prices in the GVA.[10] The minimal bid increment is varied to adjust the number of rounds in *i*Bundle, and with the values used allocative efficiency varies between 93% and 100%.

`Adj-Pivot*` computes prices closer to GVA prices than the minimal CE prices in Decay and Random, and minimal CE prices are equal to GVA prices in WR (where there is typically a single agent in the final allocation). Prices remain quite far from GVA prices in the Uniform problem set because second-best allocations are typically quite different from optimal allocations, and Assumption B often fails.

Related Work

There have been a number of recent proposals to achieve incentive-compatibility and allocative efficiency with less

[10]We do not focus on the auctioneer's winner-determination time in this paper, but note that T_{gva} is 362s, 9.1s, 1791s, and 138s (on a 450MHz Pentium) for problems (a – d), i.e. the run time for *i*Bundle in WR is small despite the considerable slow-down in comparison with the GVA.

computation than the GVA, focusing on sealed-bid auctions in special cases (Lehmann *et al.* 1999; Kfir-Dahav *et al.* 1998; Nisan & Ronen 1999).

For iterative auctions in particular, previous work has focused on careful control of prices during an auction, so that the auction terminates with GVA prices. Positive results exist only for special cases (Demange *et al.* 1986; Gul & Stacchetti 1997; Ausubel 1997). *i*Bundle(*3*) with `Adjust*` solves all of these problems because Assumption B holds with myopic best-response bids.

Bikchandani & Ostroy (1998) provides additional motivation and background for our work, formulating linear programs for combinatorial resource allocation problems and relating primal and dual solutions to competitive equilibrium outcomes. Wurman & Wellman (1999) provide useful background on equilibrium prices in bundle auctions.

Milgrom (1999) presents examples of strategic-manipulation in simultaneous ascending-price auctions on individual items, and identifies the search for strategy-proof iterative combinatorial auctions as an important open problem.

Conclusions

We have proposed a new method, "proxy agents and price adjustment", to make iterative auctions more robust to strategic manipulation. This is important given the computational advantages of iterative auctions over sealed-bid auctions for bidding agents, because of dynamic price-discovery coupled with incremental computation on agents' values for different items or bundles of items.

The method introduces proxy bidding agents and adjusts the final prices in an iterative auction towards Vickrey prices. We characterize necessary and sufficient conditions on agents' bids and prices to obtain dominant strategy truth-revelation without a sealed-bid auction, describe a dynamic test for an auctioneer to detect a Vickrey outcome, and relate the conditions to agents' valuation functions.

We proposed both an optimal procedure `Adjust*` and an approximate procedure `Adj-Pivot*` to reduce prices after the auction terminates. The `Adj-Pivot*` approximation is both fast and effective. An interesting open empirical problem is to understand the level of approximation to GVA prices that is "good enough" to prevent most opportunities for strategic manipulation.

Finally, this work suggests a method to design an iterative Generalized Vickrey Auction: keep the auction open until every agent in the optimal allocation is also in all revenue-maximizing allocations without bids from each agent in the optimal allocation, or bids its value. It might be useful to keep *i*Bundle open for longer, past the first round in which a competitive equilibrium outcome is computed, and increase the prices for bundles. Paradoxically, higher prices when *i*Bundle terminates will allow lower adjusted prices.

Acknowledgments

This research was funded in part by National Science Foundation Grant SBR 97-08965.

References

Ausubel, L. M. 1997. An efficient ascending-bid auction for multiple objects. Tech. report, University of Maryland.

Bertsekas, D. P. 1990. The auction algorithm for assignment and other network flow problems: A tutorial. *Interfaces* 20(4):133–149.

Bikchandani, S., and Ostroy, J. M. 1998. The package assignment model. Tech. report, Anderson School of Management and Department of Economics, UCLA.

Clearwater, S. H., ed. 1996. *Market-Based Control: A Paradigm for Distributed Resource Allocation*. World Scientific.

Demange, G.; Gale, D.; and Sotomayor, M. 1986. Multi-item auctions. *J. of Political Economy* 94(4):863–872.

Gul, F., and Stacchetti, E. 1997. English and double auctions with differentiated commodities. Tech. Report 97-02, University of Michigan.

Kfir-Dahav, N. E.; Monderer, D.; and Tennenholtz, M. 1998. Mechanism design for resource bounded agents. Tech. report, Technion.

Lehmann, D.; O'Callaghan, L.; and Shoham, Y. 1999. Truth revelation in rapid, approximately efficient combinatorial auctions. In *Proc. ACM Conf. on Electronic Commerce (EC-99)*.

Milgrom, P. 1999. Putting auction theory to work: The simultaneous ascending auction. *J. of Political Economy* 108. To appear.

Nisan, N., and Ronen, A. 1999. Algorithmic mechanism design (extended abstract). In *Proc. 31st Annual Symposium on Theory of Computing (STOC99)*.

Papadimitriou, C. H., and Steiglitz, K. 1982. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall.

Parkes, D. C., and Ungar, L. H. 2000. Iterative combinatorial auctions: Theory and practice. In *Proc. 18th National Conference on Artificial Intelligence (AAAI-00)*. To appear.

Parkes, D. C. 1999a. Optimal auction design for agents with hard valuation problems. In *Proc. IJCAI-99 Workshop on Agent Mediated Electronic Commerce*. Stockholm.

Parkes, D. C. 1999b. *i*Bundle: An efficient ascending price bundle auction. In *Proc. ACM Conf. on Electronic Commerce (EC-99)*.

Rothkopf, M. H.; Pekeč, A.; and Harstad, R. M. 1998. Computationally manageable combinatorial auctions. *Management Science* 44(8):1131–1147.

Sandholm, T. 1993. An implementation of the Contract Net Protocol based on marginal-cost calculations. In *Proc. 11th National Conference on Artificial Intelligence (AAAI-93)*, 256–262.

Sandholm, T. 1996. Limitations of the Vickrey auction in computational multiagent systems. In *Second International Conference on Multiagent Systems (ICMAS-96)*, 299–306.

Sandholm, T. 1999. An algorithm for optimal winner determination in combinatorial auctions. In *Proc. 16th International Joint Conference on Artificial Intelligence (IJCAI-99)*, 542–547.

Varian, H., and MacKie-Mason, J. K. 1995. Generalized Vickrey auctions. Tech. report, University of Michigan.

Vickrey, W. 1961. Counterspeculation, auctions, and competitive sealed tenders. *J. of Finance* 16:8–37.

Wellman, M. P.; Walsh, W. E.; Wurman, P. R.; and MacKie-Mason, J. K. 1999. Auction protocols for decentralized scheduling. *Games and Economic Behavior*. To appear.

Wurman, P. R., and Wellman, M. P. 1999. Equilibrium prices in bundle auctions. In *Proc. AAAI-99 Workshop on Artificial Intelligence for Electronic Commerce*, 56–61.