ITERATIVE COMBINATORIAL AUCTIONS: ACHIEVING ECONOMIC AND COMPUTATIONAL EFFICIENCY

David Christopher Parkes

A DISSERTATION PROPOSAL

in

Computer and Information Science

Presented to the Faculties of the University of Pennsylvania

in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy

2000

Lyle H. Ungar
Supervisor of Dissertation

Val Tannen
Graduate Group Chair

Acknowledgements

Thanks to my advisor, Professor Ungar, for his belief in my abilities, willingness to listen, and patience. Thanks to Dean Harker for his vision, encouragement and support. Thanks to my first tutor, Professor Bird, for providing early inspiration and a perfect introduction to the mathematical foundations of computer science. Thanks to my friends, and to my sister, for giving constant and unquestioning love and support. Finally, to my parents for allowing me to follow my dreams. This has been the most stimulating and rewarding time of my life.

Abstract

ITERATIVE COMBINATORIAL AUCTIONS:
ACHIEVING ECONOMIC AND COMPUTATIONAL EFFICIENCY

David Christopher Parkes

Supervisor: Lyle H. Ungar

This thesis presents new auction-based mechanisms to coordinate systems of self-interested and autonomous agents, and new methods to design such mechanisms and prove their optimality.

Computation is increasingly carried out on open networks, with self-interested programs ("agents"), competing to derive the most utility for users. In addition, new interconnectivity and distributed computing is changing business practices. It is now possible to implement dynamic mechanisms to trade goods and services, both business-to-consumer (B2C) and business-to-business (B2B), and remove the inefficiencies in traditional marketplaces. Auctions offer great promise as simple and robust dynamic mechanisms for efficient resource allocation. There are already on-line consumer auctions, and nascent auctions for B2B trade in the supply-chain. However the majority of on-line auctions remain simple variations on standard auctions.

My thesis is that it is necessary to take an explicitly computational approach to auction design. Auctions will be populated with automated bidding agents, and only auctions that are robust and present simple optimal bidding strategies will be successful. In addition, *combinatorial auctions* are important in many important applications, but present inherently hard computational problems for agents and for auctioneers.

A combinatorial auction allows agents to bid for bundles of items. Consider a manufacturer that needs either components A and B, or just component C; consider a mobile agent that needs an interval of compute time; consider a train that needs a bundle of

departure and arrival times on tracks across its route. Combinatorial auctions present two key computational problems: (1) the *winner-determination* problem, to compute a revenue-maximizing set of bids, is $\mathcal{NP}$-hard; (2) agents often have hard valuation problems, for example local optimization problems, to compute the value of different bundles.

I propose *i*Bundle, an *iterative* combinatorial auction, that is economically efficient under a reasonable assumption about agent bidding strategies, minimizes agent valuation work, and allows the auctioneer to introduce approximate solutions for winner-determination.

Iterative auctions allow agents to compute *incremental* values for items or bundles of items, in response to bids from other agents, and avoid valuation altogether on high priced bundles. In addition, the auctioneer can tradeoff economic efficiency with computational efficiency. Increasing the bid increment decreases the number of rounds and the number of winner-determination problems to solve. Approximate algorithms for winner-determination can be introduced while maintaining incentives for agents to follow the same bidding strategy.

*i*Bundle is the first iterative combinatorial auction that is optimal for a reasonable bidding strategy, in this case myopic best-response to current prices. Its optimality is proved through a fundamental connection with primal-dual optimization theory. *i*Bundle maintains feasible primal and dual solutions to the resource allocation problem, the allocation and prices in the auction. The strong duality theorem of linear-programming states that primal and dual solutions are optimal if and only if they satisfy *complementary slackness* conditions, which express logical constraints between their values. Best-response bids from agents in *i*Bundle provide enough information for the auctioneer to adjust prices so that the auction provably terminates with primal and dual solutions that satisfy complementary slackness conditions.

The primal-dual interpretation of *i*Bundle also suggests a method to boost the "strategy-proofness" of the auction, adjusting the price after *i*Bundle terminates towards prices that make myopic best-response an optimal strategy for rational self-interested agents. When successful, together with proxy bidding agents which constrain agent bidding strategies to (possibly untruthful) best-response bidding strategies, this "proxy and adjust" method makes *i*Bundle an iterative, optimal, and non-manipulable auction.

Contents

List of Tables

List of Figures

Preface

This dissertation proposal has a slightly novel form. The early chapters provide an extended introduction, and an overview of my work. Subsequent chapters review important literature in economic mechanism design, and introduce primal-dual optimization theory, which provides a road-map for the design of optimal iterative auctions in hard distributed problems.

Later chapters on *i*Bundle, and methods to prevent strategic-manipulation in iterative auctions, can be viewed as "extended abstracts" of a collection of papers that I attach as an Appendix. This document, taken with the attached papers, provides an accurate description of the work already completed towards my dissertation. In addition, at the end of the Introduction, I summarize my current contributions and describe the work that I propose to complete for my full dissertation.

Papers in Appendix:

1. David C. Parkes and Lyle H. Ungar. Bounded-rational compatible auctions. Under review, *Journal of Artificial Intelligence Research*, 2000.

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

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

4. David C. Parkes and Lyle H. Ungar. Preventing strategic manipulation in iterative auctions: Proxy agents and Price-adjustment. In *Proc. 18th National Conference on Artificial Intelligence (AAAI-00)*, 2000. To appear.

Chapter 1

Introduction

Computation is no longer restricted to the mainframe, or even to the PC; computation is increasingly ubiquitous, powerful processors are found in handheld computers, mobile phones, automobiles, even wrist watches. The web browser provides users on thin-clients with instant access to air schedules on all the Worlds' airlines, and powerful servers for search and query optimization. Mobile agents can migrate over networks of computers, searching for information, or simply consuming computational resources. Moreover, the engineering complexity of systems is decentralized to multiple developers and competing applications, networks are *open*, and programs ("agents") need not be cooperative. In fact, applications are more likely to be *self-interested*, competing to derive the most utility for users.

More recently, the Internet and electronic commerce has generated renewed interest in electronic marketplaces and auction systems, both as dynamic mechanisms to sell items to individuals and as systems for business-to-business transactions. Many retailers have online consumer auctions, e.g. `www.onsale.com`, and there are nascent auctions for procurement in the supply-chain, e.g. `www.freemarkets.com`. Forrester Research predicts that online business-to-business (B2B) trade in e-marketplaces, e.g. auctions and exchanges, will increase from \$54.7 billion to \$1,417 billion over the next 5 years, from 13.5% to 53% of B2B e-commerce trade, to form 10% of total B2B trade by 2004 (U.S. data). Electronic marketplaces will intermediate more B2B transactions and at finer levels than in current practice. However, at present the vast majority of online auctions are simple variations on the traditional English auction, an ascending-price single-item auction

Auctions provide useful mechanisms for resource allocation problems with autonomous and self-interested agents, respecting the autonomy and information decentralization in open systems. Applications range from distributed task allocation, to procurement in the supply-chain, to multi-agent scheduling problems. Typical characteristics include distributed information about agents' local problems and multiple conflicting goals. Auctions can minimize communication within a system, and generate optimal (or near-optimal) solutions that maximize the sum value over all agents.

Away from the rarefied atmosphere of the traditional auction house, auctions can extend to multiple items, and to complex decision spaces. In addition, with agent-mediation many participation costs are either reduced or eliminated. Fixed prices, useful in static markets with many agents and stationary aggregate demand statistics, often lead to suboptimal outcomes in dynamic multiagent problems. Similarly, bilateral negotiated outcomes and long-term agreements in the supply chain are too static and often suboptimal. Auctions can compute prices to support optimal solutions without *a priori* information about the preferences of agents in a system.

My thesis is that it is necessary to take an *explicitly computational approach* to auction design. Not only will auctions become increasingly agent-mediated— with automated bidding, winner-determination and price-adjustment —but auctions will also be applied to solve difficult distributed problems. Potential computational complexity exists for bidding agents that must compute strategies to participate optimally in new marketplaces, and in the implementation of mechanisms by auctioneers. Limited and costly computation requires a rethinking of traditional auction theory, because direct generalizations of auctions that work well in small problems can fail in large distributed systems.

This dissertation presents work that is especially important in systems with the following characteristics, which are very different from the assumptions made in traditional auction theory:

[A1] Agents have hard valuation problems and limited or costly computation.

In many important problems agents must solve difficult problems to compute values for items before bidding. For example, consider an auction-based system for distributed task allocation where agents need to reformulate local plans to compute costs for performing additional tasks.

[A2] Agents demand bundles of items.

Bundles are important in many real-world problems: consider a manufacturer that needs either components A and B, or just component C; consider a mobile agent that needs an interval of compute time; consider a train that needs a bundle of departure and arrival times on tracks across its route.

In comparison, most existing auction theory assumes: that agents know *a priori* their values for items before bidding; and that agents demand single items, multiple identical items, or have additive values for items.

As an example, consider an auction-based system for a scheduling problem, with distributed agents competing for the access to shared resources. An auction-based mechanisms for an $\mathcal{NP}$-hard scheduling problem, must either expect to compute approximate solutions or expect to solve $\mathcal{NP}$-hard problems. Only mechanisms that remove economic inefficiencies *and* handle computational complexities will succeed. The ultimate impact of this work will be judged by the simplicity, robustness, and performance of the auction-based systems that follow from the new insights and results.

As I address computational issues, I strive to retain important auction properties, in particular:

[P1] Allocative efficiency. A good resource allocation should maximize the sum value over all agents.

[P2] Incentive-compatibility. An agent's optimal bidding strategy is truth-revelation, whatever the bids of other agents.

Allocative efficiency is a measure of the total quality of an allocation, taking all agents as equal and seeking to maximize the total value over the system, i.e. maximize the "social welfare". This metric is meaningful in many distributed problems, equivalent to minimizing the total costs of delay over all trains in a train-scheduling problem, or maximizing the total value of computation and bandwidth in an application to competition for network resources between mobile agents.

Incentive compatible mechanisms are important because they are *self-enforcing* such that it is no single agent can successfully manipulate the outcome by misstating its true values for different items, or outcomes. Strategic manipulation is undesirable because it can reduce the allocative efficiency of an outcome, and also because it is inherently computationally complex. Moreover strategic manipulation is often hard to detect; for example an auctioneer typically has no way to know whether an agent is misstating its preferences, e.g. over-reporting or under-reporting its value for a certain type of outcome.

Consider a couple of examples of systems that are not incentive-compatible:

- A recent report stated that pilots into Heathrow on flights from the Far East tend to misstate medical emergencies or fuel shortages on their arrival, to jump the queue of holding planes. The problem with the current system is that their is little incentive *not* to misstate conditions on board a place. One remedy that is being considered is to require pilots to complete a full "emergency" report on landing if such a request is made, making its use for strategic manipulation less attractive.

- In performing simulations of auction-based systems for my dissertation I have accessed a shared cluster of workstations, Eniac-2000. A simple scheduler uses a queue to control jobs waiting for compute time. There are no "prices" associated with jobs, but shorter jobs receive precedence over longer jobs, and earlier jobs over later jobs. One simple way to manipulate the scheduler is to check the queue, and whenever no jobs are waiting in the queue release my current jobs and resubmit jobs with new extended deadlines.

First, I review existing economic mechanism-design theory. I introduce the Groves family of mechanisms, that represent the complete set of incentive-compatible mechanisms in a quite general sense. As applied to auctions, the Vickrey auction and its extension to problems with bundles of items, the *generalized Vickrey auction*, are examples of Groves mechanisms. In addition, I introduce the central *revelation-principle* of mechanism design, and suggest its shortcomings in systems with agents that have limited computation, hard valuation problems [A1], and need bundles of items [A2].

The revelation-principle states that it is sufficient to consider *direct-revelation* mechanisms in which agents report (possibly untruthfully) *all* of their private-information. This is clearly infeasible in many interesting applications, in the absence of unlimited computation and communication. I introduce a new property, *bounded-rational compatibility*, which describes auctions that in comparison allow agents to participate optimally with *approximate values* for some outcomes. Iterative auctions are especially useful: intuitively it is easier to respond to prices than submit a sealed-bid for items.

I introduce primal-dual optimization theory, which provides an important theoretical framework for designing iterative auctions with provable properties. The strong-duality theorem of linear-programming states that feasible primal and dual solutions are optimal if and only if they satisfy *complementary-slackness* conditions, which express constraints on their values. In the context of an iterative auction that maintains a provisional allocation, and prices for items or bundles of items, the allocation can be interpreted as a primal solution and the prices can be interpreted as a dual solution. With a reasonable assumption about agents' bidding strategies it is possible to adjust prices in the auction so that the allocation and prices satisfy complementary-slackness conditions when the auction terminates, and are therefore optimal solutions.

Continuing, I present my main contribution: an *ascending-price* combinatorial auction, *i*Bundle. Combinatorial auctions allow agents to bid for bundles of items directly, and generate prices on bundles. Although combinatorial auctions can be approximated by multiple auctions on single items, this often results in inefficient outcomes. In *i*Bundle agents can adjust their bids in response to bids from other agents, as the auctioneer updates a provisional allocation and bundle prices.

*i*Bundle is the first iterative combinatorial auction that is optimal for any reasonable bidding-strategy, in this case *myopic* best-response bids. This relaxes desirable property [P2], replacing full incentive-compatibility with *myopic* incentive-compatibility: it computes efficient allocations with agents that bid to maximize their self-interest in response to the prices in the current round, but ignore the possibilities of manipulating the outcome of the auction in future rounds. I prove that *i*Bundle maintains feasible primal and dual solutions to a linear program formulation of the "top-level" resource-allocation problem, and terminates with solutions that are optimal by the strong duality theory of linear programming.

Finally, it is possible to use primal-dual theory to compute the outcome of the generalized Vickrey auction— an incentive-compatible Groves mechanism for the combinatorial resource allocation problem —from agents' bids during an iterative auction. In application of this "proxy-agent and price-adjustment" technique to *i*Bundle, I have a number of encouraging initial results. When successful *i*Bundle retains its desirable computational properties, but inherits incentive-compatibility, such that it is an agent's optimal strategy to truthfully report its values for items to the auctioneer. My hope is that future work will extend *i*Bundle to an *iterative generalized Vickrey auction*, such that truthful bidding is optimal, whatever the bids of other agents.

I believe that the primal-dual framework holds great promise for the design of useful auction-based solutions in other problems, indeed many existing algorithms in mathematical programming have interesting market interpretations.

1.1 Current Status

Let me briefly review the work that I have completed within each of these areas, and outline my main contributions.

1.1.1 Bounded-Rational Compatible Auctions

I propose a new auction property, bounded-rational compatibility, which describes auctions that allow agents to place optimal bids without computing exact values for all items. This is important in problems with agents that have limited computation and hard valuation problems. Iterative auctions present a useful special case of bounded-rational compatible (BRC) auctions. In an iterative auction an agent can adjust its bids in response to bids from other agents as prices change, and compute incremental values for different outcomes, to support optimal decisions. In comparison, in a single-shot sealed-bid auction an agent must perform all computation up-front, for example submitting bids that might be consistent with all possible outcomes.

Contributions:

- Proposed that cognitive costs in valuing items explains the presence of ascending-price auctions on-line in favor of sealed-bid auctions [PUF99].

- Demonstrated that strategic equivalence (e.g. between the English and Vickrey auctions) is not sufficient for auctions to have equal performance with agents that have hard valuation problems [Par99a].

- Proposed bounded-rational compatibility, a property of auctions that allow agents to follow an optimal bidding strategy with approximate values for items or bundles of items [PU00a].

- Proved that all BRC auctions that also compute optimal allocations with enough agent computation are iterative auctions; characterized sufficient properties for bounded-rational compatibility [PU00a].

- Proposed metrics to compare the performance of BRC auctions, completed an experimental comparison of auctions in simple multi-item allocation problems [PU00a].

- Proved that agents with approximate values for items continue to have truth-revealing optimal bidding strategies in strategy-proof auctions [PU00a].

1.1.2 Iterative Combinatorial Auctions

*i*Bundle is an ascending-price combinatorial auction in which agents can bid on bundles of items and the auctioneer adjusts prices on bundles and maintains a provisional allocation as bids are received. *i*Bundle is the first *iterative* combinatorial auction to compute optimal allocations for any reasonable agent bidding strategy, in this case agents with myopic best-response bidding strategies. *i*Bundle satisfies property P1 (allocative-efficiency) but relaxes property P2 (incentive-compatibility), it remains possible for a rational agent with lookahead to manipulate the outcome of the auction.

The auction is bounded-rational compatible, and also permits the auctioneer to tradeoff computation for allocative-efficiency while maintaining incentives for agents to follow the same bidding strategies. In comparison, the generalized Vickrey auction (see Chapter 5), which is the only other known optimal combinatorial auction, is not bounded-rational compatible and loses its strategy-proofness with approximate algorithms for winner-determination.

Contributions:

- Proposed *i*Bundle, the first iterative combinatorial auction for any reasonable agent bidding strategy. Proved the optimality of *i*Bundle within a primal-dual framework, as applied to a linear-program formulation of the combinatorial resource-allocation problem [PU00b].

- Completed experimental tests in some simulated hard problem designs, demonstrated that the auction achieves 100% allocative efficiency for small enough bid increments [Par99b].

- Measured savings in "agent valuation work" compared to the generalized Vickrey auction [Par99b].

- Completed initial computation tests, demonstrated orders-of-magnitude speed-up in comparison with the generalized Vickrey auction in some hard problems; suggested cache-based techniques to speed-up winner-determination; tested a simple approximate winner-determination algorithm [PU00b].

1.1.3 Price-Adjustment to Vickrey Prices

In its basic form, an agent in *i*Bundle can *manipulate* the outcome of *i*Bundle, for example placing jump bids, signaling false intentions, or waiting to bid until the end of the auction. I only prove that *i*Bundle is allocatively-efficient for agents that submit myopic best-response bids to current prices.

Recently, I have proposed a new method "proxy agents and adjust" to make iterative auctions more robust to strategic manipulation, by adjusting agents' prices after the auction terminates towards prices in that make myopic best-response a sequentially-rational strategy. The idea is to compute Vickrey prices from agents' bids during the auction. The proxy agents constrain agents' bids to (possibly untruthful) myopic best-response bids to current prices.

My insight is that primal-dual theory presents a method, `Adjust*`, to adjust prices towards minimal competitive equilibrium prices, from which Vickrey prices can be computed. In application to *i*Bundle, when successful *i*Bundle retains the computational advantages of an iterative auction, but inherits the incentive-compatibility of the generalized Vickrey auction.

Contributions [PU00c]:

- Proved that Vickrey prices can be computed from "enough" minimal competitive equilibrium prices.

- Proposed proxy bidding agents to convert an iterative auction into a progressive direct revelation mechanism, such that agents report (possibly untruthfully) their values to the auctioneer (via the proxy agents), but provide information continuously as necessary during the auction.

- Proposed a method to use primal-dual theory to adjust prices after an auction terminates towards minimal competitive equilibrium prices.

- Proposed an approximate method to adjust prices after an auction terminates towards minimal competitive equilibrium prices, based on provisional allocations computed during previous rounds in the auction.

- Derived necessary conditions to compute minimal competitive equilibrium prices; derived necessary and sufficient conditions to compute Vickrey prices.

- Proposed a dynamic test that allows an auctioneer to determine dynamically whether an auction terminates with Vickrey prices.

- Classified some resource allocation problems in which `Adjust*` will compute Vickrey prices.

- Completed initial experimental tests for "proxy and adjust" with *i*Bundle, with promising results.

1.2 Proposed Work

Here is an outline of the further work that I propose to complete as part of my full dissertation:

- Characterize easy special cases of *i*Bundle, with polynomial time winner-determination, derive *i*Bundle price-update rules for problem-specific bid languages.

- Characterize Problems in which *i*Bundle with `Adjust*` computes Vickrey prices, and is therefore robust to strategic manipulation.

- Derive upper-bounds on possible gains from strategic behavior as approximations are introduced into *i*Bundle with price-adjustment.

- Experimental: Test methods to speed-up *i*Bundle; e.g. caching, ϵ-scaling, and approximate winner-determination algorithms.

- Develop an "iterative GVA" extension of *i*Bundle with `Adjust*`, that always terminates with the Vickrey outcome

- Implement an on-line prototype of the final auction. Issue a challenge for users to submit problems and agent strategies that can manipulate the auction.
- Complete the "application" study of *i*Bundle, e.g. to train-scheduling or some other real-world problem.
- Map the auction to a reverse auction, e.g. an auction by a manufacturer to purchase components from competing suppliers.
- Understand how impossibility theorems (e.g. Myerson-Sattherwaite) relate to the design of new auctions: (i) bundle auctions with sellers that have non-zero reservation prices; (ii) two-sided bundle auctions.

1.3 Outline

Chapter 2 motivates my work, presenting some illustrative problems in distributed systems and in electronic commerce that can be solved with combinatorial auctions. The applications are characterized by bundled items and hard agent valuation problems.

Chapter 3 reviews related work in artificial intelligence, economics, game theory and algorithm design. Chapter 4 introduces bounded-rational compatible (BRC) auctions, and present a simple example of how it is possible to structure auctions to allow agents to bid without computing exact values for items. I summarize my main results, providing some intuition about the characteristics of BRC auctions.

Chapter 5 provides a introduces the literature on economic mechanism design, presenting the Groves mechanisms, the generalized Vickrey auction, the revelation principle and an important impossibility result.

Chapter 6 introduces primal-dual optimization theory, and makes the connection between the design of iterative auctions and primal-dual algorithms. I also present linear-program formulations of the combinatorial resource allocation problem, which compute optimal allocations as primal solutions and corresponding bundle prices as dual solutions.

Chapter 7 presents *i*Bundle, describing the bidding rules in the auction, a best-response bidding strategy for agents, and the auctioneer's price-update and winner-determination problems. I summarize some encouraging experimental results in hard resource allocation problems from the literature.

Chapter 8 makes a connection between primal-dual theory and mechanism design. It is possible to compute Vickrey payments from minimal prices that satisfy complementary-slackness conditions with the optimal allocation, which opens up the method presented in Chapter 9 to adjust prices after an iterative auction terminates towards prices in the Vickrey auction.

Chapter 9 introduces the two stage method, "Proxy Agents and Price Adjustment", which can make iterative auctions robust to strategic manipulation. As an application of the methodology I consider *i*Bundle, and characterize sufficient conditions on agents' valuation functions for `Adjust*` to compute GVA prices.

Finally, Chapter 10 introduces a working prototype of an application of *i*Bundle-style price-updates to a distributed train scheduling problem, and Chapter 11 concludes.

Chapter 2

Mechanisms for Self-Interested Agents: Applications

My basic assumption is that agents have private information, and are uncooperative: they can not be expected to follow a particular protocol if deviations can be beneficial and go undetected and unpunished. The problem addressed in this thesis, in its broadest sense, is one of *mechanism design*, or designing the "rules of encounter" between agents [RZ94a]. I assume that while a system-designer can control the types of messages that agents can send, and state the methods that will be used to compute outcomes based on messages, it is *not* possible to control the strategies (or protocols) that agents follow in sending messages (so long as they are within the system rules).

Iterative combinatorial auctions are useful when: (1) agents demand bundles of items; (2) agents have hard local optimization problems to compute their value for items;(3) the information to compute the value of items is an agent's *private information*. The applications presented below all have these properties.

Auctions provide useful mechanisms for resource allocation problems with autonomous and self-interested agents. Typical applications include task assignment and distributed scheduling problems, and are characterized with distributed information about agents' local problems and multiple conflicting goals [Wel93, Cle96]. Auctions can minimize communication within a system, and generate optimal (or near-optimal) solutions that maximize the sum value over all agents.

More recently, electronic commerce has generated new interest in auction-based systems, both as dynamic mechanisms to sell items to individuals, and as systems for business-to-business transactions. At present, the vast majority of online auctions are simple variations on the standard auctions. In addition, simple extensions of standard auctions do not scale to combinatorial problems.

2.1 Resource Allocation in Multi-agent Systems

Many coordination and negotiation problems in multi-agent systems can be formulated as resource-allocation problems, where the allocation of items to agents represents a negotiated agreement between multiple self-interested agents. All that is required is a fixed set of items, and agents with private valuation problems such they do not care about the final allocation received by other agents.

1. *Distributed scheduling.* The supplier controls machines that can be configured to perform particular jobs, agents with jobs bid time slots on machines.

2. *MBA course registration.* A business school has a finite supply of courses and class times. Agents represent students that demand bundles of courses that are conflict-free and satisfy requirements for graduation.

3. *Train scheduling.* A track scheduler controls access to a shared track network, enforcing safety constraints. Agents represent trains with desired departure and arrival times, that must acquire a sequence of track times to complete their route, minimizing for example costs of late arrival and fuel costs en-route.

4. *FCC spectrum rights.* The government sells spectrum rights to telecommunications companies. Each company demands spectrum rights across geographically consistent areas, and must formulate a business plan and predict projected earnings to compute the value of a particular bundle.

The provable optimality of auction mechanisms can also be useful in *cooperative* systems, where agents are trusted to participate truthfully in a protocol.

Consider, for example, using a combinatorial auction as a simple and robust mechanisms for collaborative planning between cooperative autonomous agents [HG00]. Combinatorial auctions allow roles within a team to be conditioned on various constraints, for example time constraints, to protect the feasibility of local commitments.

2.2 Electronic Commerce Applications

The general problem of a distributed system with self-interested agents that (a) have hard local optimization problems and (b) demand bundles of items is ubiquitous in e-commerce.

To focus ideas, it is useful to consider a single *supplier* with a set of items (e.g. jobs, resources) for sale, and a number of agents with values for bundles of items. An auction-based mechanism can allow the supplier to determine an allocation of items to agents which maximizes economic efficiency.

Figure 2.1 illustrates the structure of a general problem, shown in this example with four agents.

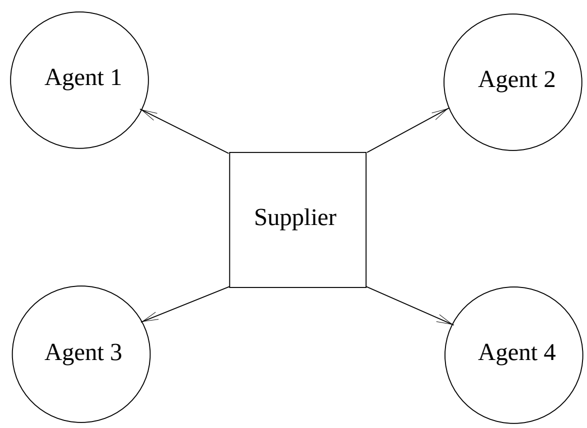


Figure 2.1: Illustrating the General Problem Structure.

It is often a reasonable assumption that the supplier (or auctioneer) seeks to maximize allocative efficiency, even when the auctioneer is self-interested and wants to maximize revenue. The basic argument is that can be made in the presence of alternative markets is that if the auctioneer does not compute efficient allocations then agents will go elsewhere. Ausubel [AC98] proves that the auctioneer maximizes profit with an efficient mechanism in the presence of after-markets.

The examples below illustrate some of the new marketplaces that are enabled by new connectivity and ubiquitous computational devices. I focus on examples in business-to-business (B2B) e-commerce.

1. *Supply-chain.* The supplier sells electronic components to manufacturing plants. Agents demand batches of components of compatible types, and must compute local manufacturing plans to determine the value of a batch of components.

2. *Power auction.* An electricity generator sells units of power (time, quantity) to manufacturing plants. A plant must formulate an alternate local plan to determine the tradeoff between, for example, expensive power now and cheap power later.

3. *Flights.* An airline auctions flights, to users that want a single pair of matching departure and return flights.

4. *Task allocation.* The supplier has a set of tasks to contract to distributed self-interested contractors. Consider, for example, packages to be delivered or computer systems to be assembled. The contractors have non-linear costs for tasks and therefore want to bid for suitable bundles of tasks. For example, in a task delivery problem the marginal cost of delivering an additional task to the same location is zero.

Chapter 3

Related Work

This work lies at the boundary between artificial intelligence, economics, game theory, and algorithm design. For now, I limit my attention to the literature in artificial-intelligence, auction-design, and economics. The large and relevant literature in economic mechanism design, that studies the problem of implementing desirable outcomes in systems of self-interested agents, is presented in Chapter 5.

3.1 Artificial Intelligence

Early work in distributed artificial intelligence, e.g. the work of Lesser *et al.* on blackboard-based systems assumed cooperative distributed agents, and motivated the need for multi-agent decision making from communications limitations that necessitated that decisions could not be taken centrally. See Weiß[Wei98] for an introduction to the literature. Early studies emphasized the importance of computational and communication efficiency, see also Yokoo [Yok95].

In comparison, early research into mechanisms to coordinate systems of self-interested agents focused on mechanism-design to encourage truthful participation, and ignored computational issues. For example, Rosenschein & Zlotkin [RZ94b] applied game theory techniques to negotiation over task-allocation, but assumed agents can solve exponentially many NP-hard problems. The work applied methods from game-theory to systems of computational agents. Rosenschein & Zlotkin's mechanisms are not market-based methods, there is no mechanisms to transfer utility (e.g. cash) between agents.

Wellman [Wel93] was an early proponent of market-based computation. Wellman proposed a *market-oriented* method, WALRAS, to coordinate agents, with an application to a multi-commodity flow problem. The WALRAS system has been applied to other problem domains, see Wellman [Wel96] for a later summary. In common with related work on market-based control [Cle96], this work mainly ignores the agent valuation problem, often assuming that agents have simple analytic utility functions.

Sandholm [San93, San96] has considered the agent valuation problem in markets. Sandholm [San93] implemented a ContractNet [DS88] based system for a distributed task allocation problem, with agents that bid on the basis of marginal values for tasks. The system, Traconet, allowed agents to bid with approximate values and continue to deliberate during the auction.

In subsequent work Sandholm & Lesser [SL96] propose a framework for *leveled commitment* contracts between agents, which allows agents to decommit from a contract. As noted by the authors, this is useful with agents that have approximate values for tasks and continue to refine their beliefs after striking initial contracts. For example, agents can correct early mistakes as they continue to compute values for tasks. The technique allows agents to integrate local deliberation with negotiation between many other agents. However, the focus is on a *decentralized* system, while our work considers techniques for auctions— with a centralized auctioneer.

Sandholm & Lesser [SL97] study a coalition formation problem, in which the problem of computing the value of a coalition structure is complex because it requires determining an optimal assignment of tasks to agents in the coalition. However, the authors do not design a mechanism that allows approximate information about coalition values. Instead it is assumed that agents predict value perfectly. There is no attempt to *integrate* the formation and valuation problems.

Sandholm [San96] demonstrates that the strategy-proofness of an auction can break when agents have approximate values for items and options to continue computation or submit bids. An agent can make a better decision about whether or not to perform further computation about the value of an item if it is well informed about the bids from other agents. I claim that this loss of strategy-proofness is *useful*. Indeed, bounded-rational compatible auctions provide agents with information about other agents, for example via prices in an ascending-price auction, to allow them to deliberate about their values based on information about likely outcomes in the auction.

Parunak *et al.* [PWS98] propose a system, MarCon, for *interactive decision-support*, in which people provide continual information about local constraints and goals as a global collaborative plan is formulated. The work shares many of our goals, allowing deliberation as information is exchanged about local constraints of all agents in the system. Users state broad initial preferences, and add more details as necessary.

There is a growing literature on auctions for e-commerce. See [KF98, HV99] for an introduction, and [MGM99, GK99] for an emphasis on *agent*-mediated auctions. Wellman *et al.* [WWWMM99] propose ascending-price auctions for hard distributed-scheduling problems, in which agents need bundles of time on machines. Another study proposes an auction protocol to coordinate agents in supply chain problems [WW99a].

3.2 Multi-Item Auctions

Although not *bundle auctions*, in the sense that agents cannot submit bids of the form "I only want item A if I also get item B", it is useful to mention a few auctions that compute optimal allocations in problems in auctions for multiple items, for restricted classes of valuation functions. This section describes auctions which are *not* incentive compatible, in the sense that they do not necessarily compute Vickrey payments. Incentive-compatible auctions are reviewed later, in Chapter 8, when I discuss my approach to make *i*Bundle incentive compatible.

Kelso & Crawford [KC82] propose an ascending-price auction for problems with agents that have gross substitutes preferences. Gross substitutes is a technical condition that states that an agent that demands good j at price $p(j)$ will continue to demand good j if the prices for other goods increase. Gross substitutes implies that agents have subadditive valuation functions[1] [GS97b], although subadditive is not sufficient for GS [BM97], and decreasing returns for multiple units of identical items. Recently, Gul & Stacchetti [GS97a] propose a bundle auction that terminates with minimal competitive-equilibrium prices when agents have gross-substitutes preferences (see the next section).

Bertsekas [Ber81, Ber88, Ber90] developed a mechanism for the assignment problem, called AUCTION. The proof technique, with primal-dual theory, was very influential in our proof of the optimality of *i*Bundle.

[1]The value for all packages is no greater than the minimal sum of values for a partition of the package.

3.3 Bundle Auctions

Bundle auctions have been proposed for problems in organizational theory, scheduling problems, and for the FCC spectrum auction. I define a bundle auction as an auction that allows agents to link a bid for one or more items with a bid for another item, i.e. to bid directly for bundles of items. A bundle auction does not necessarily compute bundle prices, but prices for a bundle might be the sum of the prices of individual items, or some other amount.

Rassenti *et al.* [RSB82] performed early work in combinatorial auctions, with an application of a simple sealed-bid bundle auction for the problem of allocating airport slots, where airlines value take-off and landing slots in pairs. Agents submit sealed XOR bids for bundles of take-off and landing slots. Agents submit sealed XOR bids for bundles of take-off and landing slots. The auction computes *linear prices* that approximately clear the market, given agent bids. Finally, agents can place *bids* and *asks* for individual slots in a secondary market, to cleanup their final allocation. Although the auction design is fairly ad-hoc, empirical results with human bidders suggest that the market can achieve high efficiency with experienced bidders.

Graves *at al.* [GSS93] generalized the auction to a course registration auction, extending the auction to multiple rounds, and allowing a limited number of trades in an after market.

AUSM [BLP89] allows agents to bid for arbitrary bundles of items, and maintains a revenue-maximizing allocation. There are no pricing rules, and agents must coordinate their own bids. Theoretical analysis is difficult because of the flexible auction rules, but see Milgrom [Mil99]. AUSM has reasonable performance empirically, see Ledyard et al. [LPR97].

The recent FCC spectrum auction generated a lot of debate among economists about auction mechanisms for the bundling problem. Spectrum licenses have non-additive value in bundles because of network synergies from spatially-coherent geographical regions. The final FCC auction design was a variant on a simultaneous ascending-price auction that allowed agents limited decommitment rights, and placed participation constraints on agents to enable information exchange via prices during the auction [MM96]. The goal was to allow agents to find a good "fit" between their demand sets and the demand sets of other bidders, and win coherent bundles of spectrum licenses.

Recently, DeMartini et al. [DKLP98] proposed RAD, an auction that allows agents to place XOR bids on bundles but generates prices on items. It is a multi-round variant of Rassenti et al.'s sealed-bid auction. Although promising empirical results have been presented, there are no theoretical results on its allocative efficiency. RAD borrows from the FCC auction design, agents must re-submit winning bids, and there are activity rules to encourage information revelation early in the auction and encourage coordinate bidding.

Bykowsky et al. [BCL00] demonstrate the *exposure* and *existence* problem with linear prices for the general bundling problem. Bykowsky et al. also identify the "threshold problem" for bundle auctions, where smaller bidders must coordinate bids to outbid a larger bidder.

Ausubel [Aus97] has proposed an ascending-price auction for an auction of multiple identical items, in which agents bid for bundles of items. The auction computes efficient allocations for agents with subadditive valuation functions.[2] The auction also terminates with prices that support Vickrey payments, and therefore is optimal for agents with sequentially-rational bidding strategies.

Gul & Stacchetti [GS97a] propose an ascending-price multi-item auction that finds the *minimal* competitive equilibrium prices when agents have gross-substitutes preferences. Agents bid for bundles in each round (all the bundles that maximize utility at the current prices), and the auction prices items. The auctioneer selects a revenue-maximizing allocation in each round, and increases prices on a *minimally overdemanded set of items*. Gul & Stacchetti [GS97b] show that the minimal competitive equilibrium prices for GS preferences do not always support Vickrey payments, as is the case for unit-demand preferences.

The Ascending k-Bundle Auction (AkBA) [Wur99, chapter 5] family of iterative combinatorial auctions share many of iBundle's computational advantages over the GVA. The main difference is that AkBA uses a linear program to update prices between rounds, and never charge discriminatory prices. A1BA, thought to be the most promising of the family, is not believed to support optimal allocations in all problems for any reasonable agent bidding strategy.

The auctioneer's winner-determination problem in combinatorial auctions is NP-complete [RPH98], although efficient algorithms exist for special cases. Recent results suggest that search algorithms can solve large problems quite quickly in the average case [San99,

[2]Subadditive preferences are such that the value of a package is no greater than the minimal sum of values for all partition of the package, i.e. decreasing returns for multiple units of identical items.

FLBS99, ATY00].

3.4 Economics & Game Theory

While self-interested agents, i.e. people, are ubiquitous in traditional economic systems, it is only recently that studies in economic theory have considered the effects of costs of participation on the performance of different mechanisms for resource allocation. Costs are suggested for example, in bid preparation and in information acquisition. However, almost all models assume that all participation decisions are made as a one-shot decision before an auction starts, and the models cannot capture the important idea that agents may continue to incur costs as an auction proceeds. Important exceptions, see below, are in the studies of Ehrman & Peters [EP94] and Milgrom & Weber [MW82].

In one of the few models to allow agents to enter sequentially, Ehrman & Peters [EP94] compare the performance of different auctions for agents with one-shot participation costs. The authors show that a *sequential posted-price auction* is useful for high costs of participation because it controls participation, through controlling the number of agents that are offered the item.

Similarly, in a model of *affiliated* values, in which the value of one agent for an item is partially related to the value of other agents, Milgrom & Weber [MW82] show that the English auction outperforms other auctions. The information during the auction, from agents' bids and decisions to leave the auction, allow an agent that remains in the auction to refine its estimate of value. Bids from other agents directly improve an agent's valuation. In comparison, in our model bids from other agents provide information that improve an agent's metadeliberation. Milgrom & Weber also show that providing expert appraisals always improves performance (the "linkage principle"), which is analogous to providing free computational resources in our model.

Chapter 4

Bounded-Rational Compatible Auctions

This chapter provides background theory describing auctions that are desirable when agents have hard valuation problems and limited computation. My characterization of bounded-rational compatible (BRC) auctions provides a good counterpoint to the revelation-principle, which I turn to in the next chapter. In essence, BRC auctions allow agents to bid with approximate values for items or bundles of items. It is important to allow agents to avoid deliberation wherever possible if auctions are to provide useful solutions in complex distributed problems. This chapter summarizes work presented in the attached paper *Bounded Rational Compatible Auctions* [PU00a].

4.1 Agent Decision Problem

Bounded-rational compatibility isolates an important computational property of auctions when agents have costly or limited computation, and cannot compute exact values for items before bidding. Figure 4.1 presents a simple model of an agent's decision problem within an auction. Assume that the problem can be separated into *valuation*, to compute the value of different items, and *bidding*, to compute an optimal bid. This separation is useful because each problem is well-defined by itself, and it isolates the valuation problem from the auction environment. Valuation can be solved with decision analysis tools and optimization methods that are independent of the particular auction.

Before turning to valuation, note that the bidding problem can be hard, in particular when an agent with information about the bidding strategies of other agents can manipulate the outcome of the auction. Counterspeculation and game-theoretic reasoning is difficult. However, it is possible to design *strategy-proof* auctions [MCWG95] in which the same bidding strategy is optimal for an agent whatever the strategy of other

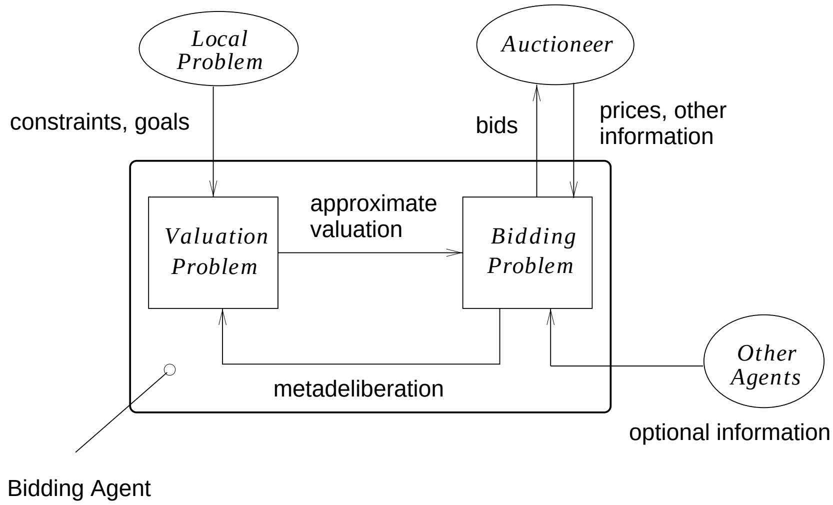


Figure 4.1: The Agent Decision Problem.

agents. Examples of standard auctions with this property include the Vickrey auction, which is a second-price sealed-bid auction, and the English auction, which is a first-price ascending-bid auction [MM87].

While seeking to retain strategy-proofness, we place a new emphasis on the valuation problem, which can also be difficult. It is possible to simplify an agent's valuation problem without introducing new complexity to the bidding problem. Although an agent's valuation problem is isolated from the auction, it is possible to design auctions that allow an agent to *avoid* unnecessary computation on its valuation problem and bid with approximate valuations. The key is an agent's *metadeliberation* procedure, which estimates the value of further computation [RW91]. This is the glue that connects auction design with the valuation problem.

In many important e-commerce problems agents must solve difficult problems to compute values for items before bidding [San93, PUF99, Par99a]. This is very different to an assumption made in traditional auction theory, which assumes that agents know *a priori* their values for items before bidding.

Let $v_i(S) \geq 0$ denote agent i's value for bundle $S \subset G$ of items, where G is a set of discrete items. The valuation problem is to compute the valuation function $v_i : 2^G \rightarrow \mathbb{R}_+$:

DEFINITION 4.1 Valuation Problem. The valuation problem for agent i is to compute $v_i(S)$ for all bundles $S \subseteq G$, for valuation function $v_i \in V_i$, where V_i is the set of all possible valuation functions for agent i.

For example, consider an auction-based system for distributed task allocation where agents need to reformulate local plans to compute costs for performing additional tasks; or an auction-based system for allocating landing times at an airport, where the valuation problem is to compute the value of a time slot based on local constraints such as the availability of support and maintenance crews, gate availability, and costs for late arrival.

Example: eBay

On-line auctions such as eBay, www.ebay.com, for consumer-to-consumer e-commerce present a real-world example of auctions with separate valuation and bidding problems: people value items, and eBay provides automated bidding agents that monitor auctions and place bids. In an ascending-price auction, the proxy agents are configured with a user's *reservation value*, the maximum she will pay for an item, and bid while the price is below that value. Interestingly, the proxy agents do not convert ascending-price auctions into sealed-bid auctions because they can inform a user by e-mail when her reservation value has been reached, and accept *updated* values. This allows the user to deliberate further about her value for the item, but only if that is required by the current price in the auction, and makes the auction with proxy agents bounded-rational compatible.

4.2 Bounded-Rational Compatible Definitions

Intuitively, an auction is bounded-rational compatible when it allows an agent to follow an optimal bidding strategy without computing its exact values for different items or bundles of items. The "degree" of bounded-rational compatibility, *informed-*, regular, or *strong-* depends on the information that an agent has about other agents and the number of problems in which it can bid with approximate values. The definitions are formalized in Parkes & Ungar [PU00a].

DEFINITION 4.1 Bounded-rational compatible (BRC). An auction is bounded-rational compatible if an agent with no information about the bidding strategies of other agents and an approximate valuation function can bid optimally in some non-trivial bidding problem.

In other words, at least in some non-trivial problems, an agent can place a bid for an item (or bundle of items) that is optimal for *all* possible exact solutions to its valuation problem. By definition, an *optimal* bid maximizes an agent's expected utility, given its knowledge about the auction and the bidding strategies of other agents.

BRC auctions allow agents with hard valuation problems to use costly or limited computation more effectively, and place more accurate bids than in non-BRC auctions.

An auction that *announces ask prices* before agents bid (e.g. a posted price auction) or while agents bid is BRC. Ascending-price auctions such as the English auction are particularly useful because prices are adjusted dynamically, so that agents can participate with less computation *and* it is still possible to find prices that support optimal allocations. As prices adjust they provide agents with implicit information about the values that other agents have for values, and allow agents to direct deliberation toward outcomes that are consistent with good system-wide solutions.

The yard stick for a bounded-rational compatible auction is that an agent can bid optimally with an approximate valuation in some bidding problem. This provides a clean theoretical characterization which is independent of agents' deliberation and metadeliberation procedures, and is directly applicable when agents have limited amounts of free computation. BRC auctions are also useful for agents with *costly* computation. An agent with costly computation and metadeliberation can avoid computation on value in at least those problems where it has an optimal bid with its current approximate valuation.

In an *informed* BRC auction an agent that is informed about the bids of other agents can bid optimally with approximate values.

DEFINITION 4.2 Informed Bounded-rational compatible (iBRC). An auction is informed bounded-rational compatible if a perfectly informed agent and an approximate valuation function can bid optimally in some non-trivial bidding problem.

Intuitively, if an agent is informed about other agents then it can avoid computation on its valuation problem even without any information from the auctioneer, because it can begin to predict the likely bids and outcomes of the auction.

In a strong bounded-rational compatible auction an uninformed agent can always bid optimally with an approximate valuation. Although useful computationally, I prove that strong bounded-rational compatibility is incompatible with allocative-efficiency.

DEFINITION 4.3 Strong Bounded-rational compatible (sBRC). An auction is strong bounded-rational compatible if an agent with no information about the bidding strategies of other agents and an approximate valuation function can bid optimally in all bidding problems.

Example: The English Auction

The English auction is a single-unit ascending price auction. The auctioneer announces an ask price p, and increases the ask price by a small increment ϵ while more than one agent bids for the item at the current price. The auction terminates when there is only a single bidder, and the item is sold to that agent for the final price.

The English auction is bounded-rational compatible. Figure 4.2 illustrates an example scenario in an English auction. The lines represent lower and upper bounds that agents 1–5 have computed for their value of a single item. In the English auction, a simple ascending-price auction, all agents have an optimal strategy to bid while the ask price is below their lower bound, and drop out of the auction when the ask price is above their upper bound. Prices between the bounds require further deliberation.

Given the approximate values in Figure 4.2, agents 3 and 5 will bid the ask price up to just above agent 5's lower bound, the second-highest lower bound. At this price, agents 1 and 2 can drop out of the auction without computing their exact values, while agents 3, 4 and 5 must perform further deliberation. This bidding problem demonstrates the BRC property of the English auction for agents 1 and 2.

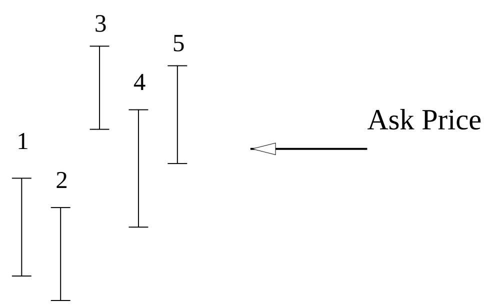


Figure 4.2: Example Scenario in the English Auction.

LEMMA 4.1 *The English auction is bounded-rational compatible.*

PROOF. Let v_i denote agent i's value, and $(\underline{v_i}, \overline{v_i})$ an approximate solution to its valuation problem, where $\underline{v_i}$ and $\overline{v_i}$ are lower and upper bounds on value. Consider a bidding problem where the ask price p, which is $\epsilon > 0$ above the highest bid from another agent, is less than the agent's lower bound on value, such that $p < \underline{v_i}$. The agent can following bidding strategy "bid if the ask price is less than $\underline{v_i}$", and this will be an optimal bidding strategy in this problem. ▮

LEMMA 4.2 *The English auction is not* strong *BRC.*

PROOF. By contradiction. Assume the English auction is strong BRC, so that an agent with valuation v_i and approximate valuation $v_{\mathrm{app},i}$ can bid optimally in all bidding problems. Let p denote the highest outside bid in the auction. Therefore an optimal bidding strategy b_i^* must bid $p + \epsilon$ when $p < v_i$, but make no bid in response to this price when $p > v_i$. However this is not a possible bidding strategy for an agent with an approximate value, because it implies that the agent can compute its true value, because the agent can make an accurate response to $p = v_i - \epsilon$ and $p = v_i + \epsilon$ for all values v_i. ∎

Example: The Vickrey Auction

The Vickrey auction [Vic61] is a second-price sealed-bid auction for a single item. The item is sold to the agent with the highest bid, but for the price of the second-highest bid. The Vickrey auction is interesting because it promotes competition between agents, it is an agent's optimal strategy to reveal its true value for an item. I prove in the next chapter that the Vickrey auction is *incentive-compatible*, or "strategy-proof". However, the Vickrey auction is *not* bounded-rational compatible. An agent must reveal its true *and exact* value to the auctioneer, any other bid and the agent risks paying too much for the item or missing a good price.

LEMMA 4.3 *The Vickrey auction is informed BRC.*

PROOF. For all valuation problems, let $(\underline{v_i}, \overline{v_i})$ denote lower and upper bounds on value. Let p denote the highest outside bid from another agent, and assume that agent i knows this information. Consider a bidding problem with $p < \underline{v_i}$. This is a non-trivial problem because the optimal bid in this problem depends on an agent's value for the item. Now, the set of optimal bids is $b_i^* = [p+\epsilon, \infty]$, and it is possible for the agent to bid optimally with strategy "bid $\underline{v_i}$ if $\underline{v_i} > p$". ∎

LEMMA 4.4 *The Vickrey auction is* not *bounded-rational compatible.*

Proof. By contradiction, assume that for all valuation problems an uninformed agent with an approximate valuation function $v_{\mathrm{app},i}$ can bid optimally in some non-trivial bidding problem. However, this requires that the agent bids $b_i^* = v_i$ (this is the optimal bid for an uninformed agent, irrespective of the bids that other agents actually make.) This is a contradiction because the agent cannot compute this bid without also computing its exact valuation function. ∎

4.3 Some Theoretical Results

We have already seen an example of an auction, the English auction, which is bounded-rational compatible and able to compute optimal allocations with agents that can compute accurate enough values for the item.

Theorem 4.1 (Sufficiency). *There are auctions, for example iterative auctions, that are bounded-rational compatible and allocatively-efficient with agents that can compute accurate enough values for items.*

In fact, the only BRC auctions that are also allocatively efficient are iterative auctions.

Theorem 4.2 (Necessary). *An auction that is bounded-rational compatible and allocatively-efficient (in a rich enough environment) must be iterative, i.e. allow agents to adjust their bids in response to bids placed by other agents.*

There is a limit to the ability of an auction to both compute optimal global allocations *and* allow agents to submit bids with approximate valuations.

Theorem 4.3 (Impossibility). *No* strong *BRC auction, which allow agents to bid with approximate values in all problems, can also be allocatively-efficient.*

For example, a posted-price auction is strong BRC but not allocatively-efficient (unless the auctioneer is perfectly informed about the values of all agents).

4.4 Retaining Truthful Bidding

It is important that BRC auctions can reduce the amount of computation that agents need to perform on valuation *without* introducing new computational complexity to the bidding problem.

Some auctions, such as the Vickrey auction, are *incentive-compatible*: it is optimal for an agent to bid its true value for items irrespective of the bids placed by other agents. I review some of the extensive literature on incentive-compatibility in the next chapter.

Sandholm [San96] showed that strategy-proofness can break when agents have approximate valuations, in the following sense: an agent that is informed about the bids that another agent will place can optimize its local computation about value, and submit more useful bids in the auction. For example, even though the generalized Vickrey auction is strategy-proof for agents that know their values for items, if an agent with uncertain values and limited computation knows one item will receive a very high bid from some agent, then it can devote its computation to refining its value for the other item. It can be helpful for an agent with uncertain values to speculate about the bids of other agents, even in a strategy-proof auction.

I argue that this is a *useful* loss of strategy-proofness, because it allows an agent to make effective use of its local computation, and improves economic and computational efficiency. BRC auctions are useful precisely because they allow agents to reason about the outcome of the auction, and implicitly about the bids of other agents.

It remains useful to retain the property in incentive-compatible auctions that agents cannot manipulate the outcome through strategic *bidding*, even when the agents have approximate valuations. Truthful bidding is useful in terms of both computational and economic efficiency.

Fortunately, I have a positive result for a large class of agents, the class of *prototypical* bidders. Truthful bidding continues to be optimal in strategy-proof auctions even with agents that have approximate values for items, although strategy-proofness does not hold in general if one also considers agents' computational actions.

Theorem 4.4 (Truthful Bidding with Approximate Values). *Prototypical bidders with approximate valuation functions have truthful dominant bidding strategies in a strategy-proof auction.*

Prototypical bidders include risk-neutral agents with quasi-linear utility functions; it is optimal for a risk-neutral agent to follow the optimal bidding strategy for the expected value of items given an approximate valuation function with distributional information on the true value of the item. Prototypical bidders also include agents that bid based on worse-case or best-case values, given their approximate values.

The optimal bidding strategy for prototypical bidders in a strategy-proof auction, such as the Vickrey auction, is truth-revelation; an agent should truthfully reveal its approximate value for items whatever the bids placed by other agents.

Examples: Retaining Incentive-Compatibility

The Vickrey auction is strategy-proof, it is optimal for an agent to bid truthfully for all bids from other agents. Now consider the bidding strategy of an agent with an approximate valuation function.

- $v_{\text{app},i} = U(5, 10)$, i.e. the agent believes that its true value is uniformly distributed between 5 and 10. The optimal strategy for a risk-neutral agent with $u_i(v_i - p) = v_i - p$ is $b_i = 7.5$, this maximizes expected utility for all bids from other agents (given the approximation $v_{\text{app},i}$).

- Approximate valuation computes bounds such that $5 \leq v_1 \leq 10$. Consider an agent that bids with value $v_i' = 5$, i.e. a pessimistic agent that bids assuming a worst-case value. The optimal strategy with assumed value $v_i' = 5$ is $b_i = 5$, for all bids from other agents.

4.5 Metrics to Quantify Performance Tradeoffs

I propose metrics *bounded-efficiency* and *bounded-computation* to compare the performance of different auctions in a particular problem at design time. The metrics assume a model of agent deliberation procedures, computational resources, and valuation problems. I introduce a "lazy deliberation and eager bidding" model of auction participation, and define:

- Bounded-efficiency $Perf_{\mathcal{A}}(C_{\max})$ of auction $\mathcal{A}$ is the allocative-efficiency achieved with agents that have limited computation budget $C_{\max}$.

- Bounded-computation $Comp_{\mathcal{A}}(C_{\max})$ of auction $\mathcal{A}$ is the average computation performed by agents with computation budget $C_{\max}$.

The metrics are closely related to the theoretical characterization of BRC auctions. For example, the ratio of bounded-computation to computation budget provides a direct measure of the bounded-rational compatibility of an auction. I prove the following result:

THEOREM 4.5 *Auction $\mathcal{A}$ is bounded-rational compatible iff the bounded-computation $Comp_{\mathcal{A}}(C_{\max}) < C_{\max}$ for some budget $C_{\max} < C_{\text{exact}}$, where C_{exact} is the average per-agent computation that is required for all agents to compute exact values for all items.*

For example, in the Vickrey auction $Comp_{\mathcal{A}}(C) = C$ for all $C \leq C_{\text{exact}}$, while in the English auction $Comp_{\mathcal{A}}(C') < C'$ for some $C' < C_{\text{exact}}$ because agents can bid optimally without exact values for items.

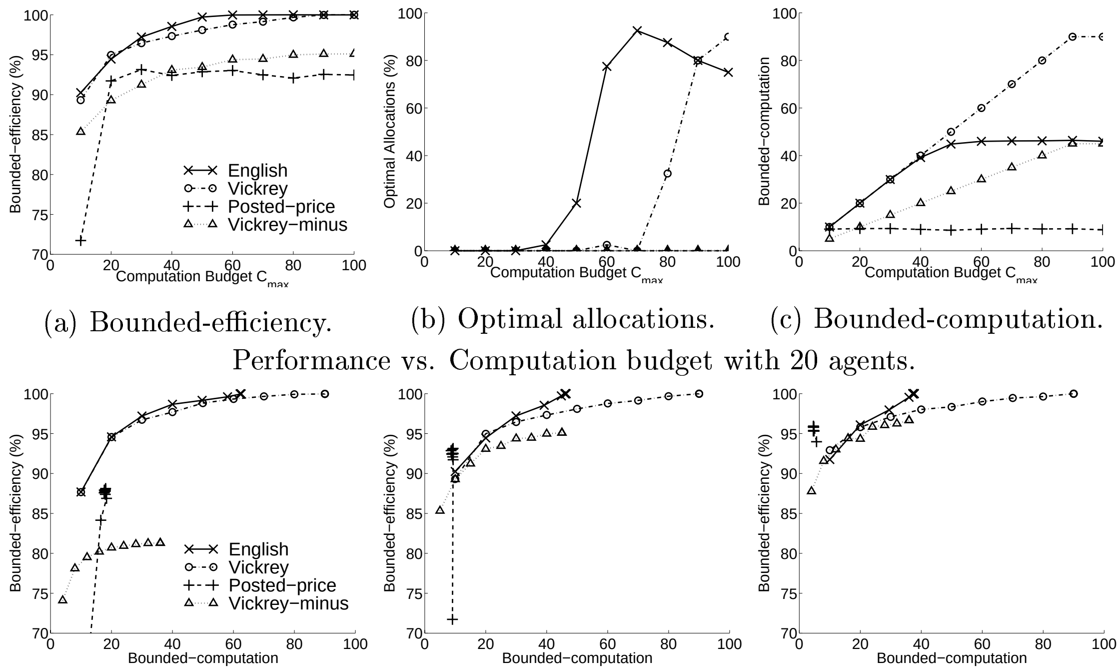
4.6 Experimental Results

Computational analysis in simple resource allocation problems demonstrates the importance of BRC auctions, especially when agents demand bundles of items. In Parkes & Ungar [PU00a] we compare the performance of the English, Vickrey, Posted-price (with a items offered at a fixed price to agents sequentially) and a Vickrey-minus auction (with a proportion of agents selected at random and blocked from participation).

Figure 4.3 plots experimental results for an additive-value multi-item problem with 10 items, with agents that have values $v_i(S) = \sum_{j \in S} v_i(j)$ for items j in bundle S, and values $v_i(j) = U(0, 10)$, i.e. selected from a uniform distribution between 0 and 10.

The agents have a simple model of deliberation: an agent maintains an upper and lower bound on value, and computes new bounds with an uncertainty of $1 - \alpha$ of the current bounds in a single computation step, for a "deliberation effectiveness" $1 - \alpha$ between 0 and 1. The mean of the new bounds is uniformly distributed with respect to the mean of the current bounds.

Figures 4.3 (a – c) plot the bounded-efficiency, number of optimal allocations, and bounded-computation for the problem with 20 agents. Figures 4.3 (d – f) plot bounded-efficiency versus bounded-computation for 5, 20 and 50 agents. I set the posted price $p^* = 5.6, 8.4$ and 8.8 for each problem size, and drop 50% of agents in Vickrey-minus.



(a) Bounded-efficiency. (b) Optimal allocations. (c) Bounded-computation.

Performance vs. Computation budget with 20 agents.

(d) 5 agents. (e) 20 agents. (f) 50 agents.

Bounded-efficiency vs. Bounded-computation.

Figure 4.3: Additive-value problem.

The English auction performs better than the Vickrey auction for agents with the same computation budget, as shown for example in Figure 4.3 (a) and (b). With 20 agents, for medium budgets, $30 \leq C_{\max} \leq 80$, the bounded-efficiency is greater in the English auction, and the auction computes more optimal allocations. For intermediate computation budgets the agents in the English auction can use prices to make good decisions about how to allocate computational resources.

Furthermore, the agents in the English auction compute 100% efficient allocations with 49% less computation than the agents in the Vickrey auction. This is illustrated in Figure 4.3 (c) and (e). For large computation budgets the agents in the English auction can avoid computation altogether.

Figures 4.3 (d – f) show that as the number of agents increases from 5 to 50, the agents in the English auction are able to avoid more computation on average, computing 100% efficient allocations are computed with 31%, 49% and 58% less agent computation than in the Vickrey auction.

Figures 4.3 (d – f) also show that the posted-price auction performs especially well as the number of agents increases, achieving bounded-efficiency of 88%, 93% and 96% for 5, 20, and 50 agents, and with 20%, 10% and 5% of the computation in the Vickrey auction.

The Vickrey-minus auction eliminates 50% of the agents from the auction, and reduces the average computation by 50%. For large numbers of agents it performs almost as well as the Vickrey auction, with 97% bounded-efficiency for 50 agents.

The number of optimal allocations does not reach 100% in either the Vickrey or English auctions, see Figure 4.3 (b). This is because the agents do not refine their value for items beyond uncertainty $\overline{v}_i(j) - \underline{v}_i(j) < \Delta_{\min}$, where $\Delta_{\min}$ is a small positive value that provides enough accuracy for 100% allocative efficiency, but not enough accuracy for 100% optimal allocations.

Chapter 5

Economic Mechanism Design

This chapter reviews important results from the economic mechanism design literature, that relate to auction-design for complex systems of self-interested agents. For a more general introduction, MasColell *et al.* [MCWG95] provides a good reference, and see [Var95] for an introduction to mechanism-design in the context of computational agents.

My purpose is to review some key ideas from the literature, and consider how they relate to my work on auction design for complex distributed computational systems. E.g., it is relevant that the Groves mechanisms (including the generalized Vickrey auction as a special case) are unique as allocatively-efficient *and* incentive-compatible mechanisms; *and* it is important to note that the revelation principle— central to mechanism design theory —fails in systems with limited or costly computation.

There is little prior attention to the interaction between mechanism design and limited computation. Indeed, the central *revelation-principle* that states that any outcome can be implemented with a direct-revelation mechanism in which agents report their complete valuation functions, fails with limited agent computation and hard valuation problems. The unrealistic assumptions made in the revelation principle, at least in their application to organizational theory, appear to have been first made by Ledyard [Led89], and again [Led93].

Recent research has considered the problem, in particular with respect to the *implementation complexity* of the *sealed-bid* generalized Vickrey auction (see below). The work has focused on special cases, and considered approximate approaches in other problems [KDMT98, Ron99, LOS99]. As an example, Nisan & Ronen [NR99] view auctions from the perspective of algorithm-design for self-interested agents, and present problems that can be solved in polynomial-time via mechanism design and an application of Vickrey-Groves mechanisms. In related work, Ronen [Ron99] proves under quite general assumptions, that applying Vickrey-Groves mechanisms directly to an approximate solution for the auctioneer's winner-determination problem is not *incentive-compatible*, i.e. it is possible that an agent can successfully manipulate the outcome.

The work that I have completed on *i*Bundle, and the work that I propose, to make *i*Bundle strategy-proof by price adjustment towards the outcome of the generalized Vickrey auction, differs in one very important dimension. *i*Bundle is an *iterative* mechanism, which as argued in chapter 4 is important because it provides it with the bounded-rational compatible property and agents can reduce and avoid valuation computation where possible.

5.1 Preliminaries

Consider a set G of discrete items and a set I of agents. Each agent has a valuation function $v_i : 2^G \rightarrow \mathbb{R}_+$, that defines its value $v_i(S) \geq 0$ for bundles of items, $S \subseteq G$. Assume that agents have *independent values*, so one agent's value for resources does not affect another agent's value. Assume $v_i(\emptyset) = 0$, and *free disposal* of items, implying that agents have weakly increasing values for bundles, i.e. $v_i(S) \leq v_i(S')$ for all $S' \supset S$.

An agent's *utility function* defines its preferences over different outcomes of an auction. I assume that agent i has a strictly increasing von-Neumann Morgenstern utility function for money, $u_i : \mathbb{R} \rightarrow \mathbb{R}$, which is normalized by $u_i(0) = 0$. Therefore, agent i has utility $u_i(v_i(S) - p)$ for bundle $S \subseteq G$ at price $p > 0$.

Finally, I assume that agents are risk-neutral with utility functions that are quasi-linear in money, i.e. agent i has utility $u_i(S) = v_i(S) - p(S)$ for bundle S at price $p(S)$, and ignore budget effects, i.e. agents have an initial endowment of wealth large enough to buy any bundle they need.

5.2 The Combinatorial Resource Allocation Problem

The combinatorial resource allocation problem is to allocate items to maximize the sum value over all agents. Introducing $x_i(S)$ to indicate that agent i receives bundle S the straightforward integer program, [IP], is:

$$\max_{x_i(S)} \sum_S \sum_i x_i(S) v_i(S) \qquad \text{[IP]}$$

$$\text{s.t.} \quad \sum_S x_i(S) \leq 1, \quad \forall i \qquad \text{(IP-1)}$$

$$\sum_{S \ni j} \sum_i x_i(S) \leq 1, \quad \forall j \qquad \text{(IP-2)}$$

$$x_i(S) \in \{0, 1\}, \quad \forall i, S$$

where $S \ni j$ indicates a bundle S that contains item j. A feasible solution allocates at most one bundle to each agent (IP-1), and cannot allocate an item multiple times (IP-2).

Let V^*_{IP} denote the value of the optimal allocation. The quality of an allocation is measured in terms of *allocative efficiency*, the ratio of the total value of the allocation to the maximum possible total value to agents over all allocations.

Definition 5.1 Allocative efficiency. The allocative efficiency, $Eff(\mathbf{S})$, of allocation $\mathbf{S} = (S_1, \ldots, S_{|I|})$, is measured as $Eff(\mathbf{S}) = \frac{\sum_i v_i(S_i)}{V^*} \times 100\%$

Allocative efficiency is an economic measure of the total welfare in a system of agents. This measure of performance has wide applicability to problems in distributed artificial intelligence, for example in scheduling problems where the goal is to minimize the total cost of delay over an entire factory and different bidding agents represent different operating units, and in load-balancing problems where the goal is to maximize additive performance over all jobs.

Later, in chapter 6, I present a hierarchy of linear-program formulations for [IP], that are important in proving optimality properties of *i*Bundle.

5.3 Economic Mechanism Design

In multi-agent systems agents are autonomous, and have private information about their local problems and values for resources. The auctioneer does not know the values of agents, and agents cannot be trusted to reveal their values truthfully— because lying can improve their allocation of resources, for example over-stating the value of resources that are important to them.

Mechanism design looks for mechanisms that are self-enforcing, such that it is in no agent's best self-interest to manipulate the outcome by misstating their preferences for different bundles of items. The *mechanism design* problem is to implement a mechanism that will compute efficient allocations with self-interested agents, i.e. provide incentives for agents to reveal values for bundles of items *truthfully* within a mechanism that computes an efficient allocation. Self-enforcing mechanisms are important in systems with self-interested agents and private information because it prevents *undetectable* deviation. There is no easy way to know, as an auctioneer for example, whether an agent is making true claims about its value for an item or bundle of items.

Leaving aside computational considerations for the moment, desirable properties of mechanisms are:

1. Allocative-efficiency.

2. Incentive-compatibility. Each agent follows a utility-maximizing strategy; e.g. in Bayes-Nash equilibrium with the strategies of other agents, or a dominant strategy (optimal for all strategies of other agents).

3. Individual-rationality (or participation). Each agent must be able to choose whether to participate in a mechanism, i.e. no agent that participates makes an expected *loss* in utility.

In the next section I introduce the *revelation principle*, a central result in mechanism design that provided a direction for important theoretical contributions. As an important caveat, central to my thesis, I note the unreasonable computational and informational assumptions made in the revelation principle. Then, I introduce the Groves-Clarke mechanisms and the generalized Vickrey auction, which has direct application to the combinatorial resource allocation problem. Finally, I present a set of impossibility results, that define limits on what can be achieved in mechanism design for general problems.

5.4 The Revelation Principle

This section introduces the *revelation-principle*, which is a central result in mechanism design. The revelation principle can **fail** in complex distributed systems with limited computation.

A central result in mechanism design is the revelation principle. Informally, the revelation principle states that the outcome of any mechanism can be implemented as a *truth-revealing direct revelation* mechanism. A direct revelation mechanism is any mechanism in which agents (possibly untruthfully) reveal their valuation function, instead of some other indirect information (e.g. reporting ones value for an item instead of a maximum price one is prepared to pay, or instead of bidding for an item at the current price). In a truth-revealing mechanism an agent's optimal strategy is to report its true value.[1]

Of particular interest in our current discussion is a special case of the revelation principle for *dominant-strategies*, that applies to auctions in which an agent has an optimal strategy for all strategies from other agents. In the next section I present the Groves family of mechanisms, in which agents have dominant strategies. However, a similar revelation principle can be stated for other implementation concepts, e.g. Bayes-Nash equilibria.

THEOREM 5.1 (Revelation Principle). *Any mechanism (indirect or otherwise) that can be implemented with dominant-strategies can be implemented as a truth-revealing (or strategy-proof) dominant-strategy direct-revelation mechanism.*

The revelation principle was first formulated for dominant-strategy equilibria [Gib73], and later extended by Green & Laffont [GL77] and Myerson [Mye81]. MasColell *et al.* [MCWG95] provide a useful text-book introduction to the subject.

Put simply, the principle says that anything that can be achieved by an indirect mechanism can be achieved by a strategy-proof direct mechanism, in which agents reveal all of their private information truthfully. The intuitive idea is that the auctioneer, or some other "mechanism implementer" can *simulate* the entire system— the bidding strategies of agents and the rules of an indirect mechanism —with complete and perfect information about every agent. So long as the auctioneer can claim credibly to implement an agent's strategy faithfully, then it is an optimal for an agent to report its true valuation function.

5.4.1 Failure with Limited Computation

Although useful for proving theorems in mechanism design, e.g. uniqueness (Theorem 5.3) and impossibility (Theorem 5.4), it is often invalid in real systems because of limited

[1]The appropriate sense of "optimal" depends on the implementation concept, e.g. it might be dominant strategy optimal (ideally), or a utility-maximizing strategy within a Bayes-Nash equilibria.

computation and communication. Indeed, a central contribution of my work is an *iterative* combinatorial auction, which is useful precisely *because* it is not a direct-revelation mechanism. With the revelation principle, *i*Bundle is no better than the GVA, because any computational benefits to the auctioneer from *i*Bundle could be accrued by central simulation of *i*Bundle in order to implement the GVA.

Ledyard [Led89] notes that writing down one's complete preferences is considerably more difficult that reacting to a price. This observation continues to apply to agent-mediated electronic markets when agents have hard local valuation problems and there are an exponential number of different bundles of items. The revelation principle breaks in the presence of limited or costly computation for agents with hard valuation problems.

The revelation principle assumes: (a) that the auctioneer can simulate the complete "system", i.e. bidding strategies of agents and the rules of the initial, possibly indirect mechanism; (b) that agents can compute their complete valuation functions up-front and report them to the auctioneer; and (c) that agents' valuation functions are small enough to communicate to the auctioneer. With a careful implementation, it is possible to escape (a) and (c); i.e. implement local "proxy bidding agents" on the client machines of agents, with strategies programmed by the auctioneer and value information provided by the local agent but retained locally.

Assumption (b) presents a fundamental problem in many important applications of auction mechanisms to complex distributed systems, and breaks the revelation principle.

I propose the following "bounded" revelation principle, for problems in which agents have hard valuation problems and limited computation, and cannot compute their complete valuation functions and submit them to the auctioneer:

THEOREM 5.2 (Bounded Revelation Principle). *Any mechanism (indirect or otherwise) that can be implemented with dominant-strategies can be implemented as a truth-revealing (or strategy-proof) dominant-strategy* ***iterative*** *direct-revelation mechanism.*

My insight is that it "direct revelation" and "iterative" need not be mutually exclusive. For example, in my "proxy agent and price adjustment" extension to *i*Bundle to push the auction closer to incentive-compatibility, I suggest the following scheme:

The auctioneer provides each agent with a proxy bidding agent, that is either implemented on an agent's local client machine, or on the auction server. The proxy bidding *agents place bids for agents within the auction on the basis of an agent's reported valuation function. But, the proxy agents can receive continuous updates about an agent's valuation function, they need not receive all information up-front. In particular, an agent's proxy might request more information of a particular kind, e.g. a more accurate value for a particular bundle. At any time the proxy agents follow their bidding strategy to the extent permitted by the current value information provided by the agents.*

Similar motivation follows for Milgrom & Weber's [MW82] work on iterative auctions in problems with correlated values. Ascending-price auctions generate more revenue than sealed-bid auctions because agents receive information from the bids of other agents that is useful to refine their own valuations for items. When agents have correlated values the revelation principle breaks for informational reasons: the auctioneer or some central "mechanism implementer" cannot faithfully predict how an agent will adjust its values for items or bundles of items in response to bids from other agents.

5.5 Vickrey-Groves-Clarke Mechanisms

The Groves mechanisms, of which the Vickrey auction is a special case, are interesting because they characterize the *only* mechanisms that are allocatively-efficient and incentive-compatible, in quite a general sense. This *uniqueness* property motivates my attempt to implement the generalized Vickrey auction, a Groves mechanism, as an iterative auction for combinatorial problems.

Groves [Gro73] proposed a family of mechanisms, the Groves mechanisms, which in application to the combinatorial resource allocation problem are efficient and incentive-compatible, and individual-rational in special cases.[2]

In a Groves mechanism agents report values $\hat{v}_i$, and the auctioneer computes the allocation $\mathbf{S}^* = (S_1^*, \ldots, S_{|I|}^*)$ that maximizes the total *reported value.* Agent i receives bundle S_i^* and pays transfer function:

$$t_i = h_i\left(\hat{v}_1, \ldots, \hat{v}_I\right) - \sum_{j \neq i} \hat{v}_j(S_j^*)$$

[2]Groves mechanisms (and the Clarke mechanism [Cla71], see below), were first formulated to implement efficient outcomes in problems with *public goods*, where it had been perceived the "free-rider" problem was unsurmountable.

To see that the mechanism is efficient and incentive compatible, we prove that agent i maximizes its utility by reporting its true value $\hat{v}_i = v_i$ to the auctioneer. With this, allocative-efficiency follows because the auctioneer explicitly computes allocation $\mathbf{S}^*$ to maximize allocative-efficiency. The utility to agent i from bid $\hat{v}_i$ is

$$u_i(\hat{v}_i) = v_i(S_i^*) + \sum_{j \neq i} \hat{v}_j(S_j^*) - h_i(\hat{v}_1, \ldots, \hat{v}_I)$$

Ignoring the final term, because $h_i(\cdot)$ is independent of an agent's bid, a truthful bid $\hat{v}_i = v_i$ maximizes utility because the auctioneer computes allocation $\mathbf{S}^*$ to maximize the sum reported value, i.e. the first two terms.

Intuitively, the Groves mechanism *internalizes* the externality of the effect of an agent's bid on the other agents in a system, and makes self-interested agent i share the auctioneer's goal of maximizing allocative efficiency. Thus, the Groves mechanism is efficient and incentive-compatible, so long as agents must participate— for any function $h_i(\cdot)$.

Green & Laffont [GL77] prove the following important property of Groves mechanisms, as a special case of the dominant strategy revelation principle:

THEOREM 5.3 (Uniqueness). *The Groves mechanisms are* unique*, in the sense that they characterize the only efficient mechanisms with dominant-strategy truth-revelation.*

The Groves mechanisms are also *individual rational*, such that the auction is allocatively-efficient with voluntary agent participation if the $h_i(\cdot)$ price function is sufficiently small, such that $t_i < v_i(S_i^*)$ for every agent in all problems.

Clarke [Cla71] proposed a special-case of the Groves mechanisms, the *pivotal mechanism*, which defines $h(\cdot)$ to guarantee *individual-rationality*. The additional transfer payment in the Clarke pivotal mechanism is defined as:

$$h_i(\hat{v}_1, \ldots, \hat{v}_I) = \sum_{j \neq i} \hat{v}_j(S_j^{-i})$$

where $\mathbf{S}^{-i} = (S_1^{-i}, \ldots, S_I^{-i})$ is the efficient allocation of items computed with bids from every agent except agent i. Note that $h_i(\cdot)$ is independent of the bids of agent i, and therefore within the Groves family of mechanisms. With the Clarke payment, notice that $t_i \geq 0$ for all agents, because $\sum_{j \neq i} \hat{v}_j(S_j^{-i}) \geq \sum_{j \neq i} \hat{v}_j(S_j^*)$. Furthermore, the only agents with

$t_i > 0$ are pivotal agents that change the allocation to other agents by their presence, such that $S_j^{-i} \neq S_j^*$ for some $j \neq i$.

To prove individual rationality, note as a special case that for agent i with no allocation, $S_i^* = \emptyset$, the price $t_i = 0$. In addition, for an agent i in the optimal allocation:

$$\begin{aligned}
u_i(\hat{v}_i) &= v_i(S_i^*) - \sum_{j \neq i} \hat{v}_j(S_j^{-i}) + \sum_{j \neq i} \hat{v}_j(S_j^*) \\
&= \sum_{j} \hat{v}_j(S_j^*) - \sum_{j \neq i} \hat{v}_j(S_j^{-i}) \\
&= V^* - V^{-i} \\
&\geq 0
\end{aligned}$$

because the total reported value of the optimal allocation without agent i must be less than or equal to the total reported value of the optimal allocation with agent i.

In its application to the combinatorial allocation problem, and other private-values mechanism design problems, the Clarke mechanism is typically referred to as the *generalized Vickrey auction* in recognition of Vickrey's seminal work.[3]

Vickrey [Vic61] proposed a auction mechanism that encourages competition between agents: agents receive incentives to bid their true values for items. The Vickrey auction is efficient, incentive-compatible and individual-rational. The Vickrey auction is a second-price sealed-bid auction. The item is allocated to the agent with the highest bid, for the price of the second-highest bid. Truth-revelation is a *dominant strategy*, optimal for all bids from other agents. The optimal strategy of agent i is to submit its true value v_i for an item. Intuitively, this is optimal because an agent will pay just enough to bid above the highest bid of any other agent if that bid is below its value, but never pay above its value for the item.

The generalized Vickrey auction, GVA for short, computes payment

$$p_{\mathrm{gva},i} = \sum_{j \neq i} \hat{v}_j(S_j^{-i}) - \sum_{j \neq i} \hat{v}_j(S_j^*) \tag{5.1}$$

to each agent i. The GVA computes efficient allocations with incentive-compatibility,

[3] Alternative names include the Vickrey-Groves mechanism, and the Vickrey-Groves-Clarke mechanism.

from the Groves mechanism, *and* individual-rationality (or participation), from the Clarke "pivotal" transfer.

The connection between Groves-Clarke mechanisms and the Vickrey auction appears to have been first made by Forsythe & Isaac [FI82]. In the special-case of an auction for a single-item,

$$p_{\mathrm{gva},i} = \begin{cases} \hat{v}_k - 0 & \text{, if } \hat{v}_j < \hat{v}_i \\ \hat{v}_k - \hat{v}_k = 0 & \text{, otherwise.} \end{cases}$$

where $\hat{v}_j$ is the maximum reported value over all agents without agent i, i.e. $\hat{v}_k = \max_{j \neq i}\{\hat{v}_j, 0\}$. In other words, if agent i submits the highest bid it receives the item and pays the price of the second-highest bid, otherwise it does not receive the item and pays zero. Note that by the Groves mechanism, the first term in an agent's price, $\hat{v}_k$ is independent of its bid, and the second term internalizes the reported value of the allocation to the other agents.

In summary, the GVA is an incentive-compatible and individual-rational mechanism for the combinatorial resource allocation problem if: agents have quasi-linear utility functions (risk-neutral); *and* the seller has zero value for the items— but seeks to maximize allocative-efficiency.[4]

The GVA extends directly to the case of a single seller that truthfully reports its value for the items to the auctioneer; in this case the auctioneer can implement a proxy agent for the seller, that bids its value and will "buy-back" the items if the seller cannot achieve a high enough sale price in the auction.

A word of caution: the GVA is *not* robust to manipulation by a coalition of agents [San96], and may be quite easily manipulated in an Internet environment with false-name bids, where identities are particularly fluid [SYM99]. In addition, in application to the combinatorial resource allocation problem the GVA requires the solution of an $\mathcal{NP}$-hard problem, and is likely to be infeasible for large problems [RPH98].

[4]An auction that maximizes allocative efficiency will also maximize revenue under quite general conditions.

5.6 Impossibility Results

The success of the GVA makes an interesting counterpoint to two impossibility results in the economic mechanism design literature, that prove that the GVA must fail in some important more general problems, e.g. when the auctioneer does not know the value of a seller.

Although proved for direct-revelation mechanisms via the revelation-principle, the impossibility theorems continue to hold for indirect revelation mechanisms, for example in application to iterative auctions.

First, the Gibbard-Sattherwaite impossibility theorem [Gib73, Sat75], that shows that it is impossible to implement efficient outcomes with individual-rational and incentive-compatible dominant-strategy mechanisms in a general class of problems. This impossibility theorem for allocation mechanism with agents that have general preferences was first stated by Hurwicz [Hur72].

THEOREM 5.4 (Gibbard-Sattherwaite Impossibility Theorem). *If agents can have any ordinal preferences*[5] *over the outcome of a mechanism, and there are at least three different optimal outcomes over the set of all agent preferences,* then *only dictatorial mechanism can be implemented in truth-revealing dominant strategy mechanisms.*

A mechanism is *dictatorial* if there is at least one agent that always receives one of its favorite alternatives. No dictatorial mechanism can be allocatively-efficient.

The generalized Vickrey auction does not violate the Gibbard-Sattherwaite Theorem because the positive result applies only to agents with a *restricted* class of utility functions: quasi-linear and private-values, such that they do not care about the payments or allocations received by other agents.

More importantly in our present context, given our assumptions of quasi-linear utility functions and private-values, is the *Myerson-Sattherwaite impossibility theorem* [MS83].[6] We extend our concerns to include the *sum value of the net payments made by agents* in a mechanism, and require that a mechanism is allocatively-efficient and *budget-balanced* such

[5] An ordinal preference specifies the ordering between different outcomes without quantifying the differences (that would be a cardinal preference)

[6] An earlier version is stated in Green & Laffont [GL79].

that the net payment is positive. The Myerson-Sattherwaite theorem applies to *bilateral trade*, when there are buyers and sellers, and within our present context states that:

THEOREM 5.5 (Myerson-Sattherwaite). *Whenever gains from trade are possible but not certain (sometimes states as agents have values from "overlapping" intervals) and trade is bilateral then there is* ***no*** *mechanism that is allocatively-efficient, budget-balanced, individual-rational and Bayes-Nash incentive-compatible.*

Clearly, dominant-strategy mechanisms are a special case, and interpreted in terms of the generalized Vickrey Auction (Groves-Clarke mechanism), Myerson-Sattherwaite states that in problems with multiple buyers and sellers, the auctioneer:

1. Can achieve allocative-efficiency, incentive-compatibility, and individual-rationality if it expects to make a net payment *to* the agents in some problems.

2. Can achieve allocative-efficiency, incentive-compatibility, individual-rationality, and budget-balance if and only if it knows the valuation functions of all sellers or all buyers.

3. Without knowledge of the valuation functions of all sellers, or all buyers, cannot expect to implement an auction that is allocatively-efficient, incentive-compatible, individual-rational, and budget-balanced.

To reemphasize the positive results for the GVA: when the auctioneer represents a single seller, and knows the reservation prices of the seller— or more simply, if the seller has no intrinsic value for items —then the GVA is allocatively-efficient, incentive-compatible, individual-rational *and* also budget-balanced. Every agent makes a non-negative payment to the auctioneer.

5.7 The Coase Theorem and Transaction Costs in E-Markets

Finally, it is interesting to make a few comments about the Coase [Coa60] in relation to the impossibility theorems of mechanism design. The point that I make here is that private information, for example about a seller's value for an item, represents a transaction cost and limits the efficiency of markets.

Coase [Coa60] asserts the irrelevance of the initial *ownership* of items, and states that efficient outcomes will always be achieved in a market with *zero transaction* costs. We might expect this assertion to hold, at least approximately, in "frictionless" electronic marketplaces, where agent-mediation and automated auctions can eliminate many traditional transaction and search costs.

However, there remains an important transaction cost: *the cost of private information and agent self-interest.* Coases' theorem assumes that all gains to trade can be detected, which is not true if agents can benefit from misstating information about their values for different outcomes. Therefore, one can take Coases' theorem in its application to electronic marketplaces in which transaction costs are otherwise small, and argue that *incentive-compatibility*, i.e. *truth-extraction* is one of the most important problems to be solved in order to make markets truly efficient. As noted by Myerson [Mye89], the *decentralization of information imposes an efficiency cost on allocation.*

Chapter 6

Primal-Dual Theory: Towards Iterative Mechanisms

In Chapter 5, I introduced the generalized Vickrey auction (GVA), and proved that it is an allocatively-efficient, incentive-compatible and individual-rational auction for the combinatorial resource allocation problem. In the light of its uniqueness (Theorem 5.3) and the revelation principle (Theorem 5.1) it might seem this is the perfect solution. Of course, this is not quite true.

First, the GVA is *not* bounded-rational compatible: an agent in the GVA must report its complete valuation function to the auctioneer, which is impossible if an agent with limited computation has a hard valuation problem and there are many possible bundles of items. Second, the GVA requires the solution of $\mathcal{NP}$-hard problems in the combinatorial resource allocation problem.

This chapter provides important background for my *i*Bundle ascending-price combinatorial auction. Primal-dual optimization theory provides a strong theoretical framework for the design of *iterative* auction mechanisms for the combinatorial resource allocation problem. In comparison to the GVA, iterative auctions are useful because they allow agents to compute incremental values for different outcomes in response to bids from other agents.

The connection between primal-dual theory and auction theory has been made before for simpler allocation problems, for example by Bertsekas [Ber90] for the assignment problem, but seems to be new for this problem. Bikchandani & Ostroy [BO98] introduce linear-programming models for the "package assignment problem" which have proved very influential in the application of primal-dual theory to the current problem.

Iterative auctions that follow from primal-dual theory are not automatically fully incentive-compatible. Instead, I weaken incentive-compatibility to *myopic* incentive-compatibility, and prove that an auction is allocatively-efficient with individual-rationality and agents that follow *myopic* utility-maximizing bidding strategies to current prices, ignoring future rounds of the auction.

6.1 Primal-Dual Optimization Theory

First, I provide a brief review of basic results in linear programming. See Papadimitriou [PS82] for more details.

Consider the linear program:

$$\begin{aligned} \max\ & c^T x \\ \text{s.t.}\quad & \mathrm{Ax} \leq b \\ & x \geq 0 \end{aligned} \tag{P}$$

where A is a $m \times n$ integer matrix, $x \in R^n$ is a n-vector, and c and b are $n-$ and m-vectors of integers. vectors are column-vectors, and notation c^T indicates the *transpose* of vector c, similarly for matrices. The primal problem is to compute a feasible solution for x that maximizes the value of the objective function.

The dual program is constructed as:

$$\begin{aligned} \min\ & b^T y \\ \text{s.t.}\quad & \mathrm{A^T y} \geq c \\ & y \geq 0 \end{aligned} \tag{D}$$

where $y \in R^m$ is a m-vector. The dual problem is to compute a feasible solution for y that minimizes the value of the objective function.

Let $V_{\mathrm{LP}}(x) = c^T x$, the value of feasible primal solution x, and $V_{\mathrm{DLP}}(y) = b^T y$, the value of feasible dual solution y.

The *weak duality theorem* of linear-programming states that the value of the dual always dominates the value of the primal:

THEOREM 6.1 (Weak-duality). *Given a feasible primal solution x with value $V_{\text{LP}}(x)$ and a feasible dual solution y with value $V_{\text{DLP}}(y)$, then $V_{\text{LP}}(x) \leq V_{\text{DLP}}(y)$.*

PROOF. Solution x is feasible, so $Ax \leq b$. Solution y is feasible, so $A^T y \geq c$. Therefore, $x \leq A^T b$ and $y \geq Ac$, and $c^T x \leq c^T A^T b = b^T AC \leq b^T y$, and $P \leq D$. ∎

The *strong duality theorem* of linear-programming states that primal and dual solutions are optimal if and only if the value of the primal equals the value of the dual:

THEOREM 6.2 (Strong-duality). *Primal solution x^* and dual solution y^* are a pair of optimal solutions for the primal and dual respectively, if and only if x^* and y^* are feasible (satisfy respective constraints)* and $V_{\text{LP}}(x^*) = V_{\text{DLP}}(y^*)$.

The strong-duality theorem of linear-programming can be restated in terms of *complementary-slackness* conditions (CS for short). In the context of auctions, in which agents' valuation functions are private information, complementary-slackness conditions provide a constructive technique to adjust primal and dual solutions towards optimality, given agents' bids.

DEFINITION 6.1 [Complementary-slackness] Complementary slackness conditions constrain pairs of primal and dual solutions. *Primal* CS conditions state $x^T(A^T y - c) = 0$, or in logical form:

$$x_j > 0 \Rightarrow A^j y = c_j \tag{P-CS}$$

where A^j denotes the jth column of A (written as a row vector to avoid the use of transpose). *Dual* CS conditions state $y^T(Ax - b) = 0$, or in logical form:

$$y_j > 0 \Rightarrow A_i x = b_i \tag{D-CS}$$

where A_i denotes the ith row of A.

The strong-duality theorem can be restated as the *complementary-slackness theorem*:

THEOREM 6.3 (Complementary-Slackness Theorem). *A pair of feasible primal, x, and dual solutions, y, are primal and dual optimal if and only if they satisfy the complementary slackness conditions.*

Proof. P-CS iff $x^T(A^Ty - c) = 0$, and D-CS iff $y^T(Ax - b) = 0$. Equating, and observing that $x^TA^Ty = y^TAx$, we have P-CS and D-CS iff $x^Tc = y^Tb$, or $c^Tx = b^Ty$. The LHS is the value of the primal, $V_{\mathrm{LP}}(x)$, and the RHS is the value of the dual, $V_{\mathrm{DLP}}(y)$. By the strong duality theorem, $V_{\mathrm{LP}}(x) = V_{\mathrm{DLP}}(y)$ is a necessary and sufficient condition for the solutions to be optimal. ∎

6.2 Primal-Dual Algorithms

A *primal-dual* algorithm computes solutions to the primal *and* dual formulations of a linear-program simultaneously, and searches for solutions that satisfy complementary slackness conditions and are therefore optimal by the strong-duality theorem of linear-programming.

In a standard primal-dual algorithm, the method is to maintain a feasible dual solution, y, and compute a solution to a *restricted primal problem*, given the dual solution. The restricted primal problem is typically formulated in one of the following ways:

1. Compute a feasible primal solution x' that minimizes the "violation" of complementary-slackness conditions with dual solution y.

2. Compute a primal solution x' that satisfies complementary slackness conditions with dual solution y, and minimizes the "violation" of feasibility constraints.

Figure 6.1 illustrates the variation in which the primal-dual algorithm maintains a feasible dual solution, y, and computes a primal solution x' to satisfy CS conditions with dual solution y and minimize feasibility violations. The algorithm terminates when the solution to the restricted primal is feasible, because the primal and dual solutions are optimal by the strong-duality theorem. Otherwise, the *dual of the restricted primal* is typically used to adjust the dual solution y so that the next restricted primal solution is "closer" to being feasible.

Primal-dual is often a useful algorithm-design paradigm for combinatorial optimization problems. Instead of solving a single hard primal solution, or a single hard dual solution, a primal-dual approach solves a sequence of restricted primal problems. Each restricted primal problem is often much simpler to solve than the full primal (or dual) problem

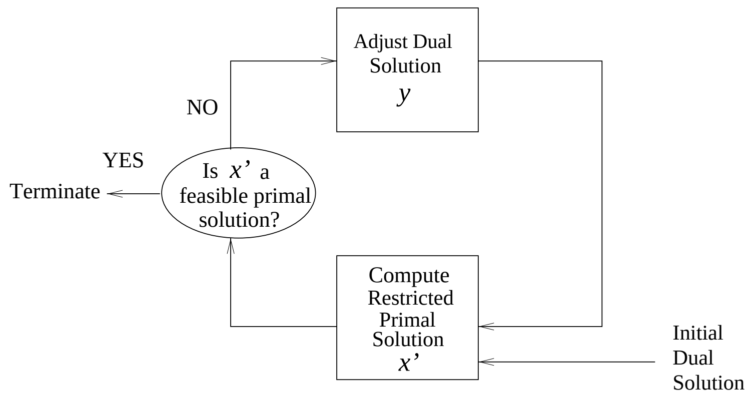


Figure 6.1: Primal-Dual Algorithm.

[PS82]. Often, it is possible to reduce a weighted optimization problem to a series of unweighted problems, that are often much easier to solve; e.g. for the assignment problem, the restricted primal becomes a max-flow problem (which in turn has a matching problem as *its* restricted-primal), these problems are successively easier to solve.

6.3 A Primal-Dual Interpretation of Ascending-Price Auctions

The Primal-Dual framework provides a methodology for the design and analysis of ascending-price auctions. The important difference between an auction-based algorithm and a traditional primal-dual algorithm is that: (a) agents are self-interested; and (b) agents have private information about their values.

A direct implication of this, for a primal problem that selects an allocation of items to maximize the sum value over agents, is that the auctioneer *cannot compute the value of the primal problem directly*. However, it is possible to test for complementary-slackness conditions based on agents' bids under reasonable conditions. An auction-based primal-dual algorithm can take the following form:

1. Compute feasible primal and dual solutions.

2. Test whether primal and dual solutions satisfy complementary-slackness conditions under a reasonable assumption about agents' bidding strategies.

3. Adjust primal and dual solutions towards solutions that satisfy complementary-slackness conditions based on agents' bids.

Figure 6.2 summarizes this interpretation of auction algorithms within primal-dual optimization theory. The primal solution corresponds to a provisional allocation, and the dual solution corresponds to prices for items or bundles of items.

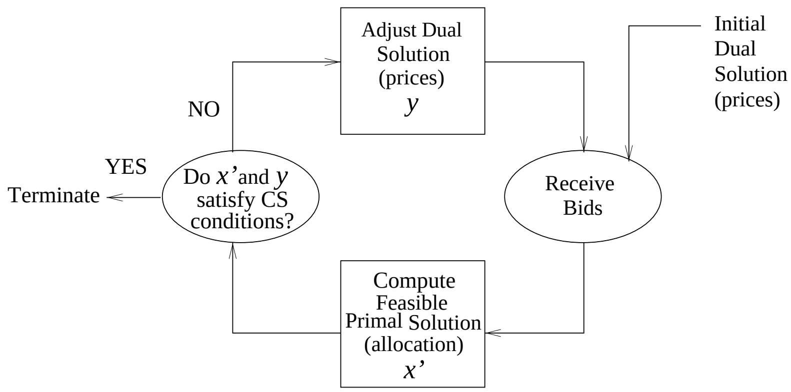


Figure 6.2: A Primal-Dual Interpretation of an Auction algorithm.

In the context of a market-based mechanism the final primal and dual solutions have natural interpretations as a *competitive-equilibrium* outcome. Intuitively, the final allocation maximizes the auctioneer's revenue and the agents' utilities at the final prices.

This section provides a top-level description of the primal and dual problems and complementary slackness conditions for linear-program formulations of an allocation problem. This leads immediately into an intuitive explanation of how an auction-based method can compute an optimal solution to an allocation problem based on agents' bids.

Definition 6.1 [Primal-problem] The primal allocation problem is to *allocate items* to agents to maximize the sum value over all agents, such that no item is allocated to more than one agent.

Definition 6.2 [Dual-problem] The dual allocation problem is to assign *prices* to items, or bundles of items, to minimize the sum of (i) each agents' maximum utility given the prices, over all possible allocations; *and* (ii) the maximum revenue over all possible allocations given the prices.

Clearly, without information on agents' values the auctioneer cannot compute an optimal primal or an optimal dual (because of term *(i)* in the dual). However, under a reasonable assumption about agents' bidding strategies (myopic best-response, see below) the auctioneer can *compute complementary-slackness conditions and adjust prices and the allocation towards optimal solutions.*

The complementary-slackness conditions between a feasible primal solution, allocation x, and a feasible dual solution, prices p, state:

(CS-1) Agents receive a bundle if and only if the bundle maximizes their utility given the prices, and has non-negative utility.

(CS-2) The allocation maximizes the auctioneer's revenue given the prices.

The details of *what the prices are*, i.e. are they linear-prices, bundle prices, bundle and discriminatory prices, and *what it means to maximize revenue at the prices*, are defined with respect to a particular linear-program formulation. I provide details for the combinatorial resource allocation problem later in this chapter, as I introduce linear-program formulations for that problem.

Duality theory can be leveraged for auction-design under a reasonable assumption about agents' bidding strategies.

DEFINITION 6.3 [Myopic Best-Response] A myopic best-response bidding strategy is to bid for all items or bundles of items that maximize utility at the current prices.

The assumption provides auctions that are often *not* fully incentive-compatible, and leave the possibility that an agent can manipulate the outcome of the auction with untruthful bidding. In special cases the final prices in the auction are also payments in the Vickrey auction, or can be adjusted towards payments in the Vickrey auction (see Chapter 9). When this occurs the myopic best-response strategy is provably optimal for strategically-rational agents. For example, Demange, Gale & Sotomayor (DGS) [DGS86] propose an iterative auction that computes Vickrey payments in the *assignment problem*, in which agents demand at most one item.

In the case of agents that want at most one item, or at most one bundle, as in the combinatorial resource allocation problem, an agent's best-response bidding strategy can be expressed by submitting an exclusive-or bid (XOR) for every bundle that maximizes utility. This is supported in the *i*Bundle ascending-price auction (see Chapter 7).

The auctioneer can test (CS-1) because the allocation must maximize *revenue* and the auctioneer knows the prices of items (recall that the auctioneer cannot test directly whether an allocation maximizes value). In addition, with best-response bidding strategies the auctioneer can test (CS-2) because every agent maximizes utility with allocation x given prices p if every agent that bids is allocated one of the bundles in its bid.

Primal-dual theory provides a road-map for auction-design, suggesting how to adjust prices and compute allocations in response to bids from agents, so that the auction terminates with an optimal allocation.

6.4 Primal-Dual Optimality in Auctions

With careful price-adjust rules and bidding-rules the auctioneer can compute a sequence of primal and dual solutions based on agents' bids that eventually satisfy complementary-slackness conditions, and terminate with an optimal solution.

I will consider the following specific instantiation of the general auction algorithm illustrated in Figure 6.2. The steps are intended to mirror what one typically expects in an auction.

1. Receive Bids.

2. Compute an allocation that: (a) only allocates bundles of items to agents that place bids for those bundles; and (b) maximizes the auctioneer's revenue at the current prices.

3. Terminate if every agent that bids receives a bundle, or agents bids are unchanged in two successive rounds.

4. Increase prices on bundles that receive bids.

Non-negative prices correspond to a feasible dual solution, and the allocation computed from agents' bids corresponds to a feasible primal solution. The termination condition is equivalent to testing for complementary-slackness between the allocation and the prices, given agents that follow myopic best-response bidding strategies. Left to define, and the challenging part, is to design a price-update rule based on bids placed by agents, the current prices, and the current allocation to move the primal and dual solutions closer to optimal solutions.

It is useful to provide two intuitive descriptions of how to adjust prices. The essential concept is that the auctioneer maintains feasible primal and dual solutions, and adjusts solutions to reduce the "violation" of the complementary-slackness conditions.

Adjust prices to achieve complementary-slackness conditions

The first primal-dual interpretation is that an auctioneer should *increase* prices on bundles that agents bid for in each round to maintain (CS-1), and to achieve (CS-2) when the auction terminates.

Proposition 6.1 *The auction will terminate with an optimal allocation if: the auctioneer increase prices on at least some bundles that receive bids,* and *increases prices so that the allocation computed from agents' bids in the next round will maximize the auctioneer's revenue over all possible allocations.*

Increasing prices on bundles which receive bids must eventually give (CS-2), so that every agent with positive utility for some bundle receives a bundle in the allocation. If the auctioneer can always maximize revenue at the current prices by allocating bundles according to bids received from agents then (CS-1) is true.

Intuitively, the auctioneer should *only increase prices on bundles if it is sure that it will receive bids from agents for one of the revenue-maximizing allocations in the next round.*

Adjust prices to monotonically reduce the value of the dual

Recall that when complementary-slackness conditions are satisfied, the allocation x and prices y represent primal and dual solutions with equal value, such that $V_{\mathrm{LP}}(x) = V_{\mathrm{DLP}}(y)$. The degree to which the complementary-slackness conditions are violated is represented by the *duality-gap* $V_{\mathrm{DLP}}(y) - V_{\mathrm{LP}}(x)$ during the auction.

Prices can be updated to ensure that the *value of the dual is monotonically decreasing* throughout the auction.[1] The weak-duality theorem states $V_{\mathrm{DLP}}(y) \geq V^*_{\mathrm{LP}}$, for the value V^*_{LP} of the optimal primal solution. Therefore, eventually the value of the dual equals the value of the optimal primal solution, and complementary slackness conditions are satisfied. This is illustrated in Figure 6.3.

[1]The duality gap between the primal and dual solutions is *not* necessarily monotonically decreasing.

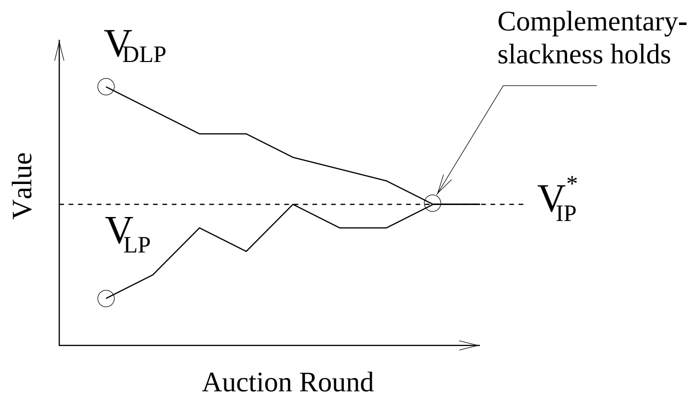


Figure 6.3: Primal-Dual Interpretation of an Ascending-Price Auction.

Although the auctioneer cannot compute the value of the dual directly, it is still possible to provably reduce the value of the dual, relative to its current value. The value of the dual is the sum of the auctioneer's maximum revenue and each agent's maximum utility at the current prices. The value of the dual will decrease across rounds of the auction if and only if:

the increase in revenue at the new prices is less than the decrease in total utility across all agents.

Intuitively, this occurs when the auctioneer increases the price on overdemanded items (or bundles of items) such that its increase in revenue at the new prices is less than the *total* decrease in maximum utility over all agents that were bidding for the bundles.

6.4.1 Example: The English Auction

The standard English auction illustrates the primal-dual framework for auction design. The English auction is an ascending-price auction for single items, where the price increases as long as more than one agent bids at the current price.

Let v_i denote agent i's value for the item. The single-unit resource allocation problem is:

$$\max \sum_i v_i x_i \qquad [\text{IP}_{\text{single}}]$$

$$\text{s.t.} \quad \sum_{\text{i}} \text{x}_{\text{i}} \leq 1$$

$$x_i \in \{0, 1\}$$

where $x_i = 1$ if and only if agent i is allocated the item, i.e. the goal is to allocate the item to the agent with the highest value. This can be solved as a linear-program, [LP$_{\text{single}}$], relaxing the integral constraint

$$\max \sum_i v_i x_i \qquad [\text{LP}_{\text{single}}]$$

$$\text{s.t.} \quad \sum_{\text{i}} \text{x}_{\text{i}} \leq 1$$

$$x_i \geq 0$$

and $V^*_{\text{LP}} = V^*_{\text{IP}}$, i.e. there is always an integral optimal solution to the relaxed problem. The dual formulation, [DLP$_{\text{single}}$], is

$$\min \pi \qquad [\text{DLP}_{\text{single}}]$$

$$\text{s.t.} \quad \pi \geq v_i, \quad \forall i$$

$$\pi \geq 0$$

The complementary-slackness conditions are

$$\begin{aligned} \sum x_i \geq 0 \quad &\Rightarrow \quad \pi = v_i, \qquad \forall i \\ \pi > 0 \quad &\Rightarrow \quad \sum x_i = 1 \end{aligned}$$

The English auction maintains price p on the item, initially $p = 0$. Agent i bids whenever $p < v_i$, and the provisional allocation sets $x_j = 1$ for one of the agents that bids in each round, and increases the price p whenever more than one agent bids.

Let the provisional allocation define a feasible primal solution, and the price define dual solution $\pi = \sum_i \max\{0, v_i - p\} + p$. This is feasible, $\pi \geq \max\{0, v_i - p\} + p \geq v_i$ for all agents i.

Assume that agents follow a myopic best-response bidding strategy, bidding for the item at the ask price whenever the price is below their value. The optimality of the English auction can be understood in two different ways:

- The English auction terminates with primal and dual solutions that satisfy CS-1 and CS-2.

  Clearly, CS-2 is satisfied throughout the auction because the item is always allocated to one of the agents. CS-1 is satisfied when the auction terminates. Let j indicate the only agent that bids at price p. Therefore $v_i - p \leq 0$ for all agents $i \neq j$ and $v_j - p \geq 0$ for agent j (because agents follow best-response bidding strategies), and $\pi = \sum_i \max\{0, v_i - p\} + p = \max\{0, v_j - p\} + p = v_j$.

- The value of the dual strictly decreases in each round of the auction. Let $m > 1$ equal the number of agents that bid in each round of the auction except the final round. For price increment ϵ, the sum maximal utility to the agents decreases by $m\epsilon$ and the maximal revenue to the auctioneer increases by ϵ, for a net change in π of $-(m-1)\epsilon$.

In fact, the English auction is a incentive-compatible, individual-rational and allocatively-efficient mechanism for the single-item allocation problem because the final price is equal to the Vickrey payment (to within ϵ).

The ascending-price combinatorial auction, *i*Bundle, computes allocatively-efficient outcomes with agents that follow myopic best-response bidding strategies. The proof is similar in structure to the proof of the English auction.

In general the final prices in *i*Bundle are *not* Vickrey payments, and *i*Bundle is not fully incentive-compatible. In Chapter 9 I introduce a method to adjust prices after *i*Bundle terminates towards payments in the generalized Vickrey auction. When successful, and with proxy bidding agents, I explain that this makes *i*Bundle an iterative equivalent of the GVA.

6.5 Linear Program Formulations for the Combinatorial Resource Allocation Problem

In the previous sections I presented primal-dual theory for *linear programs.* However, the combinatorial resource allocation problem, as formulated in [IP] (Chapter 5) is an integer program. In addition, a direct linear relaxation of [IP], replacing the integral constraints $x_i(S) \in \{0,1\}$ with non-negativity constraints will compute *fractional* allocations of items to agents.

This section introduces a hierarchy of linear program formulations for the combinatorial resource allocation problem, from Bikchandani & Ostroy [BO98]. The models provide essential background to a primal-dual interpretation of *i*Bundle with myopic best-response agents.

At each level, more problem instances are solved without fractional allocations of items as a linear-program formulation. Each formulation introduces new primal constraints, and new dual variables. The dual variables price bundles linear in the price of items, i.e. $p(S) = \sum_{j \in S} p(j)$, where $p(j)$ is the price of item j in bundle S, then price bundles directly (and possibly non-linearly), and then price discriminate, with different prices for the same bundle to different agents.

It is always possible to add enough constraints to a linear program relaxation to make the optimal solution integral, this follows from general duality results [Wol81a, Wol81b, TW81].

6.5.1 Competitive Equilibrium

There is a direct interpretation of the complementary-slackness conditions (CS-1) and (CS-2) in terms of *competitive equilibrium prices* when the optimal primal solution is integral. In the usual sense of competitive-equilibrium, by CS-1, every agent maximizes utility given the allocation and the prices, and by CS-2 the auctioneer maximizes its revenue.

Moreover, the standard allocative-efficiency properties of competitive-equilibrium prices and allocations hold, by primal-dual theory:

Theorem 6.4 *An allocation x is efficient if and only if there exists competitive equilibrium prices p, for an appropriate type of prices (e.g. linear, bundle, discriminatory).*

This is a different result from that in Wurman & Wellman [WW99b], because it uses a stronger statement of the auctioneer's revenue maximization (CS-2). Wurman & Wellman under their definition that *non-discriminatory* equilibrium bundles prices always exist to price the efficient allocation, but that it is also possible that competitive equilibrium prices support suboptimal allocations. A full primal-dual analysis, as provided in Bikchandani & Ostroy, leads to the optimality of competitive equilibrium, following directly from the strong duality theorem of linear programming.

Looking ahead, an intuitive interpretation of the "safe-bids" test in *i*Bundle to dynamically decide whether to introduce price-discrimination is that it characterizes all problems in which no competitive-equilibrium prices exist with bundle but not discriminatory prices.

Bikchandani & Ostroy [BO98] prove that discriminatory bundle prices always exist to price the efficient allocation in competitive equilibrium (although the non-discriminatory prices are often sufficient, and sometimes linear prices are sufficient).

6.5.2 A Hierarchy of Linear Programs for the Combinatorial Resource Allocation Problem

Let V^*_{LP} and V^*_{IP} denote the optimal values linear and integer programs. In general, given a relaxation [LPR] of an integer program [IP], we have $V^*_{\mathrm{LPR}} \geq V^*_{\mathrm{IP}}$.

For the combinatorial resource allocation problem the optimal solution to a direct linear program [LP_1] relaxation of [IP] (Chapter 5) allocates fractional items to agents, and $V^*_{\mathrm{LP}_1} \geq V_{\mathrm{IP}}$. Linear programs LP_2 and LP_3, each more constrained than LP_1, can be formulated [BO98] such that $V^*_{\mathrm{LP}_1} \geq V^*_{\mathrm{LP}_2} \geq V^*_{\mathrm{LP}_3}$.

Furthermore, Bikchandani & Ostroy [BO98] prove that $V^*_{\mathrm{LP}_3} = V^*_{\mathrm{IP}}$ in all instances of the combinatorial resource allocation problem. An immediate implication is that competitive-equilibrium prices always exist, although they might be discriminatory bundles prices, which correspond to the variables in the dual of LP_3.

Linear Prices

The simplest primal and dual linear program relaxations of the combinatorial resource allocation problem [IP], replacing integral constrains $x_i(S) \in \{0, 1\}$ with non-negativity $x_i(S) \geq 0$, are:

	A	B	AB
Agent 1	0	0	3
Agent 2	2^*	0	2
Agent 3	0	2^*	2

Table 6.1: Problem 1.

$$\max_{x_i(S)} \sum_S \sum_i x_i(S) v_i(S) \qquad [\mathrm{LP}_1]$$

$$\text{s.t.} \quad \sum_{\mathrm{S}} \mathrm{x_i(S)} \leq 1, \qquad \forall i \qquad (\mathrm{LP}_1\text{-}1)$$

$$\sum_{S \ni j} \sum_i x_i(S) \leq 1, \qquad \forall j \qquad (\mathrm{LP}_1\text{-}2)$$

$$x_i(S) \geq 0, \qquad \forall i, S$$

$$\min_{p(i), p(j)} \sum_i p(i) + \sum_j p(j) \qquad [\mathrm{DLP}_1]$$

$$\text{s.t.} \quad \mathrm{p(i)} + \sum_{\mathrm{j \in S}} \mathrm{p(j)} \geq v_i(S), \qquad \forall i, S \qquad (\mathrm{DLP}_1\text{-}1)$$

$$p(i), p(j) \geq 0, \qquad \forall i, j$$

With substitution $p(i) = \max_S \left\{ v_i(S) - \sum_{j \in S} p(j) \right\}$ a feasible dual solution is defined by *prices* $p(j)$ on items $j \in G$, and the value of the dual has an intuitive interpretation:

$V_{\mathrm{DLP}_1}(p(j))$ *is the sum of the maximal utility to each agent with bundles priced equal to the sum of the value of their items* plus *the auctioneer's revenue for selling every item.*

The dual variables correspond to linear prices on bundles, defining prices on items $p(j)$ for item j, and are *competitive equilibrium prices* if and only if $V^*_{\mathrm{LP}_1} = V^*_{\mathrm{IP}}$, i.e. the optimal primal allocation is integral, and maximizes agents' utility and the auctioneer's revenue at the prices. Kelso & Crawford [KC82] prove that gross substitutes preferences (see Chapter 3) are a sufficient condition for linear competitive equilibrium prices, when $V^*_{\mathrm{LP}_1} = V^*_{\mathrm{IP}}$.

Problem 1 in Table 6.1 can be solved with LP_1; $V^*_{\mathrm{LP}_1} = V_{\mathrm{IP}} = 4$. The optimal allocation is $x_2(A) = 1$ and $x_3(B) = 1$, indicated by *. To see that $V_{\mathrm{LP}_1} \leq 4$, notice that dual prices $p(A) = p(B) = 1.6$ gives a dual solution with value $V_{\mathrm{DLP}_1} = 0 + 0.4 + 0.4 + 3.2 = 4$. Remember that $V^*_{\mathrm{LP}_1} \leq V_{\mathrm{DLP}_1}$ by the weak-duality theorem of linear programming.

	A	B	C	AB	BC	AC	ABC
Agent 1	60	50	50	200*	100	110	250
Agent 2	50	60	50	110	200	100	255
Agent 3	50	50	75*	100	125	200	250

Table 6.2: Problem 2.

Bundle Prices

However, in general the value $V^*_{\mathrm{LP}_1} > V_{\mathrm{IP}}$ and the optimal primal solution makes fractional assignments to agents. For example, consider the Problem 2 in Table 6.2.

In this problem $V^*_{\mathrm{LP}_1} = 300 > V^*_{\mathrm{IP}} = 275$. The primal allocates fractional solution $x_1(AB) = 0.5, x_2(BC) = 0.5$ and $x_3(AC) = 0.5$, which satisfies constraints (LP_1-1) because $\sum S \ni j \sum_i x_i(S) \leq 1$ for all items $j \in G$. Prices $p(A) = p(B) = p(C) = 100$ solve the dual problem DLP_1.

Introducing new constraints to the "first-order" linear-program relaxation [LP_1] of [IP], gives a "second-order" linear-program [LP_2], with dual [DLP_2]. The corresponding dual variables to the new primal constraints are interpreted as *bundle prices* within an auction-based primal-dual algorithm.

$$\max_{x_i(S),y(k)} \sum_S \sum_i x_i(S) v_i(S) \qquad [\mathrm{LP}_2]$$

$$\text{s.t.} \quad \sum_{\mathrm{S}} \mathrm{x_i(S)} \leq 1, \qquad \forall i \qquad (\mathrm{LP}_2\text{-}1)$$

$$\sum_i x_i(S) \leq \sum_{k \ni S} y(k), \qquad \forall S \qquad (\mathrm{LP}_2\text{-}2)$$

$$\sum_k y(k) \leq 1 \qquad (\mathrm{LP}_2\text{-}3)$$

$$x_i(S), y(k) \geq 0, \qquad \forall i, S, k$$

$$\min_{p(i),p(S),\pi} \sum_i p(i) + \pi \qquad [\mathrm{DLP}_2]$$

$$\text{s.t.} \quad \mathrm{p(i) + p(S)} \geq v_i(S), \qquad \forall i, S \qquad (\mathrm{DLP}_2\text{-}1)$$

$$\pi - \sum_{S \in k} p(S) \geq 0, \qquad \forall k \qquad (\mathrm{DLP}_2\text{-}2)$$

$$p(i), p(S), \pi \geq 0, \qquad \forall i, S$$

where $k \in K$ is a *partition* of items in set K, and $k \ni S$ indicates that bundle S is represented in partition k. A partition is a feasible "bundling" of items, e.g. $[A, B, C]$ or $[AB, C]$, etc., and K is the set of all possible partitions, e.g. $K = \{[A, B, C], [AB, C], [A, BC], \ldots, [ABC]\}$ in Problem 2 (Table 6.2).

Constraints (LP_2-2) and (LP_2-3) replace constraints (LP_1-1), and ensure that no more than one unit of every item is allocated. The dual [DLP_2] has variables $p(i)$, $p(S)$ and π, which correspond to constraints (LP_2-1), (LP_2-2) and (LP_2-3), and constraints (DLP_2-1) and (DLP_2-2) correspond to primal variables $x_i(S)$ and $y(k)$.

Dual variables $p(S)$ can be interpreted as bundle prices, and with substitution $p(i) = \max_S \{v_i(S) - p(S)\}$, i.e. the maximal utility to agent i at prices $p(S)$, and $\pi = \max_{k \in K} \sum_{S \in k} p(S)$, i.e. the maximal revenue to the auctioneer at prices $p(S)$, then the value of the dual has an intuitive interpretation:

$V_{\text{DLP}_2}(p(S))$ *is the sum of the maximal utility to each agent with bundle prices* $p(S)$ plus *the auctioneer's maximal revenue over all feasible allocations at the prices.*[2]

The dual variables correspond to bundle prices, $p(S)$, and are *competitive equilibrium prices* if and only if $V^*_{\text{LP}_2} = V^*_{\text{IP}}$, i.e. the optimal primal allocation is integral, and maximizes agents' utility and the auctioneer's revenue at the prices. I have shown via *i*Bundle that sufficient conditions for competitive equilibrium bundle prices include: (1) agents have additive or superadditive values, i.e. $v(S \cup S') \geq v(S) + v(S')$ for non-conflicting bundles S and S'; and (2) agents demand bundles from the same partition of items, e.g. all bids are for pairs of matching shoes, or single items.

It remains possible that $V^*_{\text{LP}_2} > V^*_{\text{IP}}$, with the optimal primal allocation assigning fractional bundles to agents. However, with the new constraints $V^*_{\text{LP}_2} < V^*_{\text{LP}_1}$ and $V^*_{\text{DLP}_2} < V^*_{\text{DLP}_1}$ in some problems, i.e. the second-order linear program solves some problems not solved by the first-order linear program. Problem 2 is such an example.

Allocation $x_1(AB) = x_2(BC) = x_3(AC) = 0.5$ is *not* feasible in [LP_2] because it is not possible to allocate $y(k_1) = y(k_2) = y(k_3) = 0.5$ for $k_1 = [AB, C], k_2 = [AC, B]$ and $k_3 = [AB, C]$ without violating constraint (LP_2-3) and without this we violate constraints (LP_2-2). [LP_2] solves Problem 2, with $V^*_{\text{LP}_2} = V^*_{\text{IP}} = 275$. An optimal dual solution is given by bundle prices $p = (50, 60, 75, 190, 200, 200, 255)$, with total agent maximal utility $10 + 0 + 0$ and maximal auctioneer revenue $75 + 190 = 265$, i.e. $V_{\text{DLP}_2} = 275$ (again, this

[2]Where an allocation is 'feasible' if it allocates no fractional items, and no item more than once.

	A	B	AB
Agent 1	0	0	3^*
Agent 2	2	2	2

Table 6.3: Problem 3.

proves that [LP_2] solves Problem 2 by the weak-duality theorem of linear-programming).

Discriminatory Bundle Prices

However, in general the value $V^*_{\text{LP}_2} > V_{\text{IP}}$ and the optimal primal solution makes fractional bundle assignments to agents. For example, consider Problem 3 in Table 6.3.

In this problem $V^*_{\text{LP}_2} = 3.5 > V^*_{\text{IP}} = 3$. The primal allocates fractional bundles $x_1(AB) = 0.5$ and $x_2(A) = x_2(B) = 0.5$, which satisfies constraints (LP_2-2) and (LP_2-3) with $y(k_1) = y(k_2) = 0.5$ for partitions $k_1 = [AB, \emptyset]$ and $k_2 = [A, B]$. Prices $p(A) = 1.5, p(B) = 1.5, p(AB) = 3$ solves the dual problem DLP_2.

Introducing new constraints to the second-order linear-program relaxation [LP_2] of [IP] gives a "third-order" linear-program [LP_3], with dual [DLP_3]. The corresponding dual variables to the new primal constraints are interpreted as *discriminatory bundle prices*, with different prices for the same bundle to different agents.

$$\max_{x_i(S),y(k)} \sum_S \sum_i x_i(S) v_i(S) \qquad [\mathrm{LP}_3]$$

$$\text{s.t.} \quad \sum_S x_i(S) \leq 1, \quad \forall i \qquad (\mathrm{LP}_3\text{-}1)$$

$$x_i(S) \leq \sum_{k \ni [i,S]} y(k), \quad \forall i, S \qquad (\mathrm{LP}_3\text{-}2)$$

$$\sum_k y(k) \leq 1 \qquad (\mathrm{LP}_3\text{-}3)$$

$$x_i(S), y(k) \geq 0, \quad \forall i, S, k$$

$$\min_{p(i),p_i(S),\pi} \sum_i p(i) + \pi \qquad [\mathrm{DLP}_3]$$

$$\text{s.t.} \quad p(i) + p_i(S) \geq v_i(S), \quad \forall i, S \qquad (\mathrm{DLP}_3\text{-}1)$$

$$\pi - \sum_{[i,S] \in k} p_i(S) \geq 0, \quad \forall k \qquad (\mathrm{DLP}_3\text{-}2)$$

$$p(i), p_i(S), \pi \geq 0, \quad \forall i, S$$

where $k \ni [i, S]$ indicates that *agent-partition* $k \in K'$ contains bundle S designated for agent i. Variable $y(k)$ in [LP$_3$] corresponds to an *agent-partition* k, where the set of agent-partitions in Problem 3 is $K' = \{[(1, A), (2, B)], [(1, B), (2, A)], [(1, AB), (2, \emptyset)], [(1, \emptyset), (2, AB)]$. It is important to note that each agent can receive at most one bundle in a particular agent-partition.

The dual variables $p_i(S)$ which correspond to primal constraints (LP$_3$-2) are interpreted as *discriminatory bundle prices*, price $p_i(S)$ is the price to agent i for bundle S. As before, substitutions $p(i) = \max_S \{v_i(S) - p_i(S)\}$, i.e. the maximal utility to agent i at individual prices $p_i(S)$, and $\pi = \max_{k \in K'} \sum_{[i,S] \in k} p_i(S)$, i.e. the maximal revenue to the auctioneer at prices $p_i(S)$ given that it can allocate at most one bundle at prices $p_i(S)$ to each agent i, give the value of the dual an intuitive interpretation:

$V_{\mathrm{DLP}_3}(p_i(S))$ *is the sum of the maximal utility to each agent with bundle prices* $p_i(S)$ plus *the auctioneer's maximal revenue over all feasible allocations at the prices.*[3]

[3]Where an allocation is 'feasible' if it allocates no more than one bundle to each agent, no fractional items, and no item more than once.

The dual variables correspond to discriminatory bundle prices, $p_i(S)$, and are *competitive equilibrium prices* if and only if $V^*_{\text{LP}_3} = V^*_{\text{IP}}$ and the optimal primal allocation is integral, i.e. maximizes agents' utility and the auctioneer's revenue at the prices (where the auctioneer can sell no more than one bundle to each agent).

Bikchandani & Ostroy [BO98] prove this important theorem:

THEOREM 6.5 *The optimal solution to linear program [LP$_3$] is always integral, and therefore an optimal resource allocation; i.e.* $V_{\text{LP}_3} = V_{\text{DLP}_3} = V_{\text{IP}}$.

Therefore, *there are always competitive equilibrium bundles prices*, that maximize the agents' utility and the auctioneers's revenue given the allocatively-efficient allocation, *although these prices may need to price-discriminate between agents.*

Consider Problem 3. Allocation $x_1(AB) = 0.5$ and $x_2(A) = x_3(B) = 0.5$ is *not* feasible in [LP$_3$] because $y(k_1) = y(k_2) = y(k_3) = 0.5$ for $k_1 = [(1, AB), (2, \emptyset)], k_2 = [(1, A), (2, B)]$ and $k_3 = [(1, B), (2, A)]$ violates constraint (LP$_3$-3), but without this constraints (LP$_3$-2) are violated. In this problem $V^*_{\text{LP}_3} = V^*_{\text{IP}} = 3$. To see this, consider bundle prices $p_1 = (0, 0, 2.5)$ and $p_2 = (2, 2, 2)$, for which the value of the dual is $0.5 + 0 + 2.5 = 3$, providing $V_{\text{LP}_3} \leq 3$ by the weak-duality theorem of linear-programming.

6.6 Auctions and Duality Theory in the Combinatorial Resource Allocation Problem

The primal-dual programs introduced in the previous section relate to an iterative auction for the combinatorial resource allocation problem.

Figure 6.4 illustrates the outcome of a regular primal-dual algorithm with a linear-program formulation [LPR] that is too relaxed for a problem instance, such that $V^*_{\text{LPR}} > V^*_{\text{IP}}$. The primal-dual algorithm terminates with a fractional primal solutions, that is an infeasible allocation.

In an auction-based primal dual algorithm the primal solution remains *feasible* to [IP] in each round of the auction, because it is computed to maximize revenue over bids received from agents in each round. Therefore, if the linear-program formulation is too relaxed, when the auction terminates the final primal solution does not satisfy complementary-slackness conditions with the final dual solution, and we do not know whether it is an

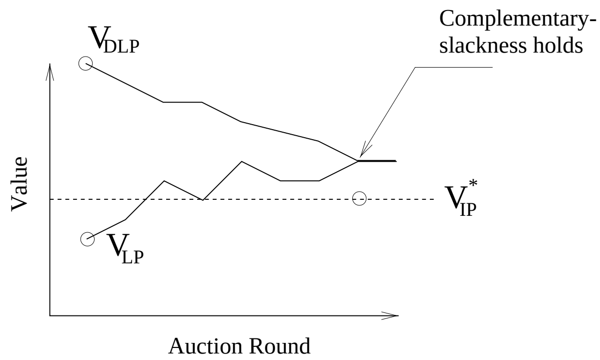


Figure 6.4: Regular Primal-Dual Algorithm, with $V^*_{\mathrm{LPR}} > V^*_{\mathrm{IP}}$

optimal allocation. Figure 6.5 illustrates the situation, with the final (integral) primal solution possibly sub-optimal.

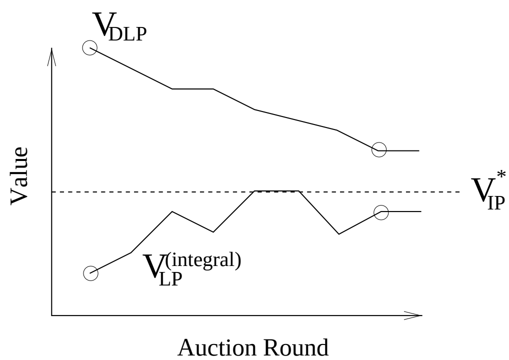


Figure 6.5: Auction-based Primal-Dual Algorithm, with $V^*_{\mathrm{LPR}} > V^*_{\mathrm{IP}}$

Finally, Figure 6.6 indicates what is the ideal case. The auction implements a primal-dual algorithm for a linear-program formulation that is not too relaxed, i.e. with prices that are non-linear and discriminatory if necessary. Taking the combinatorial resource allocation problem as an example, this scenario occurs when the prices correspond to an optimal solution to DLP_2 if $V^*_{\mathrm{LP}_1} > V^*_{\mathrm{LP}_2} = V^*_{\mathrm{IP}}$, or to an optimal solution to DLP_3 if $V^*_{\mathrm{LP}_2} > V^*_{\mathrm{LP}_3} = V^*_{\mathrm{IP}}$.

In Parkes & Ungar [PU00b] I prove that *i*Bundle computes optimal allocations. The auction uses non-discriminatory bundle-prices where possible, but introduces price-discrimination when it is possible that $V^*_{\mathrm{LP}_2} > V^*_{\mathrm{IP}}$.

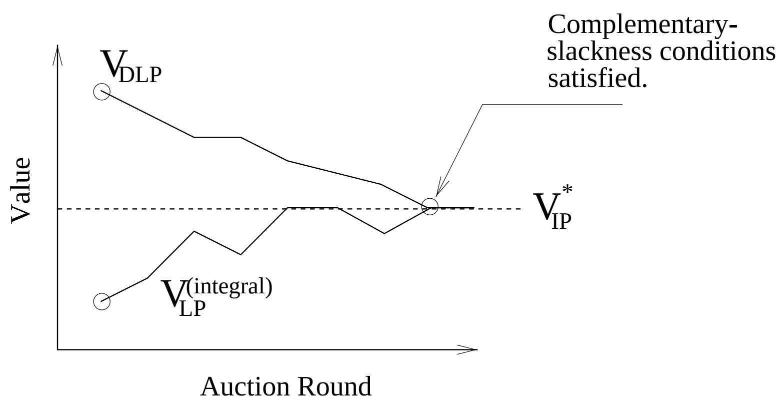


Figure 6.6: Auction-based Primal-Dual Algorithm, with $V^*_{\mathrm{LPR}} = V^*_{\mathrm{IP}}$

Chapter 7

*i*Bundle: An Iterative Bundle Auction

*i*Bundle is the first iterative bundle auction that computes optimal allocations for any reasonable bidding strategy. *i*Bundle[1] is an ascending-price auction. Agents can bid in each round of the auction. The auctioneer solves a sequence of winner-determination problems (one in each round), maintains a provisional allocation, and updates prices.

This chapter summarizes work presented in the attached papers i*Bundle: An efficient ascending-price bundle auction* [Par99b], and *Iterative combinatorial auctions: Theory and Practice* [PU00b].

The combinatorial resource allocation problem is $\mathcal{NP}$-hard. It is therefore not surprising that there are some hard computational problems embedded within *i*Bundle and the GVA. The auctioneer's winner-determination problems in both auctions is $\mathcal{NP}$-hard, by reduction from the maximal weighted-clique problem.

In *i*Bundle the auctioneer solves one winner-determination problem in each round of the auction, to compute a new provisional allocation based on adjusted bids from agents. In the GVA the auctioneer solves $O(I)$ problems, for I agents.

Although *i*Bundle is not incentive-compatible, but optimal only for myopic agents, it has a number of computational advantages over the generalized Vickrey auction:

- It is an ascending price auction, allowing agents to adjust their bids and compute incremental values for items in response to bids from other agents; i.e. it is bounded-rational compatible. In comparison, the GVA is a sealed-bid auction in which an agent's optimal strategy is to bid for, and compute the value of, *all* bundles with positive value. This is often impossible, since for G items there are 2^G bundles to value, each of which may require solving a difficult optimization problem.

[1] *i*Bundle has three variations, that differ in their price update rules. I use *i*Bundle both to refer to the family of auctions in general, and *i*Bundle(d) in particular, described below.

- The winner-determination problems in *i*Bundle tend to be smaller than in the GVA, because the agents bid for less bundles in each round of *i*Bundle than in the GVA. In addition, the provisional allocations computed by the auctioneer in each round of *i*Bundle can be cached and used to reduce computation time in future rounds.

- The auctioneer can reduce the number of rounds in the auction, and the number of winner-determination problems to solve, by increasing the minimal bid increment: which controls the speed with which prices are increased. This trades off computation with economic efficiency.

- The auctioneer can introduce approximate winner-determination algorithms, providing orders-of-magnitude speed-ups. Algorithms that satisfy *bid-monotonicity* maintain the auction's "myopic incentive-compatibility".

At present *i*Bundle has weaker truth-revelation properties than the GVA, which is an incentive-compatible auction (as described in Chapter 5). It is possible that agents in *i*Bundle can manipulate the outcome of the auction. However, in recent work I have proposed a technique to make the auction more robust to manipulation, by price-adjustment towards the outcome of the GVA after the auction terminates. This method is introduced in Chapter 9.

7.1 Auction Description

Recall that G denotes the set of items to be auctioned, I denote the set of agents, and $S \subseteq G$ denote a bundle of items. The auction proceeds in rounds, indexed $t \geq 1$. I describe the types of *bids* that agents can place, and the *allocations* and *price updates* computed by the auctioneer.

[Bids.] Agents can place exclusive-or bids for bundles, e.g. S_1 XOR S_2, to indicate than an agent wants either S_1 *or* S_2 but not both. Exclusive-or bids provide complete expressibility, but are not necessarily computationally efficient for all problems. The price-update rules for other bid languages, such as additive-or bids, can be derived from the rule for XOR bids.

Agent i associates a *bid price* $p^t_{\mathrm{bid},i}(S)$ with a bid for bundle S, non-negative by definition. The price must either be within ϵ of, or greater than, the ask price announced by the auctioneer (see below). Parameter ϵ defines the *minimal bid increment*, the minimal price increase in the auction. Agents must repeat bids for bundles in the current allocation, but can repeat the same bid price even if the ask price has increased since the previous round.[2]

[**Winner-determination.**] The auctioneer solves the *winner-determination* (WD) problem to compute a new allocation in each round, based on bids received from agents. The auctioneer allocates bundles to agents to maximize revenue, respecting XOR bid constraints and without allocating any item more than once. A number of algorithms have been proposed to solve this problem [RPH98, San99, FLBS99, ATY00]. The provisional allocation becomes the final allocation when the auction terminates.

[**Approximate Winner Determination.**] The auctioneer can also use an approximate winner-determination algorithm, and still maintain the same incentives for agents to follow best-response bidding strategies. This holds so long as the WD algorithm has the *bid-monotonicity* property, such that if an agent is allocated a bundle with bids $\mathcal{B}$ it is also allocated a bundle with bids $\mathcal{B} \cup B_1$ that include a bid for a new bundle B_1. An optimal WD algorithm trivially has this property.

[**Prices.**] The auctioneer announces an explicit *ask price* $p^t_{\text{ask}}(S)$ in round t for all bundles S that receive failed bids from agents. Other bundles are implicitly priced at least as high as the greatest price of any bundle they contain, i.e. $p(S') \geq p(S)$ for $S' \supseteq S$. The initial ask prices are zero. The prices are usually the same for all agents, but in some cases the auctioneer introduces price-discrimination based on agents' bids, when this is necessary to achieve an optimal allocation (see below). The price-update rule generalizes the rule in the English auction, an ascending-price auction for a single item. In the English auction the price is increased whenever two or more agents bid at the current price. In *i*Bundle the price on a bundle is increased when one or more agents that do not receive a bundle in the current allocation bid at (or above) the current ask price for a bundle. The price is increased to ϵ (the minimal bid increment) above the greatest failed bid price.

[2] An agent can also bid ϵ below the ask price for any bundle in any round— but then the agent cannot bid a higher price for that bundle in the future. This allows an agent to bid for bundles priced slightly above its value.

[Price discrimination.] A simple rule is used to dynamically introduce price discrimination on an agent-by-agent basis as bids are received. An agent begins to receive individual prices if it submits bids that are not *safe*, given the new allocation.

DEFINITION 7.1 Safe bids. An agent's bids are *safe* if the agent is allocated a bundle in the current allocation, or it does not bid at or above the ask price for any pair of compatible bundles S_1, S_2, such that $S_1 \cap S_2 = \emptyset$.

Intuitively, when a single agent submits unsuccessful bids for compatible bundles the general price-increase is larger than might be necessary to support the optimal allocation of items to agents. Individual ask prices are initialized to the current general prices, $p^t_{\text{ask},i}(S) = p^t_{\text{ask}}(S)$, and then increased to ϵ above an agent's bid prices whenever the agent does not receive a bundle in the current allocation.

[Termination.] The auction terminates when: [T1] all agents submit the same bids in two consecutive rounds, or [T2] all agents that bid receive a bundle.

[Best-response Bids.]

*i*Bundle computes an optimal allocation with *myopically rational* agents that play a best (utility-maximizing) response to the current ask prices and allocation in the auction. The agents are myopic in the sense that they only consider the current round of the auction. By definition, a myopic agent bids to maximize utility at the current ask prices (taking an ϵ discount when repeating a bid for a bundle in the provisional allocation or bidding for a bundle priced just above its value). The myopic best-response strategy is to submit an XOR bid for all bundles S that maximize (to within ϵ) utility $u_i(S)$ at the current prices. This maximizes the probability of a successful bid for bid-monotonic WD algorithms.

7.1.1 Example Auction Scenarios

A couple of full worked examples are presented in Parkes [Par99b].

Figure 7.1 shows four different auction scenarios that illustrate winner-determination and price-updates. In each scenario bundles ABC, CD, D and AB each receive a bid from some agent, but the scenarios differ in the agents that submit the bids. The 'boxes' indicate XOR bids from the same agent, and the 'circles' indicate the allocation selected by the auctioneer to maximize revenue. Price increases are indicated with an 'arrow'. The minimal bid increment $\epsilon = 5$. Notice that the bid prices for bundles are consistent, such that $p(ABC) \geq p(AB)$ and $p(CD) \geq p(D)$. This must be maintained in *i*Bundle.

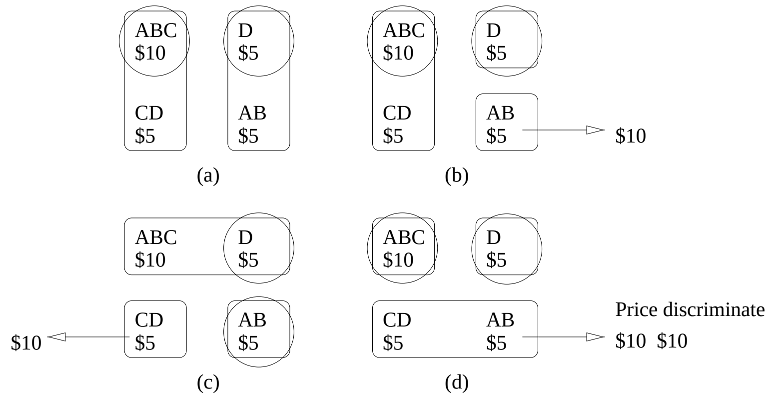


Figure 7.1: Auction Scenarios.

The auctioneer selects the same revenue-maximizing allocation in scenarios (a), (b) and (d), allocating bundles ABC and D to two different agents. In (c) the same agent bids for bundle ABC and D and the auctioneer must select another allocation, because it must respect the XOR bid constraint. Allocation D and AB is chosen in preference to ABC because it includes more agents.

In (a) the auction terminates because the revenue-maximizing allocation includes a bid from every agent that bids. In (b) the auction does not terminate because the agent that bids $(AB, \$5)$ is not happy. The ask price for AB is increased to $5 + \epsilon$, the minimal bid increment, in the next round. Scenario (c) is similar, except that the provisional allocation is different, and the price is increased on CD in the next round. In (d) the bids from the agent that receives no bundle in the provisional allocation are not *safe* because bundle CD and AB are compatible. The auctioneer introduces *price-discrimination* in the next round. The ask price remains unchanged for the 2 agents in the provisional allocation, while the ask price to the third agent is increased to $5 + \epsilon$ for both bundles CD and AB.

7.2 Theoretical Results

This section introduces the main theoretical results for *i*Bundle. Recall that $|G|$ is the number of items, $|I|$ is the number of agents, and ϵ is the minimal bid increment.

THEOREM 7.1 *(Optimality) iBundle terminates with an allocation that is within*

$3\min\{|G|,|I|\}\epsilon$ *of the optimal solution, for best-response agent bidding strategies.*

The auction is *optimal* as the bid increment approaches zero because the error-term goes to zero. The proof is inspired by a proof due to Bertsekas [Ber88] for a simpler iterative auction, and exploits the connection with *primal-dual* theory of linear programming that is outlined in Chapter 6. The full proof is presented in Parkes & Ungar [PU00b].

Intuitively, the prices are carefully adjusted in each round of the auction to maintain complementary slackness conditions (CS-2) (see Chapter 6), so that the auctioneer maximizes its revenue given the current prices and the provisional allocation. Eventually the allocation and prices satisfy (CS-1), with every agent that submits a bid receiving a bundle (or bidding for a bundle priced just above its value). The optimality of the allocation follows from the strong duality-theorem of linear-programming, and because the allocation is integral by construction throughout the auction. The worst-case error term is derived from a formulation of the complementary-slackness conditions that allows for the ϵ-indifference of agents' bidding strategies.

7.2.1 Dropping Price-Discrimination

Price-discrimination can be hard to enforce in some problems. For example, it is necessary to prevent agents entering the auction under multiple pseudonyms, and also to prevent the transfer of items in an after-market. In a simpler variation, *i*Bundle(2), the auctioneer never tests for bid-safety and never introduces price discrimination [Par99b]. From Theorem 1, this variation is optimal when bids are safe:

Theorem 7.2 i*Bundle(2) terminates with an allocation that is within* $3\min\{|G|,|I|\}\epsilon$ *of the optimal solution when bids are* safe*, for best-response agent bidding strategies.*

As an example, bids are safe if each agents bids for a set of *conflicting* bundles in every round of the auction.

I can prove that *i*Bundle solves the following problems without price discrimination: (1) Every agent demands different bundles; (2) Agents have additive or superadditive values, i.e. $v(S \cup S') \geq v(S) + v(S')$ for non-conflicting bundles S and S'; (3) The bundles that receive bids throughout the auction are from a single partition of items, e.g. bids are for pairs of matching shoes or individual items.

In experimental tests *i*Bundle performs well without price discrimination in many hard problems, achieving an average of 99% allocative efficiency [Par99b] compared to only 82% allocative efficiency from non-combinatorial auctions in the same problems.

7.3 Computational Analysis

As an iterative auction, *i*Bundle has many computational advantages for *agents* over the sealed-bid GVA. In Parkes [Par99b] I present results that demonstrate savings in agent valuation work in *i*Bundle.

However, the winner-determination (WD) problem that the auctioneer solves in each round of *i*Bundle to compute the provisional allocation is $\mathcal{NP}$-hard, just as in the GVA. The auctioneer must solve one WD problem in each round, and a naive worst-case analysis gives $O(BV_{\max}/\epsilon)$ rounds to converge, for a total of B bundles with positive value over all agents, maximum value $V_{\max}$ for any bundle, and minimal bid increment ϵ. In the worst-case the price of a single bundle must increase by at least ϵ in each round the auction remains open, and prices are bounded by the maximum value over all agents. The number of rounds to termination is inversely proportional to the minimal bid increment. The auctioneer can solve less WD problems by increasing the minimal bid increment, for some loss in economic efficiency.

A number of optimizations are possible within *i*Bundle to speed-up computation on winner-determination in each round. First, the provisional allocation from the previous round provides a good initial solution to the WD problem, because agents must re-bid bundles received in the previous round. This allows pruning of the search for a revenue-maximizing allocation. An additional saving in computation time is achieved by limiting search to an allocation at least ϵ better than the value of the allocation in the previous round. Moreover, although each intermediate WD problem in *i*Bundle may be intrinsically more difficult than each WD problem in GVA because all agents bid at similar prices for bundles (Andersson *et al.* [ATY00]), the problems are typically much smaller than in the GVA.

The auctioneer only announces price *increases* in each round, and need not maintain explicit prices for all possible bundles. Bid prices are verified dynamically in each round, to check that bids are at least as large as the ask price of all contained bundles. With a simple sorted-list implementation, the total work in checking each bid is *linear* in the number P of bundles that have explicit ask prices. Similarly, prices can be maintained in linear-time in P for each new price increase. In addition, $P \leq B$, with agents that have values for B bundles, because only bundles that receive bids can receive explicit ask prices.

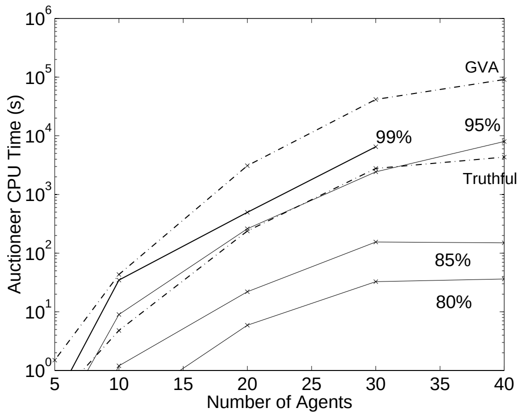
7.4 Experimental Results

This section presents experimental results to compare *i*Bundle's computation with the GVA in some hard resource allocation problems. The results only show part of the picture, comparing the *auctioneer*'s computation without comparing the work done by agents in valuation (see Parkes [Par99b] for this). In these experiments we assume that agents know *a priori* their values for bundles without computation.

Problem sets *Decay*, *Weighted-random* and *Random* are taken from [San99]. Each problem defines a distribution over the agents' values for bundles of items. Agents have XOR valuation functions, i.e. they want at most one bundle. The number of items $|G| = 50$, and I scale the problems by increasing the number of agents from 5 to 40, with values for 10 bundles per agent. In the Decay problem I set Sandholm's parameter $\alpha = 0.85$. The Random and Weighted-random problems are quite easy, while the Decay problems tend to be harder because agents bid for smaller bundles and the optimal allocation requires *coordination* across a number of agents [San99, ATY00].

I implemented a variation on Sandholm's depth-first branch-and-bound search algorithm to solve winner-determination (WD) in both *i*Bundle and the GVA, with a new heuristic to compute better upper-bounds for XOR bids. Further savings in computation time are achieved by using the provisional allocation from the previous round as an initial solution to the WD problem, and also limiting search to an allocation at least ϵ better than the value of the allocation in the previous round. Further computational savings should come from using all previous provisional allocations in an initial stage of winner-determination in the next round of the auction.

I present results for *i*Bundle(2), the auction variation without price discrimination. Performance is measured for different minimal bid increments, the bid increment is adjusted to give allocative efficiency of 80%, 85%, 95% and 99% ($\pm$1%). To test the performance of the auction with an approximate WD algorithm I used a greedy algorithm for WD in [LOS99]. This satisfies bid-monotonicity, so that the agents receive incentives to follow the same bidding strategy.



(a)

Problem		GVA	*i*Bundle			Approx-Bundle
			≃ 90%	≃ 95%	≃ 99%	
Decay	Eff (%)	100	91.5	94.9	98.3	85.1
67.3%[d]	T_{wd}[a] (s)	41700	831	2400	5650	0
13.4[e]	T_{pr}[b] (s)	–	26	34.5	44	39.2
	Comm[c]	18.8	221	306	394	377
WR	Eff (%)	100	90.7	94.9	99.2	79.4
71.5%	T_{wd} (s)	3	0.6	1.7	6	0
1	T_{pr} (s)	–	5.4	11.5	40.9	12.2
	Comm	18.1	20.5	52.1	144	53.1
Rand	Eff (%)	100	89.3	97	99	95.8
37.8%	T_{wd} (s)	68	4.4	7.4	11	0
11.2	T_{pr} (s)	–	6.5	9.7	12.1	12.9
	Comm	18.7	49.5	66.4	82.6	85.6

(b)

Figure 7.2: Performance of *i*Bundle in some hard resource allocation problems. (a) The Decay problem. (b) The Decay, WR, and random problems.
[a] Auctioneer WD time. [b] Price-update time. [c] Communication cost (k bit). [d] Alloc. eff. of a sealed-bid auction with a greedy WD algorithm and truthful agents. [e] Average number of agents in the optimal solution.

Figure 7.2 (a) plots the total auctioneer winner-determination and price-update time[3] in *i*Bundle(*2*), the GVA, and a sealed-bid auction with truthful agents in the Decay problem set.[4] The results are averaged over 10 trials.

The performance improvement of *i*Bundle over GVA is striking, achieving at least one order of magnitude improvement with 99% efficiency and three orders of magnitude with 85% efficiency. For up to 95% efficiency the bounded-rational compatibility and myopic truth-revelation in *i*Bundle is "free", with the auction's run-time approximately the same as for a sealed-bid auction with truthful agents.

Figure 7.2 (b) compares *i*Bundle with the GVA in all 3 problem sets. *i*Bundle has less WD time at 95% allocative efficiency than the GVA in all problems. In some problems the price-update is quite expensive, for example in Weighted-random because the bids are for large bundles and it is necessary for the auctioneer to check price consistency against the

[3]Time is measured as user time on a 450 MHz Pentium Pro with 1024 MRAM), with *i*Bundle coded in C++.

[4]The GVA is intractable in some instances, in which case its run-time is estimated as the average number of agents in the optimal allocation times the time for the single WD problem in the truthful auction.

price of all included bundles. There is also a communication cost penalty in using *i*Bundle compared with the GVA in these problems,[5] because of repeated bids across a number of rounds. This would change in problems with agents that have values for many bundles because all values must be reported in the GVA, or in easier problems because *i*Bundle can terminate after only a few rounds of bidding.

Finally, the performance of *i*Bundle with the greedy WD algorithm is noteworthy: *i*Bundle with approx-WD achieves allocative efficiency of 85.1% in Decay with at least a *1000-fold reduction* in WD time. I propose to test other approximate algorithms, to find an algorithm that boosts allocative efficiency further and continues to save on computation.

7.5 Proposed Work

I propose to: (a) characterize easy special cases of *i*Bundle, with polynomial time winner-determination; (b) derive price-update rules for problem-specific bid languages; (c) test methods to speed-up *i*Bundle, e.g. caching, ϵ-scaling, and approximate winner-determination algorithms.

[5] I assume that bids and price information in *i*Bundle must only specify a bundle, because bids are usually at the current ask price, and ask prices only increase by the minimal bid increment. I also assume a broadcast network infrastructure for price updates. A bundle is specified with $|G|$ bits. In the GVA a bid specifies both a bundle and a value. I assume that values require 10 bits, enough to specify a value to 3 significant figures ($\log_2(1000) \simeq 10$.)

Chapter 8

Relating Primal-Dual Analysis to Vickrey Payments

In this chapter I will provide a brief motivation for the methods presented in Chapter 9, which attempt to use primal-dual theory to *adjust prices in* i*Bundle after the auction terminates towards Vickrey payments.*

I have recently proved a positive result, that shows that Vickrey payments can be computed from minimal competitive equilibrium prices, and therefore potentially via primal-dual theory in the context of an auction.

This is important, because if one can compute Vickrey payments at the end of an iterative auction the auction inherits the incentive-compatibility of the generalized Vickrey auction (see Chapter 5), and is a *non-manipulable* iterative auction.

8.1 Minimal Competitive Equilibrium Prices

It is useful to restate the definition of competitive equilibrium, introduced in Chapter 6 with respect to complementary-slackness conditions.

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 I 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 of linear programming:

THEOREM 8.1 *In an auction that terminates in competitive equilibrium, minimal prices that satisfy complementary slackness with the final allocation are minimal competitive equilibrium prices.*

In competitive equilibrium (CE for short) the value of the primal of the associated linear program equals the value of the dual. Recall that the value of the dual is the sum maximal utility over all agents at the prices plus the maximal revenue to the auctioneer.

$$\sum_i v_i(S_i^*) = \sum_i \max_S \{0, v_i(S) - p_i^*(S)\} + \max_S Rev(S, p^*)$$

where $Rev(S, p^*)$ denotes the revenue for allocation S at prices p^*. This equality of the value of the primal and the dual follows from complementary-slackness conditions (CS-1) and (CS-2).

The *minimal* competitive-equilibrium prices $\underline{p}$ are prices that minimize the auctioneer's total revenue in competitive-equilibrium. The minimal competitive equilibrium prices can be computed as a linear-program: they are competitive equilibrium while complementary-slackness conditions are maintained with the optimal allocation:

$$\begin{aligned} \min_p \quad & Rev(S^*, p) \\ \text{s.t.} \quad & \text{(CS-1)}(p, S^*) \\ & \text{(CS-2)}(p, S^*) \end{aligned}$$

Let $\underline{p}_i(S_i^*)$ denote a minimal competitive equilibrium price for agent i that receives bundle S_i^* in the optimal allocation. Bikchandani & Ostroy [BO98] prove the following result:

THEOREM 8.2 *Minimal competitive equilibrium prices* $\underline{p}_i(S_i^*) \leq p_{\text{gva},i}$, *for all agents* i *that receive bundle* $S_i^* \neq \emptyset$ *in the optimal allocation, where* $p_{\text{gva},i}$ *is the agent's Vickrey payment.*

The Vickrey payment $p_{\text{gva},i}$ (equation 5.1) is computed as the total reported negative effect that the presence of agent i has on the other agents, given that agents report truthful values. I described this in some detail in Section 5.

So, the minimal competitive equilibrium prices are an *upper-bound* on the Vickrey payments. At the end of this chapter I summarize some special cases in which the minimal prices are equal to the Vickrey payments. Intuitively, Vickrey payments are priced in a competitive equilibrium when agents are more like substitutes for each other than complements [BO98].

In fact, minimal competitive-equilibrium prices compute the Vickrey payment if and only if the minimal CE prices are *unique* in terms of $\underline{p}(i, S_i^*)$, i.e. the prices for bundles that agents receive.

Minimal competitive-equilibrium prices are often not unique: the total revenue to the auctioneer can be minimized with more than one set of prices, with each set providing a different distribution of surplus over the agents.

In Parkes & Ungar [PU00c] I prove the following important theorem:

THEOREM 8.3 *The Vickrey payment to agent* i *can be computed as the smallest price to agent* i *over all minimal competitive equilibrium prices.*

PROOF. (Constructive). Essentially, the proof is to show that it is possible to construct a set of prices that are minimal CE prices and price the Vickrey payment to any agent j in the optimal allocation. To compute the Vickrey payment for agent j, construct prices $p_i(S) = v_i(S)$ for all agents $i \neq j$, and $p_j(S) = \max\{0, v_j(S) - \Delta_j\}$, with $\Delta_j = v_j(S_j^*) - p_{\text{gva},j}$. ∎

The implication of this result is that it is always possible to compute GVA prices as the smallest price to each agent over "enough" minimal CE prices.

In the next chapter I introduce "proxy and adjust", which aims to compute the price-discount to each agent after *i*Bundle terminates to adjust the payment from each agent so that it equals the payment it would make in the Vickrey auction.

Previous attempts to design incentive-compatible (i.e. non-manipulable) iterative auctions relied and carefully adjusting prices *during* the auction, to give Vickrey payments on termination. I believe that this idea— using primal-dual theory to compute minimal CE prices and possibly GVA payments after an auction terminates —is both novel and powerful, especially in combination with the other methods discussed in this thesis. I am able to derive sufficient conditions to be able to compute minimal CE prices, and necessary and sufficient conditions on prices and bids to be able to compute Vickrey prices.

8.2 Example

In this section I present a couple of examples to illustrate how it is possible to compute Vickrey payments from minimal CE prices.

Figure 5 continues scenarios (b) and (d) from Figure 3. Now, let us assume that the values shown represent agent's values for bundles. E.g. in (b) agent 1 has $v_1(ABC) = 50, v_1(CD) = 50$.

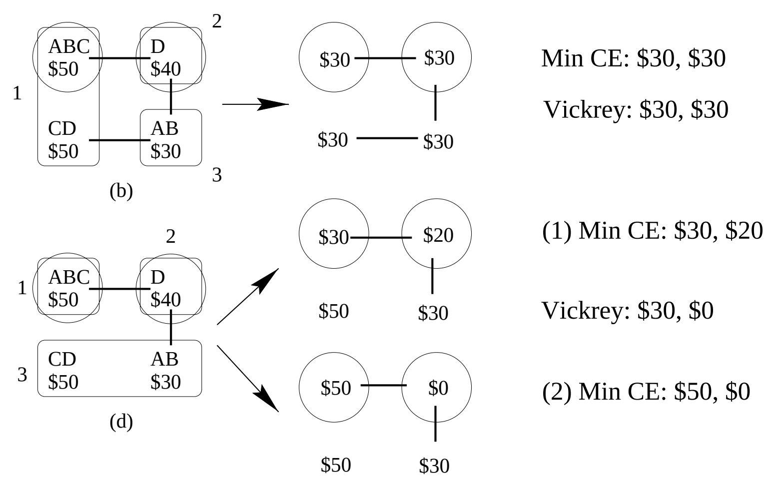


Figure 5: Price Adjust Scenario.

The agents are numbered (next to the 'boxes'), and the 'lines' indicate compatible bundles in an allocation, for example it is possible to allocate CD and AB in (b) but not in (d), because they are both bids from agent 3 in (d). The optimal allocation is ABC to agent 1 and D to agent 2 in both scenarios, written $[(ABC, 1), (D, 2)]$.

In (b) the Vickrey payments are $p_{\text{gva},1} = 70 - 40 = 30$ and $p_{\text{gva},2} = 80 - 50 = 30$. In

(d) the Vickrey payments are $p_{\text{gva},1} = 70 - 40 = 30$ and $p_{\text{gva},2} = 50 - 50 = 0$.

The "arrow" shows the effect of computing minimal competitive-equilibrium prices in each problem.

In (b) the value of the auctioneer's minimal revenue in competitive equilibrium is \$60, which is equal to the total revenue at the Vickrey payments. The minimal CE prices are unique, and compute the Vickrey payments. The minimal CE prices are $\underline{p}_1(ABC) = \$30, \underline{p}_1(CD) = 30$, $\underline{p}_2(D) = 30$, and $\underline{p}_3(AB) = 30$. At these prices (CS-1) holds (notice that the price for bundles ABC and CD for agent 1 is reduced by the same amount, and the price for bundle AB to agent 3 is left unchanged). Furthermore, (CS-2) holds because the auctioneer can receive no more that \$60 from any feasible allocation at the final prices. It is easy to see that these are minimal CE prices, any further reduction to prices and another allocation has more revenue.

In (d) there are two interesting minimal CE prices. In both cases the prices to agent 3 are left unchanged, $\underline{p}_3(CD) = 50, \underline{p}_3(AB) = 30$. This is necessary to maintain (CS-1), because this agent receives no bundle in the optimal allocation. In competitive equilibrium the revenue from the optimal allocation, ABC to agent 1 and D to agent 2, must be at least \$50 to maintain (CS-2) because this is the revenue that the auctioneer can achieve with allocation CD to agent 3. In addition, the price to agent 1 must remain greater than \$30, the price $\underline{p}_3(AB)$ else allocation D to agent 2 and AB to agent 3 will achieve more revenue than the optimal allocation. In case (1) the minimal CE prices are $\underline{p}_1(ABC) = 30, \underline{p}_2(D) = 20$, in case (2) the prices are $\underline{p}_1(ABC) = 50, \underline{p}_2(D) = 0$.

Although no *single* set of CE prices compute the Vickrey payment to every agent (this is impossible because total CE revenue must be at least \$50, but total GVA payment is \$30), the smallest CE price over all minimal CE prices equals the GVA payment for each agent; i.e. $p_{\text{gva},1} = \min\{30, 50\} = 30$ and $p_{\text{gva},2} = \min\{20, 0\} = 0$.

8.3 Prior Results

The minimal competitive equilibrium prices are known to equal the Vickrey payments in some special cases of the combinatorial resource allocation problem. Moreover, auctions have been designed to terminate with the Vickrey outcome in these problems.

Crawford & Knoer [CK81] present an auction for agents with linear-additive preferences

[CK81], that is essentially a simultaneous ascending-price auction on individual items, and prove that the auction terminates in competitive equilibrium with Vickrey payments.

Leonard [Leo83] demonstrated that minimal CE prices in the assignment problem are equal to Vickrey payments. In the assignment problem each agent wants at most a single item, i.e. $v_i(S) = \max_{j \in S} v_i(j)$ for items j in bundle S, and value $v_i(j)$ for item j. Leonard proposed a "pseudo-market" procedure to compute the Vickrey payments efficiently in a *sealed-bid auction* for agents with linear-additive or unit-demand preferences.

Demange *et al.* [DGS86] proposed an iterative auction to compute Vickrey payments and the optimal allocation in the assignment problem, with best-response agent bids. The auction allows agents to place exclusive-or bids on items. In each round of the auction the auctioneer determines a provisional allocation that maximizes revenue and increases prices on a *minimal over-demanded subset* of items.[1]

More recently, Ausubel [Aus97] has proposed an ascending-price auction for the multiple identical items assignment problem, that provably generates efficient allocations and Vickrey payments with subadditive agent valuation functions, i.e. decreasing-returns. In general minimal CE prices do not equal Vickrey payments. For example, Gul & Stacchetti [GS97a] show that *gross-substitutes* (in which an agent that demands good j at price $p(j)$ will continue to demand good j if the prices for other goods increase) is not sufficient for the min CE prices to equal the Vickrey payments.

[1] Recently, Sankaran [San94] has addressed the computational aspects of DGS-exact and suggested an efficient method to compute the provisional allocation and the price increases, using the Ford-Fulkerson max-flow algorithm.

Chapter 9

Proxy-Agents and Price-Adjustment

This chapter introduces a new technique to make iterative auctions, and *i*Bundle in particular, more robust to strategic manipulation by agents. I summarize work presented in the attached paper *Preventing strategic manipulation in iterative auctions: Proxy agents and Price-adjustment* [PU00c].

Continued research into this "proxy and price-adjust" method forms a large part of the work that I propose as the remainder of my dissertation.

At present *i*Bundle leaves open the possibility of agent manipulation, for example placing jump bids, signaling false intentions, or waiting to bid, all of which can reduce economic efficiency.

The proposed method is called "proxy-agents and price-adjustment". The idea is to leave the auction largely unchanged and adjust prices *after* the auction terminates based on information received from agents' bids towards prices in the generalized Vickrey auction (GVA). 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. If successful, *i*Bundle can retain its computational advantages *and* inherit the strategy-proofness of the GVA.

One might think that if an iterative auction terminates with an optimal allocation and GVA prices with agents that place truthful best-response bids to prices throughout the auction (*myopic-implementation*), then truthful best-response would be a dominant strategy for self-interested agents. In fact, manipulation remains possible with a non best-response strategy [GS97a]. However, truth revelation to *proxy bidding agents* at the auctioneer is optimal, with proxy agents that place best-response bids to current prices based on (possibly untruthful) value information provided by agents. Myopic-implementation of the Vickrey outcome with proxy bidding agents gives an auction that is *iterative, optimal, and strategy-proof.* Agents can still provide incremental information about values and perform incremental computation. A bounded-rational compatible auction with proxy bidding agents remains bounded-rational compatible, e.g. with the proxy agents in eBay (see discussion in Section 2).

Consider agents that follow a *myopic best-response bidding strategy*, i.e. bid to maximize utility in the current round, taking prices as fixed. It is useful to define the *myopic-implementation* of the Vickrey outcome (i.e. optimal allocation & Vickrey prices) in an iterative auction:

DEFINITION 9.1 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, such that agent i has value $v_i(S)$ for bundle S, and p is the current prices in the auction. Call this a *truthful best-response bidding strategy.* Also, let $BR(\hat{v}_i, p)$ denote an *untruthful* best-response bidding strategy for agent i, for some valuation function $\hat{v}_i \neq v_i$. The following result is immediate from the incentive properties of the GVA:

THEOREM 9.1 *Truth-revelation is a dominant best-response bidding strategy in an iterative auction $\mathcal{A}$ that myopically-implements the Vickrey outcome if all agents must follow a (possibly-untruthful) best-response bidding strategy.*

Truth-revelation, i.e. following a best-response strategy for $\hat{v}_i = v_i$, is optimal if all agents are constrained to some best-response bidding strategy. I suggest providing *proxy agents* at the auctioneer to constrain bidding strategies. Agents must bid through the proxy agents, which follow a best-response bidding strategy based on reported information about an agent's valuation function. Agent i provides (possibly untruthful) and incomplete information $\hat{v}_{\text{app},i}$ about its value $\hat{v}_i$ to its proxy-agent, just enough to allow the proxy-agent to place an optimal bid (given the reported value) at all stages of the auction. For example, if the current ask price for an item is p, the proxy agent will only bid if it has an approximate value for the item that bounds the true value above the price. This can be defined probabilistically for approximations that are not hard bounds. Proxy agents

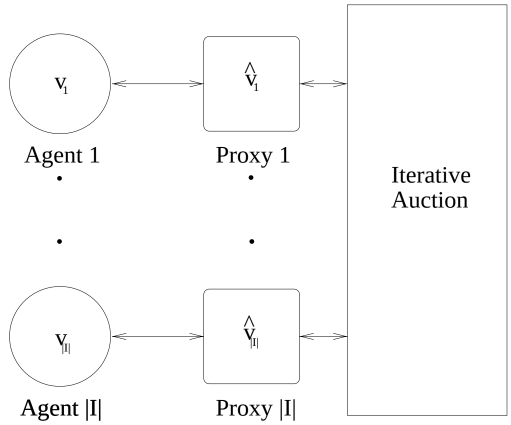


Figure 9.1: Proxy Bidding Agents.

provide the following important result:

THEOREM 9.2 *Introducing myopic best-response proxy bidding agents to an auction that myopically-implements the Vickrey outcome creates an auction where truth-revelation is a dominant strategy.*

Proxy agents retain the computational advantages of iterative auctions because agents can provide incremental information about value; an iterative auction with proxy bidding agents remains bounded-rational compatible.

The architecture is illustrated in Figure 9.1. Agent i provides incomplete information, $\hat{v}_{\text{app},i}$ 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 $\hat{v}_{\text{app},i}$ 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.

Most iterative auctions are *indirect* mechanisms, where agents do not explicitly reveal their values, but respond to prices, e.g. English auction, first-price ascending. Most *direct-revelation* mechanisms, when agents report their values for items, are *single-shot*, e.g. Vickrey auction. There is a middle-ground: an *iterative direct-revelation mechanism*, where There is always at least one valuation function $\hat{v}_i$ at the end of the auction that would implement the bidding strategy for the entire auction.

9.1 Computing GVA prices from CE prices

As discussed in the previous chapter, it is possible to compute Vickrey prices from minimal CE prices. In some problem instances the minimal CE prices are equal to Vickrey prices, while in other problems the Vickrey prices can be computed as the smallest prices over all minimal CE prices.

My insight is that a primal-dual formulation of the problem of computing minimal CE prices allows minimal CE prices to be computed in an auction, on the basis of agents' bids during the auction. Bids from myopic best-response agents can provide enough information to formulate the complementary slackness conditions between primal and dual problems [PU00c].

This allows the computation of minimal CE prices after an auction terminates, based on bids placed by agents. Reduce prices while agents continue to maximize utility with the final allocation, *and* the final allocation continues to maximize the auctioneer's revenue.

In Parkes & Ungar [PU00c] I propose procedure `Adjust*` to adjust prices within this framework, and prove necessary and sufficient conditions for computing GVA prices in an auction. Let I^* denote the set of agents in the optimal allocation, V^* denote the auctioneer's revenue with allocation S^*, and $V^{-i}(\hat{P})$ denote the auctioneer's maximal revenue without allocating a bundle to agent i, computed at the final prices $\hat{P}$. Procedure `Adjust*` computes the price-discount to each agent i in the optimal allocation.

`Adjust*:`

`for each` $i \in I^*$

$\Delta_i = \min\{V^* - V^{-i}(\hat{P}),\ \ p_i(S_i^*)\}$;

I have been able to characterize conditions on agents bids and final prices in *i*Bundle in which `Adjust*` computes the Vickrey outcome. Put simply (*all* conditions must hold):

(A.1) The price for a bundle that an agent not in the final allocation receives in a second-best allocation must be no greater than that agent's value for the bundle.

(A.2) Agents must bid for all bundles that they receive in second-best allocations.

(B) The price for the bundle that an agent receives in the optimal allocation must be equal to that agent's value for the bundle if the agent is not in every second-best allocation.

THEOREM 9.3 *Conditions A.1, A.2 and B are necessary and sufficient conditions on agents' bids and prices for* `Adjust*` *to compute* GVA *prices.*

PROOF. See Parkes & Ungar [PU00c]. ∎

I also propose a test that allows an auctioneer to determine whether `Adjust*` computes GVA prices.The Vickrey-Test is sufficient but not necessary for GVA prices.

DEFINITION 9.1 [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.

9.1.1 Example: Adjusting Prices Towards Vickrey 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 $p_{\mathrm{gva},1} = 40 - 40 = 0$ and $p_{\mathrm{gva},2} = 50 - 30 = 20$.

Now, consider using `Adjust*` to reduce prices in each scenario. In both cases initial prices are competitive equilibrium prices, and best-response bids satisfy Assumptions A.1 and A.2 with the prices. `Adjust*` computes GVA prices in Scenario 2, but not scenario 1.

(Scenario 1) Prices are $p_1 = \{25, 0, 25\}$, $p_2 = \{0, 25, 25\}$ and $p_3 = \{0, 20, 40\}$. `Adjust*` computes prices $\underline{p}_1(A) = 25 - (50 - 40) = 15$ and $\underline{p}_2(B) = 25 - (50 - 45) = 20$. Agent 2 pays its GVA price because agent 1 is 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 $\underline{p}_2(B) = 20$, equal to GVA prices.

9.2 Application to *i*Bundle

In application to *i*Bundle(*3*), the variation that maintains price-discrimination throughout the auction, conditions A.1 and A.2 always hold, and I have a partial characterization of problems in which condition B holds: 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.

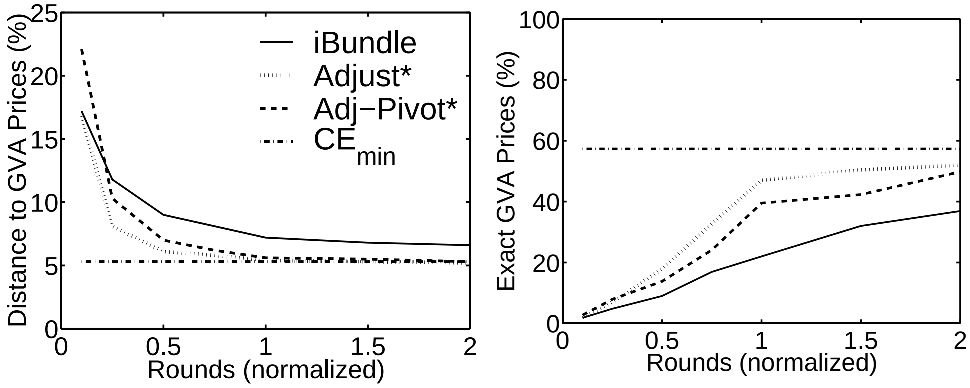
9.2.1 Experimental Results

Initial experimental results demonstrate that minimal CE prices are often close to GVA prices, and also show that an approximate algorithm `Adj-Pivot*` for `Adjust*` works well in practice (see Parkes & Ungar [PU00c]). The algorithm `Adj-Pivot*` formulates `Adjust*` as a linear-program, and uses provisional allocations computed by the auctioneer during the auction to approximate $V^{-i}(\hat{P})$, the maximal revenue to the auctioneer without allocating a bundle to agent i.

Figure 9.2 (a) 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 from Parkes [Par99b]. `Adj-Pivot*` is an approximate algorithm for `Adjust*` with negligible computation.

I tested *i*Bundle with different bid increments to vary the number of rounds to termination, and averaged 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, I also plot the distance between the minimal CE prices and the GVA prices. Distance between prices is measured with respect to a L_1 norm, and computed as a fraction of the total value of the optimal allocation.

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.



(a) Distance.

(b) Fraction Correct.

Figure 9.2: Average performance of *i*Bundle with price-adjustment `Adjust*` and `Pri-Adjust*` in problems PS 1–12.

Figure 9.2 (b) plots the fraction of all problems in which the prices are within 2% of the GVA prices. 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.

It is noteworthy that the approximate method `Adj-Pivot*` is as effective as `Adjust*` for small bid increments.

9.3 Proposed Work

Paradoxically, it might be useful to keep the auction open longer so that agents bid more for the optimal allocation, because this *decreases* the final price that agents pay after price-adjustment. One approach is to have the auctioneer dynamically detect an optimal allocation, and then take a position and keep the auction open for a few additional rounds.

This presents an exciting prospect for future work, because the truth-revelation of the GVA with the incremental computation of *i*Bundle will provide a powerful mechanism for resource allocation in many B2B e-commerce applications. I propose to continue to research this "proxy and price-adjust" method, and try to develop an *iterative generalized Vickrey auction.*

I also propose to: (a) characterize problems in which *i*Bundle with `Adjust*` computes

Vickrey prices, and is therefore robust to strategic manipulation; and (b) derive upper-bounds on possible gains from strategic behavior as approximations are introduced into *i*Bundle with price-adjustment.

Chapter 10

Train Scheduling

This chapter reviews an application of iterative combinatorial auctions to a train scheduling system, with self-interested trains that compete for the right to run on a shared track network. Each train has a schedule that specifies optimal departure and arrival times at stations along its route, a cost function that specifies the cost of arriving and departing late (or early), and physical constraints such as maximum speed. The network is controlled by a number of dispatchers. It is unrealistic to assume that a single dispatcher schedules a complete track network, because there is often local and dynamically changing knowledge of track and weather conditions, that can affect the types of schedules that are safe.

I propose an auction-based system to compute a global schedule over all trains, that maximizes the total value (or minimizes the total cost), and ensures train *safety* such that there are no collisions and trains remain a minimum safe distance apart. The track network consists of three key elements: *stations*, that define possible initial and final locations for trains; *single-track sections*, on which trains are not allowed to pass or overtake; and *yards* which have a fixed capacity and allow trains to pass and overtake. In some parts of the network there are parallel tracks, formed from two single-track sections. To control problem-solving complexity I limit trains to pass but not overtake on double track sections. The present system only models linear track topologies without forks.

I have structured the auction as follows. Each dispatcher implements an auction with *i*Bundle-style price-updates for its local track region. A bid (t_a, t_d, S_a, S_d, p) to a dispatcher specifies an arrival time t_a at station S_a, departure time t_d from station S_d and a bid price p, which is the amount the train will pay for the schedule. Trains can place a set of XOR bids with each dispatcher, indicating that they want at most one schedule. Trains also inform the dispatchers at the start of an auction about their physical constraints, for example their maximum speed. The dispatchers operate simultaneous auctions, and trains bid in parallel across multiple dispatchers, to minimize the sum cost of late arrivals and departures and the cost of purchasing a schedule.[1] Our auction extends a previous auction for train scheduling, BICAP [BP96], in a number of important ways.

In each round a dispatcher computes a revenue-maximizing schedule that is safe, such that all trains can make their arrival and departure times without collisions. A bid for a pair of times (t_a, t_d) implicitly specifies a set of feasible (time, location) sequences (or bundles), based on the physical constraints of a train (e.g. its speed). I formulate the winner-determination problem as a mixed-integer program, and use a commercial optimization package, OSL, to solve in each round. The solution computes a time-location schedule for each train that specifies when the train must pass between key parts of the network to ensure its safety and the safety of other trains.

Each dispatcher uses *i*Bundle-style price-update rules. The ask price for a bundle (t_a, S_a, t_d, S_d) is increased to ϵ above a train's bid price whenever the train receives no allocation in response to bids in a particular round. Price-consistency implies that the price of all bundles is at least as large as the price of all *less constrained* bundles, where bundle (t_a, S_a, t_d, S_d) is *less constrained* than bundle (t'_a, S_a, t'_d, S_d) if $t_a \leq t'_a$ and $t_d \geq t'_d$. Trains must re-bid any bundles that they are allocated in the next round. Dispatchers have continuous auction-closing, so that bundles are sold to dispatchers if they remain allocated continuously for a fixed number of rounds. This is important in an on-line scheduling system where new trains continuously arrive into the system.

10.1 Proposed Work

I have performed initial experiments with trains that follow a simple myopic best-response bidding strategy. In future work I propose to understand the computational scaling properties of the auction-based system compared to traditional centralized optimization methods [KHC91, KH95, Hal93]. There also remains some work to define an agent's optimal bidding strategy within the auction.

[1] There is a potential "exposure problem" [BCL00] for trains because it is possible to purchase incompatible bundles across multiple dispatchers. The problem occurs because there is not a single auction for all items. One possible solution is to introduce a *secondary-market* in which trains can sell back unwanted track rights.

Chapter 11

Conclusions

Auctions offer great promise as mechanisms for optimal resource allocation in complex distributed systems with autonomous and self-interested agents. However, limited and costly computation necessitates a rethinking of traditional auction theory, because direct extensions of auctions that work well in small problems can fail in complex distributed systems. My thesis is that it is necessary to take an explicitly computational approach to auction design. The value of auctions in e-commerce will depend on the ability to maintain the desirable properties of auctions, for example economic efficiency, robustness, and simplicity, as methods are introduced to allow tractable computation. If computational issues are successfully addressed than auctions can generate many new economic efficiencies, in particular in business-to-business (B2B) e-commerce.

One important consideration is that agents can have difficult valuation problems, for example in auctions for job scheduling in a dynamic workflow environment, where an agent's costs to perform additional jobs depends on its local commitments and its availability of local resources. A new auction property, bounded-rational compatibility, identifies auctions in which agents can bid optimally with approximate values for items. For example, iterative auctions are bounded-rational compatible, and allow agents to provide incremental information to an auctioneer and perform incremental computation. It is possible to compute optimal allocations with approximate agent valuations in well-designed auctions.

Another important consideration is that agents often need bundles of items, and have values for bundles that are not equal to the sum of the values of the items in the bundle. I have proposed *i*Bundle, a new iterative combinatorial auction. *i*Bundle allows agents to bid for bundles of items and change their bids in response to price increases and bids from other agents. *i*Bundle computes an optimal allocation with myopic agents that follow a best-response bidding strategy in response to current prices. *i*Bundle also allows the auctioneer to tradeoff computation and economic efficiency, with approximate winner-determination algorithms and larger minimal bid increments, while continuing to provide incentives for self-interested myopic agents to bid truthfully in each round.

At present *i*Bundle leaves open the possibility of agent manipulation, for example placing jump bids, signaling false intentions, or waiting to bid, all of which can reduce economic efficiency. I propose to develop a new method, "proxy and adjust" to adjust the prices in iterative auctions retrospectively after the auction terminates towards prices in the generalized Vickrey auction (GVA). The GVA is a sealed-bid combinatorial auction in which it is optimal for an agent to bid its true value for bundles of items. I hope that it will be possible to unite the strategy-proofness of the GVA with the incremental computation of *i*Bundle, to provide a powerful mechanism for resource allocation in complex distributed problems.

A running theme throughout my work is a link between optimization and auction design. I use optimization theory to prove the economic efficiency of *i*Bundle with best-response agent bids, and also to adjust the prices after the auction terminates towards Vickrey prices based on agents' bids. I believe that this conceptual framework holds great promise for the design of useful auction-based solutions.

Bibliography

[AC98] Lawrence M Ausubel and Peter Cramton. The optimality of being efficient. Technical report, University of Maryland, 1998.

[ATY00] Arne Andersson, Mattias Tenhunen, and Fredrik Ygge. Integer programming for auctions with bids for combinations. In *Forthcoming, Proc. ICMAS'00.*, 2000.

[Aus97] Lawrence M Ausubel. An efficient ascending-bid auction for multiple objects. Technical report, University of Maryland, 1997.

[BCL00] Mark M Bykowsky, Robert J Cull, and John O Ledyard. Mutually destructive bidding: The FCC auction design problem. *Journal of Regulatory Economics*, mar 2000.

[Ber81] D P Bertsekas. A new algorithm for the assignment problem. *Math. Progr.*, 21:152–171, 1981.

[Ber88] D P Bertsekas. The auction algorithm: A distributed relaxation method for the assignment problem. *Annals of Operations Research*, 14:105–123, 1988.

[Ber90] D P Bertsekas. The auction algorithm for assignment and other network flow problems: A tutorial. *Interfaces*, 20(4):133–149, 1990.

[BLP89] J S Banks, J O Ledyard, and D Porter. Allocating uncertain and unresponsive resources: An experimental approach. *The Rand Journal of Economics*, 20:1–25, 1989.

[BM97] S Bikchandani and J W Mamer. Competitive equilibrium in an exchange economy with indivisibilities. *Journal of Economic Theory*, 74:385–413, 1997.

[BO98] S Bikchandani and J M Ostroy. The package assignment model. Technical report, Anderson School of Management and Department of Economics, UCLA, 1998.

[BP96] Paul J Brewer and Charles R Plott. A binary conflict ascending price (BICAP) mechanism for the decentralized allocation of the right to use railroad tracks. *Int. Journal of Industrial Organization*, 14:857–886, 1996.

[CK81] V P Crawford and E M Knoer. Job matching with heterogeneous firms and workers. *Econometrica*, 49:437–450, 1981.

[Cla71] E H Clarke. Multipart pricing of public goods. *Public Choice*, 11:17–33, 1971.

[Cle96] Scott H Clearwater, editor. *Market-Based Control: A Paradigm for Distributed Resource Allocation*. World Scientific, 1996.

[Coa60] R Coase. The problem of social cost. *Journal of Law and Economics*, 3:1–44, 1960.

[DGS86] G Demange, D Gale, and M Sotomayor. Multi-item auctions. *Journal of Political Economy*, 94(4):863–872, 1986.

[DKLP98] C DeMartini, A M Kwasnica, J O Ledyard, and D Porter. A new and improved design for multi-object iterative auctions. Technical Report SSWP 1054, California Institute of Technology, 1998. Revised March 1999.

[DS88] R Davis and R G Smith. Negotiation as a metaphor for distributed problem solving. In A Bond and L Gasser, editors, *Readings in Distributed Artificial Intelligence*, pages 333–356. Morgan Kaufmann, CA, 1988.

[EMN89] John Eatwell, Murray Milgate, and Peter Newman, editors. *Allocation, Information and Markets*. W. W. Norton, 1989.

[EP94] Carl Ehrman and Michael Peters. Sequential selling mechanisms. *Economic Theory*, 4:237–253, 1994.

[FI82] Robert Forsythe and R Mark Isaac. Demand-revealing mechanisms for private good auctions. In V L Smith, editor, *Research in Experimental*

Economics, volume 2, pages 45–61. JAI Press, Inc., Greenwich, Conn., 1982.

[FLBS99] Y Fujishima, K Leyton-Brown, and Y Shoham. Taming the computational complexity of combinatorial auctions: Optimal and approximate approaches. In *Proc. 16th International Joint Conference on Artificial Intelligence (IJCAI-99)*, pages 548–553, 1999.

[Gib73] Alan Gibbard. Manipulation of voting schemes: A general result. *Econometrica*, 41:587–602, 1973.

[GK99] Amy Greenwald and Jeffrey O Kephart. Shopbots and pricebots. In *Proc. 16th International Joint Conference on Artificial Intelligence (IJCAI-99)*.

[GL77] J Green and J J Laffont. Characterization of satisfactory mechanisms for the revelation of preferences for public goods. *Econometrica*, 45:427–438, 1977.

[GL79] J R Green and J J Laffont. *Incentives in Public Decision Making.* Amsterdam: North Holland, 1979.

[Gro73] Theodore Groves. Incentives in teams. *Econometrica*, 41:617–631, 1973.

[GS97a] F Gul and E Stacchetti. English and double auctions with differentiated commodities. Technical Report 97-02, University of Michigan, 1997.

[GS97b] F Gul and E Stacchetti. Walrasian equilibrium without complementarities. Technical report, University of Michigan, 1997.

[GSS93] R L Graves, L Schrage, and J Sankaran. An auction method for course registration. *Interfaces*, 23(5):81–92, 1993.

[Hal93] Susan Fraley Hallowell. *Optimal Dispatching Under Uncertainty: With Application to Railroad Scheduling.* PhD thesis, The Wharton School, University of Pennsylvania, 1993. OPIM TR 93-12-02.

[HG00] Luke Hunsberger and Barbara J Grosz. A combinatorial auction for collaborative planning. In *Forthcoming, Proc. ICMAS'00.*, 2000.

[Hur72] Leonid Hurwicz. On informationally decentralized systems. In C. McGuire and Roy Radner, editors, *Decision and Organization: A Volume in Honor of Jacob Marchak*. North-Holland, 1972.

[HV99] Michael H Huhns and José M Vidal. Online auctions. *IEEE Internet Computing*, 3(3):103–105, 1999.

[KC82] A S Kelso and V P Crawford. Job matching, coalition formation, and gross substitutes. *Econometrica*, 50:1483–1504, 1982.

[KDMT98] Noa E Kfir-Dahav, Dov Monderer, and Moshe Tennenholtz. Mechanism design for resource bounded agents. Technical report, Technion, 1998. Available at: http://iew3.technion.ac.il:8080/˜dov/noa90.ps.

[KF98] Manoj Kumar and S I Feldman. Internet auctions. Technical report, IBM Institute of Advanced Commerce, 1998.

[KH95] David R Kraay and Patrick T Harker. Real-time scheduling of freight railroads. *Transportation Research-B*, 29B(3):213–229, 1995.

[KHC91] David R Kraay, Patrick T Harker, and B Chen. Optimal pacing of trains in freight railroads: Model formulation and solution. *Operations Research*, 39:82–99, 1991.

[Led89] John O Ledyard. Incentive compatibility. In Eatwell et al. [EMN89], pages 141–151.

[Led93] John O Ledyard. The design of coordination mechanisms and organizational computing. *Journal of Organizational Computing*, 1:121–134, 1993.

[Leo83] H B Leonard. Elicitation of honest preferences for the assignment of individuals to positions. *Journal of Political Economy*, 91:461–479, 1983.

[LOS99] Daniel Lehmann, Liadan O'Callaghan, and Yoav Shoham. Truth revelation in rapid, approximately efficient combinatorial auctions. In *Proc. ACM Conf. on Electronic Commerce (EC-99)*.

[LPR97] J O Ledyard, D Porter, and A Rangel. Experiments testing multiobject allocation mechanisms. *Journal of Economic and Management Strategy*, 6(3):639–675, 1997.

[MCWG95] Andreu Mas-Colell, Michael D Whinston, and Jerry R Green. *Microeconomic Theory*. Oxford University Press, 1995.

[MGM99] Pattie Maes, R Guttman, and A Moukas. Agents that buy and sell: Transforming commerce as we know it. *Communications of ACM*, 1999.

[Mil99] Paul Milgrom. Putting auction theory to work: The simultaneous ascending auction. *Journal of Political Economy*, 108, 1999. To appear.

[MM87] R. Preston McAfee and John McMillan. Auctions and bidding. *Journal of Economic Literature*, 25:699–738, June 1987.

[MM96] R P McAfee and J McMillan. Analyzing the airwaves auction. *J Econ Perspect*, 10:159–175, 1996.

[MS83] R B Myerson and M A Sattherwaite. Efficient mechanisms for bilateral trading. *Journal of Economic Theory*, 28:265–281, 1983.

[MW82] P Milgrom and R Weber. A theory of auctions and competitive bidding. *Econometrica*, 50:1089–1122, 1982.

[Mye81] R B Myerson. Optimal auction design. *Mathematics of Operation Research*, 6:58–73, 1981.

[Mye89] Robert B Myerson. Mechanism design. In Eatwell et al. [EMN89], pages 191–206.

[NR99] Noam Nisan and Amir Ronen. Algorithmic mechanism design (extended abstract). In *Proc. 31st Annual Symposium on Theory of Computing (STOC99)*, 1999. Full version: http://www.cs.huji.ac.il/~amiry.

[PS82] C H Papadimitriou and K Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, 1982.

[Par99a] David C Parkes. Optimal auction design for agents with hard valuation problems. In *Proc. IJCAI-99 Workshop on Agent Mediated Electronic Commerce*, July 1999. Stockholm.

[Par99b] David C Parkes. *i*Bundle: An efficient ascending price bundle auction. In *Proc. ACM Conf. on Electronic Commerce (EC-99)*.

[PU00a] David C Parkes and Lyle H Ungar. Bounded rational compatible auctions. *Journal of Artificial Intelligence Research*, 2000. Submitted for publication.

[PU00b] David C Parkes and Lyle H Ungar. Iterative combinatorial auctions: Theory and practice. In *Proc. 18th National Conference on Artificial Intelligence (AAAI-00)*. To appear.

[PU00c] David C Parkes and Lyle H Ungar. Preventing strategic manipulation in iterative auctions: Proxy agents and price-adjustment. In *Proc. 18th National Conference on Artificial Intelligence (AAAI-00)*. To appear.

[PUF99] David C Parkes, Lyle H Ungar, and Dean P Foster. Accounting for cognitive costs in on-line auction design. In P Noriega and C Sierra, editors, *Agent Mediated Electronic Commerce (LNAI 1571)*, pages 25–40. Springer-Verlag, 1999. Earlier version appeared at the Agents'98 Workshop on Agent Mediated Electronic Trading, 1998.

[PWS98] V Parunak, A Ward, and J Sauter. A systematic market approach to distributed constraint problems. In *(ICMAS-98)*, pages 455–456. Poster session. Extended version: Tech. rep. CEC-010, Center for Electronic Commerce, ERIM, MI. Available at: http://www.erim.org/cec/rappid/icmas98.pdf.

[Ron99] Amir Ronen. A note on strategy-proof approximation mechanisms. Technical report, Hebrew University of Jerusalem, 1999. Draft. Available at: http://www.cs.huji.ac.il/~ronen.

[RPH98] M H Rothkopf, A Pekeč, and R M Harstad. Computationally manageable combinatorial auctions. *Management Science*, 44(8):1131–1147, 1998.

[RSB82] S J Rassenti, V L Smith, and R L Bulfin. A combinatorial mechanism for airport time slot allocation. *Bell Journal of Economics*, 13:402–417, 1982.

[RW91] Stuart Russell and Eric Wefald. Principles of metareasoning. *Artificial Intelligence*, 49:361–395, 1991.

[RZ94a] Jeffrey S Rosenschein and Gilad Zlotkin. Designing conventions for automated negotiation. *AI Magazine*, 1994. Fall.

[RZ94b] Jeffrey S Rosenschein and Gilad Zlotkin. *Rules of Encounter*. MIT Press, 1994.

[San94] Jayaram K Sankaran. On a dynamic auction mechanism for a bilateral assignment problem. *Mathematical Social Sciences*, 28:143–150, 1994.

[San93] T Sandholm. An implementation of the Contract Net Protocol based on marginal-cost calculations. In *Proc. 11th National Conference on Artificial Intelligence (AAAI-93)*, pages 256–262, 1993.

[San96] T Sandholm. Limitations of the Vickrey auction in computational multiagent systems. In *Second International Conference on Multiagent Systems (ICMAS-96)*, pages 299–306, 1996.

[San99] T Sandholm. An algorithm for optimal winner determination in combinatorial auctions. In *Proc. 16th International Joint Conference on Artificial Intelligence (IJCAI-99)*, pages 542–547, 1999.

[SL96] Tuomas W Sandholm and Victor R Lesser. Advantages of a leveled commitment contracting protocol. In *Proc. 14th National Conference on Artificial Intelligence (AAAI-96)*, pages 126–133, July 1996.

[SL97] Tuomas W Sandholm and Victor R Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94(1–2):99–137, 1997.

[SYM99] Y Sakurai, M Yokoo, and S Matsubara. A limitation of the generalized Vickrey auction in electronic commerce. In *Proc. 17th National Conference on Artificial Intelligence (AAAI-99)*, July 1999.

[Sat75] M A Sattherwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10:187–217, 1975.

[TW81] Jorgen Tind and Laurence A Wolsey. An elementary survey of general duality theory in mathematical programming. *Mathematical Programming*, 21:241–261, 1981.

[Var95] Hal R Varian. Economic mechanism design for computerized agents. In *Proc. USENIX Workshop on Electronic Commerce*, 1995.

[Vic61] W Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16:8–37, 1961.

[Wei98] G Weiß, editor. *Multi-Agent Systems.* The MIT Press, Cambridge, MA, 1998.

[WW99a] William E Walsh and Michael P Wellman. Efficiency and equilibrium in task allocation economies with hierarchical dependencies. In *Proc. 16th International Joint Conference on Artificial Intelligence (IJCAI-99)*, pages 520–526.

[Wel93] Michael P Wellman. A market-oriented programming environment and its application to distributed multicommodity flow problems. *Journal of Artificial Intelligence Research*, 1:1–23, 1993.

[Wel96] Michael P Wellman. Market-oriented programming: Some early lessons. In Clearwater [Cle96], chapter 4, pages 74–95.

[WWWMM99] M P Wellman, W E Walsh, P R Wurman, and J K MacKie-Mason. Auction protocols for decentralized scheduling. *Games and Economic Behavior*, 1999. To appear.

[Wur99] P R Wurman. *Market Structure and Multidimensional Auction Design for Computational Economies.* PhD thesis, University of Michigan, 1999.

[WW99b] P R Wurman and M P Wellman. Equilibrium prices in bundle auctions. In *Proc. AAAI-99 Workshop on Artificial Intelligence for Electronic Commerce*, pages 56–61, 1999.

[Wol81a] Laurence A Wolsey. Integer programming duality: Price functions and sensitivity analysis. *Mathematical Progamming*, 20:173–195, 1981.

[Wol81b] Laurence A Wolsey. A resource decomposition algorithm for general mathematical programs. *Mathematical Programming Study*, 14:244–257, 1981.

[Yok95] Makoto Yokoo. Asynchronous weak-commitment search for solving distributed constraint satisfaction problems. In *Proc. 1st International Conference on Principles and Practice of Constraint Programming (LNCS 976)*, pages 88–102. Springer-Verlag, 1995.