Fundamental Aspects of Privacy and Deception in Electronic Auctions

Felix Brandt

Institut für Informatik
der Technischen Universität München

Fundamental Aspects of Privacy and Deception in Electronic Auctions

Felix Brandt

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:		Univ.-Prof. Dr. Stefan Kramer
Prüfer der Dissertation:	1.	Univ.-Prof. Dr. Dr. h.c. Wilfried Brauer
	2.	Univ.-Prof. Dr. Martin Bichler

Die Dissertation wurde am 15.05.2003 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 21.07.2003 angenommen.

Kurzfassung

Auktionen spielen eine wesentliche Rolle im elektronischen Handel. Regierungen versteigern Lizenzen für Funkfrequenzen (wie kürzlich die UMTS-Mobilfunklizenzen), Millionen Bieter nehmen an Internet-Versteigerungen teil und Aufgabenverteilung in Multiagentensystemen sowie Beschaffung von Waren oder Dienstleistungen für Unternehmen erfolgen häufig durch Auktionen. Diese Arbeit gliedert sich in zwei Teile. Während sich der erste Teil mit strategischem Bietverhalten beschäftigt, werden im zweiten Teil Aspekte der Sicherheit und des Datenschutzes in Auktionen behandelt.

Es gibt Auktionsmechanismen, die bewiesenermaßen zu einer Güterverteilung führen, die den Nutzen aller beteiligten Personen (Käufer und Verkäufer) maximiert, und die strategisches Bieten überflüssig machen. Um diese Eigenschaften zu garantieren, werden jedoch einschränkende Annahmen im zu Grunde liegenden theoretischen Modell gemacht. So werden beispielsweise Absprachen zwischen Bietern (sog. Kollusionen) oder betrügerische Auktionatoren nicht betrachtet. In der vorliegenden Arbeit wird diese Liste um sog. „antisoziale“ Agenten ergänzt. Das Ziel dieser Agenten ist, neben der Maximierung ihres eigenen Nutzens, den Nutzen ihrer Konkurrenten zu minimieren. Es werden optimale Bietstrategien für antisoziale Agenten präsentiert und die Unmöglichkeit der Konstruktion eines Auktionsmechanismus gezeigt, der oben genannte Grundeigenschaften in Gegenwart von mindestens einem antisozialen Agenten aufweisen kann.

Im zweiten Teil dieser Arbeit wird der Schutz von verdeckten Geboten (beispielsweise bei Ausschreibungen) untersucht. Üblicherweise wird die Vertraulichkeit von verdeckten Geboten durch die Zuhilfenahme eines Dritten, dem Auktionator, sicher gestellt. Die Richtigkeit des Auktionsergebnisses und der tatsächliche Schutz der Gebote hängen allerdings vollkommen von der Vertrauenswürdigkeit dieser Instanz ab. Der Hauptbeitrag dieser Dissertation ist die schrittweise Entwicklung von verteilten Protokollen, die Auktionsmechanismen nachbilden ohne sich auf vertrauenswürdige Instanzen zu stützen. Dies wird unter anderem mit kürzlich entwickelten, kryptographischen Verfahren wie „multiparty computation“ und der Grundidee, die Bestimmung des Auktionsergebnisses auf die Bieter zu verteilen, erreicht. Die vorgestellten Protokolle sind sicher, unabhängig davon wie viele Bieter ihr Wissen teilen. In einigen können die Gebote selbst mit unbeschränktem Rechenaufwand nicht aufgedeckt werden.

Acknowledgements

First of all, my sincere thanks are due to Prof. Brauer, my supervisor. Without the pleasant working atmosphere provided at chair VII and his unconditional support, this work would not have been possible. Moreover, his comments greatly improved the presentation of this thesis.

I am much obliged to Prof. Bichler for interesting discussions and serving as an additional referee. Gerhard Weiß, co-author of some joint papers and leader of our research group, introduced me to auctions. So, in a way, he can be made responsible for the following pages. Thanks for the exciting and fruitful collaboration.

Many thanks have to go to my colleagues Michael Rovatsos, Klaus Stein, and Felix Fischer for proof-reading. The most exhaustive proof-reading was done by my girl-friend Monika. Thank you so much and sorry for countless days (and especially nights) that I spent working on this thesis.

Thanks to my colleagues Volker Baier, Markus Holzer, Matthias Nickles, and Stefan Römer, and former colleagues Claudia Brand, Dieter Bartmann, Till Brychcy, Astrid Kiehn, Clemens Kirchmair, Barbara König, Alexandra Musto, Peter Rossmanith, and Michael Sturm. They all contributed to the convenient atmosphere mentioned above.

Moreover, I am grateful to Masayuki Abe, Ivan Damgård, Hiroaki Kikuchi, Helger Lipmaa, and James Peck for helpful emails. Further thanks go to Marcus Martenstein from `econia`.

Abstract

Auctions have become a major phenomenon of electronic commerce. Governments use auctions to sell spectrum licenses, millions of users trade goods in Internet auctions, and task assignment in multiagent systems as well as procurement in the "real world" is handled via auctions. The contribution of this dissertation is two-fold. The first part addresses strategic bidding whereas the second part deals with privacy issues.

There are auction mechanisms that have been proven to lead to allocations of goods that maximize utility among participants (bidders and sellers) and to remove counter-speculation in the bid-preparation process. However, the theoretical model makes several restrictive assumptions to achieve those properties. For example, agreements between bidders ("bidder collusion") or untruthful auctioneers are not considered. This thesis extends this list by adding the notion of "antisocial" agents, i.e., agents that intend to maximize their own utility while minimizing their competitors' utilities. We present optimal antisocial bidding strategies and show that it is impossible to construct an auction mechanism that provides the properties mentioned above in the presence of at least one antisocial agent.

In the second part of this thesis, the privacy of sealed-bid auctions is investigated. Traditionally, privacy is obtained by consulting a trusted third-party, the auctioneer. However, the correctness of the outcome and the privacy of bids completely depend on the trustworthiness of the auctioneer. The major contribution of this dissertation is the incremental development of distributed protocols that emulate auction mechanisms without relying on any trusted party. This is achieved by applying recently developed cryptographic techniques of secure multiparty computation and distributing the determination of the auction outcome on bidders. The proposed protocols are secure despite any collusion of bidders. Some of them even provide partial privacy against computationally unbounded adversaries.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Auctions have become a major phenomenon of electronic commerce. In contrast to popular belief, auctions have not been invented for the purpose of entertainment and excitement alone. They represent mechanisms to solve a crucial economic problem. A person that wants to gainfully sell a good faces two problems:

- To whom does he deliver the good?
- And how much should he demand for it?

The assignment problem and pricing problem arise because there is a knowledge asymmetry between buyers and the seller. Each buyer has some individual information on a good's value but neither he nor the seller know the exact valuations of other potential buyers. Even in the simplest case (the so-called "independent private-value model", see Section 2.1.1), when a buyer's valuation does not depend on others' valuations, it is a non-trivial task to determine the appropriate price of a good. It may seem straightforward and optimal to have each potential buyer submit his valuation, award the good to the bidder who submitted the highest bid, and make this bidder pay the amount that he bid. This mechanism is called "first-price sealed-bid" auction (see Section 2.2.2). The main problem with this type of auction is that it is *not* a bidder's best strategy to submit his true valuation. In fact, as we will see in Section 3.3.1, bidders are best off bidding somewhat less than their private valuation, depending on their estimation of other bidders' valuations. It is obvious that this uncertainty may lead to sub-optimal outcomes. The field of mechanism design provides other, more sophisticated, mechanisms that solve this and various other problems (see Chapter 3). Giving up the private-value model, selling several goods at once, or allowing more than one seller obviously makes the task of finding an appropriate mechanism even more complex.

In short, auctions are mechanisms that determine the optimal pricing of goods while assigning the goods to buyers who value them the most. From an economic point of view, auctions thus belong to the field of pricing mechanisms. According to [Cas67] there are three common ways of pricing goods:

1. fixed pricing

2. private treaty pricing (**1** : **1** bargaining)

3. auctions (**1** : **n**, i.e. one seller and many buyers, or **m** : **n**, i.e. many sellers and many buyers)

Historically, bargaining is the oldest form of pricing and is still common in emerging countries and even in advanced societies at pre-retail level. It turned out that negotiating each price one-on-one is rather impractical for larger businesses. Fixed or “take-it-or-leave-it” pricing is much more convenient when some knowledge on the buyers’ valuations is known to the seller. On the other hand, fixed pricing is unsuitable when the seller has little, or no knowledge at all, about these valuations. The only way for consumers to express their preferences is to either buy or to refuse buying. The seller can then make a price adjustment based on these observations, resulting in another “take-it-or-leave-it” offer. Due to this inflexibility, *auctioning* off hard-to-value goods like art, antiques, and ground estates became popular. However, with the extension from classic marketplaces to corporate, national, and world-wide businesses, auctions became infeasible for the selling of mass products. As a consequence, fixed pricing has risen to the most popular pricing method in developed economies. Yet, with the emergence of large computer networks like the Internet, suddenly the logistical problems of large-scale auctions became tractable. The success story of `ebay`, `amazon`, and many other virtual marketplaces began.

In a simplified way, Figure 1.1 illustrates the flexibility of dynamic pricing (bargaining and auctions) in contrast to fixed pricing[1]. The variety of economic transactions that are conducted through auctions today is huge. Governments sell treasury bills, foreign exchange, mineral rights, radio spectrum rights, and firms to be privatized via auctions. Many types of contracts are awarded by procurement (reverse) auctions. Ground estates, houses, agricultural produce, livestock, art, antiques, and collectibles are commonly sold by auction. This list could be endlessly extended by the whole spectrum of goods that are sold in Internet auctions since the late nineties.

[1] In reality, dynamic pricing can flexibly react on more factors than just varying production cost, e.g. consumer demand or the underlying market structure.

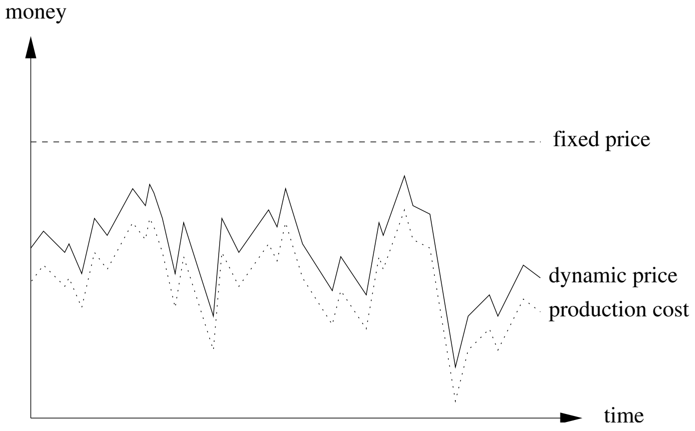


Figure 1.1: Fixed vs. dynamic pricing

1.1 A Brief History of Auctions

One of the earliest reports on auctions states that marriageable women have been sold in Babylon around 500 B.C. via auctions. Interestingly, bidding sometimes started at a negative price, meaning that “buyers” could receive a monetary compensation for marrying supposedly unattractive women. In ancient Rome, auctions were used for commercial trade in a building called “atrium auctionarium”. Unfortunately, little is known about the auction system used by the Romans. However, it is believed that they used some type of ascending English auction (see Section 2.2.1) as the word “auction” is derived from the Latin word “augere” which means “to increase”. One of the most bizarre and biggest auctions to date, was held by the Praetorian Guard after having killed the previous emperor Pertinax. Literally, the entire Roman empire was on auction in 193 A.D. Didius Julianus outbid all his competitors and the guard declared him the new emperor. After being in power for just two months he was put to death by the legions of his rival Septimius Severus who gained power and seized the capital. This tragic event can be explained by an effect called the “winner’s curse” (see Section 2.2.2) [KT01].

After at least hundred years of increasing occurrences of auctions in Great Britain, the most-prominent classic auction houses Sotheby’s and Christie’s were established in 1744 and 1766, respectively. In early America, auctions became popular for selling second-hand household furnishings, farm utensils, domestic animals, and, unfortunately, slaves. The nineteenth century saw the rise of fruits, vegetables, and flower auctions in the Netherlands and fish auctions in Germany.

Today, more than 62 million users are participating in auctions conducted by the world's largest Internet auction house ebay alone (Table 1.1)[2]. Furthermore, there are countless business-to-business (B2B) marketplaces and procurement auctions.

Registered Users	$62,000,000$
Listings	$638,000,000$
Gross Merchandise Sales (per day)	$\$ \ 50,000,000$
Net Revenue	$\$ \ 1,214,000,000$

Table 1.1: ebay company information 2002 [eba03]

1.2 Preference Aggregation

From an abstract point of view, auctions can be seen as a special case of preference aggregation mechanisms. Whenever a group of agents intends to come to a decision that affects the entire group, they need to aggregate their individual preferences[3], i.e., they have to compromise in order to find *global* preferences. The aggregation of conflicting preferences is one of the central topics of economics and multiagent systems. Two major problems have been considered in this context so far [MWG95, Var99].

Social choice problem The problem is to find a function that "*fairly*" aggregates conflicting preferences. The most important theorem in this context, Arrow's impossibility theorem[4], states that it is impossible to find such a function when preferences are unrestricted. When only allowing restricted preferences like so-called single-peaked preferences, fair social choice functions can be specified.

Mechanism design problem In order to be able to apply a social choice function, agents need to reveal their preferences. The mechanism design problem is to construct mechanisms that urge self-interested agents to reveal preferences *truthfully*. A mechanism is said to *implement* a social choice function if it leads to the same outcome as the social choice function and agents are best off submitting their true preferences. Similarly

[2] Interestingly, Germany is the fastest-growing market for ebay world-wide (including the U.S.).

[3] unless the group is led by a dictator

[4] As a consequence of Nobel Prize Laureate Kenneth J. Arrow's celebrated theorem, there are no sufficiently fair voting systems with more than two candidates.

to Arrow's theorem, the Gibbard-Satterthwaite theorem (Theorem 3.2) states the impossibility of finding such a mechanism for general preferences. However, there are solutions for restricted sub-domains, e.g. the Clarke tax mechanism for quasilinear preferences (see Section 3.2).

Traditionally, the existence of a central institution that receives all preferences and resolves the mechanism is assumed. In auctions, this party is usually called the auctioneer. However, as there is no reason to fully trust this third-party, neither the correctness of the result nor the privacy of the individual inputs can be guaranteed. Especially, incentive-compatible, i.e. truth-promoting, mechanisms might deter agents from participating as they require the submission of true valuations. Confidentiality of these valuations is essential for future negotiations and its revelation can be disastrous. This leads to a specification of the "preference protection problem".

Preference protection problem The problem, introduced in this thesis, is to enable the correct execution of a mechanism without trusted third-parties while preventing agents to learn preferences of other participants.

Figure 1.2 shows the resulting three-layer model. A protocol *emulates* a mechanism which in turn *implements* a social choice function.

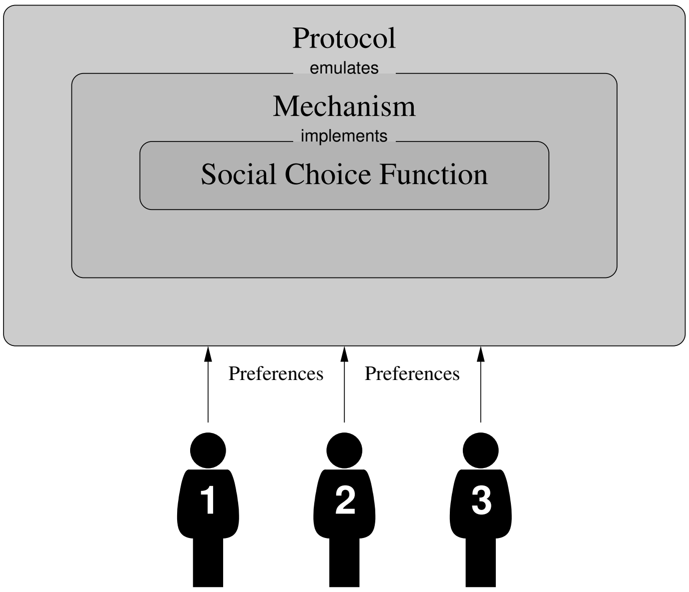


Figure 1.2: A three-layer model of preference aggregation

In the following, the abstract notions and problems described so far will be broken down to the case of single-unit auctions. First, the *social choice problem* for such auctions can be specified as follows. In a group of agents that have differing values for a good to be sold, to whom should the good be awarded? Preferences are severely restricted because a bidder is only able to evaluate a single outcome: the good being awarded to him. For this reason, the social choice problem can easily be solved by awarding the good to the agent who values it the most[5]. In order to identify the agent who values the good the most, agents need to reveal their individual values. So far, agents have no incentive to reveal their values truthfully. In fact, agents are best off submitting a bid as high as possible. The *mechanism design problem* is to make it an agent's optimal strategy to submit his true private value. This can be achieved by assigning payments to the agents that are based on their submitted bids. It turns out that assigning a payment that equals the second highest bid to the agent who values the good the most, the so-called Vickrey auction, is a very strong mechanism: An agent is always best off submitting his private value, no matter what all other agents do. The *preference protection problem* for auctions, whose solution is the main goal of this thesis, can be stated as follows:

> *Is there an interaction protocol that yields the outcome of the Vickrey auction without requiring agents to reveal their valuations to anybody?*

Usually, agents submit their values in a closed envelope (as "sealed bids") to the auctioneer. After having opened all envelopes, the auctioneer is able to compute the auction outcome. This approach is very critical for two reasons: The confidentiality of bids remains in the hands of the auctioneer and there is no way for the auctioneer to prove the correctness of the outcome (without revealing all bids). The preference protection problem is addressed in Chapters 5 and 6. Protocols in Section 5.3 are based on the incremental opening of commitment values whereas Chapter 6 introduces "secure multiparty computation", a sub-field of cryptography, in order to solve the preference protection problem.

Figure 1.3 shows the embedding of this work within classic fields of computer science and economics. There are numerous other research fields that investigate aspects of (electronic) auctions like operations research, data mining, or marketing which are not covered in this thesis.

[5] As we will see in Section 3.3.3, computational problems can arise when determining the winners in combinatorial auctions, however.

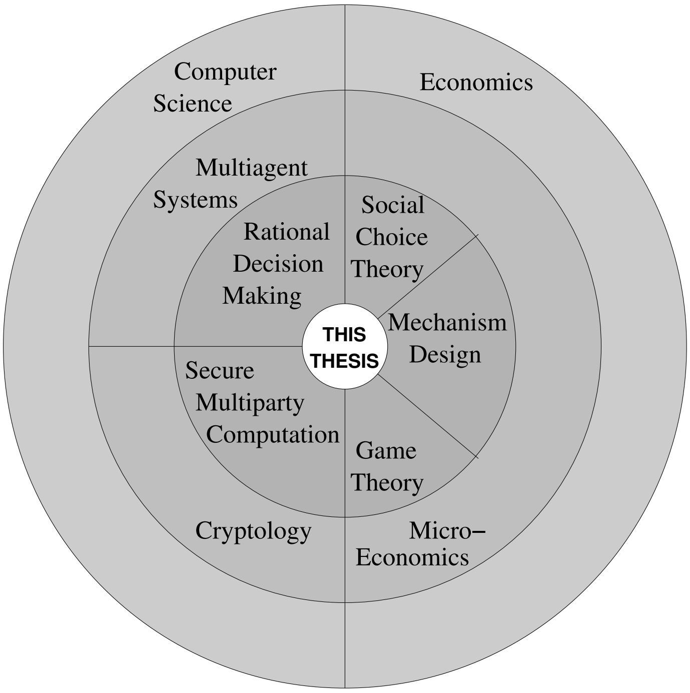


Figure 1.3: Related Fields

1.3 About this Thesis

This dissertation consists of two parts. The first part addresses strategic bidding and introduces antisocial agents whereas the second part deals with privacy issues and contains the secure auction protocol vX-share as a highlight. The remainder of the thesis is structured as follows.

Chapter 2 The second chapter defines basic models that enable the theoretical analysis of auctions and explains the most common auction types.

Chapter 3 This chapter introduces the economic fields of social choice theory and mechanism design. It explains how the Gibbard-Satterthwaite impossibility theorem is classically circumvented by assuming quasilinear preferences and proposes the Clarke tax mechanism as a solution to the mechanism design problem in this context. While Section 3.1, 3.2, and 3.4 compactly summarize the relevant economic literature, Section 3.3 represents a consequent application of these insights to the case of auctions.

Chapter 4 After introducing several well-known examples of deceptive behaviour in auctions like bidder collusion, shills, and sniping, the novel notion of "antisocial" agents is proposed. In the remainder of this chapter, the effects of this antisocial attitude in auctions are investigated theoretically and practically. Some of the obtained results have been previously published in [Bra00, BW01b, BW01a].

Chapter 5 The second part of this thesis begins by motivating the need for privacy protection in auctions and states the two main desiderata for secure auction protocols: privacy and correctness. The key observation of Section 5.2 is that no third-party can be fully trusted. For this reason, auctioneers are completely omitted and the determination of the auction outcome is designated to bidders themselves. Section 5.3 presents our first approach to provide privacy without trusted third-parties. The classification of collusion forms and the three partial revelation protocols have been published in [Bra01].

Chapter 6 This chapter establishes a general connection between preference aggregation and secure multiparty computation which is the most far-reaching contribution of this thesis [Bra03c, Bra03b]. For this purpose, essential properties like *weak robustness* and *full privacy* are introduced. Moreover, several concrete secure auction protocols that do not require any trusted third-parties are proposed. The most efficient protocol, vX-SHARE, privately computes Vickrey auction outcomes in a *constant* number of rounds. The protocols have been previously published in [Bra02a, Bra02b, Bra03a]. At the end of Chapter 6, a choice of existing secure auction protocols is briefly discussed.

Appendices Two of the most important underlying cryptographic concepts of the proposed protocols are explained in Appendix A: the discrete exponential function and commitment schemes. Appendix B briefly introduces two software programs that were implemented during the work on this dissertation.

Part I

Strategic Bidding

Chapter 2

Auctions

In order to enable the theoretical analysis of auctions, the various aspects of auctions have to be captured in models. The essential part of modeling is how bidders value goods and how they appraise money. This chapter explains the three basic value models: private, common, and correlated value, and the three basic agent models: risk-neutral, risk-averse, and risk-seeking. A presentation of the four most common auction types (English, 1st-price sealed-bid, Vickrey, and Dutch) is followed by short descriptions of three popular contemporary auction formats: consumer Internet auctions, B2B reverse auctions, and spectrum license auctions.

More extensive overviews are presented in [Kle99, Wol96]. The classic results can be found in [MM87, MW82, Mil89]. A chapter about auctions in the context of game theory is included in [Ras95].

2.1 Auction Models

It is often assumed that an agent's utility is linear in its wealth. Such agents are called "risk-neutral". However, in practice, it can be observed that agents, in particular humans, have diminishing value of money. They are "risk-averse". The reader can quickly check if he is risk-averse by asking himself whether he would prefer a guaranteed amount of $ 1,000,000 over a fifty-fifty chance of getting $ 2,000,000. Risk-averse agents prefer the former, whereas risk-neutral agents are indifferent. There would be no insurance companies if most of us were not risk-averse. The categorization can be completed by introducing "risk-seeking" agents. These agents would choose the fifty-fifty chance of receiving $ 2,000,000 in the example above. Figure 2.1 shows the agents' utility functions subject to their wealth. Risk-neutral agents have linear utility functions, while the utility functions of risk-averse and risk-seeking agents are concave and convex, respectively.

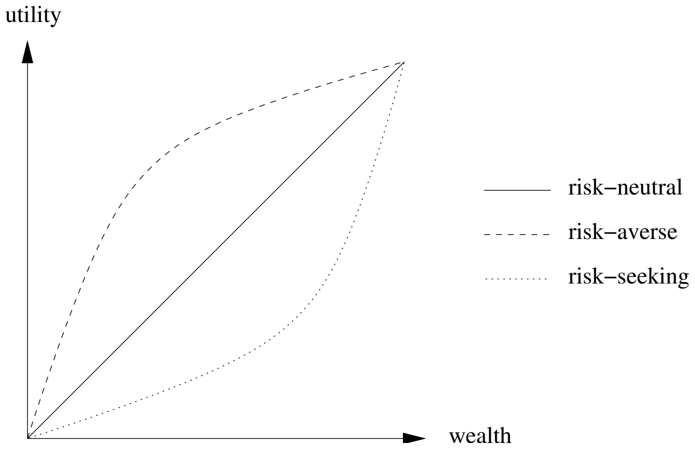


Figure 2.1: Risk models

As mentioned in Chapter 1, a key feature of auctions is the presence of asymmetric information. This asymmetry can be modeled in several ways to capture different types of goods to be sold. We will distinguish between the value and the valuation of a good in the following. An agent's *value* of a good exactly prescribes how much the good is worth to him. As the agent might not know the true value of a good (while bidding in an auction), he needs to compute an estimated value: his *valuation*.

The first two value models presented in the following are special cases of the third general model. In the remainder of this thesis, we will assume the private-value model and risk-neutral agents unless otherwise noted.

2.1.1 Private Values

In the private-value model, bidders have private, independent valuations of the object to be sold. They do not have to estimate their values. Thus, values and valuations are equal in this model. An example is the selling of a piece of cake that will be eaten after the auction. The good's purpose is to be consumed by the buyer. He does not intend to resell it to other bidders. As the value is independent of other bidders' values, knowing all other values in advance will not change one's own valuation. However, this knowledge might likely affect a bidder's *strategy* (depending on the auction type). Figure 2.2 illustrates this model by depicting three bidders who all precisely know how much the good is worth to them (v_i).

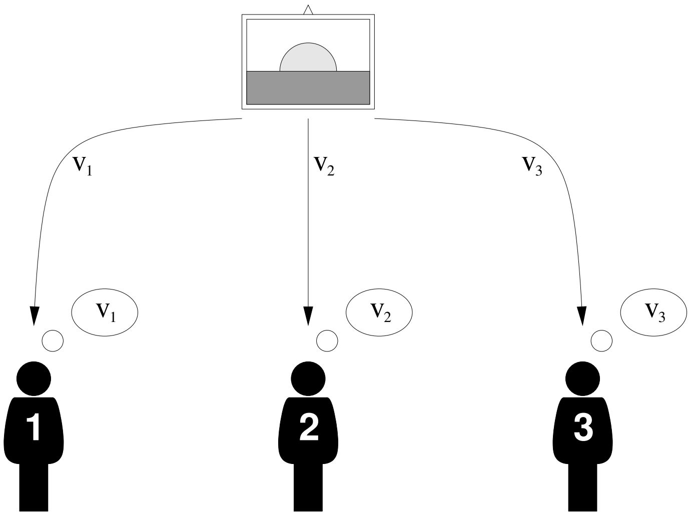


Figure 2.2: Private independent values model

2.1.2 Common Value

In the common-value model, the good to be sold has exactly the same value to all bidders. However, the bidders have differing *valuations* depending on private information that is available to them. To give an example, the selling of a jar filled with coins belongs to this category. Obviously, the real value is identical for all bidders. However, each bidder has a different valuation based on his private estimation. These estimations are based on internal information and so-called “signals” that are available to subsets of agents.

In the setting of Figure 2.3, the good has identical value v to all three agents. Yet, all bidders have their own valuation functions $\hat{v}_i(\cdot)$ subject to internal knowledge and signals. Bidder 1 has access to signal s_1 and s_2, whereas bidder 2’s valuation function is founded on signal s_2 alone. The third bidder has no access to any of the signals.

The auctioning of treasury bills or oil tracts are well-known examples of common-value auctions. In a famous experiment [BS83], jars filled with coins have been auctioned off to students at Boston university. The secret value of each jar was exactly \$ 8. After a series of auctions, it turned out that the average bid was \$ 5.13. The average winning bid, however, was \$ 10.01. It follows that the average winning bidder *lost* money by winning an auction. This important problem is called “winner’s curse”. All reasonable auction types have in common that the bidder with the highest valuation wins the auction. However, the true value of the good, which is equal to all bidders,

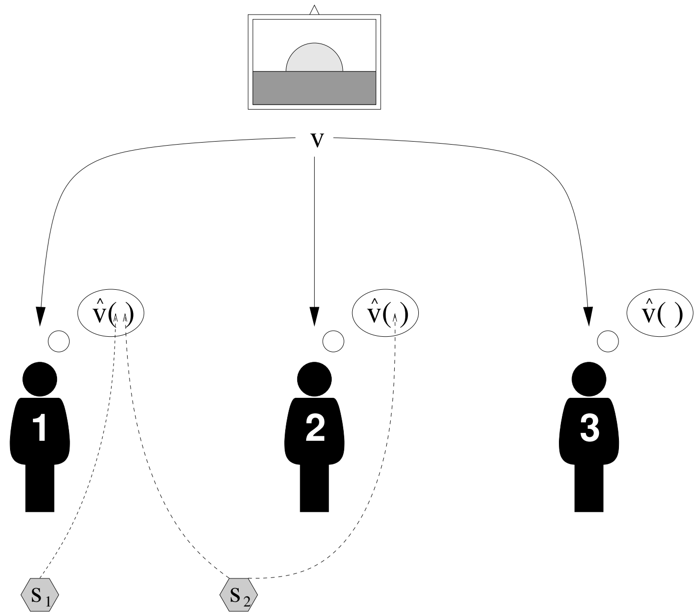


Figure 2.3: Common-value model

statistically lies somewhere in the middle of the different valuations. As a consequence, the bidder that overestimated the good the most is declared the winner. He likely paid too much for the good which in turn leads to bidders biasing their bids downwards. The strategic implications of the winner's curse are far-reaching and hard to analyze [LK02]. The winner's curse is made responsible for low profits of oil and gas corporations on drilling rights in the Gulf of Mexico that have been sold in auctions [Ras95].

2.1.3 Correlated Values

Finally, the correlated value model is a general model that includes the previous two as special cases. Bidders have private estimations regarding their real values. These values may differ and can depend on other bidders' values. The correlated-value model is the one most likely to come across in real-world auctions. However, auction theory mostly uses the private-value model (and sometimes the common-value model) as it simplifies theoretical analysis.

2.2 Auction Types

2.2.1 English

The English or "ascending open-cry" auction is by far the most prominent auction type. The price is raised successively until only one bidder remains. This bidder wins the good at the final price. There are various ways to conduct English auctions. The seller can announce prices, or the bidders can call out prices themselves. Moreover, many different realizations of bid increments are possible. The most common variation used in auction theory is the so-called Japanese auction. The price rises progressively while bidders quit the auction one after another. Bidders can observe when someone quits and no bidder is able to re-enter the auction.

2.2.2 1$^{\text{st}}$-Price Sealed-Bid

In a 1$^{\text{st}}$-price sealed-bid auction, each bidder independently submits a single sealed bid. The bidder that submitted the highest bid is awarded the good and pays the amount that he specified in his bid. This type of auction is frequently used in procurement scenarios where competing contractees submit bids and the *lowest* bidder is awarded the contract. More generally, in a regular auction there is one seller and many buyers whereas in a so-called *reverse* auction there are many sellers and one buyer.

The winner's curse problem that we mentioned in Section 2.1.2 appears in another form in 1$^{\text{st}}$-price sealed-bid auctions, even in the private-value model. The winner of an auction can easily figure out that he could have won the auction by bidding less, namely slightly more than the second highest bid (which is unknown to him).

2.2.3 Vickrey

The Vickrey or "2$^{\text{nd}}$-price sealed-bid" auction was proposed by Nobel Prize Laureate William S. Vickrey in 1961 [Vic61]. It is almost identical to the previous auction type. The only difference is that the winning bidder has to pay the amount of the *second* highest bid[1]. Figure 2.4 shows how the Vickrey auction works (discrete bars represent bids).

In the private-value model, the Vickrey auction has a dominant strategy, which means that if an agent applies this strategy he receives the highest possible payoff, no matter which strategies are used by other bidders. The

[1] If two or more bidders tie for the highest bid, the winner is picked at random and has to pay the amount of his bid (because it is equal to the second highest bid in this case).

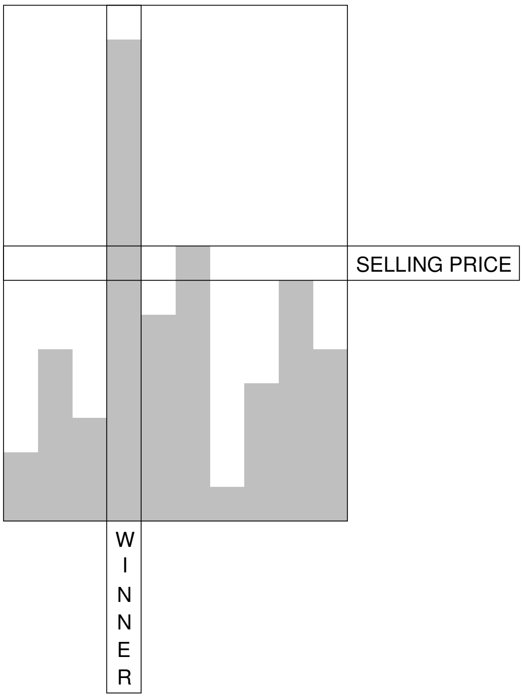


Figure 2.4: An example Vickrey auction

dominant strategy is to bid one's true valuation of a good. Even if an agent knows all the other bids in advance, he still does best by bidding his private value. This leads to a so-called "dominant-strategy equilibrium" (to be formally defined on page 29): All agents are best off using the same strategy[2]. It can easily be seen by case analysis why bidding the private value is an optimal strategy.

Theorem 2.1 (Dominant strategy in Vickrey auctions)
Assuming the private-value model, it is a dominant strategy to bid one's true valuation of a good in Vickrey auctions.

Proof: Given two agents, A and B, and their corresponding private values v_a and v_b, the profit or utility of each agent can be written as a function of

[2] If bidders don't have to estimate their private values, the dominant strategy equilibrium exists independently of risk neutrality [Ras95].

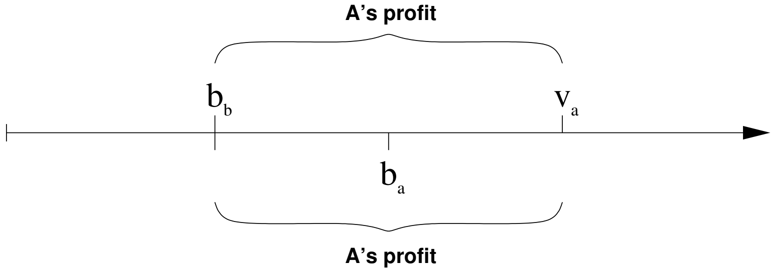
submitted bids (b_a and b_b).

$$u_a(b_a, b_b, v_a) = \begin{cases} v_a - b_b & \text{if } b_a \geq b_b \\ 0 & \text{if } b_a \leq b_b \end{cases} \qquad u_b(b_a, b_b, v_b) = \begin{cases} 0 & \text{if } b_a \geq b_b \\ v_b - b_a & \text{if } b_a \leq b_b \end{cases} \tag{2.1}$$

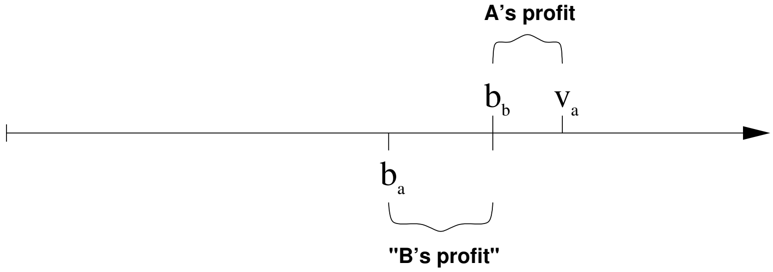
We consider agent A and investigate the possible profits he would make by not bidding his private value[3]. It suffices to model only one opposing agent B representing the entire competition because A only cares whether he wins or loses. He does not draw distinctions between his fellow bidders.

If A bids less than his private value ($b_a < v_a$) there are three subcases conditional on agent B's bid b_b. (The outcome that would have occurred if A had bid v_a is displayed at the top of figures. The outcome shown at the bottom of figures relates to the case in which he deviates from the dominant strategy[4].)

i) $b_b < b_a < v_a$: A wins the auction, but does not make more profit then the profit he would have made when bidding v_a, because the price remains b_b and his profit is still $v_a - b_b$.



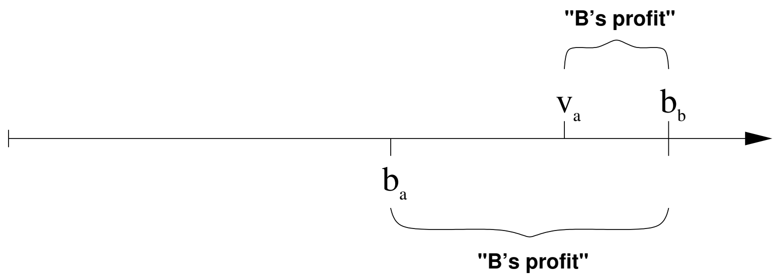
ii) $b_a \leq b_b < v_a$: A loses the auction, instead of winning it by bidding v_a. If $b_a = b_b$, he might win the auction but does not make more profit than when bidding v_a.



[3] There are certainly shorter proofs for this theorem, but case vi) of the following case analysis is essential for introducing antisocial bidders in Chapter 4.

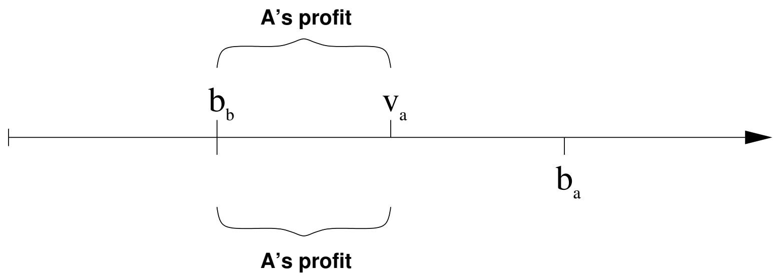
[4] In the figures, B's profit is in quotation marks because this only marks his profit if he bid his private value. The proof, however, works no matter which strategy B adopts.

iii) $b_a < v_a < b_b$: B wins the auction and has to pay less money than if A had bid v_a.

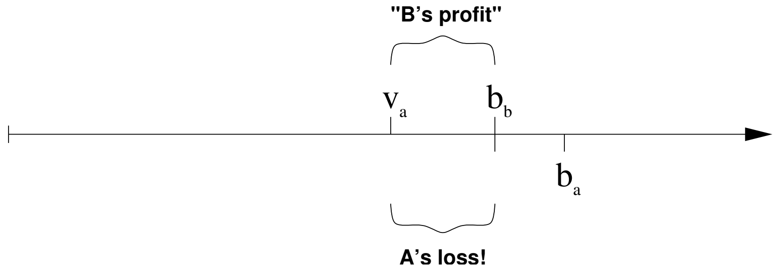


If A bids $b_a > v_a$ the following cases describe the resulting situations.

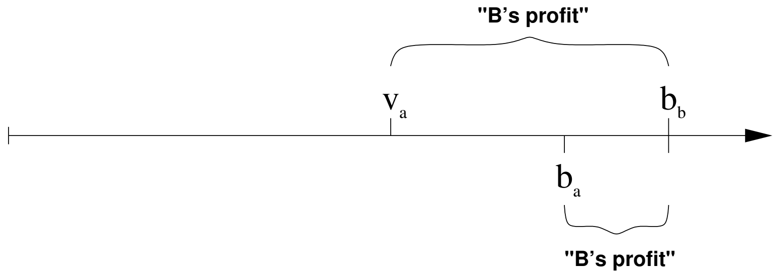
iv) $b_b < v_a < b_a$: A wins, but has to pay the same amount of money (b_b) that he would have paid by bidding v_a.



v) $v_a < b_b \leq b_a$: A wins, but has to pay more than the good is worth to him, i.e., he is losing $b_b - v_a$. If $b_a = b_b$, he might lose the auction.



vi) $v_a < b_a < b_b$: A loses and reduces B's profit by $b_a - v_a$.



Concluding, bidding anything else than v_a cannot yield more profit than bidding the true valuation v_a. □

Obviously, this extremely simplifies the bid preparation, due to the absence of wasteful counter-speculation which is required for example in 1st-price auctions. Surprisingly, the Vickrey auction is rarely used in practice. Reasons for the Vickrey auction's sparseness will be given in Chapter 5.

Similar to the 1st-price sealed-bid auction, the Vickrey auction can be used as a reverse auction, e.g. to assign tasks. Contractees submit bids that indicate how much money they want to receive for accomplishing the task. The cheapest contractee wins the auction and receives the amount submitted by the second cheapest bidder.

In the private-value model, the Vickrey and the English auction are *strategically equivalent*, i.e., there is a mapping from Vickrey auction strategies to English auction strategies and vice versa (see Table 2.1). Bidding b in a Vickrey auction and "bidding as high as b" in an English auction yield exactly the same outcome. This only holds in the private-value model as bid information revealed during an English auction affects bidders' strategies. Bidders tend to bid more in English auctions.

2.2.4 Dutch

In a Dutch auction, the auctioneer announces a decreasing bid starting with the highest possible price. The price decrease can happen continuously or in discrete intervals. The first bidder that stops the auction by expressing his willingness to pay is awarded the contract for the amount of the actual bid. The Dutch auction's name originates from the selling of Dutch flowers where an electronic device with buzzers connected to a clock is used to implement the Dutch auction mechanism. Fish are sold in Israel in a similar way, as is tobacco in Canada. Dutch auctions are particularly suitable for perishable goods that lose value *during* the auction[5]. [Wol96] describes an interesting form of the Dutch auction "in disguise" that is used to sell clothes in the United States. Items are sold at a fixed price *minus* a discount that depends on how many weeks an item is on the shelf. Thus, the price constantly decreases (until some minimum price is reached).

Interestingly, it turns out that the Dutch auction and the 1st-price sealed-bid auction are *strategically equivalent*, i.e. the Dutch auction generates exactly the same outcome as the 1st-price sealed-bid auction. This holds independently of the assumed value model. The continuous price decrease

[5] Obviously, this is not covered by the theoretical value models described at the beginning of this chapter.

reveals no information that bidders could use to update their valuations and the highest bidder wins the auction.

Figure 2.1 summarizes the equivalences of the presented auction types. Auction types listed in the same row are strategically equivalent in the given value model. The 2nd-price Dutch auction (see page 79) is listed in brackets because it is not used in practice (as far as the author knows). Due to these equivalences, we will use the term "1st-price" auctions for Dutch and 1st-price sealed-bid auctions, and "2nd-price" auctions for English and Vickrey auctions.

Value model	Sealed-bid	Ascending	Descending
any	1st-price	—	Dutch
private-value	Vickrey	English	(2nd-price Dutch)

Table 2.1: Strategic equivalences of major auction types

2.3 Contemporary Auctions

In recent times, the rise of computers and the Internet gave birth to countless virtual marketplaces and modern auction sites. Besides well-known consumer Internet auctions, a vast amount of goods and services is sold in B2B reverse auctions. Recently, the selling of next generation mobile phone spectrum licenses via auctions across Europe gained much attention. Moreover, auctions are used to sell electrical power (on a daily basis in the United Kingdom), treasury bills, or oil field drilling rights.

2.3.1 Consumer Internet Auctions

Since the formation of `ebay` in 1995, various Internet auction houses attract millions of users world-wide. The type of articles to be sold ranges from CDs and mobile phones to cars and pieces of real estates on the moon. English auctions are predominantly used by Internet auction houses, though there are subtle differences in the particular auction rules. Some auction houses (e.g. `ebay`) allow the submission of bids until some pre-determined point of time whereas others (e.g. `amazon`) only stop the auction when no bid has been submitted for a certain period of time. The former seemingly attracts a phenomenon called "sniping" (see Section 4.1.2).

There are two interesting additions that have recently been made to `ebay`'s auction rules. The first is the disposition of "bidding agents", i.e. agents that keep raising one's bid if necessary until a private value that can be specified by the user is reached. This removes an interested bidder's obligation to constantly monitor an auction and adjust his bid accordingly. The following excerpt explains what happens if all participants are using bidding agents.

> "From a game theoretic point of view, the 'agents' in traditional Internet auctions convert the auction protocol from an English auction to a Vickrey auction: the participant with the highest willingness to pay gets the item at the price of the willingness to pay of the second highest participant. This is an interesting real world manifestation of the revelation principle [Theorem 3.1]. It states that any outcome that can be supported in equilibrium via a complex protocol can be supported in an equilibrium via a protocol where the agents reveal their types truthfully in a single step. The proof is based on having the new protocol incorporate a virtual player for each real world participant such that the virtual player will find and play the best strategy for the original complex protocol on behalf of the real world participant—given that the participant reveals his preferences to the virtual player. Because the virtual player will play optimally for the participant, the participant is motivated to reveal his preferences truthfully. Each 'agent' in current Internet auctions is a materialization of such a theoretical virtual player". [San00]

It is important to note that, using a bidding agent, one might reveal unnecessary information. When monitoring an auction oneself, it may turn out to be unnecessary to bid at all due to submitted bids that are higher than one's valuation.

A second innovation is `ebay`'s so-called "private auction" in which the identities of bidders that (temporarily) submitted the highest bid are not disclosed. Of course, this method does not apply cryptographic means (as described in Chapters 5 and 6) and there is apparently no need in using *real* privacy-enhancing techniques for the type of articles sold on `ebay`. However, it is quite evident that privacy completely depends on the trustworthiness of the auction provider in this case.

As mentioned before, Internet auctions are extremely successful nowadays. The only shortcoming that could be attributed to these auctions is the growing number of fraud cases and the inability to prevent fraud in general (see Chapter 4).

2.3.2 B2B Reverse Auctions

Instead of using conventional means of acquiring supplies, many companies have begun to rely on B2B (business-to-business) reverse auctions to source their business requirements and bring about cost savings in the supply chain. As mentioned earlier, all of the auction types proposed in Section 2.2 can be used as a "reverse auction". More specifically, a company wanting to source products or services holds an auction in order to buy these products from the bidder with the *lowest* prices and best terms. Typically, a small number of pre-qualified suppliers are invited to participate in an auction, and bid against one another in order to win a supply contract. Two characteristic examples for such B2B auctions, held by the German B2B marketplace `econia`, are the purchase of 6,500 personal computers (including monitors and printers) for a major German bank or the acquisition of 700,000 cotton carrying bags for a trading concern.

According to a recent report from Forrester Research, B2B reverse auctions are predicted to generate $ 745.8 billion in sales by 2004.

2.3.3 3G Spectrum License Auctions (UMTS)

The third generation (3G) mobile phone spectrum licenses, also called "Universal Mobile Telecommunications System“ (UMTS), have been assigned from 1999 to 2001. Few countries (e.g. Spain) used administrative reviews (so-called "beauty contests") to award licenses. Most countries decided to sell licenses via auctions, some of which turned out to be the most revenue generating auctions in modern times. Apart from the Danish sealed-bid auction[6], all other countries used some type of simultaneous ascending auction (SAA) for different spectrum blocks[7]. However, there were subtle differences in the applied auction types and rules. While the UK and German auctions performed very well (in terms of revenue generation: they earned 37.5 billion Euro and 50.8 billion Euro, respectively), some countries like The Netherlands, Austria and Switzerland performed very poorly which was mostly due to dramatically falling valuations of UMTS licenses (estimated value shrank to about one-tenth in a single year [Kle02a]). This deterred many potential bidders to take part in such auctions at all and can clearly be seen by looking at the Swiss auction where the number of bidders shrank from nine to four, just weeks before the auction. As there were just four licenses for sale and

[6] in which all (four) winners had to pay the amount of the fourth highest bid which was the only bid revealed to the public

[7] Each spectrum block is sold in a separate English auction. All auctions are synchronized and take place simultaneously.

it was foreseeable that missing competition will lead to a disastrous result, the Swiss government postponed the auction and tried to change its rules, in vain. Adding the fact that the reservation price (the "minimum bid") was ridiculously low (due to the positive experiences made in the UK and Germany), Switzerland earned just 2% of the revenue generated by the German auction *per capita*. A sealed-bid auction might have attracted more bidders. However, SAAs have been preferred in most countries for three reasons [dVV01]:

- The U.S. FCC (Federal Communications Commission) has a long history of successful spectrum bandwidth auctions using SAAs.
- Open-cry auctions are completely transparent to the public. It is difficult to accuse the auctioneer of favouritism (see Chapter 5).
- Spectrum licenses strongly relate to the common-value model defined in Section 2.1.2. Open-cry auctions generate more revenue than sealed-bid auctions in the common-value model.

In the following, we address two observations that have been made in the German UMTS auction and that relate to the focus of this thesis. The German (and Austrian) auction differed from other 3G license auctions in that the number of licenses to be sold was not fixed. Financially strong bidders were able to acquire more spectrum blocks than others and thus prevent weak competitors from attaining a license at all. In both auctions, the number of licenses to be allocated was between four and six (in contrast to all other auctions where it was fixed to either four or five). As price arrangements in auctions are usually illegal, it has become common practice in ascending auctions to use the last digits[8] of bids to signal one's intentions or to attempt to coordinate actions [GRW02]. In fact, three major providers (German Telekom, Mannesmann-Vodafone, and 3G (which changed its name to "Quam" later)) constantly used the digit "6" in their public bids without any obvious reasons. This behaviour has been interpreted as an attempt to indicate the willingness to settle with a market structure of six licenses [GRW02]. Interestingly, German Telekom later signalled the digit "5" which some observers explained by Telekom's decision to crowd out another bidder and indicating that purpose to Mannesmann-Vodafone, the other major provider. As a matter of fact, German Telekom kept raising the price at a

[8] The smallest bid increment in the German UMTS auction was set to DM 100,000 for this reason. Furthermore, only the highest bid and the corresponding bidder have been revealed by the auctioneer after each bidding round. Apparently, this did not prevent signalling.

point of time at which only six bidders were remaining and it was in Telekom's hands to stop the auction. After a while, Telekom gave up to crowd out one of the weak competitors and the record number of six licenses was awarded.

Two conclusions can be drawn from theses events. First, signalling is a serious problem in SAAs (that can be avoided by (partly) using sealed-bid auctions instead), and secondly, the behaviour of German Telekom can be explained by the "antisocial attitude" as described in Section 4.3. Experts wondered why Telekom gave up to crowd out competitors at some point because in the end its behaviour only resulted in higher prices for every participant [Kle02a, GRW02]. However, the situation could be explained by stating that Telekom invested money in order to inflict losses to weak competitors that may not be able to cope with the high amounts spent for the licenses. Actually, as this is written, one of the weak providers (Mobilcom) is desperately trying to sell its license and another one (Quam) discontinued its operative business. Experts believe that the only two telecommunications providers being able to survive the current crisis for sure are German Telekom and Mannesmann-Vodafone.

There is an ongoing debate why some of the 3G spectrum auction schemes apparently failed and others did not, e.g., there are assertions that German Telekom's objectives were affected by the fact that it was majority-owned by the German government. More details on the 3G auctions can be found in [Kle02a, dVV01, GRW02, Kle02b, BE00].

Chapter 3

Microeconomic Foundations

This chapter provides the theoretical basis for the analysis of auctions and auction strategies and the construction of desirable auction mechanisms. Microeconomic theory in general and especially game theory has emerged as an indispensable fundament of agent research as it formally describes systems of rational, self-interested agents. Our view on microeconomics will be limited to some basic game-theoretic solution concepts and mechanism design. More extensive overviews are included in [MWG95, Var99, Ras95]. A nice and short introduction to mechanism design and its applications to electronic commerce can be found in [Var95]. [Par01] contains a very good introduction and deals with the computational aspects of mechanism design. [RZ94, NR99] provide further examples of computer science problems that can be approached by using game theory and mechanism design.

We will first consider social choice under the simplistic assumption of complete information and then, for the major part of this chapter, assume a model of incomplete information. Most of the proofs will be omitted[1]. $[n]$ denotes the set of natural numbers less or equal than n ($[n] = \{1, 2, \dots, n\}$).

3.1 Social Choice and Incentives

Consider a group of n self-interested agents that has to make a collective decision that affects all agents. After having agreed on a "social choice function" that prescribes which decision is to be made subject to the agents' private preferences over the different outcomes, e.g. the choice that maximizes total utility, the agents run into the following problem.

As their preferences are private information, they need to reveal them in

[1] We did not include Arrow's impossibility theorem despite its beauty because it does not hold in allocation scenarios in which utilitarian social welfare is maximized.

order to be able to determine the social choice. But what if lying about one's preferences can lead to a higher individual utility than telling the truth? The "mechanism design problem" is to construct a mechanism that implements a social choice function while meeting miscellaneous, useful demands, e.g. pointlessness of lying or allocative efficiency. A mechanism defines the possible actions of the agents and a mapping from the agents' actions to the outcome.

Agents' preferences can be modeled in various ways, e.g. by defining a preference relation or by assigning values of utility for specific outcomes[2]. The latter is appropriate when allocating goods. Furthermore, it is quite useful to define the "utility function" in dependence of an agent's "type". An agent's *type* $\theta_i \in \Theta_i$ specifies the individual preferences of the agent.

The utility of an agent depends on the outcome of a social choice function (or a mechanism) and his type, and prescribes the agent's benefit from a given outcome.

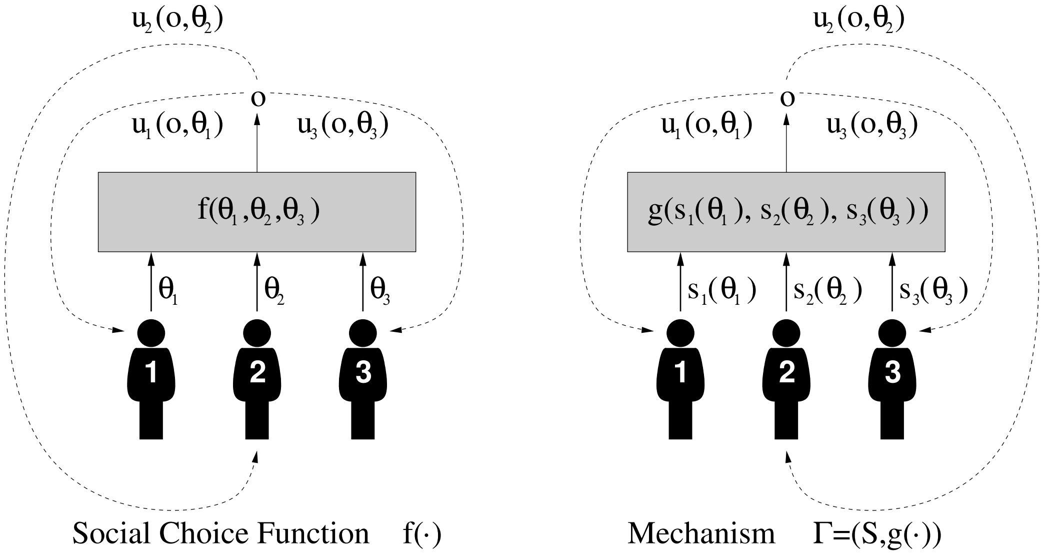
Definition 3.1 (Utility)
$u_i(o, \theta_i)$ is called the *utility* of agent i for the outcome $o \in \mathcal{O}$ given his type θ_i.

We henceforth assume that agents are expected utility maximizers. To save space, we will use the abbreviations $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ and $\Theta = \Theta_1 \times \Theta_2 \times \cdots \times \Theta_n$.

Definition 3.2 (Social choice function)
A *social choice function* $f : \Theta \mapsto \mathcal{O}$ assigns an outcome $f(\theta) \in \mathcal{O}$ to each possible profile of agents' types $\theta \in \Theta$.

The left part of Figure 3.1 illustrates that a social function computes the outcome based on the individual types. Agents receive utility depending on the outcome and their personal type. One of the most important features of a social choice function is that we want it to yield outcomes that are socially preferable.

[2] Every rational, i.e. complete and transitive, preference relation (on a finite set of alternatives) can be mapped to a utility function.



Social Choice Function $f(\cdot)$

Mechanism $\Gamma=(S,g(\cdot))$

Figure 3.1: Social choice

Definition 3.3 (Pareto-optimality)
A social choice function $f(\theta) = o$ is *Pareto-optimal* (or Pareto-efficient or ex post efficient) if $\forall i \in [n], o' \in \mathcal{O}, \theta \in \Theta$ with $o' \neq o$

$$u_i(o',\theta_i) > u_i(o,\theta_i) \quad \Rightarrow \quad \exists j \in [n] : \; u_j(o',\theta_j) < u_j(o,\theta_j) \quad .$$

In other words, a social choice function is Pareto-optimal if, given its outcome, no agent could be made better off without reducing another agent's utility. A nice property of Pareto-optimality is that it is independent of the actual utility values; only the utility ordering is considered.

Unfortunately, there are usually numerous Pareto-optimal outcomes. For example, a dictatorial social choice function that selects the outcome that gives a single agent, the dictator, the highest utility is trivially Pareto-optimal. In order to be able to measure the "quality" of social choice functions more precisely, the notion of social welfare has been introduced. Social welfare functions aggregate the individual utility functions into a single function that describes the "social utility". A reasonable restriction seems to only allow welfare functions that are increasing in each individual's utility. We will stick with the classic utilitarian setting of social welfare which is defined as the sum of all individual utility functions. The most desirable outcome maximizes social welfare and thus is also Pareto-optimal[3].

Definition 3.4 (Social welfare)
A social choice function $f(\theta) = o$ is *social-welfare-maximizing* if $\forall o' \in \mathcal{O}$, $\theta \in \Theta$

$$\sum_{i=1}^{n} u_i(o, \theta_i) \geq \sum_{i=1}^{n} u_i(o', \theta_i) \quad .$$

So far, we only regarded social choice as a function of agents' preferences. In the following, we will extend the model to allow for strategic revelation of false preferences which will lead us to the mechanism design problem.

Definition 3.5 (Strategy)
A *strategy* $s_i(\theta_i) \in S_i$ defines the action an agent will take in every possible state of a mechanism, given its type θ_i.

Strategies can be deterministic (pure) as well as stochastic (mixed). It is deliberately left open *what* strategies are precisely. They can be single numbers or complex sets of rules. Like above, we will use the short forms $s = (s_1, s_2, \ldots, s_n)$ and $S = S_1 \times S_2 \times \cdots \times S_n$.

Definition 3.6 (Mechanism)
A *mechanism* $\Gamma = (S, g(\cdot))$ consists of n strategy sets S_i and an outcome function $g : S \mapsto \mathcal{O}$.

Analogous to the definition of a social choice function (Definition 3.2), a mechanism yields an outcome. This time, the outcome depends on the individual strategies rather than the true preferences (see Figure 3.1). Note that the execution of a mechanism can be a lengthy process of actions and counter-actions by the agents based on their strategies.

[3] Of course, there are other conceivable social welfare functions, e.g. the minimax (or Rawlsian) function that defines welfare as the utility of the worst off agent, but the utilitarian setting especially makes sense in allocation scenarios that we are considering.

Definition 3.7 (Mechanism implementation)
A mechanism $\Gamma = (S, g(\cdot))$ *implements* a social choice function $f(\cdot)$ if there is an equilibrium strategy profile $(s_1^*(\cdot), s_2^*(\cdot), \ldots, s_n^*(\cdot))$ for Γ such that $\forall \theta \in \Theta$

$$g(s_1^*(\theta_1), s_2^*(\theta_2), \ldots, s_n^*(\theta_n)) = f(\theta) \quad .$$

The mechanism design problem is to find a mechanism that implements a given social function “in equilibrium”. In the following, we will present several equilibrium concepts. An equilibrium or *solution concept* specifies when a strategy profile is “best” for all players.

The strongest equilibrium concept possible is that of dominant strategies. In a dominant strategy equilibrium each agent is best off using his equilibrium strategy no matter which strategies the other players choose. We will use the notation $s_{-i} = (s_1, s_2, \ldots, s_{i-1}, s_{i+1}, s_{i+2}, \ldots, s_n)$ to denote the strategy profile without i's strategy s_i. $S_{-i} = S_1 \times S_2 \times \cdots \times S_{i-1} \times S_{i+1} \times S_{i+2} \cdots \times S_n$ is defined analogically.

Definition 3.8 (Dominant-strategy equilibrium)
The strategy profile s is in *dominant-strategy equilibrium* of mechanism $\Gamma = (S, g(\cdot))$ if $\forall i \in [n],\ \theta \in \Theta,\ s_i' \in S_i,\ s_{-i} \in S_{-i}$

$$u_i(g(s_i(\theta_i), s_{-i}(\theta_{-i})), \theta_i) \geq u_i(g(s_i'(\theta_i), s_{-i}(\theta_{-i})), \theta_i) \quad .$$

A dominant-strategy equilibrium is very robust, because an agent's strategy is independent of available information on other agents. Obviously, dominant-strategy equilibria do not always exist. A weaker, less demanding solution concept is that of a Nash equilibrium, named after Nobel Prize Laureate John F. Nash. In a Nash equilibrium, there is no reason to deviate from the equilibrium strategy as long as all other players choose their equilibrium strategy. In a way, the equilibrium strategies are well-balanced.

Definition 3.9 (Nash equilibrium)
The strategy profile s is in *Nash equilibrium* of mechanism $\Gamma = (S, g(\cdot))$ if $\forall i \in [n],\ \theta \in \Theta,\ s'_i \in S_i$

$$u_i(g(s_i(\theta_i), s_{-i}(\theta_{-i})), \theta_i) \geq u_i(g(s'_i(\theta_i), s_{-i}(\theta_{-i})), \theta_i) \quad .$$

In contrast to the dominant-strategy equilibrium, a Nash equilibrium is generally not unique and, like with dominant strategies, there are cases in which no Nash equilibrium exists. However, when allowing mixed, i.e. randomized, strategies, it has been proven that there is at least one Nash equilibrium in any “game”. Nevertheless, the concept of a Nash equilibrium is somewhat useless in the context of mechanism design as it requires agents to have complete information about the other agents' preferences (and their rationality) in order to be able to identify the equilibrium. If preferences were common knowledge, there would be no need to design a mechanism. The asymmetry of information demands a solution concept that is based on beliefs about others' preferences rather than knowledge.

Definition 3.10 (Bayesian Nash equilibrium)
The strategy profile s is in *Bayesian Nash equilibrium* of mechanism $\Gamma = (S, g(\cdot))$ if $\forall i \in [n], \theta \in \Theta, s'_i \in S_i$

$$u_i^E(g(s_i(\theta_i), s_{-i}(\theta_{-i})), \theta_i) \geq u_i^E(g(s'_i(\theta_i), s_{-i}(\theta_{-i})), \theta_i)$$

where u_i^E is the expected utility over an assumed distribution of types.

In other words, a Bayesian Nash equilibrium is a Nash equilibrium with incomplete information. Another, less prominent[4], solution concept is the maximin equilibrium [Ras95]. A maximin equilibrium solution consists of strategies in which a single agent chooses the strategy that maximizes his utility given the worst possible combination of strategies selectable by other agents.

[4] It is listed here because it will be used in Theorem 4.2.

Definition 3.11 (Maximin equilibrium)
The strategy profile s is in *maximin equilibrium* of mechanism $\Gamma = (S, g(\cdot))$ if $\forall i \in [n], \theta \in \Theta, s'_i \in S_i$

$$\min_{s_{-i} \in S_{-i}} (u_i(g(s_i(\theta_i), s_{-i}(\theta_{-i})), \theta_i)) \geq \min_{s_{-i} \in S_{-i}} (u_i(g(s'_i(\theta_i), s_{-i}(\theta_{-i})), \theta_i)) \quad .$$

The concept of implementation allows us to transfer basic properties of social choice functions to mechanisms. In the remainder of this thesis, we refer to a Pareto-optimal or social-welfare-maximizing *mechanism* as a mechanism that implements a Pareto-optimal or social-welfare-maximizing *social choice function*, respectively.

It may seem almost impossible to identify implementable social choice functions because the set of possible mechanisms is extremely large. Fortunately, there is a theorem called the *revelation principle* (Theorem 3.1) which tells us that we can restrict our attention to direct-revelation mechanisms, i.e. mechanisms where the agents are asked to reveal their types in a single step. In other words, the only strategies available in a direct-revelation mechanism are to submit a claim about one's preferences.

Definition 3.12 (Direct-revelation mechanism)
A mechanism $\Gamma = (S, g(\cdot))$ is called a *direct-revelation mechanism* if $S = \Theta$.

Furthermore, the revelation principle tells use that every social choice function that can be implemented by an arbitrary mechanism in equilibrium, can also be implemented by a direct-revelation mechanism in which the equilibrium strategy is to submit one's preferences *truthfully*.

Definition 3.13 (Incentive-compatibility)
A direct-revelation mechanism Γ is *incentive-compatible* if there is an equilibrium $(s_1^*(\cdot), s_2^*(\cdot), \ldots, s_n^*(\cdot))$ in which $\forall i \in [n],\ \forall \theta_i \in \Theta_i :\ s_i^*(\theta_i) = \theta_i$.

The underlying equilibrium concept is deliberately left unspecified; it can be a dominant-strategy equilibrium or Bayesian Nash equilibrium.

If a direct-revelation mechanism is an incentive-compatible implementation of a social choice function, the outcome rule of the mechanism is equal to the social choice rule: $g(\theta) = f(\theta)$.

Theorem 3.1 (Revelation principle)
For any mechanism Γ that implements a social choice function $f(\cdot)$ in equilibrium, there is a direct-revelation mechanism implementing $f(\cdot)$ incentive-compatibly.

Proof: The proof is based on the fact that the direct-revelation mechanism can "simulate" the strategies and outcome rule of the non-direct mechanism (see e.g. [MWG95] for details). □

The first version (for dominant strategies) of this outstanding result was found by Allan Gibbard in 1973. There also is a formulation for Bayesian Nash equilibria.

If the equilibrium concept is that of dominant strategies, incentive-compatibleness is of particular importance, because it lies in each agent's interest to assist in selecting a socially preferable outcome of the social choice function independently of knowledge about other agents' preferences or their rationality. Such a mechanism is called *strategy-proof*.

Definition 3.14 (Strategy-proofness)
A direct-revelation mechanism Γ is *strategy-proof* if there is a dominant-strategy equilibrium $(s_1^*(\cdot), s_2^*(\cdot), \ldots, s_n^*(\cdot))$ in which $\forall i \in [n]$, $\forall \theta_i \in \Theta_i$: $s_i^*(\theta_i) = \theta_i$.

Concluding, the revelation principle states the following. Assume we have a social choice rule that computes a desired outcome given the agents' true preferences. If this social choice function can be implemented by a mechanism in equilibrium, i.e., there is a strategy profile that is optimal in some way, then it is possible to construct a single-shot mechanism in which revealing your preferences truthfully is the optimal strategy. The "optimality" of strategies depends on the corresponding equilibrium concept with dominant-strategies being truly optimal as submitting forged preferences will never give you more utility.

Please notice that the revelation principle does not tell us how to find such a direct-revelation mechanism. Neither does it say that the resulting mechanism is computationally efficient for the mechanism infrastructure or the agents. In fact, having to determine one's entire set of preferences can be a computational problem, for instance in combinatorial auctions (see Section 3.3.3). Additionally, truthful preference submission poses a problem of privacy. An incentive-compatible mechanism gives much more information to the mechanism infrastructure than a possibly equivalent non-direct mechanism does.

Following the previous positive result, now comes a very significant impossibility result that unfortunately renders it impossible to design strategy-proof mechanisms for unrestricted preferences.

Theorem 3.2 (Gibbard-Satterthwaite impossibility theorem)
If there are at least two agents ($n > 1$), three different outcomes ($|\mathcal{O}| > 2$), and no restrictions in agents' preferences, then a social choice function can only be implemented by a strategy-proof mechanism if and only if it is dictatorial.

Proof: The direction from right to left can easily be seen. Any dictatorial social choice function can be implemented by a strategy-proof mechanism because the outcome to be chosen directly correlates to a single agent's preferences. The other direction is not trivial (see e.g. [MWG95]) and has a great (negative) impact on mechanism design. □

3.2 The Clarke Tax Mechanism

Fortunately, non-dictatorial mechanisms are not completely impossible, because preferences may belong to a restricted domain, invalidating one of the conditions of Theorem 3.2. One of these restricted domains is that of quasilinear preferences[5].

Definition 3.15 (Quasilinearity)
Agent i's utility function $u_i(\cdot)$ is *quasilinear* if it is of the form

$$u_i(o, \theta_i) = w_i(x, \theta_i) + \pi_i \quad .$$

The outcome in this special case is of the form $o = (x, \pi_1, \pi_2, \ldots, \pi_n)$ where x is an element of a finite set X, to be called the "project choice", and π_i is a transfer term assigned to agent i. The valuation function $w_i(x, \theta_i)$ yields the utility that agent i derives from project choice x given his type θ_i.

[5] Another possibility to circumvent the negative results of Theorem 3.2 is to choose the dictator at random, thus providing ex ante fairness. This will not lead to a social-welfare-maximizing outcome and is questionable for obvious reasons.

When π_i is positive, agent i receives money. If it is negative, he has to make a payment. This kind of utility function is called “quasilinear” because it is partly linear in the transfer term π_i. These payments enable the transfer of utility among participants and can thus be used to influence an agent’s optimal strategy. As we will see in Proposition 3.3, a clever setting of payments can lead to a dominant-strategy equilibrium.

A quasilinear embodiment of utility allows us to evaluate social choice and payments separately: An important attribute of social choice functions for quasilinear preferences is *allocative efficiency* whereas the essential property of payments is *budget-balance.* $\pi(\theta)$ is an abbreviation for $(\pi_1(\theta), \pi_2(\theta), \ldots, \pi_n(\theta))$.

Definition 3.16 (Allocative efficiency)
A social choice function $f(\theta) = (x(\theta), \pi(\theta))$ is (allocatively) *efficient* if $\forall x' \in X, \theta \in \Theta$

$$\sum_{i=1}^{n} w_i(x(\theta), \theta_i) \geq \sum_{i=1}^{n} w_i(x', \theta_i) \quad .$$

If the sum of transfer terms π_i is negative, the Clarke tax mechanism yields a surplus of money. This surplus can be paid to any outside party, institution, or mechanism infrastructure as long as none of the involved agents gets it. It has to vanish from the system. For this reason, it seems to be desirable that the sum of all payments is zero (making an outside party or money burning unnecessary) or, if this is not possible, non-positive. Otherwise, if $\sum_{i=1}^{n} \pi_i > 0$, a subsidy would be needed to pay for the execution of the mechanism.

Definition 3.17 (Budget-balance)
Social choice function $f(\theta) = (x(\theta), \pi(\theta))$ is *budget-balanced* or *weakly budget-balanced* if $\forall \theta \in \Theta$

$$\sum_{i=1}^{n} \pi_i(\theta) = 0 \quad \text{or} \quad \sum_{i=1}^{n} \pi_i(\theta) \leq 0 \quad \text{, respectively.}$$

Social welfare is maximized when the social choice function is efficient and budget-balanced.

PROPOSITION 3.1 (EFFICIENCY & BUDGET-BALANCE)
A social choice function $f(\theta) = (x(\theta), \pi(\theta))$ is social-welfare-maximizing if and only if it is allocatively efficient and budget-balanced.

Proof:

$$\sum_{i=1}^{n} u_i(o,\theta) \overset{3.15}{=} \sum_{i=1}^{n} w_i(x,\theta_i) + \sum_{i=1}^{n} \pi_i(\theta) \overset{3.17}{=} \sum_{i=1}^{n} w_i(x,\theta_i) \overset{3.16}{\geq}$$

$$\sum_{i=1}^{n} w_i(x',\theta_i) \overset{3.15+3.17}{=} \sum_{i=1}^{n} u_i(o',\theta_i) \quad \forall x' \in X, o' \in \mathcal{O}$$

□

As all social-welfare-maximizing social choice functions are Pareto-optimal, efficiency and budget-balance imply Pareto-optimality.

In the case of quasilinearity, there is a unique family of direct-revelation mechanisms, the so-called *Groves mechanisms*, named after Theodore Groves, that are strategy-proof and efficient. Different members of this family of mechanisms make differing trade-offs across budget-balance and individual rationality. A mechanism is individually rational if an agent receives always more utility from participating in the mechanism than from not participating.

DEFINITION 3.18 (INDIVIDUAL RATIONALITY)
A mechanism Γ implementing social choice function $f(\cdot)$ is *individually rational* if $\forall \theta \in \Theta, i \in [n]$

$$u_i(f(\theta),\theta_i) \geq \bar{u}_i(\theta_i) \quad .$$

$\bar{u}_i(\theta_i)$ is the utility of agent i when not participating in the mechanism.

Technically, this is *ex post* individual rationality. Sometimes, it is useful to use *interim* individual rationality, which means that *expected* utility is always higher than utility when not participating.

The Clarke tax or pivotal mechanism is the most prominent member of the Groves family. It provides individual rationality[6] and simultaneously max-

[6] Individual rationality is provided if the participation of an agent does not reduce the outcome set $\mathcal{O}$ and if $\forall i \in [n], \theta_i \in \Theta_i : \; w_i\left(x^*_{-i}(\theta_{-i}), \theta_i\right) \geq 0$.

imizes the payments made by the agents to the mechanism. Weak budget-balance is achieved whenever that is possible in a strategy-proof and efficient mechanism.

Definition 3.19 (Clarke tax mechanism)
In the *Clarke tax mechanism*, the payments from the mechanism to agents are defined by

$$\pi_i(\theta) = \sum_{j \neq i} w_j(x^*(\theta), \theta_j) - \sum_{j \neq i} w_j \left(x^*_{-i} \left(\theta_{-i} \right), \theta_j \right)$$

where

$$x^*(\theta) = \arg\max_{x \in X} \sum_{i=1}^{n} w_i(x, \theta_i)$$

is the efficient project choice and

$$x^*_{-i}(\theta_{-i}) = \arg\max_{x \in X} \sum_{j=1, j \neq i}^{n} w_j(x, \theta_j)$$

is the project choice that would have been taken without agent i.

In the Clarke tax mechanism, agents that are pivotal, i.e. agents whose presence changes the outcome, internalize the externality they pose on other agents by paying a tax. Non-pivotal agents do not have to make any payments. The payment structure in the Clarke tax mechanism provides an incentive to reveal one's preferences truthfully.

Theorem 3.3
The Clarke tax mechanism is efficient, strategy-proof, individually rational, and weakly budget-balanced[a] for agents with quasilinear preferences.

[a] Weak budget-balance holds if any agent can be removed without having a negative effect on the best choice available to the remaining agents.

Proof: See e.g. [MWG95]. □

Concluding, any rational agent should participate in a Clarke tax mechanism, because this can have no negative effect on his utility (individual

rationality). Moreover, it is impossible to increase one's utility by submitting untruthful preferences (strategy-proofness) and the derived social choice maximizes total value (efficiency). Finally, the mechanism needs no subsidy to work (weak budget-balance).
Despite these impressive theoretical properties, the Clarke tax mechanism has some weaknesses. It does not maintain budget-balance which implies that it is not Pareto-optimal since the tax revenue has to vanish from the system. In settings with a large number of agents, the probability that a single agent is pivotal is not very high (there might be no pivotal agents at all). As only pivotal agents pay taxes, this problem can lose its importance in large groups or societies. Another problem of the Clarke tax mechanism is, that it is not coalition-proof. Colluding agents can coordinate their untruthful preference revelations to gain more utility (see Section 4.1.3 for an example). And finally, preferences are assumed to be quasilinear which implies that

- agents are risk-neutral (see Section 2.1),
- they value the possible outcomes independently of the preferences of other agents (see Section 2.1.1), and
- they do not consider the utility of other agents (see Section 4.3).

3.3 Extended Example: Auctions

In this section, the theoretical framework presented in the previous sections is applied to the auction problem. We consider a single seller who possesses a number of goods and n buyers that intend to buy these goods. Auctions can be seen as social choice scenarios, where outcomes represent different allocations of goods. An allocation is efficient if it maximizes the sum of reported values.

In the private-value model (see Section 2.1.1), the utility of risk-neutral agents is quasilinear; it is defined as the valuations minus the costs of the goods like in Equation 2.1 on page 17. According to the revelation principle, we can restrict our attention to sealed-bid auctions.

3.3.1 Single-Unit Auctions

In the simplest case, there is an indivisible, single good that has to be allocated to one of n agents. As a consequence, there are n outcomes. Table 3.1 shows the different "project choices" for an auction with three bidders. This setting is equivalent to a voting situation in which three possible choices are

"1 gets the good", "2 gets the good", or "3 gets the good". Each voter can only express his value for the outcome relating to him receiving the good.

	Allocation			Values			
Choice	1	2	3	1	2	3	Total value
1	×			v_1	0	0	v_1
2		×		0	v_2	0	v_2
3			×	0	0	v_3	v_3

Table 3.1: Choices in an example single-unit auction

An agent's type is equal to his valuation of the good ($\theta_i = v_i$). This valuation represents how he values outcome i, i.e. the good being awarded to him. His strategy is represented by his bid ($s_i = b_i$). The project choice can be seen as a vector $x = (x_1, x_2, \ldots, x_n)$ where $x_i = 0$ if bidder i lost the auction and $x_i = 1$ if bidder i won it. There can be only one winner ($\sum_{i=1}^{n} x_i = 1$). An outcome $o = (x, \pi_1, \pi_2, \ldots, \pi_n)$ consists of the project choice and the transfer terms for the individual bidders. The utility function takes the quasilinear form

$$u_i(o, v_i) = w_i(x_i, v_i) + \pi_i = x_i v_i + \pi_i \quad . \tag{3.1}$$

In auctions, we also need to model the seller. He can be seen as an outside party that derives no value from the good and collects all the payments. In this case, his utility is also quasilinear and takes the following form[7].

$$u_0(o) = \pi_0 = -\sum_{i=1}^{n} \pi_i \tag{3.2}$$

Given these utility functions and the reasonable assumption that we intend to achieve a socially desirable outcome, the "project choice" x can be determined straight-forward.

Proposition 3.2 (Social-welfare-maximizing auctions)
Every auction in which the bidder who submitted the highest bid is awarded the good is social-welfare-maximizing.

[7] The seller has no possibility to influence the auction outcome in this model. If possible, it would be in his interest to submit a bid by himself or set a "reservation price" (see Section 3.3.1 and 3.4).

Proof: We can neglect the payments because Equation 3.2 ensures budget-balance ($\sum_{i=0}^{n} \pi_i = 0$). All possible auction outcomes $o = (x, \pi)$ are Pareto-optimal since there is always only a single bidder whose utility is positive. The more specific measure of social welfare is appropriate here. According to Proposition 3.1, social welfare is maximized if the outcome is allocatively efficient. This clearly is only the case when the good is awarded to the bidder who values it the most (see also Figure 4.12).

$$\max_{o \in \mathcal{O}} \left(u_0(o) + \sum_{i=1}^{n} u_i(o, v_i) \right) = \max_{o \in \mathcal{O}} \left(\sum_{i=1}^{n} x_i v_i + \sum_{i=1}^{n} \pi_i - \sum_{i=1}^{n} \pi_i \right) = v_{h_1(v)}$$

$h_i(\cdot)$ is a function of the bid profile $b = (b_1, b_2, \ldots, b_n)$ or value profile $v = (v_1, v_2, \ldots, v_b)$ that yields the index of the ith highest bid or value, respectively. If two or more bidders have the highest bid/value in common, the bidder with the lowest index is chosen. □

As a consequence, an auction mechanism that implements a social-welfare-maximizing social choice function in equilibrium should award the good the bidder who *declared* the highest value for the good, namely the winner that submitted the highest bid.

$$x_i(b) = \begin{cases} 1 & \text{if } i = h_1(b) \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

Auctions are individually rational if no payment is assigned to losing bidders and if the winner's payment never exceeds his valuation ($-\pi_i \leq v_i$).

Now that we have outlined a social choice function that provides a social-welfare-maximizing outcome, we investigate how setting the payment rule affects bidding strategies.

1st-Price Sealed-Bid Auction

It seems reasonable to assign a payment to the winner that equals his bid and no payments to losing bidders.

$$\pi_i(b) = -b_i x_i(b) \tag{3.4}$$

It can easily be seen that there is no dominant strategy in this case by the "winner's curse" argument (see Section 2.2.2).

Proposition 3.3 (Impossibility of 1st-price dominant strategy) There is no dominant-strategy equilibrium in the 1st-price sealed-bid auction.

Proof: Suppose there are two bidders, A and B, and their private values are $v_b < v_a$. If B bids $b_b < v_a$, A's optimal, i.e. utility-maximizing, strategy is to bid

$$b_a^* = \arg\max_{b_a} \begin{pmatrix} v_a - b_a & \text{if } b_a > b_b \\ 0 & \text{else} \end{pmatrix} = b_b + \varepsilon$$

where ε is the smallest possible bid increment, e.g. $\varepsilon = 1$ when bids have to be integers. As this strategy completely depends on B's bid b_b, it cannot be dominant because dominant strategies are optimal no matter which strategies the opponents choose. □

There is a Bayesian Nash equilibrium when it is general knowledge that valuations are drawn from a uniform distribution.

Theorem 3.4 (1^{st}-price sealed-bid Bayesian Nash equilibrium)
In a 1^{st}-price sealed-bid auction with n risk-neutral bidders whose valuations are independently and uniformly distributed in the interval $[0, \bar{v}]$, bidding

$$b_i = \frac{n-1}{n} v_i + \varepsilon$$

is in Bayesian Nash equilibrium.

Proof: Let us consider the strategy for bidder i. If bidder i does not have the highest value, we do not need to model his strategy, because he will lose anyway. He can maximize his utility by bidding slightly above his expectation of the second highest value conditional on his bid being the highest. The probability that bidder j's value v_j, which is uniformly distributed in the interval $[0, v_i]$, equals an arbitrary value v lower than v_i is $\frac{1}{v_i}$. The probability that v_j is less or equal than v is $\frac{v}{v_i}$. Thus, the probability that v is the second highest value is

$$\left(\frac{1}{v_i}\right)\left(\frac{v}{v_i}\right)^{n-2} \quad . \tag{*}$$

The probability that one of the $n-1$ other bidders has the second highest value v is $n-1$ times expression (*). The expected value of v is

$$E(v) = \int_0^{v_i} v(n-1)\left(\frac{1}{v_i}\right)\left(\frac{v}{v_i}\right)^{n-2} dv = \frac{n-1}{v_i^{n-1}} \int_0^{v_i} v^{n-1} dv = \frac{n-1}{n} v_i \quad .$$

Concluding, bidder i should bid $\frac{n-1}{n} v_i + \varepsilon$ where ε is the smallest possible bid difference. □

Vickrey Auction

Since we have quasilinear preferences, we can apply the Clarke tax mechanism (Definition 3.19) to achieve strategy-proofness in the good-allocation scenario that we are considering. The transfer term π_i is defined as the summed up values (excluding i's) of the project choice minus the summed up values (again, excluding i's) of the choice that would have been taken without bidder i. In single-unit auctions, there is just one winner which simplifies this definition. If bidder i does not win the auction and thus is not pivotal, he pays nothing because the minuend and the subtrahend of the transfer term are equal. The winner of an auction "receives" the others' valuation of him winning the auction, i.e. zero, minus the total value of the choice that would have been taken without him, i.e. the second highest bid.

$$\pi_i(b_1, b_2, \ldots, b_n) = -b_{h_2(b)} x_i(b_1, b_2, \ldots, b_n) \tag{3.5}$$

According to Proposition 3.3, the mechanism induced by this payment rule is strategy-proof. This has also been proven by case analysis in Theorem 2.1. It follows that the Vickrey auction even leads to an efficient outcome when values are drawn from *different* probability distributions (asymmetric bidders). This is not the case in 1^{st}-price auctions which can be demonstrated by the following example.

Suppose there are two bidders, A and B, whose valuations are uniformly drawn from intervals $[\underline{v}_a, \overline{v}_a]$ and $[\underline{v}_b, \overline{v}_b]$, respectively. Assume that $\overline{v}_a < \underline{v}_b$. B's optimal strategy is to bid slightly more than A's expected value, namely $\frac{\overline{v}_a - \underline{v}_a}{2} + \varepsilon$. A can bid whatever he wants as long as his bid is lower than his private value. This can lead to A winning the auction inefficiently. The fragility of Bayesian Nash equilibria, in contrast to robust dominant-strategy equilibria, is emphasized by this example.

Seller's Revenue

So far, we have concentrated on allocative efficiency rather than maximization of the seller's expected revenue. It may seem that the Vickrey auction produces less revenue than the 1^{st}-price sealed-bid auction because the selling price is the second highest bid instead of the highest[8]. On the other hand, rational bidders have to bid less than their valuations in 1^{st}-price auctions (see Theorem 3.4). It surprisingly turns out that both auctions generate exactly the same expected revenue[9] in the case of risk-neutral bidders in the

[8] This *false* assertion was even included in early literature on auctions, e.g. [Cas67].

[9] When values are drawn from a uniform distribution in the interval $[0, \overline{v}]$, the expected price is $\frac{n-1}{n+1}\overline{v}$, independently of the applied auction type.

private-value model. This is one of the most celebrated theorems of auction theory.

Theorem 3.5 (Revenue Equivalence)
All four major auction protocols (English, Dutch, 1$^{\text{st}}$-price sealed-bid, Vickrey) lead to the same expected revenue if agents are risk-neutral and have private, independent values drawn from a common distribution[a].

[a] This can be generalized to any auction in which the highest bidder is awarded the item and thus also holds for other, less common auction types like the "all-pay" auction in which each bidder has to pay what he bid.

Proof: See e.g. [Wol96, Kle99]. □

In other words, all reasonable auction types result in the same outcome *on average*. This does not imply that all auction types are the same. A dominant-strategy equilibrium like in the Vickrey auction is still much more desirable than a 1$^{\text{st}}$-price auction's Nash equilibrium.

Revenue equivalence breaks down when removing any of the conditions stated in Theorem 3.5. When bidders are risk-averse, 1$^{\text{st}}$-price auctions yield more revenue than 2$^{\text{nd}}$-price auctions. The opposite holds, when the seller is risk-averse as the variance of the selling price is higher in 2$^{\text{nd}}$-price auctions. If there are more than two bidders in the common-value or correlated-value model, i.e. valuations are *not* independent, the English auction generates the highest revenue, followed by the Vickrey auction and then the 1$^{\text{st}}$-price auction (Table 3.2). This is not surprising because the open-cry character of English auctions tends to increase bidders' valuations.

risk-averse bidders	risk-averse seller	non-private values
1$^{\text{st}}$-price, Dutch	Vickrey, English	English
Vickrey, English	1$^{\text{st}}$-price, Dutch	Vickrey
		1$^{\text{st}}$-price, Dutch

Revenue decreases from top to bottom.

Table 3.2: Seller's revenue

Roger Myerson [Mye81] investigated auctions that maximize the seller's expected revenue. When bidders are symmetric, the construction of such an auction can be reduced to finding an appropriate *reservation price*, i.e. a minimum bid set by the seller, in a Vickrey auction. If private values are uniformly distributed in the interval $[0, \bar{v}]$, the optimal reservation price is $\frac{\bar{v}+v_0}{2}$ where v_0 denotes the private value of the seller[10]. As a consequence, even when we assume that the seller derives *no* value from the good (like we did in Equation 3.2 on page 38), he should set a positive reservation price. Setting a reservation price that is higher than the seller's valuation can obviously lead to inefficient allocations and thus invalidates the conditions of Theorem 3.5 (the good is not always awarded to the agent who values it most). As we will see in Section 3.4, there is no social-welfare-maximizing and strategy-proof allocation mechanism when the seller is allowed to set a reservation price (Theorem 3.6).

3.3.2 Multi-Unit Auctions

Another convenient auction protocol, due to Vickrey [Vic61] and later rediscovered in [WWW98], can be constructed by applying the Clarke tax mechanism to the case when there are M units of the same item for sale and each bidder has a demand of exactly one unit. The number of choices is $\binom{n}{M}$ and a bid values the $\binom{n-1}{M-1}$ choices in which the corresponding bidder is awarded a unit (see Table 3.3 for an example with three bidders and two units).

	Allocation			Values			
Choice	1	2	3	1	2	3	Total value
1	×	×		v_1	v_2	0	$v_1 + v_2$
2	×		×	v_1	0	v_3	$v_1 + v_3$
3		×	×	0	v_2	v_3	$v_2 + v_3$

Table 3.3: Choices in an example multi-unit auction

There are M winners ($\sum_{i=1}^{n} x_i = M$) that each get one unit of the good. These are the bidders that submitted the M highest bids, because the sum of their bids is maximal which leads to a social-welfare-maximizing outcome.

$$x_i(b) = \begin{cases} 1 & \text{if } i \in \bigcup_{j=1}^{M} \{h_j(b)\} \\ 0 & \text{otherwise} \end{cases} \tag{3.6}$$

[10] Surprisingly, the optimal setting of the reservation price does *not* depend on the number of bidders n.

Like in the previous section, the Clarke tax mechanism is applied to achieve strategy-proofness in this allocation scenario. According to Definition 3.19, a winning bidder receives the summed up values of the $M-1$ other winners minus the summed values of the allocation that would have been chosen without his participation, i.e. the allocation in which he is replaced by the $(M+1)$st highest bidder (see Figure 3.2). This yields

$$\pi_i(b) = \left(\sum_{j=1, h_j(b) \neq i}^{M} b_{h_j(b)} - \sum_{j=1, h_j(b) \neq i}^{M+1} b_{h_j(b)} \right) x_i(b) = -b_{h_{M+1}(b)} x_i(b) \quad . \quad (3.7)$$

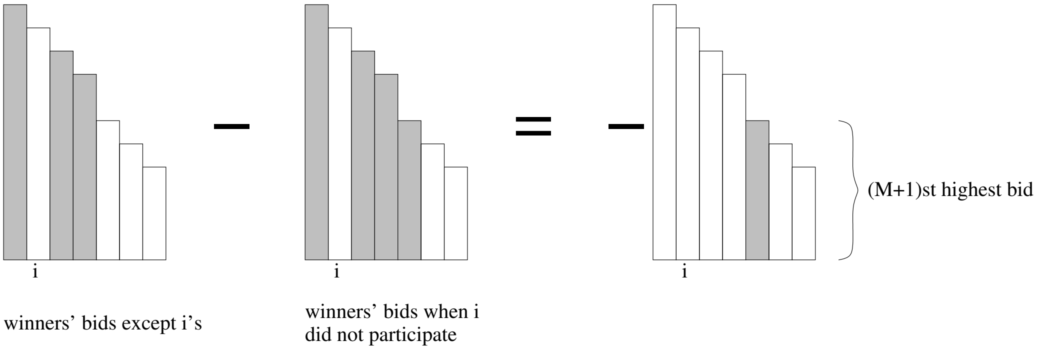


Figure 3.2: Payment assigned to winner i in a multi-unit auction

All winners have to pay the amount of the $(M+1)$st highest bid. For this reason, the resulting auction is sometimes called "$(M+1)$st-price auction" or "uniform-price" auction. The Vickrey auction is a special case for the selling of a single unit ($M=1$).

It might seem like this mechanism gives the seller less revenue than the repeated selling of M goods in single-unit Vickrey auctions. In the first auction, the highest bidder is awarded a unit for the second highest bid and quits. The second highest bidder becomes the winner in the next auction for the third highest price and so on. However, the mechanism *consisting of* successive Vickrey auctions is not strategy-proof. In the description above we assumed that "submitting your private value in consecutive auctions until you win" is a dominant strategy. Given that all bidders participate like described, it is certainly wiser to just take part in the final auction and pay less. As a matter of fact, there is no dominant strategy for the described mechanism.

If we advance to the case when bidders are allowed to bid on specific amounts of units, we already have a special case of the most general auction mechanism: the combinatorial auction.

3.3.3 Combinatorial Auctions

In a combinatorial auction, m different items are sold in a single auction. Bidders can express their willingness to pay for sets of goods. This is desirable when values of items are non-additive, i.e., they are either complementary (a bundle of items is worth more than the sum of its parts) or substitutable (a bundle of items is worth less than the sum of its parts). Figure 3.4 illustrates a combinatorial auction with two goods ($\times$ and $\circ$).

	Allocation			Values			
Choice	1	2	3	1	2	3	Total value
1	$\times\circ$			$v_1^{\times\circ}$	0	0	$v_1^{\times\circ}$
2	$\circ$	$\times$		v_1°	$v_2^{\times}$	0	$v_1^{\circ} + v_2^{\times}$
3	$\circ$		$\times$	v_1°	0	$v_3^{\times}$	$v_1^{\circ} + v_3^{\times}$
4	$\times$	$\circ$		$v_1^{\times}$	v_2°	0	$v_1^{\times} + v_2^{\circ}$
5		$\times\circ$		0	$v_2^{\times\circ}$	0	$v_2^{\times\circ}$
6		$\circ$	$\times$	0	v_2°	$v_3^{\times}$	$v_2^{\circ} + v_3^{\times}$
7	$\times$		$\circ$	$v_1^{\times}$	0	v_3°	$v_1^{\times} + v_3^{\circ}$
8		$\times$	$\circ$	0	$v_2^{\times}$	v_3°	$v_2^{\times} + v_3^{\circ}$
9			$\times\circ$	0	0	$v_3^{\times\circ}$	$v_3^{\times\circ}$

Table 3.4: Choices in an example combinatorial auction

In a way, a combinatorial auction is the mother of all auctions and can be used to allocate any kind of resources among agents. In fact, they are relevant for scheduling, logistics, and network computation. The Clarke tax mechanism for this particular problem is sometimes called *generalized Vickrey auction* (GVA). Although the Clarke tax mechanism guarantees several important properties (strategy-proofness, efficiency, individual rationality, and weak budget-balance), it poses some problems in combinatorial auctions due to its "direct revelation nature".

The number of possible bundles is $2^m - 1$. In the example auction of Figure 3.4 each bidder must submit three ($= 2^2 - 1$) bids: one for each item and one for the bundle consisting of both items. This leads to the following computational problems.

- Each bidder needs to compute his value for exponentially many bundles.
- Finding a combination of bids that is allocatively efficient is an $\mathcal{NP}$-complete optimization problem (*winner determination* is an instance

of the *weighted set packing problem* [San99]).

- In order to compute the Clarke tax payments, several more $\mathcal{NP}$-complete problems have to be solved.

Be aware that despite the $\mathcal{NP}$-hardness of the general combinatorial auction winner determination problem, combinatorial auctions are tractable in many special cases. For example, when allowing only bundles that contain at most two items, winner determination *is* tractable. We will now give two further important examples of tractable combinatorial auctions [Ten00].

General multi-unit auction A *constant* number of goods, of which there are arbitrarily many, indistinguishable units, is to be sold. Bidders submit how they value combinations of any given number of each good.

Linear goods auction An ordered list of items is sold. Bidders can submit bids for blocks of items (without "holes"). This type of auction can be useful for the selling of one-dimensional arrays like radio spectrums, time slots, or parts of a seashore.

Besides the computational aspects of determining the winners and the appropriate prices, the following problems arise[11].

- The submission of exponentially many bids of which only a fraction is needed to compute the auction outcome wastes communication resources.

- The complete revelation of preferences is unnecessary and raises privacy questions.

Due to these problems, there has recently been a large body of research on non-direct (iterative) mechanisms that lead to the GVA outcome (e.g. [Par01, CS02a]). On the other hand, there have been recent advances in efficient winner determination algorithms (e.g. [SSGL01]). Generally, combinatorial auctions are currently the most active field of auction theory [Kle99].

3.4 Further Important Results

When further generalizing auctions to *exchanges*[12] (markets with sellers and buyers), it becomes apparent that the "trick" we used in Equation 3.2 on

[11] These problems are existent in single- and multi-unit auctions as well, but they have much more relevance in combinatorial auctions.

[12] So-called *double auctions*, stock markets, and many other scenarios belong to this category.

page 38 to achieve budget-balance does not work anymore because sellers assign values to goods. Furthermore, sellers are able to actively participate in exchange mechanisms. The following theorem (that is accounted to Hurwicz in [Par01]) declares the impossibility of a social-welfare-maximizing and strategy-proof exchange mechanism with quasilinear preferences.

Theorem 3.6 (Hurwicz impossibility theorem)
There is no strategy-proof mechanism that implements an efficient, budget-balanced social choice function for simple exchange economies with quasilinear preferences.

Proof: See e.g. [Par01]. □

As if this result was not negative enough, the impossibility of a *Bayesian Nash* incentive-compatible exchange mechanism under quite reasonable assumptions has been proven as well.

Theorem 3.7 (Myerson-Satterthwaite impossibility theorem)
There is no Bayesian Nash incentive-compatible mechanism that implements an efficient, budget-balanced, and individual-rational social choice function for simple exchange economies, even with quasilinear preferences.

Proof: See e.g. [MWG95]. □

To give an elementary example, let us consider that agent A wants to sell a good to agent B. Both have private valuations of the good (drawn from a common distribution) and need a Bayesian Nash incentive-compatible mechanism that determines whether the good is sold and, if so, compute the selling price. Other reasonable conditions are:

- The good is only sold if B values it higher than A (efficiency).
- Money is transferred only between A and B (budget-balance).
- A's and B's expected utility for participating in the mechanism is at least as high as when not participating (individual rationality).

According to Theorem 3.7, there is no mechanism that satisfies all of those (modest) desiderata. As a consequence, the best we can achieve is to provide two of the three mentioned properties in an exchange mechanism. The

Clarke tax mechanism, for example, achieves efficiency and individual rationality, but lacks budget-balance. A similar mechanism, the so-called dAGVA mechanism provides efficiency and budget-balance, but lacks individual rationality. Furthermore, in contrast to the Clarke tax mechanism, the dAGVA mechanism is only Bayesian Nash incentive-compatible.

Chapter 4

Fraud and Deception

According to the FBI [FBI02], auction fraud is the most stated offense at the Internet Fraud Complaint Center (IFCC) (see Figure 4.1). In 2001, the *average* loss per consumer complaint in the case of auction fraud was $ 395. Most of these frauds involve non-delivery of goods or money, false statements about goods, or manipulation of reputation systems (like `ebay`'s) [Ben01]. Escrow services (e.g. by the auctioneer) and insurances can prevent some of those problems.

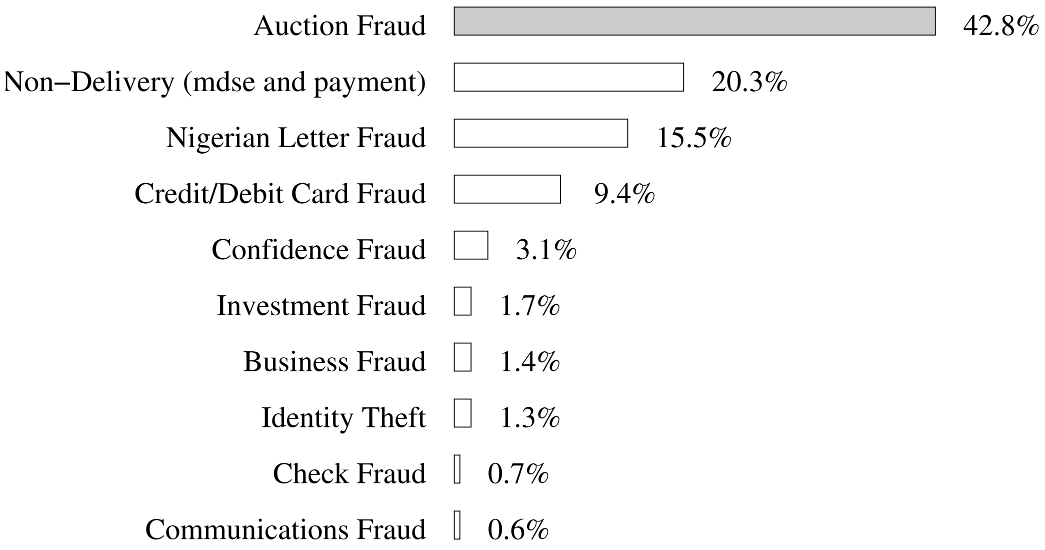


Figure 4.1: Top ten IFCC complaints (2001) [FBI02]

On the other hand, there are ways of "cheating" in an auction that are not necessarily illegal. We make a distinction between *fraud*, i.e. antinomian cheating, and *deception*, i.e. undesirable behaviour that can be blamed to poor auction design. Here is an example of the latter case (taken from [McM94]). In 1993, the Australian government auctioned off two licenses for satellite-television services. They used 1$^{\text{st}}$-price sealed-bid auctions. In one of those auctions, an unknown company called UCOM won a license for A$ 212 million (beating a consortium of Rupert Murdoch, Kerry Packer, and Telecom Australia among others). All the bids were revealed and the government hailed the auction outcome as opening up “a whole new era”, bringing new firms into the closed shop of Australian television industry. It turned out that UCOM decided to default on their bid, which resulted in the second best bidder being awarded the corresponding license. But UCOM submitted the second highest bid as well! Moreover, they submitted the third highest, fourth highest and so forth. After defaulting on several more bids, they finally paid A$ 117 million for the license (saving A$ 95 million). Shortly afterwards, they sold the license, earning A$ 21 million. The poor auction rules (free defaulting) resulted in at least a year’s delay into pay TV in Australia.

In this chapter, we will focus on deception, i.e. undesirable strategic bidding that is not covered by the theoretic models or that cannot be prevented in real auctions for practical reasons. After mentioning some classic phenomena like bidder collusion, “shills”, or “sniping”, we analyze antisocial bidding in detail. Antisocial agents are defined as agents who, in order to outperform their competitors, have an incentive to reduce their competitors’ profit. The chapter closes with some brief experimental results that show the (negative) impact antisocial agents can have in auctions and a theorem that basically states that it is not possible to construct an auction mechanism that provides elementary properties in the presence of antisocial agents.

4.1 Deceptive Bidding

In the drastic example given above, the deceptive behaviour of UCOM could have been prevented by better designed auction rules. In this section, we will present forms of deception that can not be easily prohibited.

4.1.1 Shills

Shills are bids that are placed by the seller under fictious names or by recruiting other people to bid in order to raise the price of a good. Shills can only occur in open-cry ascending (English) auctions. Furthermore, shills only make sense if the good to be sold somehow relates to the common-value oder correlated-value model, i.e., bidders valuations depend on another. Placing a shill bid then makes other bidders overestimate the good's value. Although shills are illegal in many auctions, they are ubiquitous because it is almost impossible to detect them.

Electronic auction houses often charge fees that are defined as a fraction of the selling price. Placing shills in such auctions has to be evaluated carefully because winning the auction accidently (by placing a shill bid too high) will result in deficits (and the seller inefficiently keeping the item). Most auctions allow the seller to influence the auction outcome in a limited way by setting a reservation price (see Section 3.3.1). However, he is not capable of changing that value once the auction started.

Sellers placing shills and antisocial bidders (to be defined in Section 4.3) share the same goal, i.e. to increase the selling price. Yet, they have different motivations: The seller simply wants to increase his revenue whereas the antisocial bidder intends to weaken his competitors.

4.1.2 Sniping

As mentioned in Section 2.3.1, most Internet auction houses (like `ebay`) use English auctions that last for a fixed period of time. It turned out that most users – despite the possibility of using "bidding agents" as described in Section 2.3.1 – bid at the very end of an auction, mostly in the last seconds, giving other bidders no time to react. One explanation for this behaviour is the intention to keep one's valuation private. This is of particular relevance when the private-value model does not hold as any public bid might make other bidders increase their valuations. Actually, the only difference between using the bidding agent and sniping is the information revealed by the bidding agent. There are dozens of commercial programs and websites (like `esnipe`) that bid in the very last seconds of an auction on the behalf of the user. `ebay` has filed several lawsuits against the makers of such sniping agents. It is interesting to observe that, like bidding agents, sniping converts the English auction into a Vickrey auction.

> "For example, if sniping and sniping agents become even more widespread on eBay than they are today, eBay would be gradually transformed into a sealed bid second price auction. If a large part of the bidding action were taking place on third-party sites like esnipe, eBay would face a number of choices. One would be to recapture the sniping market by offering a sniping option on eBay itself. (Under this option, last minute bids submitted in advance directly to eBay could all be counted at the same time, immediately after the auction close, thus giving bidders certainty

> both that their bids would be successfully transmitted, and that there would be no time for other bidders to react.) Of course, if all bidders used this option, the auction would be precisely a sealed bid auction. eBay, and sellers who list items for sale on eBay, might prefer not to encourage this development (for example if they believe that bidders are likely to bid more in auctions in which they can form some estimates of how much other bidders value the item for sale)". [OR02]

It is a well-known fact that open-cry auctions generate more revenue than sealed-bid auctions in scenarios that do not belong to the private-value model (see Table 3.2). However, generating more revenue certainly does not lie in *bidders'* interests. For this reason, if bidders have the possibility to transform an auction into a sealed-bid auction (like in `ebay`'s auctions), they will do so. In the auctions conducted at `amazon` for example, extra time is added whenever a late bid is submitted. As a consequence, sniping is almost non-existent in `amazon` auctions. [RO02, OR02] give reasons for late bidding that occurs even in the private-value model.

4.1.3 Bidder Collusion

A fundamental deficiency that affects all auction types is bidder collusion (sometimes called "bidding rings"). Auction mechanisms are build upon the principles of competition and asymmetry of information. If this asymmetry is removed, bidders can manipulate the action outcome and increase their utility. For example, in a 1$^{\text{st}}$-price sealed-bid auction, bidders can form a coalition and coordinate their bids by having all bidders but one bid zero. The remaining bidder bids an arbitrarily small amount. After having won the auction, he can share some of the savings he made with cooperating bidders. However, the other bidders have a substantial incentive to cheat on the agreement by not bidding zero as they have the possibility to receive the good at a bargain price. The situation is different in 2$^{\text{nd}}$-price auctions. In this case, the designated winning bidder does not need to adjust his bid downwards. He can bid his private value while all other bidders bid zero (or any other small amount). There is no incentive for any of the colluding bidders to break the agreement because it is impossible for them to obtain the good without a loss. This is why it is said that collusions are "self-enforcing" in 2$^{\text{nd}}$-price auctions [Rob85].

A collusion of bidders is faced with the problem of determining their highest bidder, preferably using a strategy-proof mechanism in order to avoid strategic behaviour. In [GM87], this problem is solved by running a pre-auction in which it is every colluder's optimal strategy to submit his valuation truthfully. The highest bidder is designated as the winner. If this agent also wins the main auction, he has to pay the difference between the selling price and the second highest bid of the pre-auction to the remaining collusion members. Efficient collusion schemes for 1$^{\text{st}}$-price and 2$^{\text{nd}}$-price sealed-bid auctions have been identified in [MM92] and [GM87, MZ91], respectively. The former even deals with the case when side payments are impossible. [LBST00] contains a more general approach that covers arbitrary auction mechanisms and parallel executions of these mechanisms.

According to the US Justice Department's antitrust chief, bidder collusion by highway contractors increased the cost of building roads by ten percent or more [MM92]. [McM91] explains a widespread form of bidder collusion in Japan's public-works contracting.

In combinatorial auctions, a new subform of collusion is particularly interesting. Colluding agents can increase their profit by introducing new bidders to the auction. In fact, it is sufficient to submit several bids under different names to manipulate the auction outcome. Clearly, in virtual marketplaces this form of deception is almost undetectable since identifying the true origins of bids is extremely hard. The general behaviour is called "false-name bidding" and has been extensively studied by Yokoo et al. They were able to prove that there is no "false-name-proof" combinatorial auction mechanism that satisfies allocative efficiency [YSM03].

4.2 Insincere Auctioneer

The problem of an insincere auctioneer obviously belongs to the fraud category and is important when bidders pay prices that are different than their bids, which implies that it is a particular problem in Vickrey auctions. Consider a Vickrey auction with three sealed bids: 10, 20, and 30. The auctioneer might tell the winning bidder that the second highest bid was 29. In a sealed-bid auction, this bidder has virtually no means to verify the correctness of the auctioneer. Even when the auctioneer is forced to prove that the second highest bid is indeed a submitted bid by showing a digital signature of the corresponding bidder, he can cheat with the help of a fellow bidder who signs a bid *after* the auctioneer opened the other bids. We will discuss solution concepts to this problem in Chapter 5.

4.3 Antisocial Agents

In most economic models as well as multiagent applications it is assumed that the objective of an agent is to maximize his absolute profit without caring for the profits made by other agents. However, in many real-world applications, it is more realistic to assume that some agents try to gain as much money (or utility) as possible *relative* to others (their competitors). In other words, in many scenarios it is wise to take into consideration the availability of "antisocial agents," that is, agents who are willing to reduce the profit of competitors.

4.3.1 Suffering when Others Win

When having another look at Table 3.1 on page 38, it stands out that bidders are indifferent to choices, in which they do not get the good. When supposing that bidders obtain negative utility if another bidders wins the auction, the table can be modified to look like Table 4.1. As bidders are assumed to be symmetric, bidder i cannot have preferences on who gets the item, but he is able to assign negative value a_i to the cases when *someone else* does. We are now assuming that the mechanism allows to express one's preferences on every possible project choice. The utility function has the following quasilinear form.

$$u_i(o, v_i, a_i) = x_i v_i - (1 - x_i) a_i + \pi_i \tag{4.1}$$

	Allocation			Values			
Choice	1	2	3	1	2	3	Total value
1	×			v_1	$-a_2$	$-a_3$	$v_1 - a_2 - a_3$
2		×		$-a_1$	v_2	$-a_3$	$-a_1 + v_2 - a_3$
3			×	$-a_1$	$-a_2$	v_3	$-a_1 - a_2 + v_3$

Table 4.1: Choices in an example single-unit auction with "anti-bids"

The Clarke tax mechanism yields $\pi_i(b) = \sum_{j=1}^{n} a_j - \sum_{j=1}^{n} a_j + a_{h_2(b)} + b_{h_2(b)} = a_{h_2(b)} + b_{h_2(b)}$ as a payment for the auction winner. This outcome can obviously be reached more easily if each bidder bids $b_i = v_i + a_i$ in a regular Vickrey auction. However, this is not possible when bidders are able to distinguish between fellow bidders, i.e., they have different (negative) values for each other bidder winning the auction (see [JMS96]).

So far we have merely taken into account that a bidder derives negative utility if someone else *wins* an auction. In the next section, we will treat bidders that consider their rivals' *profit*.

4.3.2 Suffering from Others' Utility

As a starting point for the formalization of antisocial utility, it appears to be reasonable to assume that an antisocial agent wants to maximize the difference between his profit and the gain of his competitors; this means that the own profit on the one hand and the other agents' losses on the other hand are considered to be of equal importance from the point of view of this antisocial agent. In a two-player scenario, this view captures the antisocial agent's intention to be better than his rival. To achieve a higher degree of flexibility in describing and analyzing antisocial agents, it is useful to think of different degrees of antisociality like "aggressive antisociality" (where it is an agent's objective to harm competitors at any cost) and "moderate antisociality" (where an agent puts somewhat more emphasis on his own profit rather than the loss of other agents).

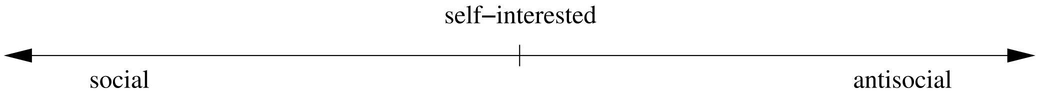


Figure 4.2: Simplified scale of social behaviour

These considerations lead to the formal specification of an antisocial agent (or an agent's antisocial attitude) as an agent who tries to maximize the weighted difference of his own profit and the profit of his competitors. We used the term "profit" instead of "utility" in the description above to motivate that antisocial behaviour can indeed be rational. Formally, an antisocial agent can be defined as follows.

Definition 4.1 (Antisocial agent)
An *antisocial agent* intends to maximize the utility given by the equation

$$u_i^A(o,\theta,d_i) = (1-d_i)u_i(o,\theta_i) - d_i \sum_{j=1,j\neq i}^{n} u_j(o,\theta_j) \quad ,$$

where $d_i \in [0,1]$ is a parameter called *derogation rate*.

The derogation rate is crucial because it formally captures, and allows to modify, an agent's degree of antisocial behavior. It is obvious that this formula covers "regular" agents by setting $d_i = 0$. We say that agent i is antisocial if $d_i > 0$ (see Figure 4.2). If d_i is greater than 0.5, hurting others has greater priority than helping yourself. A purely destructive agent is defined by $d = 1$. We say an agent is *balanced antisocial* if $d_i = 0.5$, e.g., his own utility and the utility of his competitors are of equal importance. Please note that the above definition assumes that an antisocial agent knows the types and utility functions of other agents. The utility functions are unproblematic because we will only consider quasilinear utility in simple auctions. We will dicuss methods to estimate and learn types, i.e. private values, later in this chapter. Other possible embodiments of antisocial utility functions include the *average* non-negative utility of competitors

$$u_i^A(o, \theta_i, d_i) = (1 - d_i)u_i(o, \theta_i) - d_i \frac{\sum_{j=1, j\neq i}^{n} u_j(o, \theta_j)}{|\{j \mid u_j(o, \theta_j) > 0\}|} \tag{4.2}$$

or the *maximum* utility of any rival.

$$u_i^A(o, \theta_i, d_i) = (1 - d_i)u_i(o, \theta_i) - d_i \max\{u_j(o, \theta_j)\}_{j=1, j\neq i}^{n} \tag{4.3}$$

However, we will stick with Definition 4.1 because we only consider single-unit auctions, which means that all utilities except the winner's are zero.

4.3.3 Antisociality and Vickrey Auctions

In this Section, the implications of antisocial utility in Vickrey auctions are theoretically investigated. In order to simplify such analysis, we first investigate a setting with *complete information*, i.e., all types are publicly known.

Like in Section 3.3, types θ_i are private values v_i of a good in the following. Utility u_i is defined by the quasilinear function in Equation 3.1 and $v = (v_1, v_2, \ldots, v_n)$.

$$u_i^A(o, v, d_i) = (1 - d_i)(x_i v_i - \pi_i) - d_i \sum_{j \neq i} (x_j v_j - \pi_j) \tag{4.4}$$

We will refer to an agent's regular, non-antisocial utility $x_i v_i - \pi_i$ as his "profit" in the following. Let us now consider an auction with two bidders, A and B. Similar to Equation 2.1 on page 17, the antisocial utility of agent A can be written as a function of the bids b_a and b_b.

$$u_a^A(b_a, b_b, v_a, v_b, d_a) = \begin{cases} (1 - d_a)(v_a - b_b) & \text{if } b_a \geq b_b \\ -d_a(v_b - b_a) & \text{if } b_a \leq b_b \end{cases} \tag{4.5}$$

The first striking consequence of the new definition of utility is that the Vickrey auction's dominant-strategy equilibrium breaks down for antisocial agents.

Proposition 4.1 (Antisocial non-strategy-proofness)
Bidding one's private value is *not* a dominant strategy for antisocial agents in Vickrey auctions.

Proof: It suffices to construct a single case in which deviating from the dominant strategy can lead to higher profit. Consider case vi) of the proof of Theorem 2.1 on page 18. We now assume that agent A's derogation rate d_a is greater than 0 and that $v_b - v_a > \varepsilon$. A is not able to effectively win the auction, but the price agent B pays completely depends on A's bid. So, if A carefully adjusts his bid upwards, he is capable of reducing B's utility. Supposing that A knows B's private value v_b, his optimal strategy would be to bid $v_b - \varepsilon$ (see Figure 4.3), which reduces B's profit to the absolute minimum of ε.

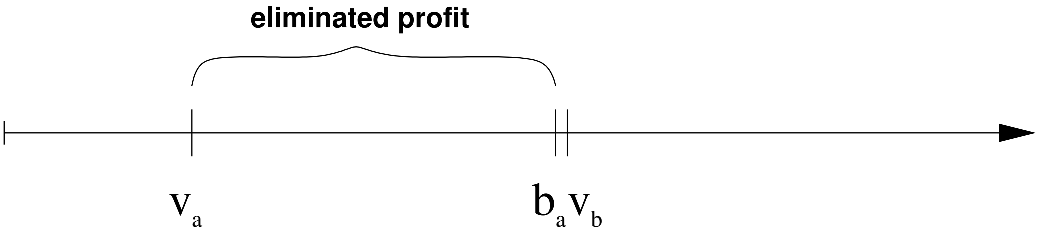


Figure 4.3: A reduces B's profit to a minimum

We can assume $b_b = v_b$ without loss of generality, because a dominant strategy should yield the highest utility for *all* possible strategies by other agents. When applying the "dominant strategy" ($b_a = v_a$), A's antisocial utility is

$$u_a^A(v_a, v_b, v_a, v_b, d_a) = -d_a(v_b - v_a) \quad .$$

However, bidding $b_a = v_b - \varepsilon$ yields more utility,

$$u_a^A(v_b - \varepsilon, v_b, v_a, v_b, d_a) = -d_a\varepsilon \quad ,$$

if $v_b - v_a > \varepsilon$. □

Please note that non-antisocial agents are best off bidding their private values, even when antisocial agents are present (otherwise the strategy would not be dominant). It has merely been shown that the dominant strategy is not optimal for antisocial agents. This raises the question whether there is a dominant-strategy equilibrium for antisocial agents at all. The answer is no if agents are not purely destructive.

Proposition 4.2 (Impossibility of antisocial dominant strategy) There only is a dominant Vickrey auction strategy for antisocial agents when their derogation rate d is 1.

Proof: Clearly, if $d = 1$, bidding the highest possible value will always yield zero utility which is the highest utility a purely antisocial agent can achieve. To prove the rest of the statement, let us again consider the case of the previous proof and assume, without loss of generality, $v_b - v_a > \frac{2d_a+1}{1-d_a}$.
If B bids v_b, A's optimal strategy is to bid $v_b - \varepsilon$ resulting in $-d_a\varepsilon$ utility. If B bids $v_b - 2\varepsilon$ however, this strategy is sub-optimal because it only yields the negative utility $(1-d_a)(v_a - v_b + \varepsilon)$ which is lower than $u_a^A(v_b - 3\varepsilon, v_b - 2\varepsilon, v_a, v_b, d_a) = -d_a(3\varepsilon)$.

$$\begin{aligned}
& (1-d_a)(v_a - v_b + \varepsilon) > -d_a(3\varepsilon) \\
\Leftrightarrow\quad & (v_b - v_a)(d_a - 1) + \varepsilon - d_a\varepsilon > -3d_a\varepsilon \\
\Leftrightarrow\quad & v_b - v_a > \frac{2d_a+1}{1-d_a}
\end{aligned}$$

□

It is self-evident to seek weaker equilibria that might fit this scenario. In the previous two propositions, the other bidders' utility functions and in particular their derogation rates were irrelevant, because we only dealt with dominant-strategy equilibria. In the proof of Proposition 4.2, agent B bid slightly less than his private value v_b, rejecting a possible gain of ε and making A lose $v_b - v_a - 2\varepsilon$. This behaviour makes perfect sense if B is antisocial as well.

As a consequence, if A's derogation rate d_a is 0.5, A should only bid $v_a + \frac{v_b - v_a}{2} - \varepsilon$ to be safe from being underbid by B (see Figure 4.4). If B still undercuts A's bid, he waives more money than A loses. If $d_b = 0.5$ as well, B's best strategy is to bid $v_b - \frac{v_b - v_a}{2}$.

Returning to the case of more than two bidders, the following bidding strategy seems to be "safe" for an antisocial agent i. We still assume the (unrealistic) model of complete information ($v_{h_1(b)}$ is the highest private value,

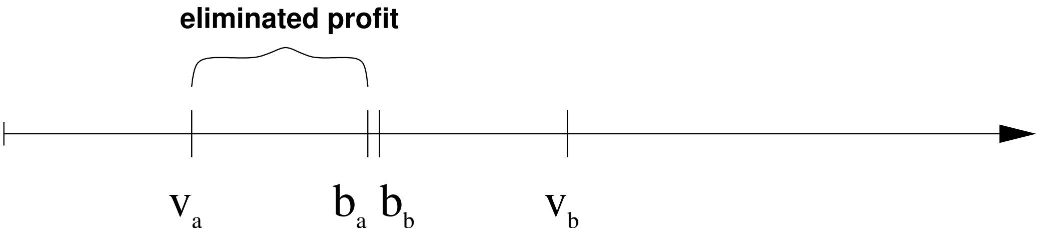


Figure 4.4: Careful antisocial bidding

$v_{h_2(b)}$ the second highest)[1].

$$b_i = \begin{cases} v_i - d_i(v_i - v_{h_2(v)}) & \text{if } i = h_1(v) \\ v_i + d_i(v_{h_1(v)} - v_i) & \text{else} \end{cases} \tag{4.6}$$

Theorem 4.1 (Balanced antisocial Nash equilibrium)
In Vickrey auctions with balanced antisocial bidders ($\forall i \in [n] : \quad d_i = 0.5$), the strategy defined by Equation 4.6 is in *Nash equilibrium.*

Proof: According to Definition 3.9, the assumption states that under the supposition that all agents apply this strategy, there is no reason for a single agent to deviate from it. We consider the utility of agent A. It suffices to take only one opposing agent B into account as A does not differentiate between the individual bidders and the Vickrey auction has a sole victor.
According to Equation 4.6, agent B's strategy is to bid right in the middle of both private values.

$$b_b = \begin{cases} v_b - \frac{1}{2}(v_b - v_a) & \text{if } v_a \leq v_b \\ v_b + \frac{1}{2}(v_a - v_b) & \text{if } v_a \geq v_b \end{cases} = \frac{v_a + v_b}{2}$$

The antisocial utility of agent A takes the following form and is depicted in Figure 4.5.

$$u_a^A(b_a, b_b, v_a, v_b) = u_a^A\left(b_a, \frac{v_a + v_b}{2}, v_a, v_b\right) \overset{4.5}{=} \begin{cases} \frac{v_a - v_b}{4} & \text{if } b_a \geq \frac{v_a + v_b}{2} \\ \frac{b_a - v_b}{2} & \text{if } b_a \leq \frac{v_a + v_b}{2} \end{cases}$$

[1] The margin ε can be omitted here. When two or more bidders share the winning bid, the winner is picked at random.

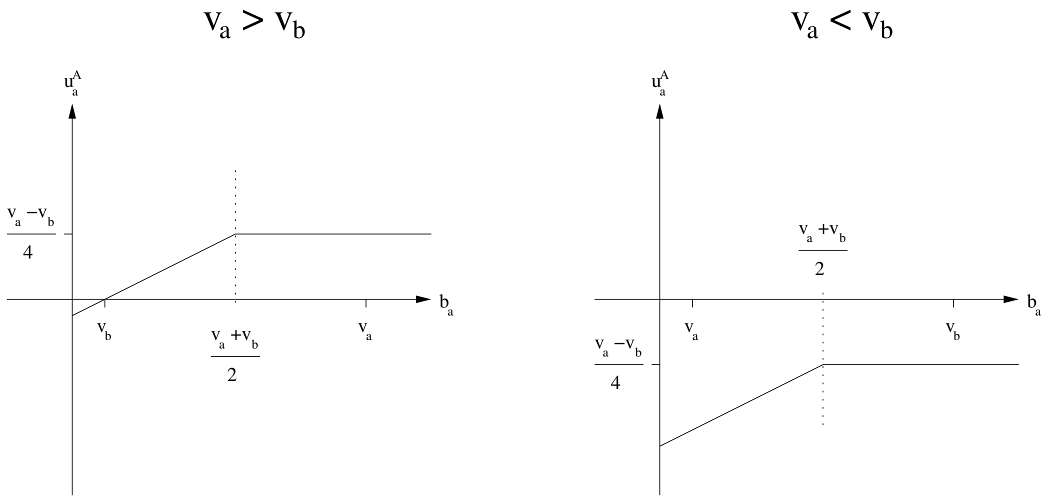


Figure 4.5: A's utility ($d_a = 0.5$)

$$\max_{b_a} u_a^A \left(b_a, \frac{v_a + v_b}{2}, v_a, v_b \right) = \frac{v_a - v_b}{4} \qquad \Rightarrow b_a \geq \frac{v_a + v_b}{2}$$

Concluding, if A bids more than $\frac{v_a+v_b}{2}$, he only receives equal utility; if he bids less, his utility is diminishing. □

When allowing arbitrary derogation rates, a weaker equilibrium concept is appropriate (see Definition 3.11 on page 31).

Theorem 4.2 (Antisocial maximin equilibrium)
The bidding strategy defined by Equation 4.6 is in *maximin equilibrium* (for arbitrary derogation rates in Vickrey auctions).

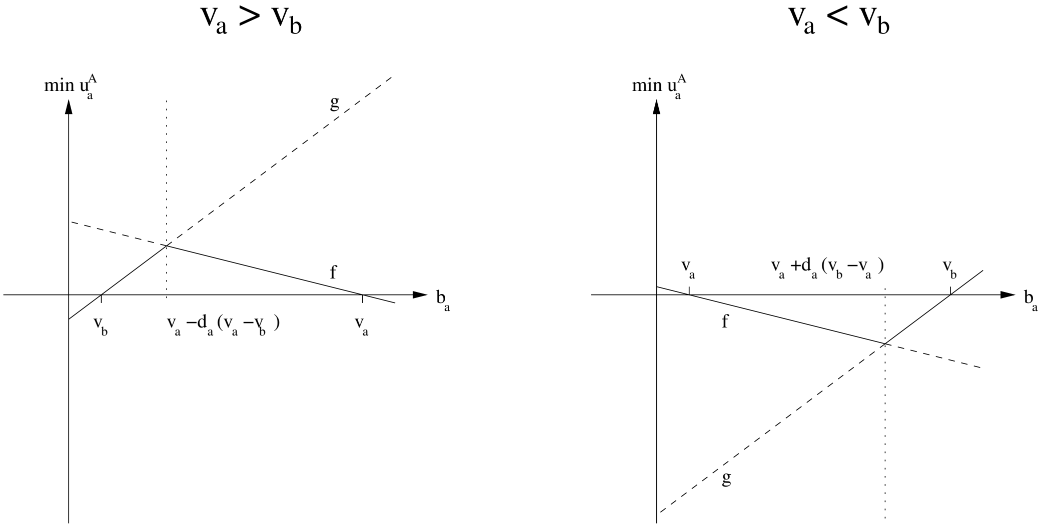
Proof: It is claimed that the strategy is an optimal strategy to reduce the possible losses that occur in worst case encounters (Definition 3.11). "Worst-case" means that the other bidders (represented by a single agent B again) try to reduce agent A's utility as much as possible.

$$\min_{b_b} u_a^A(b_a, b_b, v_a, v_b, d_a) \overset{4.5}{=} \min_{b_b} \begin{pmatrix} (1-d_a)(v_a - b_b) & \text{if } b_a \geq b_b \\ -d_a(v_b - b_a) & \text{if } b_a \leq b_b \end{pmatrix}$$

$$\overset{(*)}{=} \min\{\underbrace{(1-d_a)(v_a - b_a)}_{f(b_a)}, \underbrace{-d_a(v_b - b_a)}_{g(b_a)}\}$$

B's bid b_b can be eliminated in step $(*)$ because the term of the first case is minimal if $b_b = b_a$. f yields the minimum profit if A wins and g yields the minimum profit if he loses the auction. In the following, we consider the maximum of these minima (see Figure 4.6).

$$\max_{b_a} \min_{b_b} u_a^A(b_a, b_b, v_a, v_b, d_a) = \max_{b_a} \min\{f(b_a), g(b_a)\}$$



$v_a > v_b$

$v_a < v_b$

Figure 4.6: A's minimum utility

Due to the fact that f is decreasing and g is increasing, the maximin equilibrium point can be computed by setting $f(b_a) = g(b_a)$.

$$\begin{aligned} f(b_a) = g(b_a) &\Leftrightarrow (1 - d_a)(v_a - b_a) = -d_a(v_b - b_a) \\ &\Leftrightarrow v_a - b_b - d_a v_a + d_a b_b = -d_a v_b + d_a b_a \\ &\Leftrightarrow b_a = v_a + d_a(v_b - v_a) \end{aligned}$$

□

4.3.4 Bidding Strategies with Incomplete Information

On the basis of the theoretical foundations of the previous section, we now develop antisocial bidding strategies that can actually be used in realistic environments.

In contrast to the previous section, we consider a setting of *incomplete information* in the following. In the general case, an agent does not know the private values of other bidders, but he has several possibilities to figure out these values.

1. by estimation based on common knowledge
2. by learning from previous auctions
3. by means of espionage (e.g. bribing or colluding with the auctioneer)

We will now present strategies for the first two cases. The latter case can be prevented by techniques presented in Chapter 5 and 6.

Uniform Distribution

General assumptions about the distribution of unknown values can be used to turn the Nash equilibrium of Theorem 4.1 into a Bayesian Nash equilibrium with incomplete information.

Corollary 4.1 (Antisocial Bayesian Nash equilibrium)
The bidding strategy

$$b_i = \begin{cases} v_i + d_i \left(\frac{n-1}{n}\bar{v} - v_i\right) & \text{if } v_i < \frac{n-1}{n}\bar{v} \\ \left(1 - \frac{d_i}{n}\right) v_i & \text{else} \end{cases}$$

is in Bayesian Nash equilibrium if private values are uniformly distributed in the interval $[0, \bar{v}]$ and derogation rates are uniformly distributed in the interval $[0, 1]$.

Proof: The expected value of another bidder's derogation rate is 0.5. We can therefore modify the (complete information) strategy from Theorem 4.1. If i's private value is not the highest value, bidder i needs to adjust his bid upwards according to Equation 4.6. This is the case if his value is less than the expected highest value of the other bidders: $\frac{n-1}{n}\bar{v}$.

$$v_i + d_i \left(\frac{n-1}{n}\bar{v} - v_i\right)$$

If he possesses the highest private value, he needs to bid less than his private value.

$$v_i - d_i\left(v_i - \frac{n-1}{n}v_i\right) = \left(1 - \frac{d_i}{n}\right)v_i$$

□

Revealing Private Values by Underbidding

We now consider a multiagent task-assignment scenario that we have extensively investigated in [BW99, BW00a, BW00b, BBW00]. A fixed number of tasks is auctioned by using reverse Vickrey auctions. After they have been assigned and executed, the same tasks are auctioned again. This procedure repeats for many rounds.

Zero-Bidding Suppose a balanced antisocial agent loses an auction in the first round. When the same task is auctioned for the second time, he bids zero. As a consequence, he wins the auction[2], and receives an amount equaling the second lowest bid, which is the private value of the cheapest agent (supposing this agent applied the dominant strategy). Thus, he is able to figure out the needed private value and can place his next bid right in the middle between the two private values. Using this technique, he loses the difference between both values once, but can safely cut off 50% of the competitor's profit for all following auction rounds. If the total number of rounds is high enough, the investment pays.

In a scenario where all other agents simply follow the dominant bidding strategy and no counter-speculation is needed, an effective bidding strategy for an antisocial agent who lost in the first round looks like this.

1. Bid 0 (p=received price)

2. Bid $v_i + d_i(p - v_i)$ in all following rounds

Step-by-Step Approach Bidding zero is elegant but dangerous, especially if more than one agent is applying this strategy. In this case, one of the zero-bidding agents wins the auction, but is paid no money at all (because the second lowest bid is zero as well), thus producing a huge deficit. Moreover, he does not learn information on private values. It's safer to reduce a bid from round to round by a small margin s until the lowest bid is reached. Figure 4.7 displays the modified strategy.

[2] unless some other agent bids zero as well

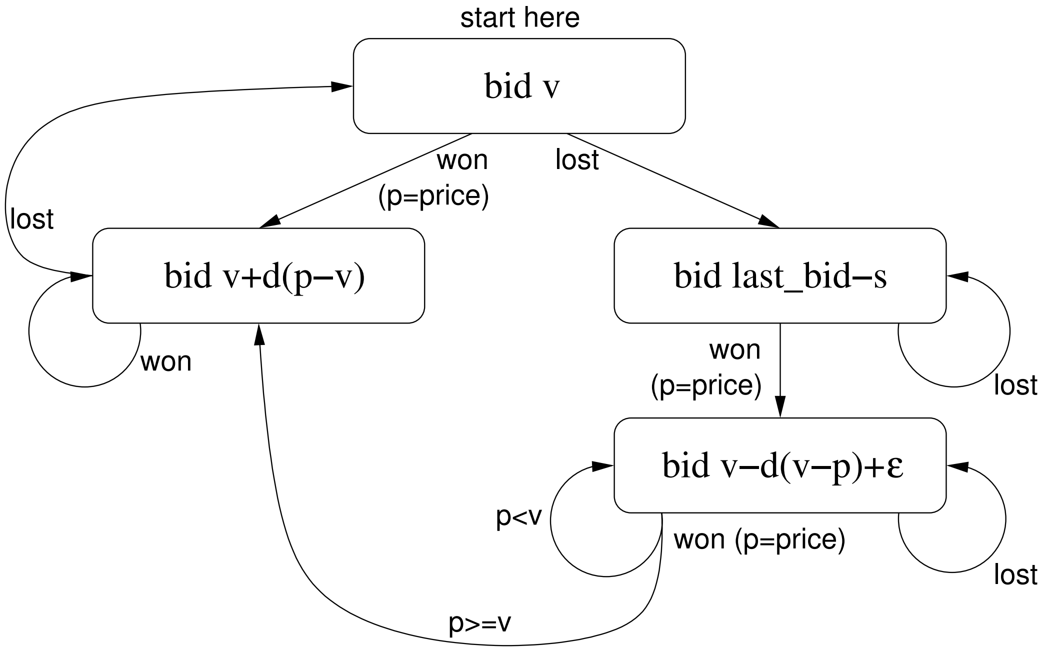


Figure 4.7: Antisocial strategy for repeated reverse Vickrey auctions

If the step size s equals the private value ($s = v$), this algorithm emulates the aggressive zero-bidding strategy. The algorithm works somewhat stable in dynamic environments where agents can vanish and new ones appear from time to time. However, the strategy is not very robust, e.g., if two balanced antisocial agents apply this strategy, the more expensive agent is only able to reduce the winning agent's profit by 25% because he is usually not able to figure out the real private value of the cheaper agent in time.

Generally, a careful agent should use a small step size s in order to be safe that the competitor already suffered huge losses before he makes negative profit himself. A reasonable setting of s depends on the number of rounds, the distribution of private values and the derogation rate.

For example, let us consider the case of two agents A and B ($v_a > v_b$) and an appropriate setting of the step size s_a when the number of total auction rounds is unknown. Let A's step size be a fraction of the difference of both private values: $s_a = \frac{v_a - v_b}{r}$. It is now possible to compute how many rounds are needed to ensure that the loss inflicted to B is greater than the loss induced by underbidding him. This yields an upper bound for s_a.

$$\begin{aligned} & d_a \sum_{i=0}^{r-1} \frac{i}{r}(v_a - v_b) \geq (1-d_a)(v_a - v_b) \\ \Leftrightarrow \quad & d_a \frac{r-1}{2} \geq 1 - d_a \\ \Leftrightarrow \quad & r \geq \frac{2-d_a}{d_a} \Rightarrow \quad s_a \leq \frac{d_a}{2-d_a}(v_a - v_b) \end{aligned}$$

In reality (model of incomplete information), v_b is unknown to agent A. However, assuming that all private values are uniformly distributed, the expected value of v_b is $\frac{v_a}{2}$. This implies the following inequation:

$$s_a \leq \frac{d_a}{4-2d_a} v_a \tag{4.7}$$

For example, the step size s_a for an agent with derogation rate $d_a = 0.5$ should be lower or equal than $\frac{1}{6}$ of his private value. This result can not easily be generalized to other cases as it just takes two bidders into account. If there is more than one bidder that intends to harm agent B and the bidders do not arrange, the situation gets much more complicated. Besides, we assumed that agent B constantly applies the dominant strategy ($d_b = 0$).

Leveled Commitment Contracting If the task execution contracts are not binding and can be breached by paying a penalty (leveled commitment contracting [SL95, SL96, AS98, SSN99, BBW00]), the unavoidable loss an agent produces by underbidding the cheapest competitor can be reduced by breaking the negative contract. Due to the fact that the only reason for closing that deal is to figure out the private value of another agent, the agent has no incentive to really accomplish the task. Therefore, a contractee will break the contract if the loss he makes by accepting the contract is greater than the penalty he pays by breaking the deal. Supposing the common definition of a penalty as a fraction of the contract value [SL96, BBW00], agent i is better off breaching the contract if

$$p \leq \frac{v_i}{pr+1} \tag{4.8}$$

with p being the actual task price and $pr \in [0;1]$ the penalty rate. To give an example, under the assumption that $pr = 0.25$, an agent should break a contract if the task price is less or equal than $\frac{4}{5}$ of his private value. When the distribution of prime costs is uniform, an antisocial agent is better off breaking a contract in 80% of all possible cases.

4.3.5 Experimental Results

The experimental scenario investigated in this section is based on the ABC implementation described in Appendix B.1. There is a number of *contractees* (CE_i) who are willing to execute tasks. Contractees associate prime costs with task execution and are interested in tasks whose prices are higher than their own costs. All prices and bids are integer values ($\varepsilon = 1$).

Whenever the selling of a task is announced, each interested contractee calculates and submits one sealed bid. The contractee who submitted the lowest bid is declared the winner of the auction, and the *second lowest* bid is taken as the price of the announced task; the contractee is paid this price and executes the task. If there are two or more equal winning bids, the winner is picked randomly. As mentioned above, this kind of auctioning is called a reverse Vickrey auction. As a contractee wants to earn money for handling tasks, his private value of a task is his prime costs plus ε.

In contrast to the setting described in Appendix B.1, it is assumed that each contractee can execute as many tasks as he wants during one round. Antisocial strategies can also be used in conjunction with *leveled* commitment, but in order to keep things simple we only consider full commitment contracting here.

	Task 1	Task 2	Task 3
CE_1	70	50	30
CE_2	50	30	70
CE_3	30	70	50

Table 4.2: Fair cost table

	Task 1	Task 2	Task 3	Task 4
CE_1	38	47	39	67
CE_2	43	84	23	49
CE_3	81	10	22	69
CE_4	98	66	18	67

Table 4.3: Random cost table

Identical Contractees

Table 4.2 contains the prime costs of three contractees with exactly identical abilities. Each contractee has one task, that he can handle for the cheapest price. If all three truly bid their private values for 100 rounds, each one would gain \$ 21 · 100=\$ 2100.

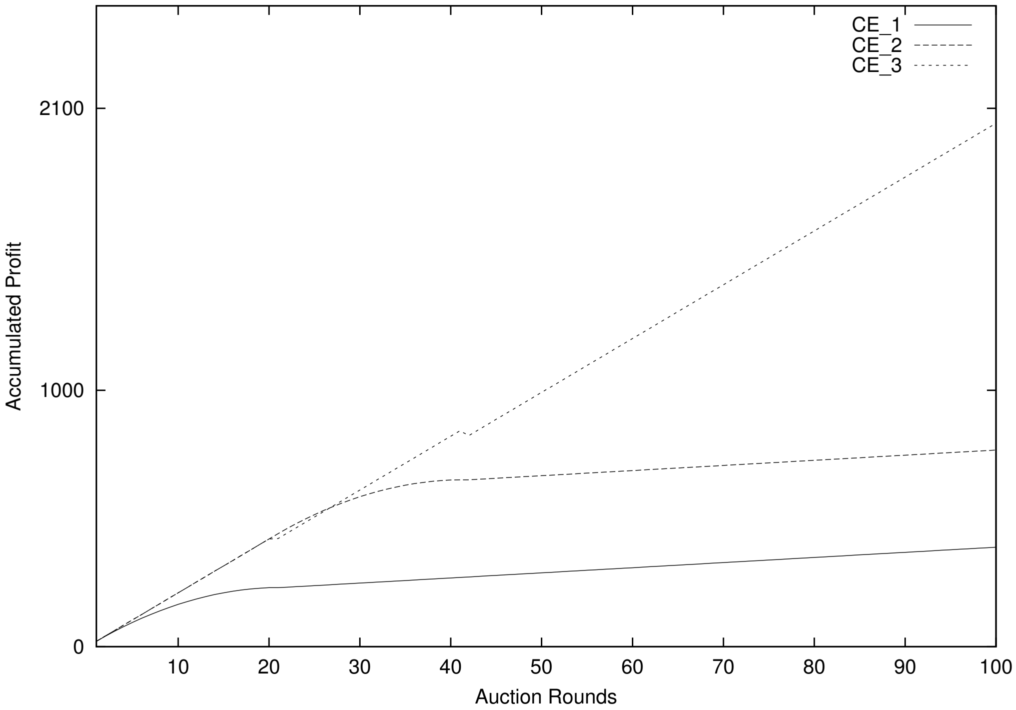


Figure 4.8: Identical contractees, CE_3 is antisocial $(d_3 = 1, s_3 = \varepsilon)$

Figure 4.8 shows the profits accumulated by the contractees in 100 rounds. CE_1 and CE_2 apply the dominant strategy and bid their prime costs plus one. CE_3, however, is antisocial and tries to harm his competitors by reducing their profits to a minimum. As CE_3 is the only antisocial agent and because his derogation rate is 1, he could use a very large step size, e.g., $s_3 = v_3$. We chose a careful step size setting $(s_3 = \varepsilon)$ for two reasons. First of all, CE_3 may not know he is the only antisocial bidder, and secondly, this setting superiorly visualizes how the antisocial strategy works. In contrast to the normal case (all contractees apply the dominant strategy and make equal profits), CE_3 outperforms his rivals by losing only \$ 60. The summed up profit of the entire group of contractees is reduced by more than 50% by actions from a single agent who himself only loses a negligible amount. This emphasizes the particular vulnerability of Vickrey auctions to antisocial bidding.

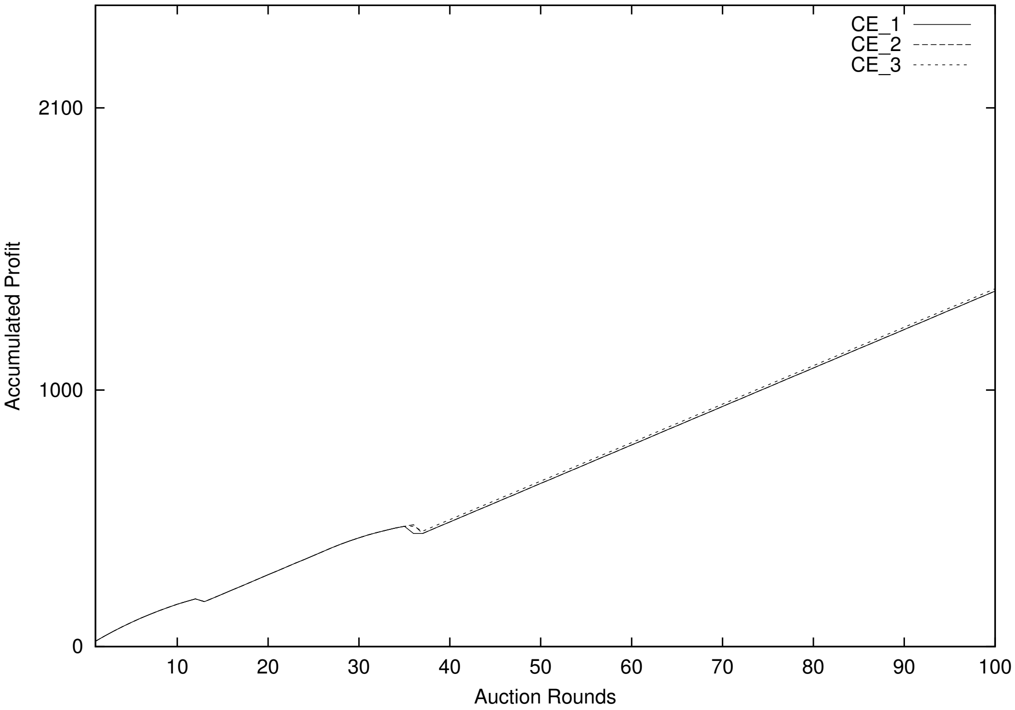


Figure 4.9: Identical antisocial contractees $(d_i = 0.5, s_i = \varepsilon)$

It might appear confusing at the first glance that an agent who does not care for his own profit at all $(d_3 = 1)$ nevertheless makes the highest profit. This effect can be explained by the conservative strategies of his fellow bidders. CE_3 *risks* his entire profit in order to hurt CE_1 and CE_2, but as both are completely "harmless", i.e. not antisocial, he keeps his gain.

If all three contractees are antisocial, overall performance breaks down (Figure 4.9). The agents almost cut off 50% of profits of their rivals.

Random Contractees

In order to examine the performance of antisocial behavior in a more realistic scenario that does not use artificial prime costs, experiments with a random cost table (Table 4.3), that includes four contractees of varying quality, have been conducted.

Figure 4.10 shows the accumulated profit for 200 rounds if one contractee (CE_4, the weakest of them) uses an antisocial strategy. He effectively minimizes the profit of his competitors after figuring out their prime costs. In round 493 he surpasses CE_3 and thus becomes the most successful contractee, even though he has the poorest abilities, i.e. the lowest prime costs compared to the competition.

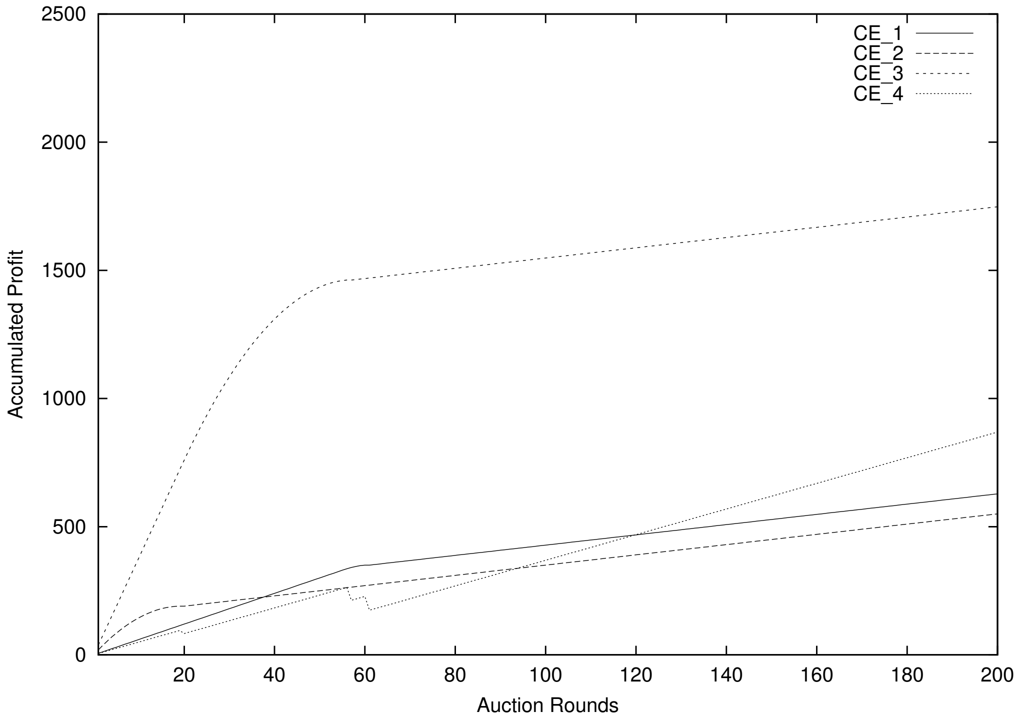


Figure 4.10: Random contractees, CE_4 is antisocial $(d_4 = 1, s_4 = \varepsilon)$

Figure 4.11 shows the profit development for four antisocial contractees. The final profit ranking (CE_3, CE_2, CE_1, then CE_4) does not differ from the result for dominant strategies. However, their overall profits are reduced by 31% to 54% compared to the profits they would accumulate when none of them were antisocial.

4.3.6 Further Implications

As the Vickrey and the English auction are strategically equivalent in the private-value model (see Table 2.1 on page 20), antisocial strategies can be used in English auctions as well. The presence of antisocial agents also affects the bidding strategies in 1st-price sealed-bid auctions. However, in contrast to the Vickrey auction, antisocial bidding can only yield to more (antisocial) utility when making negative profit oneself. The Vickrey auction's 2nd-price policy enables easy price manipulation. If an antisocial bidder knows the highest bid in a Vickrey auction, he can reduce the winning bidder's utility without losing anything. This is not possible in 1st-price auctions.

Although there is no dominant-strategy equilibrium in the Vickrey auction when agents are antisocial, there might be an auction mechanism in

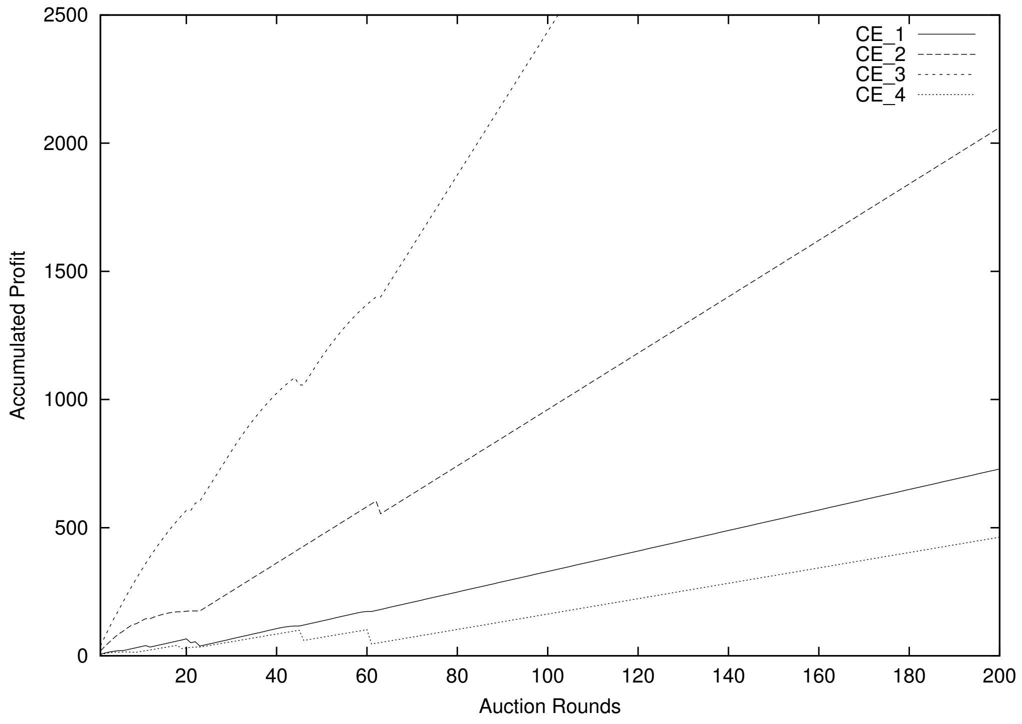


Figure 4.11: Random antisocial contractees ($d_i = 0.5, s_i = \varepsilon$)

which agents submit their private values and derogation rates in dominant-strategy (or Bayesian Nash) equilibrium and that yields a social-welfare-maximizing outcome. As antisocial utility is clearly not quasilinear (compare Definition 3.15), we cannot find a strategy-proof mechanism by applying the Clarke tax mechanism. Anyhow, this does not rule out the existence of such a mechanism (or at least an incentive-compatible mechanism). The following theorem states that there is no auction mechanism that provides basic reasonable properties in the presence of antisocial agents.

Theorem 4.3 (Impossibility of "antisocial-proof" auction)
There is no individually rational auction that maximizes social welfare in equilibrium if at least one of the bidders is antisocial.

Proof: By saying "auction" we mean a mechanism that implements a social choice function that allocates a single good, and that yields outcome $o = (x, \pi)$ where x prescribes who is awarded the good and π is a vector of transfer terms (see Section 3.3.1). We furthermore assume that the seller collects all the payments made by bidders. In order to prove the statement, we outline a social choice function with the desired properties. It turns out that these constraints completely determine the underlying mechanism by prescribing a particular payment rule. Truth-telling must be an equilibrium strategy in this mechanism in order to implement the corresponding social choice function. As this is not the case, the statement is proven by contradiction.

Lemma 4.1
Social welfare in an auction with at least one antisocial agent is maximized if bidders' payments are infinitely high.

Proof: Let us consider two bidders A and B with private values v_a and v_b ($v_a < v_b$, without loss of generality). At least one of the bidders is antisocial which means that $d = d_a + d_b > 0$. If B is awarded the good, the utilities of A, B, and the seller are as follows.

$$u_a = (1-d_a)\pi_a - d_a(v_b+\pi_b), \qquad u_b = (1-d_b)(v_b+\pi_b) - d_b\pi_a, \qquad u_0 = -\pi_a - \pi_b$$

Let U_b be the social welfare, i.e. the sum of individual utilities, when B is awarded the good ($x_b = 1$). We consider U_b subject to π_a and π_b as we seek transfer terms that maximize welfare.

$$\begin{aligned} U_b(\pi_a,\pi_b) = u_a + u_b + u_0 = (v_b+\pi_b)(1-d_a-d_b) + \pi_a(1-d_a-d_b) - \pi_a - \pi_b = \\ = (v_b + \pi_a + \pi_b)(1-d_a-d_b) - \pi_a - \pi_b \end{aligned}$$

As the social welfare only depends on the *sum* of transfer terms, we set $\pi = \pi_a + \pi_b$.

$$U_b(\pi) = (1-d)(v_b + \pi) - \pi = (1-d)v_b - d\pi$$

Similarly, we can compute the social welfare when A is awarded the good ($x_a = 1$).

$$U_a(\pi) = (1-d)v_a - d\pi$$

Please note that transfer terms are negative as bidders pay to receive the good.

In order to obtain an outcome that maximizes social welfare, we need to compute $\max_\pi(U_a(\pi), U_b(\pi))$. It turns out that this is not possible because it would require infinitely high payments by bidders.

$$\arg\max_\pi \begin{pmatrix} (1-d)v_b - d\pi & \text{if } x_b = 1 \\ (1-d)v_a - d\pi & \text{if } x_a = 1 \end{pmatrix} = -\infty$$

Figure 4.12 shows that welfare is linear increasing in $-\pi$, no matter who is awarded the good[3].

[3] In the case of non-antisocial utilities, two dashed horizontal lines at v_a and v_b denote "regular" social welfare. It can be clearly seen that, no matter how π is set, awarding the good to B is the social-welfare-maximizing outcome in that case.

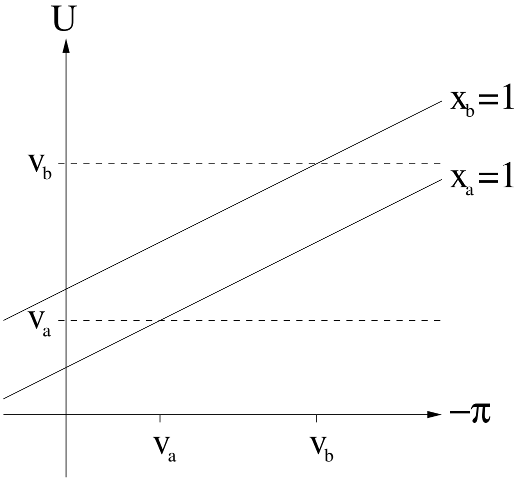


Figure 4.12: (Anti-)social welfare

If there were more than two bidders, the figure would include more and steeper parallel welfare straight lines because the sum of derogation rates d would be higher. Like in the case of non-antisocial agents (see Proposition 3.2), social welfare is highest when the good is delivered to the bidder who values it the most. However, the payments cannot be high enough[4]. □

An auction that assigns infinitely high payments to bidders is obviously not individually rational. When introducing individual rationality to the model, a feasible payment rule can be found.

Lemma 4.2
In an individually rational and social-welfare-maximizing auction with at least one antisocial agent, the winning bidder has to pay his private value of the good.

Proof: Assigning payments to losing bidders cannot be individual rational because participation would always result in less utility (even for antisocial agents). As a consequence, we can focus on finding the highest possible payment for the winning bidder that still ensures individual rationality. We will now prove by complete induction that this optimal payment is the winner's private value, independent from all derogation rates.
Induction start: If there is just a single bidder with private value v and derogation rate d, his utility from participating is $u = (1-d)(v+\pi)$ and his utility

[4] Even when excluding the seller from the social welfare measurement, payments are infinitely high if $\sum_{i=1}^{n} d_i > 1$.

when not participating is zero. Thus, participation is individual rational as long as $-\pi \leq v$. In order to maximize social welfare, the payment needs to be as high as possible, i.e. $\pi = -v$.
Induction step: Let us assume that in an auction with n bidders, the winning bidder has to pay his private value ($\pi' = -v'$). When introducing an additional highest bid, the corresponding bidder's utility from not participating is $\bar{u} = -d(v' + \pi') = 0$ and his utility from participating is $u = (1 - d)(v + \pi)$, which, as above, implies that $\pi = -v$ in order to maximize social welfare. □

Concluding, the auction mechanism has to assign a payment to the winning bidder that equals his private value in order to implement the desired social choice function. Interestingly, the specific derogation rates are irrelevant as long as $d > 0$. The highest private value can only be known to the mechanism infrastructure if truth-telling would be an equilibrium strategy in 1$^{\text{st}}$-price sealed-bid auctions. However, we have seen in Chapter 3 (see Proposition 3.3 and Theorem 3.4) that this is not the case. □

Part II

Privacy Protection

Chapter 5

Security and Partial Revelation

Sealed-bid auctions are desirable auction mechanisms in many areas because they require just a single round of bidding, and thus save bandwidth and time. The main advantage of sealed-bid auctions, however, is the protection of participants' preferences. Depending on the application, these preferences can be extremely sensitive information, e.g. valuations in large-scale B2B auctions or prime costs in procurement reverse auctions. Privacy is of particular importance in auctions with software agents. As the internationally recognized economist Hal Varian puts it:

> "Hence *privacy* appears to be a critical problem for 'computerized purchasing agents'. This consideration usually does not arise with purely human participants, since it is generally thought that they can keep their private values secret. Even if *current* information can be safeguarded, records of past behaviour can be extremely valuable, since historical data can be used to estimate willingness to pay. What should be the technological and social safeguards to deal with this problem"? [Var95]

Furthermore, in scenarios where communication between the participants is not allowed (see Section 2.3.3), sealed-bid auctions prohibit the placing of signals in public bids. A drawback of open-cry auctions like the English auction is the possibility of identifying other bidders, especially the highest bidder, even during the auction process. Members of a bidder collusion can prevent a non-member from winning. Additionally, colluding agents will perceive when a bidder breaks their agreement and are thus able to fine or punish this agent. For these reasons, it can be said that open-cry auctions support bidder collusion [Mea87]. Moreover, in the private-value model, there is no reason to use open-cry auctions at all (except for transparency) as disclosed bids do not change bidders' valuations. E.g., as stated in Section 2.2.3, the English and the Vickrey auction are strategically equivalent in the private-value model.

The strategy-proof Vickrey auction seems to be the ideal sealed-bid auction mechanism. However, despite its impressive theoretical properties, the Vickrey auction is rarely used in practice. It is generally agreed [RTK90, RH95, San96, San00] that the Vickrey auction's sparseness is due to two major reasons:

- the fear of an untruthful auctioneer and
- the reluctance of bidders to reveal their true valuations.

The winner of an auction has to doubt whether the price the auctioneer tells him to pay actually *is* the second highest bid. The auctioneer could easily make up a "second highest" bid to increase his (or the seller's) revenue (see Section 4.2)[1]. William Vickrey himself identified this flaw in the 1961 paper in which he introduced the Vickrey auction:

> "It would be necessary to show the second-best bid to the successful top bidder so that he would be able to assure himself that the price he is being asked to pay is based upon a bona fide bid. To prevent the use of a 'shill' to jack the price up by putting in a late bid just under the top bid, it would probably be desirable to have all bids delivered to and certified by a trustworthy holder, who would then deliver all bids simultaneously to the seller. [...] If corruption of this order cannot be prevented, then this would constitute a serious disadvantage of the second-price method". [Vic61]

In addition to a possibly insincere auctioneer, bidders have to reveal their private values to the auctioneer. There are numerous ways to misuse these values by giving them away to other bidders or the seller. It remains in the hands of the auctioneer whether the auction really is a *sealed*-bid auction. Revelation of bids can be disastrous due to its possible relevance for subsequent negotiations and because criminal sellers or antisocial bidders might use this information, even in the very same auction, in order to increase their utility. In other words, the downside of the existence of a dominant strategy

[1] Even in 1st-price sealed-bid auctions, an untruthful auctioneer could manipulate the auction outcome by determining a winner that did not submit the highest bid. However, since auctioneers usually receive a fraction of the selling price and because this behaviour can be prevented by publicly announcing the selling price, this type of fraud is less significant.

that urges bidders to submit their values truthfully is the fact that a single instance, the auctioneer, receives all these private values[2].

There are various ways how value information can be used strategically. We distinguish the following types of collusive agreements (as depicted in Figure 5.1, the auctioneer is treated like a mediator between bidders and the seller).

- auctioneer/seller (A/S)
- auctioneer/bidder(s) (A/B)
- bidder/bidder (B/B)

B/B collusion can be seen as the most common type of collusion. As the English auction, the Vickrey auction is in particular vulnerable to B/B collusions, i.e., agents that team up to eliminate rivalry, resulting in lower selling prices (see Section 4.1.3).

A classic example of A/S collusion is an auctioneer that overstates the second highest bid to increase the seller's revenue. Another example is an auctioneer that declares a non-existent winning bidder due to too low bids.

An often neglected form of collusion is A/B collusion, e.g., an auctioneer that collaborates with the winning bidder and therefore intends to understate the selling price, or an auctioneer that sells private values to antisocial bidders. Collusions involving the auctioneer (A/S and A/B) are of particular interest in the context of information privacy because they allow agents to receive sensitive information from the auctioneer.

Concluding, in sealed-bid auctions, bidders have to trust the auctioneer that their bids are treated *confidentially* and all participants (bidders and seller) have to rely on the auctioneer selecting the *correct* outcome (see Figure 5.1).

In the remainder of this thesis, we will focus on the development of protocols that compute auction outcomes (primarily from Vickrey auctions) *without* revealing unnecessary information. More than four decades ago, William Vickrey roughly described a mechanical apparatus that fulfills this task. He used a machine applied for Dutch flower auctions as a starting point.

> "As presently practiced, speed is achieved by having a motor-driven pointer or register started downward from a prohibitively high price by the auctioneer; each bidder may at any time press

[2] In 1st-price sealed-bid auctions, there is a similar problem. However, it is less significant as bids are not equal to private values. Counter-speculation leads to strategic bidding.

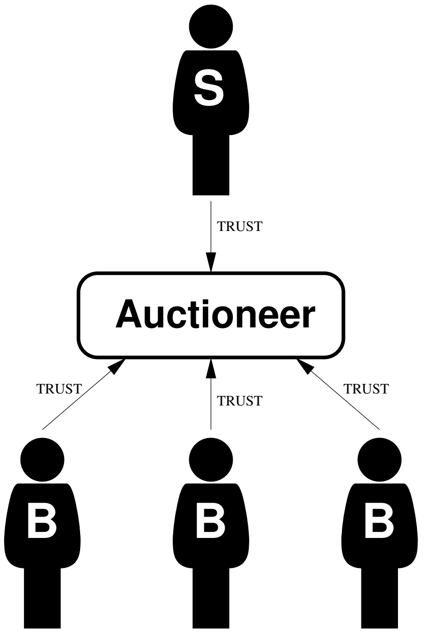


Figure 5.1: Trust centralization in traditional sealed-bid auctions

> a button which will, if no other button has been pushed before, stop the register, thus indicating the selling price, flash a signal indicating the identity of the successful bidder, and disconnect all other buttons, preventing any further signals from being activated. There would be no particular difficulty in modifying the apparatus so that the first button pushed would merely preselect the signal to be flashed, but there would be no overt indication until the second button is pushed, whereupon the register would stop, indicating the price, and the signal would flash, indicating the purchaser [. . .] An even more rapid procedure could be developed, with relatively little increase in the apparatus required, if each bidder were provided with a set of dials or switches which could be set to any desired bid, with the electronic or relay apparatus arranged to search out the two top bids and indicate the person making the top bid and the amount of the second bid". [Vic61]

Obviously, such a machine is only trustworthy if one trusts the manufacturer. Even when it is possible to assure oneself of the correctness of the machine before the auction, it might secretly be modified after that. The same holds for software programs or auctioneer agents. Cryptography is an indispensable tool to provide *provable* security.

5.1 Related Work

There has been a very fast-growing interest in cryptographic protocols for auctions during the last years. In particular, Vickrey auctions and recently the more general $(M+1)$st-price auctions attracted much attention. Starting with the work by Nurmi and Salomaa [NS93] and Franklin and Reiter [FR96], which introduced the basic problems of sealed-bid auctions, but disregarded the privacy of bids after the auction is finished, many secure auction mechanisms have been proposed, e.g. [AS02a, AS02b, BS01, Cac99, HTK98, JJ00, JS02, Kik01, KHT98, HKI03, KHAN00, Kud98, KO02, LAN02, NPS99, Sak00, SM99, SM00a, SM00b, SA99, VBD00, WI00].

When taking away all the protocols that (in their current form) are only suitable for the secure execution of *first*-price auctions or reveal (partial) information after the auction is finished [FR96, AS02b, JJ00, Kud98, NS93, Sak00, SM99, SM00a, SA99, VBD00, WI00], the remaining work can be divided into two categories.

Most of the publications rely on threshold computation that is distributed among auctioneers [HKI03, HTK98, Kik01, KHT98, KHAN00, KO02, SM00b]. This technique requires several auctioneers, out of which a fraction (mostly a majority) must be trustworthy (see Section 6.8.4). Bidders send shares of their bids to each auctioneer. The auctioneers jointly compute the selling price without ever knowing a single bid. This is achieved by using techniques like verifiable secret sharing and secure multiparty function evaluation (see Chapter 6). However, a collusion of, e.g., three out of five auctioneer servers can already exploit the bidders' trust. We argue that distributing the trust onto several distinct auctioneers does not solve the privacy problem, because you can never rule out that some of them, or even *all* of them, collude.

The remaining auction protocols prune the auctioneer's ability to forge the auction outcome and reveal confidential information by introducing a new third-party that is not *fully* trusted (see Sections 6.8.1, 6.8.3, and 6.8.6). However, all of these approaches make weak assumptions about the trustworthiness of this third-party. In [BS01, Cac99] the third-party may not collude with any participating bidder; in [AS02a, LAN02, NPS99, JS02] it is prohibited that the third-party and the auctioneer collude.

Concluding, all present work on secure auction protocols more or less relies on the exclusion of third-party collusion, may it be auctioneers or other semi-trusted institutions. Additionally, many of the existing schemes publicly announce the winner's identity and all of them publicly declare the selling price rather than making this information only visible to the seller and the winners.

5.2 Auctions without Auctioneers

As stated at the beginning of this chapter, the Vickrey (and other sealed-bid) auctions suffer from the possibility of an untruthful auctioneer and the reluctance of bidders to reveal private information. It would be nice to have an auction protocol in which it is impossible for the auctioneer to cheat and that only reveals the Vickrey auction outcome, but no additional information. Thus, the two main demands for such a protocol are *privacy of information* and *correctness of the outcome*:

Privacy It is required that no information concerning bids and the corresponding bidders' identities is revealed. The only information that naturally has to be delivered is the information that is needed to carry out the transaction, i.e.,

- the winning bidder and the seller learn the selling price, and
- the seller gets to know the winner's identity.

As [SM00a] pointed out, *anonymity* of the winners is crucial. Otherwise, a bidder that breaks a collusive agreement could be identified by his partners, thus strengthening the power of collusions. It is important to note that privacy, as defined here, includes that bids can *never* be revealed, even after the auction is finished[3].

Correctness Obviously, the winner and the selling price should be determined correctly. This requirement includes *non-repudiation* (a winning bidder cannot deny having made the highest bid) and *robustness* (no subset of malicious bidders can render the auction outcome invalid). Correctness is usually obtained by making the outcome publicly verifiable.

Privacy and correctness have to be ensured in a hostile environment as we allow every feasible type of collusion categorized at the beginning of this chapter. We assume that up to $n-1$ bidders might share their knowledge

[3] In [AS02b], it is even prohibited that bidders can prove to others how much they bid (*receipt-freeness*). We do not demand receipt-freeness because it requires untappable channels, which are hard to provide in reality.

and act as a team. This implies that each bidder can have arbitrarily many bidder sub-agents, controlled by him. Besides, the seller might collude with bidders, and any number of auctioneers or other third-parties might collude and are therefore not trustworthy. We furthermore assume that there are private communication channels and a public broadcast channel (that we will refer to as a "blackboard"). Both can be provided if one-way functions exist (see Section 6.1.1 and Appendix A.1).

As auctioneers and other third-parties cannot be trusted, we completely omit them and leave the determination of the outcome to bidders themselves. Auction protocols without auctioneers are called *bidder-resolved* in the following. All auction protocols described in this thesis have in common that the auction process is divided into two parts. In the initial phase, bidders publish their somehow encrypted bids (or bid shares) on a blackboard. Nobody is capable of opening a bid without the bidder's help. This phase ends at some pre-determined time and it is impossible to alter existing or add new bids after that deadline. Bidders are committed to their submitted bids. They are not able to decrypt them to anything else than the original value. In the protocols of the following section, the auction outcome is determined by partial decryption of bids.

5.3 Partial Revelation Protocols

Given the two-phase procedure of the previous section, nobody is able to manipulate the outcome of an auction by submitting or changing bids after learning about others' bids. What remains to be done is to determine the outcome of the auction without revealing unnecessary information. The techniques presented in this section identify the selling-price by partially opening bids. They are designed to reveal as little information as possible.

Partial revelation is achieved by the iterative opening of binary bid vectors. After having agreed on a public vector of k possible bids $\vec{p} = (p_1, p_2 \dots p_k)$, each bidder submits a bid vector $(b_{i1}, b_{i2}, \dots, b_{ik})$ that consists of commitments to k binary values denoting whether he is willing to pay a given price or not. For example, when $\vec{p} = (10, 20, 30, 40, 50)$, private value 30 is encoded to the bid vector $(C(1), C(1), C(1), C(0), C(0))$, where $C(b)$ denotes a commitment to bit b. The commitment to bids requires a cryptographic primitive called "bit commitment" (see Appendix A.2).

The bid vectors are put together to form the so-called *bid matrix* (see Table 5.1) and are published on a blackboard. Given this matrix, the goal is to find an opening sequence that rapidly locates the second highest bid by revealing as little information as possible.

	Bidder 1	Bidder 2	…	Bidder n
p_k	$C(b_{1k})$	$C(b_{2k})$	…	$C(b_{nk})$
p_{k-1}	$C(b_{1,k-1})$	$C(b_{2,k-1})$	…	$C(b_{n,k-1})$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
p_1	$C(b_{11})$	$C(b_{21})$	…	$C(b_{n1})$

Table 5.1: Bid matrix

The minimal set of bits that proves the position of the second highest bid and reveals no additional information is called the set of *essential bits* E. This set can be used to prove the auction outcome to outsiders (non-bidders) after the second highest bid p_y has been found at position (x, y) (In the case of equal winning bids, (x', y) denotes another "second-highest" bid. Otherwise, $x' = x$).

$$E \quad = \quad \{b_{xy}\} \cup \{b_{x'y}\} \cup \{b_{i,\min(y+1,k)} \;\mid\; i \in \{1, 2 \ldots n\}\}$$

Figure 5.2 shows an example bid matrix ($n = 10$, $k = 15$, $p_1 = 5, p_2 = 10 \ldots p_{15} = 75$) and the set of essential bits.

The restriction to a finite set of possible bids (or prices) $\vec{p}$ rather than real-numbered bids is not necessarily a limitation since all intervals treated by digital computers are discrete in the end. Additionally, the differences of consecutive bid prices do not have to be equal: Logarithmic scales are possible for example. In a more abstract setting, possible bid prices do not have to be numbers at all. They can be arbitrary objects that are linearly ordered. On the other hand, bid vectors obviously contain redundant information and require linear instead of logarithmic space.

In the following sections, we propose three different search procedures that locate and return the second highest bid. The framework for these procedures is given as follows:

- Phase 1: Each bidder i publishes his bid vector consisting of k committed bits.

  — *Bid submission deadline* —

- Phase 2: The following step is repeated until the second highest bid is uncovered and (if desired) until all essential bits (E) have been opened.

  - Bidder i opens his commitment to bit $C(b_{ij})$ (i and j are yielded by one of the algorithms in the subsequent sections).

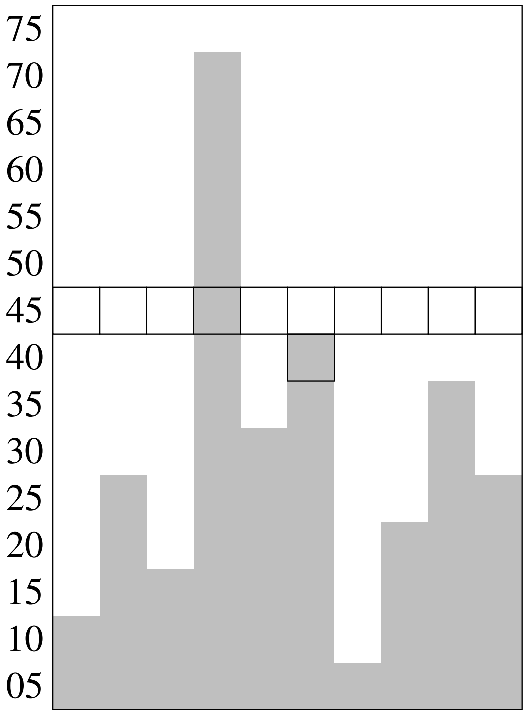


Figure 5.2: Essential bits

> If he fails to fulfill this task in time, a default bid is used and bidder i is fined, if necessary.

- The seller and the winning bidder get in contact and initiate the transaction.

We assume that bidders' indices are randomized to avoid complex randomization in the algorithms.

5.3.1 Downward Bid Search (`dbs`)

A straightforward method to open bits is to start at the highest price and open each row of bids downwards until at least two bidders are willing to pay a given price. This is similar to the *second-price* Dutch (descending) auction described on page 79.

The following algorithm fulfills this task. The algorithm is decomposed into two separate procedures (`dbs` and `dbs2`) because we will reuse the second procedure for the binary search technique in Section 5.3.3. Opened bit commitments are denoted by numbered frames in the example bid matrix in

Figure 5.3, thus illustrating the opening sequence. The search begins in the upper left corner of the bid matrix by evaluating `dbs(1,k)`.

```
procedure int dbs($i, j$)
  while $j > 0$ do
    for $n$ times do
      if $b_{ij}$ = true then
        return dbs2($i, j, \{i\}$)
      end if
      $i = i + 1$
      if $i > n$ then $i = 1$ endif
    end for
    $j = j - 1$
  end while

procedure int dbs2($i, j, F$)
  while $j > 0$ do
    for $n$ times do
      if $i \notin F \wedge b_{ij}$ = true then
        return $j$
      end if
      $i = i + 1$
      if $i > n$ then $i = 1$ endif
    end for
    $j = j - 1$
  end while
```

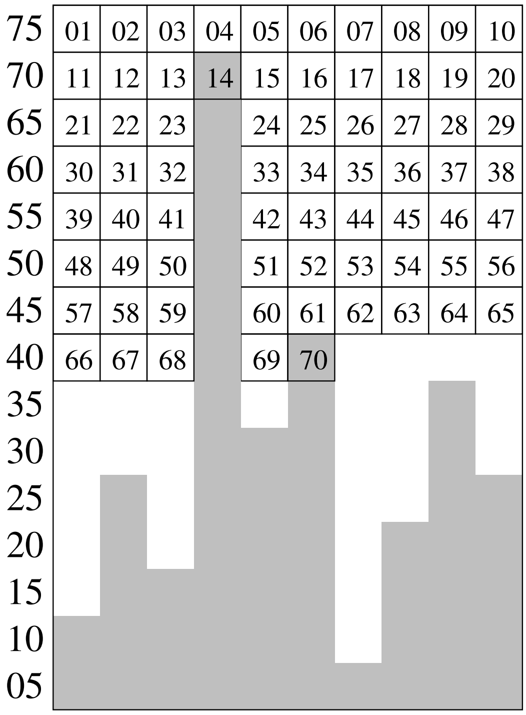


Figure 5.3: Downward bid search (`dbs`)

The maximal number of bits to open (and thus the round complexity) is $\mathcal{O}(nk)$. After the opening process, the bidders know just two out of n bids (the highest and the mandatory second highest) and have no information on other bids. Although, revealing only one private value may seem a fairly good result, a disadvantage of this procedure is that the highest bid usually requires the highest secrecy of all bids in real auctions.

Once the highest bid is revealed, the remaining bidders can falsify the selling price by refusing to open their commitment. It is therefore necessary to assign the default bit 1. As the deliberate refusal of opening a price setting commitment is in an antisocial bidder's interest, such a bidder should be fined appropriately, e.g. by paying p_j for not opening $C(b_{ij})$.

Bidders cannot take advantage of submitting *inconsistent* bid vectors, i.e., vectors that do not represent a private value like $(1, 0, 1, 0, 0)$, as only the first occurrence of a set bit counts.

5.3.2 Upward Bid Search (ubs)

The following algorithm avoids the revelation of the highest bid by opening low bids first. When searching upwards, one can skip to the next higher row when at least one bidder is willing to pay at a given price. This must not be triggered by the same bidder for two times consecutively. ubs's searching technique resembles an English auction (Figure 5.4). The search starts in the lower left corner of the bid matrix (ubs(1,1)).

```
procedure int ubs($i, j$)
  $F = \emptyset$
  while $j \leq k$ do
    $p = 0$
    $F' = \emptyset$
    for $n - 1$ times do
      if $i \notin F$ then
        if $b_{ij}$ = true then $p = 1$
        else $F' = F' \cup i$ endif
      end if
      $i = i + 1$
      if $i > n$ then $i = 1$ endif
      if $p = 1$ then break endif
    end for
    if $p = 0$ then break endif
    $j = j + 1$
    $i' = i$
    $F = F \cup F'$
  end while
  $i = i'$
  for $n - 1$ times do
    if $i \notin F \wedge b_{i,j-1}$ = true then
      return $j - 1$
    end if
    $i = i + 1$
    if $i > n$ then $i = 1$ endif
  end for
  return $j - 2$
```

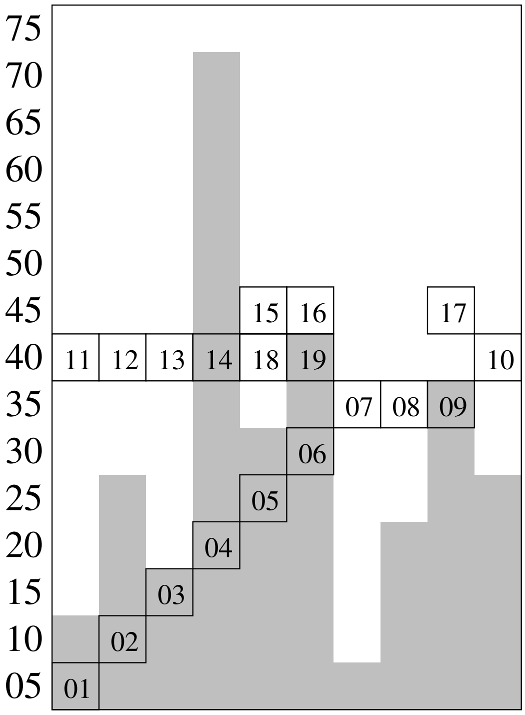


Figure 5.4: Upward bid search (ubs)

This algorithm is significantly faster than dbs. $\mathcal{O}(n + k)$ rounds are required to determine the second highest bid. Bidders learn partial information about losing bids and no information at all about the highest bid. Information about losing bids becomes more and more precise the higher the bids are. The lowest bid can be narrowed down to be in a set of at most n values. The third highest bid is barely hidden after ubs has been executed: it has to

be one out of two possible values.
The default bit 0 should be assigned to commitments that cannot be opened. It is usually not required to fine such uncooperative bidders as bidders refusing to open their commitment lose the chance of winning the auction. However, a winning bidder is able to repudiate his bid by denying to open a commitment.
Another undesirable effect is that bidders can make their bids conditional on other participants' bids by submitting inconsistent vectors. E.g., bidder 2 in Figure 5.4 can make his bid conditional on bidder 1's bid by submitting bid vector $(0, 1, 1, 1, 1, 0, 0, \ldots, 0)$. If bid 1 is greater than 5, bidder 2 bids 25. Otherwise, his bid is 0.

5.3.3 Binary Bid Search (`bbs`)

Like standard binary search, `bbs` begins in the middle of an interval by opening consecutive bids. After two set bits have been found, the row is finished and `bbs` is called recursively for the upper half of the interval. If, after having opened all bits in a row, none of them is 1, the search is continued recursively in the lower half. If exactly one set bid is found, `dbs2` is called from this point. `dbs2` reveals no additional information, except the required second highest bid. The search is initiated by executing `bbs(1,1,k,`$\emptyset$`)`.

This method is a compromise between both previous techniques. Because this algorithm uses `dbs2` to determine the second highest bid, it has the same worst-case round complexity as `dbs` (yet, the *average* round complexity is lower). Applying binary search until the end would reduce the number of opened bits to $\mathcal{O}(n \log(k))$, but this could reveal more information than needed. The search time can be further decreased by starting at the expected value of the second highest bid instead of the middle of the bid interval. `bbs` is somewhat similar to the consecutive opening of bits in standard binary radix representations of bids, but it has the advantage of uncovering less information.
Default bit 0 should be assigned in procedure `bbs` and default bit 1 in procedure `dbs2`. Like in the previous protocol `ubs`, the submission of inconsistent bid vectors allows bidders to make their bids conditional on other bids.

5.3.4 Analysis

The three suggested search techniques clearly illustrate the equivalence of Vickrey, 2$^{\text{nd}}$-price Dutch and English auctions in the private-value model. In addition, the binary search procedure is a novel method to locate the second highest bid. All protocols need more than n rounds because at least one

```
procedure int bbs($i, a, z, F$)
    $j = a + \lfloor \frac{z-a}{2} \rfloor$
    $p = 0$
    $F' = \emptyset$
    for $n$ times do
        if $i \notin F$ then
            if $b_{ij}$ = true then $p = p + 1$
            else $F' = F' \cup i$ endif
        end if
        if $p = 2$ then break endif
        $i = i + 1$
        if $i > n$ then $i = 1$ endif
    end for
    if $p = 2$ then return bbs($i, j, z, F \cup F'$)
    endif
    if $p = 0$ then return bbs($i, a, j, F$)
    else return dbs2($i, j + 1, F \cup i$) endif
```

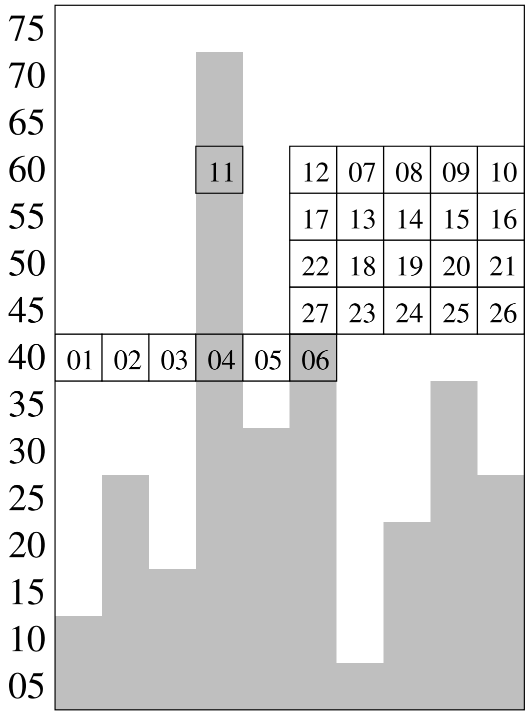


Figure 5.5: Binary bid search (bbs)

bit of each bidder has to be opened. The computational complexity is $\mathcal{O}(k)$ per bidder. When accepting more information revelation, the computational complexity can be reduced to $\mathcal{O}(\log k)$ by using radix representations of bids.

Correctness

If all bidders behave correctly, the protocols yield the correct Vickrey auction outcome. If participants refuse to open a requested commitment, the outcome is altered. Fines are used to establish an incentive to follow the protocol. Clearly, fines can only be imposed when it is possible to somehow prosecute bidders. As a consequence, bidders can not be anonymous and a legal institution is required. In theory, computationally unbounded bidders are able to arbitrarily change their commitments (see Appendix A.2) during the protocol without being detected. However, it is impossible to alter other bids, regardless of computational power. The construction of inconsistent bid vectors allows malicious bidders to submit bids that depend on other participants' bids (in a limited way). Some form of robustness is guaranteed as malicious agents can be removed from the set of bidders without any problems.

Privacy

All three proposed protocols uncover partial information about bids. In fact, *some* information is uncovered on every bid in all of the protocols. Roughly speaking, the closer a bid is to the desired second highest bid, the more precise is the revealed information. E.g., `dbs` completely uncovers the highest bid, and `ubs` almost completely uncovers the third highest bid. Besides the partial revelation, bids are unconditionally secure (see Appendix A.2).

It lies in the nature of partial revelation protocols that bidders are able to quit a protocol after having learned information. As this behaviour can only be confined by assigning penalties to uncooperative bidders which in turn is not possible in many cases, the protocols of the following chapter do not disclose *any* information before the protocol is finished.

Chapter 6

Cryptographic Protocols

In this chapter, sophisticated cryptographic techniques like secret sharing and secure multiparty computation (MPC) will be applied to enable the secret execution of auctions. Beyond the limited scope of auctions, we establish a general link between the fields of mechanism design and MPC by specifying conditions that allow the secret execution of mechanisms without any trusted parties. Unfortunately, existing generic MPC protocols are extremely inefficient. Therefore, the focus of this chapter lies in the construction of efficient special purpose auction protocols. Our main goal is to construct an efficient Vickrey auction protocol.

The first two proposed protocols, B-SHARE and MB-SHARE, introduce how MPC that is distributed on bidders can be used for the private execution of 1^{st}-price sealed-bid auctions. These plain protocols are not optimal but easy to comprehend and analyze. We then approach the more relevant and difficult problem of designing a protocol that privately computes the outcome of Vickrey, i.e. 2^{nd}-price, auctions. The Vickrey auction protocol YMB-SHARE satisfies our demand for "full privacy", i.e., privacy is guaranteed despite any collusion of participants. However, all protocols mentioned so far lack a satisfying level of robustness.

VMB-SHARE and VX-SHARE ("V" stands for "verifiable") provide robustness by making the correctness of each protocol step universally verifiable. Moreover, the most advanced protocol, VX-SHARE, just needs a constant number of rounds. Round complexity is one of the most important complexity measures in distributed protocols as interaction over computer network connections is usually the most time-consuming operation (see e.g. [GIKR01]). We therefore intend to minimize the number of rounds rather than computational complexity.

6.1 Secure Multiparty Computation

Secure multiparty computation (MPC) [CD02, FGY92, Cra00] deals with protocols that allow n parties to jointly compute a function $f(x_1, x_2, \ldots, x_n) = (y_1, y_2, \ldots, y_n)$ on their individual private inputs x_i, so that agent i only learns y_i but nothing else. A classic example is the so-called "millionaires' problem" [Yao86] in which two millionaires want to determine who is richer without revealing their wealth.

The common model defines passive adversaries (or "eavesdropping adversaries") as agents that follow the protocol but try to derive additional information. Active adversaries, on the other hand, try to violate privacy and correctness by *any* means including the sending of faulty messages. Furthermore, there are two basic security models: computational and unconditional security. The security of computational protocols is based on complexity assumptions, i.e., they are only safe against computationally polynomially bounded adversaries[1]. Unconditional (or information-theoretic) protocols, on the other hand, provide perfect security given that agents can communicate via private channels. In other words, the input of unconditional secure protocols can *never* be revealed, whereas inputs of computationally secure can be revealed, but the revelation requires computing power that should not be available for decades, centuries, or even longer[2].

Typically, secure MPC is accomplished by having each agent distribute shares of his individual input to the other participants. This has to be carried out in conjunction with a commitment scheme, so that agents can verify the consistency of shares. This primitive is called *verifiable secret sharing*. In the following, the participants verifiably evaluate a Boolean circuit representing function $f(\cdot)$ with their shares as inputs and new shares as outputs. When the evaluation of the circuit is finished, agents broadcast their resulting shares and reconstruct the final result. In the following, we will call such an MPC scheme "protocol".

Table 6.1 shows the classic results of proven bounds of adversaries tolerable in general secure multiparty computation[3]. The table entries denote how many adversaries of a given kind are tolerable at most. The results for

[1] All practical encryption techniques, symmetric and asymmetric, belong to this category.

[2] Clearly, this requires technological assumptions and complexity theory (see Appendix A.1). However, if an algorithm's running time is exponential, the problem is said to be "intractable". "Performing the exponential algorithm is futile, no matter how well you extrapolate computing power, parallel processing, or contact with superintelligent aliens". ([Sch96], page 239)

[3] $\lfloor \frac{n-1}{2} \rfloor$ active adversaries are tolerable in the unconditional case when allowing non-zero error probability and a broadcast channel.

the computational case have been proposed in [GMW87]. The bounds for unconditional adversaries have been found simultaneously by [BGW88] and [CCD88].

Adversary	polynomially bounded	unbounded
passive	$n-1$	$\lfloor \frac{n-1}{2} \rfloor$
active	$\lfloor \frac{n-1}{2} \rfloor$	$\lfloor \frac{n-1}{3} \rfloor$

Table 6.1: General secure multiparty computation bounds

A protocol is called "t-private" if a collusion of up to t agents is incapable of revealing private information. For example, according to Table 6.1, MPC that is secure against active, bounded adversaries can be at most $\lfloor \frac{n-1}{2} \rfloor$-private. As we want to distribute the emulation of a mechanism on the participants themselves, only $(n-1)$-privacy is acceptable.

Definition 6.1 (Full privacy)
A protocol is *fully private* if a coalition of $n-1$ participants can not reveal the input of the remaining agent.

Clearly, this is the highest bound possible since in the case of n colluding agents, there would be nobody left to spy on.

6.1.1 Unconditional MPC

Let us first consider unconditional multiparty computation and its applicability to secure mechanism design. Without making any assumptions, *verifiable secret sharing* can only be accomplished when more than one third of the participants are honest. Furthermore, it has been proven that the *secure computation* of essential Boolean gates like OR and AND in the unconditional model can only be achieved when a minority of (passive) adversaries are able to pool their knowledge [BGW88]. *Broadcasting*, i.e. sending one message to all other agents, is not generally possible (without a trusted third-party) because it has to be guaranteed that all agents receive the same message. It has been shown in [LSP82] that reliable broadcasting can be achieved in the presence of at most $\lfloor \frac{n-1}{3} \rfloor$ (active) adversaries in the unconditional case. Finally, agents that quit the protocol in progress render it impossible to complete the computation of $f(\cdot)$ in their absence. This is a particular problem in the final stage of a protocol as share revelation cannot be synchronized without a trusted party. As a consequence, a bidder is able to construct the result by using the shares that have been published so far and then decide not to release his share, thus leaving the other agent uninformed about the result. If a majority of participants are assumed to be cooperating, shares can be distributed in a way that allows any majority of agents to reconstruct the original values. This ensures robustness as no minority quitting the protocol can prevent the correct execution of the protocol.

Definition 6.2 (Robustness)
A protocol is (strongly) *robust* if the correct computation of function $f(x_1, x_2, \ldots, x_n)$ with private inputs $x_1, x_2, \ldots, x_n$ can be completed even when participants quit during the protocol.

Robustness obviously implies the critical property of fairness.

Definition 6.3 (Fairness)
A protocol is *fair* if no agent can learn y_i and then prevent the other participants from learning $y_1, y_2, \ldots, y_{i-1}, y_{i+1}, y_{i+2}, \ldots, y_n$.

Concluding, unconditionally secure MPC is possible if there are not more than $\lfloor \frac{n-1}{3} \rfloor$ active adversaries (see Table 6.1). Recapitulating, the reasons for thresholds in unconditional multiparty computation are:

1. Robustness, threshold: $\frac{n}{2}$
2. Feasibility of secure broadcasting, threshold: $\frac{n}{3}$
3. Feasibility of verifiable secret sharing, threshold: $\frac{n}{3}$
   ($\frac{n}{2}$ with error probability and broadcast channel)
4. Feasibility of secure OR, threshold: $\frac{n}{2}$
   (even with only passive adversaries)

Any *threshold* of trusted participants is unacceptable when requiring full privacy. However, it might be possible to make weak assumptions that allow unconditional secure MPC without thresholds. In the following, we will analyze each of the above thresholds with respect to this aspect.

Robustness

First of all, robustness against active adversaries in MPC is defined to allow correct completion of the computation even if active adversaries do not follow the protocol. Even when $\lfloor \frac{n-1}{2} \rfloor$ cheaters were forced to quit the protocol, there are enough agents left (i.e. a majority) to compute $f(x_1, x_2, \ldots, x_n)$, including the inputs of malicious participants.

When presuming that active adversaries can be "kicked out", *including* their inputs, this leads to a weaker notion of robustness.

Definition 6.4 (Weak robustness)
A protocol is *weakly robust* if the correct computation of a function $f(X)$ of inputs supplied by non-adversaries $X \subseteq \{x_1, x_2, \ldots, x_n\}$ can always be completed.

Of course, this only makes sense if $f(\cdot)$ is defined for any number of inputs up to n. A weakly robust protocol then terminates after at most $n-1$ protocol runs. If participation in a mechanism is voluntary, the outcome function $g(s_1, s_2, \ldots, s_n)$ of a mechanism is defined for an arbitrary number of inputs n. To give an example, function $f(\cdot)$ can be the outcome function of a Vickrey auction, i.e. a function that computes the identity of the highest bidder and the amount of the second highest bid given the individual bids as inputs. Clearly, this function is defined for any number of inputs greater than one.

Public verifiability of the protocol is sufficient to provide weak robustness. Unfortunately, when abandoning strong robustness, we also lose fairness. In the end of a protocol run, each participant holds a share of the result. As simultaneous publication of these shares is impossible, a malicious agent might quit the protocol after having learned the result but before others were able to learn it. There are various techniques to approximate fairness by gradually releasing parts of the secrets to be swapped (see e.g. [Yao82]). Another possibility is to introduce a third-party that publishes the outcome after it received all shares. This third-party does not learn confidential information. It is only assumed not to leave the protocol prematurely. We will see that in auctions with a single seller, it is practical to assign this role to the seller[4].

[4] This obviously leaves the possibility of a "cheating seller" who quits the protocol after having learned the (possibly unsatisfying) result. However, such a seller could be forced to sell the good for the resulting price as bidders can compute the auction outcome on their own (or with another fairness-providing third-party).

Broadcasting

Providing a secure broadcast channel eliminates the second threshold. This is not a significant restriction. In fact, a broadcast channel is required in many security models. The Internet, especially the world-wide web, can be used very well as a broadcast channel. There is no absolute guarantee that all viewers see the same web page. However, it would be very hard for a content provider to deliberately change content for specific viewers due to the quasi-anonymity of viewers.

Verifiable Secret Sharing

As we will see in Section 6.6.1, without a threshold, verifiable secret sharing can only provide *unconditional* security of either the shares' correctness or the secret, but not both. The latter seems much more practical since it means that the individuals' preferences can *never* be revealed. A malicious agent, however, can manipulate the protocol by applying super-polynomial computational power *during* the protocol. Given that protocol run times are between a few seconds and some hours in practice, this restriction seems acceptable.

Secure OR

The impossibility of securely evaluating OR (and AND) gates cannot be removed without restricting the participants' abilities[5] (either memory-wise or computational, see [CD02]). A complete characterization of functions that *can* be computed in the unconditional model without loss of privacy has been given in [Kus89]. Unfortunately, the maximum function, which is needed for auctions as well as other mechanisms, does not belong to this set. This can easily be seen by the fact that the maximum function for two one-bit inputs is equal to the OR-function.

PROPOSITION 6.1 (UNCONDITIONAL MECHANISM EMULATION)
It is impossible to emulate arbitrary mechanisms by fully private protocols in the unconditional model, even when assuming weak robustness, providing a broadcast channel, and accepting the possibility of manipulation by computationally unbounded cheaters.

[5] A noisy communications channel can also enable secure OR-gates, but this is of rather theoretical interest.

Proof: The following proof shows the impossibility of computing the OR of two bits without loss of privacy[6]. We therefore assume that two agents, A and B, intend to privately compute a mechanism outcome that is defined as the OR of their individual preference bits: $b = b_a \vee b_b$.

Without loss of generality, we assume that the computation protocol is of the following form. Each of the agents knows his private input bit b_i ($i \in \{a, b\}$) and a chosen private random bit string r_i of appropriate length. The entire protocol is uniquely determined by these initial choices. A starts the protocol by sending message m_{a1}. B replies by sending m_{b1} and so forth until A finally sends $b = b_a \vee b_b$ to B. The transcript of this conversation is called $T = (m_{a1}, m_{b1}, m_{a2}, m_{b2}, \dots, m_{at}, m_{bt}, b)$.

If one of the agents' input bits is 0, the corresponding agent can easily figure out the other agent's input bit due to the nature of the OR function: the other's bit must be equal to the result b. We will now describe how A can determine b_b, even when $b_a = 1$, thus making it possible to *always* reveal the other agent's input bit.

1. A selects $b_a = 1$ and a random bit string r_a.

2. A and B execute the protocol, resulting in transcript $T = (m_{a1}, m_{b1}, m_{a2}, m_{b2}, \dots, m_{at}, m_{bt}, b)$ (known to both parties).

3. If $b_b = 1$, B does not learn *anything* about A's input bit. As a consequence, it must be possible to generate the very same transcript for the case that $b_a = 0$.
   A now tries every possible setting of r_a to find a combination of r_a and $b_a = 0$ that leads to his original first message m_{a1}. This would result in B sending m_{b1}. If his next message would not be m_{a2}, he continues searching for an appropriate r_a until he has found a tuple $(b_a = 0, r_a)$ that would yield transcript T. If he finds such an initial random value, $b_b = 1$. Otherwise, $b_b = 0$.

It follows that the AND of two bits and all somewhat complex functions cannot be computed privately in the unconditional model. The unbounded abilities of A are essential for the proof as a computationally bounded agent certainly cannot try out all settings of r_a if the bit string has a certain length. □

Please note that like the impossibility of strategy-proof implementations for *general* preferences in the Gibbard-Satterthwaite Theorem (Theorem 3.2),

[6] The proof basically consists of an argument described in [BGW88].

Proposition 6.1 only states the impossibility of a general mapping from mechanisms to protocols, i.e., there are (many) mechanisms that cannot be emulated by a fully private protocol. However, there are some primitive mechanisms that can be emulated under the assumptions of Proposition 6.1, e.g., the sum of n input values can be computed fully privately and weakly robustly in the unconditional model if we accept the (theoretical) possibility of manipulations by computationally unbounded participants and provide a broadcast channel. This is possible because some MPC protocols work on finite rings instead on binary values. In these arithmetic protocols, additions (and thus XOR and NOT gates) are feasible while multiplication of shares is impossible without a trusted threshold assumption (multiplication could be used to build OR and AND gates).

As unconditional protocols require private channels, there is a problem of message disputes: A participant who did not send a message may claim that he did, while on the other hand, a participant may state that he did not receive a message that he in fact received. It is reasonable to isolate this conflict at the beginning of the protocol by applying the following procedure [CGS97]. The two parties agree on a symmetric, unconditional secure encryption key K, i.e. a one-time pad, and an information-theoretic secure commitment to this key, and broadcast a signed copy of the commitment. If both published commitments are equal, the two parties can henceforth communicate by broadcasting messages encrypted with the private key K. If the commitments are different, the dispute has to be resolved before the protocol itself begins.

6.1.2 Computational MPC

When allowing intractability assumptions, most of the reasons why unconditional MPC is impossible can be removed. The classic results are based on the existence of trapdoor one-way permutations[7] like the problem of factoring large composite numbers, or the decisional Diffie-Hellman problem (see Appendix A.1).

PROPOSITION 6.2 (COMPUTATIONAL MECHANISM EMULATION)
Any mechanism can be emulated by a fully private, weakly robust protocol in the computational model.

[7] All the assumptions needed in the computational model can be reduced to the existence of "oblivious transfer" which can be achieved by noisy channels, trapdoor functions, or quantum channels (see e.g. [Kil88]).

Proof: It suffices to invalidate the four reasons for threshold trust on page 94.

1. Robustness
   We assume that weak robustness (Definition 6.4) is sufficient.

2. Broadcasting
   It has been shown in [LSP82] that message signatures are sufficient to enable secure broadcasting without any trusted threshold assumptions. There are many signature schemes based on intractability.

3. Verifiable secret sharing
   It has been shown in [Ped91] that verifiable secret sharing is possible without any trusted fraction assumptions in the computational model.

4. Secure OR
   The construction in the proof of Theorem 6.1 does not work if participants have limited computational power. The first schemes that allowed the evaluation of arbitrary Boolean circuits in the computational model have been given in [GMW87].

□

The naive emulation of a mechanism can be extremely inefficient because general cryptographic multiparty computation protocols work on single bits and have excessive complexities. E.g., the general purpose MPC protocol presented in [CvdGT95] takes $\mathcal{O}(n^2l^3D)$ rounds and has a computational complexity of $\mathcal{O}(n^2l^3C)$ operations where l is a security parameter, C the size, and D the depth of the boolean circuit[8].

Recently, probabilistic homomorphic encryption has attracted attention in the context of MPC [CDN01]. It allows more efficient MPC by sharing just one secret key instead of all input values. The computation can be performed directly on encrypted values. This results in just $\mathcal{O}(D)$ rounds and $\mathcal{O}(nlC)$ sent bits. However, this is currently only possible for factorization based encryption schemes like Paillier encryption [Pai99]. The joint generation of secret keys needed for such schemes is quite inefficient [ACS02, BF97, DK01], especially when requiring full privacy. On the other hand, key generation is only needed once at the beginning of a protocol and this kind of MPC can be very effective for large circuits. It would be nice to build an MPC scheme on a discrete logarithm based encryption technique like ElGamal [ElG85] because distributed key generation is much simpler in such cryptosystems

[8] Faster implementations like [GRR98] rely on the assumption that a majority of the participants is honest.

[GJKR99]. Concluding, there is yet no efficient general purpose MPC scheme. We therefore intend to design efficient *specialized* protocols.

To enable further differentiation of protocol security against active adversaries, we introduce non-manipulability.

Definition 6.5 (Non-manipulability)
A protocol is *non-manipulable* if the malicious behavior of (computationally bounded) agents can never lead to a valid outcome that is different from the correct one.

Obviously, all protocols should be at least non-manipulable. This is an even weaker property than weak robustness. It states that manipulation by active adversaries will always be detected, but it remains unknown who prevented the execution of the protocol, making it impossible to execute the protocol again without those agents like in a weakly robust protocol. Once commitments are involved in the protocol (and this will be the case in all of the proposed protocols), *unbounded* active adversaries can manipulate the outcome by opening their commitments to a different value than they committed to.

6.2 Bidder-resolved Auctions

Like in the previous chapter, we argue that no third-party can be trusted and therefore distribute the trust onto bidders themselves. This allows us to set a new standard for privacy. In a scenario with multiple auctioneers (see Section 6.8), it cannot be ruled out that all of them collude. However, when distributing the computation on n bidders, we can assume that it will never happen that *all* bidders share their knowledge due to the competition between them. If they did, each of them would completely abandon his own privacy, resulting in a public auction. We therefore argue, that only bidder-resolved auctions provide *full privacy*, i.e., no information on any bid can be retrieved unless all bidders collude. It should be noted however, that bidders do learn the *number* of participants. Clearly, this is unavoidable because bidders have to interact in order to determine the auction outcome.

It is difficult to assure robustness in bidder-resolved auctions. However, verifiability can be used to provide weak robustness (Definition 6.4), so that malicious bidders will be detected immediately (without additional communication and information revelation) and can be excluded from the set of bidders. The protocol can then be restarted with the remaining bidders proving that their bids did not change[9]. This guarantees termination (after at most $n - M$ iterations in a $(M+1)$st-price auction) and correctness (if we agree that the removal of malicious bidders (and their bids) does not violate correctness). As malicious bidders can easily be fined in verifiable protocols and they do not gain any information, there should be no incentive to perturb the auction and we henceforth assume that a single protocol run suffices.

The naive approach of building a Boolean circuit that computes the auction outcome on binary representations of bids by applying a general MPC scheme is not feasible as those schemes are quite inefficient (see Section 6.1.2) and the circuit depth, and thus the round complexity, depends on the number of bidders and the bid size. Like in the partial revelation protocols of the previous chapter, we therefore use a vector of k possible prices (or valuations) $\vec{p} = (p_1, p_2, \ldots, p_k)$. This results in linear computational complexity but enables special purpose protocols that do not require a general MPC scheme. In fact, the most sophisticated protocol vX-SHARE has constant round complexity because it only uses additions and no multiplications. For ease of notation, b_i denotes the *index* of the corresponding bid in the price vector $\vec{p}$ rather than the bid amount itself. A framework for bidder-resolved auction protocols could look like this:

- The seller publicly announces the selling of a certain good by publishing
  - the good's description,
  - the amount of units to be sold,
  - the registration deadline,
  - lower and upper bounds of the valuation interval, and
  - the bid function, i.e. a function that prescribes how and how many valuations $(p_1, p_2, \ldots, p_k)$ are distributed among that interval subject to the number of bidders n (allowing linear, logarithmic, or any other form of scaling)

  on a blackboard.

- Interested bidders publish their id's on the blackboard.

  — *registration deadline* —

- The bidders jointly compute the winners and the selling price.

[9] This is not mandatory as their should be no reason to strategically change a bid after a bidder has been excluded (assuming the private-value model).

6.3 Protocol B-SHARE

Let us consider 1^{st}-price auctions at first. In a 1^{st}-price sealed-bid auction, each bidder submits a sealed bid and the highest bidder wins the auction by paying the amount of his bid. Thus, n bidders need to secretly compute the maximum of n values. There already is a fully private first-price auction protocol: the Dutch auction (see Section 2.2.4). The auctioneer announces a decreasing bid from round to round starting with the highest possible price. The first bidder that stops the auction by expressing his willingness to pay is awarded the good for the amount of the actual bid. This might take some time (depending on the number of possible bids), but no information except the selling price is revealed.

One might wonder if there are more efficient protocols that take less than k rounds. The following simple protocol B-SHARE ("B" stands for "bid") only needs a constant number of rounds to emulate a 1^{st}-price sealed-bid auction. Bidder i submits k binary values denoting whether he is willing to pay a given price p_j ($b_{ij} \neq 0$) or not ($b_{ij} = 0$). After that, bidders jointly compute the sum of their submitted numbers for each possible price.

$$B_j = \sum_{i=1}^{n} b_{ij} \tag{6.1}$$

Besides in $\langle \mathbb{Z}_l, + \rangle$, this function could be computed in any other finite Abelian group, e.g. $\langle \{0,1\}^l, \text{XOR} \rangle$. Harkavy et al [HTK98] were the first to propose this kind of protocol. However, they distributed bids on m auctioneers instead on bidders.

In order to protect his bid, each bidder decomposes his b_{ij} value into n addends, so-called *additive shares*, that are spreaded among the bidders. Due to the commutativity of addition, the sum of each bidder's shares is B_j (see Figure 6.1). The ith additive share of b_{aj} is denoted by b_{aj}^{+i}.

6.3.1 Formal Description

The following protocol steps have to be executed by bidder a. $i \in [n]$ and $j, b_a \in [k]$. All calculations take place in the finite Abelian group $\langle \mathbb{Z}_l, + \rangle$ with neutral element 0.

Create codes

1. Choose Y_{aj} for each j at random and commit to each Y_{aj}.

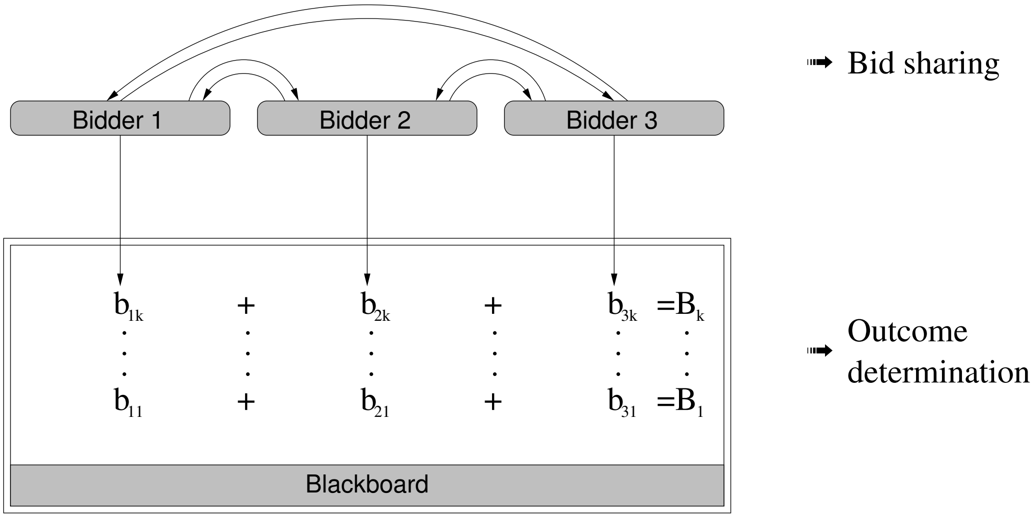


Figure 6.1: B-SHARE

Share bids

2. Choose b_{aj}^{+i} for each j and i, so that

$$\sum_{i=1}^{n} b_{aj}^{+i} = \begin{cases} Y_{aj} & \text{if } j \leq b_a \\ 0 & \text{else} \end{cases} .$$

3. Send b_{aj}^{+i} for each j to bidder i for each $i \neq a$.

4. Receive b_{ij}^{+a} for each $i \neq a$ and j.

5. Publish a commitment to $b_j^{+a} = \sum_{i=1}^{n} b_{ij}^{+a}$ for each j.

6. After all commitments have been revealed, publish b_j^{+a} for each j.

Outcome Determination

7. Compute $B_j = \sum_{a=1}^{n} b_j^{+a}$ for each j by using the published b_j^{+a}.

8. If $B_j = Y_{aj}$ for any j, then bidder a won the auction. The selling price p_w for $w = \min\{j|\, B_j = 0\} - 1$ is visible to all bidders. The winner can prove that he won by opening his commitment to Y_{aw}.

Let us illustrate B-SHARE by an example.

Example: All computations take place in the additive group $\mathbb{Z}_{11}$. There are three bidders ($n = 3$) and four possible valuations ($k = 4$): $\vec{p} = (10, 20, 30, 40)$. $b_1 = 1$, $b_2 = 3$, and $b_3 = 1$. Each bidder chooses vector $\vec{y_i} = (Y_{i1}, Y_{i2}, \ldots, Y_{ik})$, ($\vec{y_1} = (4, 10, 3, 5)$, $\vec{y_2} = (8, 1, 5, 9)$, $\vec{y_3} = (2, 8, 10, 7)$), and commits to it. He then generates his bid vector $\vec{b_i} = (b_{i1}, b_{i2}, \ldots, b_{ik})$ according to $\vec{y}$ and b_i, and creates a 3-partition of $\vec{b_i}$.

$$\vec{b_1} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 4 \end{pmatrix} = \vec{b}_1^{+1} + \vec{b}_1^{+2} + \vec{b}_1^{+3} = \begin{pmatrix} 5 \\ 1 \\ 7 \\ 8 \end{pmatrix} + \begin{pmatrix} 2 \\ 8 \\ 6 \\ 0 \end{pmatrix} + \begin{pmatrix} 4 \\ 2 \\ 9 \\ 7 \end{pmatrix}$$

$$\vec{b_2} = \begin{pmatrix} 0 \\ 5 \\ 1 \\ 8 \end{pmatrix} = \vec{b}_2^{+1} + \vec{b}_2^{+2} + \vec{b}_2^{+3} = \begin{pmatrix} 9 \\ 1 \\ 2 \\ 6 \end{pmatrix} + \begin{pmatrix} 3 \\ 7 \\ 5 \\ 3 \end{pmatrix} + \begin{pmatrix} 10 \\ 8 \\ 5 \\ 10 \end{pmatrix}$$

$$\vec{b_3} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 2 \end{pmatrix} = \vec{b}_3^{+1} + \vec{b}_3^{+2} + \vec{b}_3^{+3} = \begin{pmatrix} 4 \\ 10 \\ 0 \\ 3 \end{pmatrix} + \begin{pmatrix} 6 \\ 8 \\ 2 \\ 2 \end{pmatrix} + \begin{pmatrix} 1 \\ 4 \\ 9 \\ 8 \end{pmatrix}$$

Bidder 1 keeps b_1^{+1}, sends b_1^{+2} to bidder 2 and b_1^{+3} to bidder 3. Bidder 2 and 3 do likewise. Then, each bidder sums up the two shares he received plus the one that he kept in the first place and publishes the resulting vector.

$$\vec{b}^{+1} = \sum_{i=1}^{n} \vec{b}_i^{+1} = \begin{pmatrix} 7 \\ 1 \\ 9 \\ 6 \end{pmatrix}, \quad \vec{b}^{+2} = \sum_{i=1}^{n} \vec{b}_i^{+2} = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 5 \end{pmatrix}, \quad \vec{b}^{+3} = \sum_{i=1}^{n} \vec{b}_i^{+3} = \begin{pmatrix} 4 \\ 3 \\ 1 \\ 3 \end{pmatrix}$$

Each bidder can derive the result by summing up the published vectors.

$$\vec{B} = \sum_{i=1}^{n} \vec{b_i} = \sum_{i=1}^{n} \vec{b}^{+i} = \begin{pmatrix} 0 \\ 5 \\ 1 \\ 3 \end{pmatrix}$$

The selling price 30 (the lowest price at which nobody bid) can be seen by all bidders. Bidder 2 can tell that he won the auction because the second and the third component of $\vec{B}$ are equal to the corresponding components of $\vec{y_2}$. He can prove this to the seller by showing that he committed to $y_{2,3} = 5$. The two losing bidders cannot identify the winner or reveal each other's bid, when they are acting on their own.

6.3.2 Analysis

The entire protocol is finished after three rounds (sending the shares and publishing the sums in two steps) in contrast to the Dutch auction's execution time that is linear in k.

Correctness

The publication of intermediate sums (step 6) is "synchronized" by using some form of commitment to prevent manipulation. Otherwise, the result could be forged by the bidder who releases his sum at last. As the individual codes are randomized and unknown to other bidders, the worst that can happen is that the protocol yields no winner because an active adversary did not follow the protocol. Yet, well directed manipulation is impossible in the computational model: the protocol is *non-manipulable.* There is a very small probability of failure depending on n and l if two or more bidders chose the same code for the same price or if several codes add up to zero by chance, but this is negligible for large l. In fact, the probability of failure can be reduced exponentially by increasing the code size. Computationally unbounded bidders can manipulate the outcome by forging their commitments (see Appendix A.2).

Generally, B-SHARE fails in the following cases:

- A malicious bidder does not follow the protocol (by sending faulty or no messages).
- The highest and the second highest bid are equal (tie).

Failure implies that no bidder is able to prove that he won the auction. A possible solution is to force all bidders to reveal their messages (b_{aj}^{+i}) consecutively for each j beginning at $j = k$. This procedure is executed until the cause of failure is detected. It only reveals the identity of the highest bidder. Malicious bidders (including bidders that do not participate in the failure detection) can then be fined. In other words, a Dutch auction with precommitted bids (see Section 5.3) is used as a backup solution that obviously needs a relatively high amount of additional communication rounds.

Privacy

At first glance, B-SHARE is unconditionally $(n-2)$-private because it takes $n-1$ malicious bidders to reveal the last remaining bid by inverting the computation of Equation 6.1. This is the highest level of privacy possible in the unconditional model. However, the protocol has a major flaw. The second highest bid can be read by the winner of the auction. E.g., in the example given above, bidder 2 learns that the second highest bid is 10, because the fourth component of $\vec{B}$ is the first component that does equal his corresponding Y_{2j}-value. More generally, the cth highest bid can be read by a collusion of the $c-1$ highest bidders. This flaw will be fixed in the subsequent protocol.

6.4 Protocol MB-SHARE

In order to mask the sums of the previous protocol, they can be multiplied with shared random multipliers $M_j = \prod_{i=1}^{n} m_j^{\times i}$ that are not known to any of the bidders. Like in the previous protocol, each bidder holds a share of each M_j, only that shares are multiplicative now. A similar solution was proposed in [KHAN00]. However, the implementation in [KHAN00] provides at most $\lfloor \frac{n-1}{2} \rfloor$-privacy due to an unconditionally secure multiplication technique [BGW88]. Protocol MB-SHARE, where "M" stands for the shared multiplier, uses a one-way function and the propagation of values from bidder to bidder ("ring transfer") to realize the masking (see Figure 6.2).

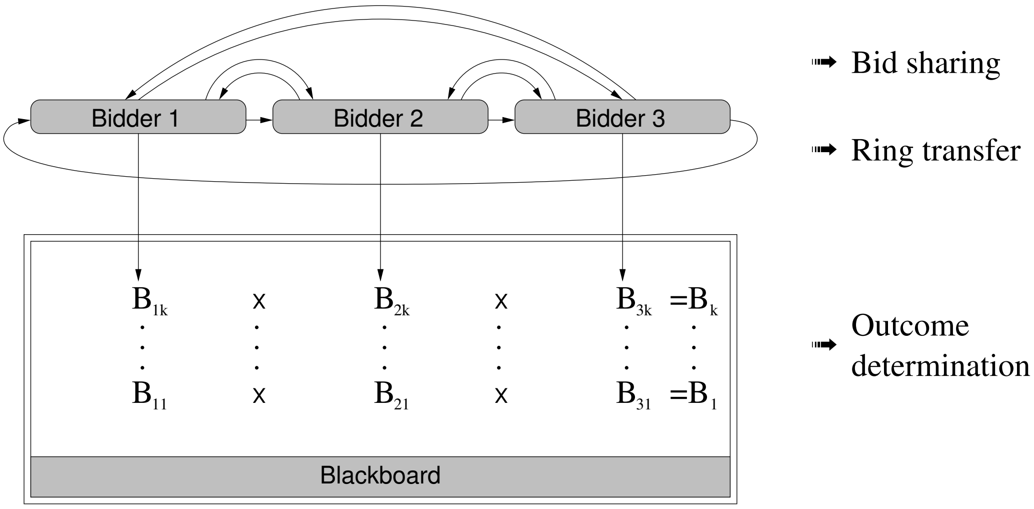


Figure 6.2: MB-SHARE

All calculations take place in a finite multiplicative group. The exponential (one-way) function ensures privacy of the shared exponents, based on the intractability of the discrete logarithm problem (see Appendix A.1). Bidders jointly compute the following function.

$$B_j = \left(\prod_{i=1}^{n} b_{ij} \right)^{\prod_{i=1}^{n} m_j^{\times i}} \tag{6.2}$$

6.4.1 Formal Description

The following protocol has to be executed by bidder a. $i \in [n]$ and $j, b_a \in [k]$. All calculations take place in the finite multiplicative group $\mathbb{G}_q$. p and q are primes so that q divides $p-1$. $\mathbb{G}_q$ is $\mathbb{Z}_p^*$'s unique multiplicative subgroup of order q (see Appendix A.1 for details). In order to enable ring exponentiation, we need an ordering on bidders. $S(i)$ returns the successor to bidder i.

$$S(i) = ((i+1) \mod n) + 1 \tag{6.3}$$

Create codes / Commit to bid

1. Choose $Y_{aj} \in \mathbb{G}_q$ and $m_j^{\times a} \in \mathbb{Z}_q^*$ for each j and r_a at random.

2. Send a commitment to bid b_a to the seller.

Share bids

3. Choose $b_{aj}^{\times i}$ for each j and i, so that $\prod_{i=1}^{n} b_{aj}^{\times i} = \begin{cases} Y_{aj} & \text{if } j \leq b_a \\ 1 & \text{else} \end{cases}$.

4. Send $b_{aj}^{\times i}$ for each j to bidder i for each $i \neq a$.

5. Receive $b_{ij}^{\times a}$ for each $i \neq a$ and j.

Ring exponentiation

6. Compute ${}_{n-1}B_j^{\times a} = \left(\prod_{i=1}^{n} b_{ij}^{\times a} \right)^{m_j^{\times a}}$ for each j and send them to bidder $S(a)$.

7. When receiving ${}_{r}B_j^{\times i}$, compute ${}_{r-1}B_j^{\times i} = ({}_{r}B_j^{\times i})^{m_j^{\times a}}$. If $r > 1$, send it to bidder $S(a)$; else, publish a commitment to $B_j^{\times i} = {}_{0}B_j^{\times i}$.
   Repeat this step until all bid shares (for each i) have been exponentiated.

8. After all commitments have been revealed, publish $B_j^{\times i}$ for each j.

Outcome determination

9. Compute $B_j = \prod_{i=1}^{n} B_j^{\times i}$ for each j by using the published $B_j^{\times i}$.

10. The selling price $p_{\min\{j|\,B_j=1\}-1}$ is visible to all bidders. The winning bidder authenticates to the seller by opening the commitment to his bid.

6.4.2 Analysis

Ring exponentiation requires n additional rounds, but after all the message complexity remains $\mathcal{O}(n)$. It lies in each bidder's interest to choose his codes Y_{ij} in $\mathbb{G}_q$ as Y_{ij}'s with low order might reveal his bid. Ring exponentiation should be synchronized and subsume messages to identical bidders in order to minimize the number of messages.

Correctness

Like B-SHARE, MB-SHARE is *non-manipulable* and has an exponentially small probability of failure. If there is more than one highest bid, a tie, the protocol will lead to an invalid outcome. It turns out that the problem of ties is hard to circumvent in this kind of protocols.

Privacy

The privacy flaw of the previous protocol has been fixed by "masking exponentiations". Moreover, due to the intractability of the discrete logarithm problem, protocol MB-SHARE is fully private in the computational model (apart from publicly declaring the selling price). Similar to B-SHARE, it takes $n-2$ *unbounded* passive adversaries to reveal all bids. However, the mentioned flaw of B-SHARE remains for unbounded adversaries: Given superpolynomial computing power, the c highest bidders can read the $(c+1)$-highest bid.

6.5 Protocol YMB-SHARE

Now, that we have found a technique to acquire the outcome of a 1st-price auction in a fully private protocol, let us consider the Vickrey auction. The previous protocol can not be adapted straightforward to 2nd-price auctions for several reasons, e.g the highest bid must be protected. Protocol YMB-SHARE is based on a new approach, where each bidder has two different codes for each price, denoting whether he is willing to pay at the given price ($(Y_i)^2$: "yes") or not (Y_i: "no"). Bidders submit shares of their bids B_{ij} that are either Y_i or $(Y_i)^2$ and jointly compute

$$B_j = \left(\prod_{i=1}^{n} b_{ij} \right)^{\prod_{i=1}^{n} m_j^{\times i}} \tag{6.4}$$

for each price j. Personalized "keys"

$$K_{ij} = \left(Y_i \prod_{h=1}^{n} Y_h \right)^{\prod_{h=1}^{n} m_j^{\times h}} \tag{6.5}$$

are jointly computed for each bidder i and price j, so that in the end only bidder i knows the value of K_{ij}. By comparing his keys with the published B_j, a bidder can find out whether he won the auction (see Figure 6.3). In order to prevent manipulation of keys, each Y_i value is jointly created by all bidders.

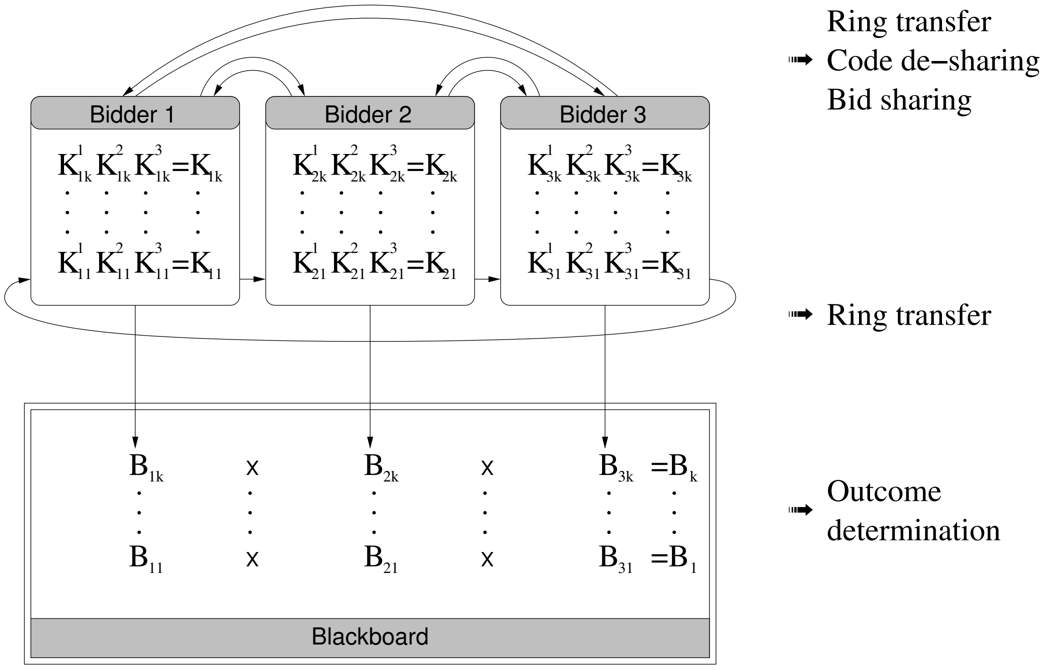

Ring transfer
Code de−sharing
Bid sharing
Bidder 1
Bidder 2
Bidder 3
$K^1_{1k} K^2_{1k} K^3_{1k} = K_{1k}$
$K^1_{11} K^2_{11} K^3_{11} = K_{11}$
$K^1_{2k} K^2_{2k} K^3_{2k} = K_{2k}$
$K^1_{21} K^2_{21} K^3_{21} = K_{21}$
$K^1_{3k} K^2_{3k} K^3_{3k} = K_{3k}$
$K^1_{31} K^2_{31} K^3_{31} = K_{31}$
Ring transfer
B_{1k} x B_{2k} x B_{3k} $=B_k$
B_{11} x B_{21} x B_{31} $=B_1$
Outcome determination
Blackboard

Figure 6.3: YMB-SHARE

6.5.1 Formal Description

Like in the previous sections, this is the step-by-step protocol specification for bidder a, taking place in the multiplicative group $\mathbb{G}_q$. $i \in [n]$ and $j, b_a \in [k]$. The successor function $S(i)$ is defined as in Equation 6.3.

Create shared codes

1. Choose $y_i^{\times a} \in \mathbb{G}_q$ for each i and $m_j^{\times a} \in \mathbb{Z}_q^*$ for each i and j at random.

Compute keys (using ring exponentiation)

2. Compute ${}_nK_{ij}^{\times a} = \left(y_i^{\times a} \prod_{h=1}^{n} y_h^{\times a} \right)^{m_j^{\times a}}$ for each j and i and send them to bidder $S(a)$.

3. When receiving ${}_rK_{ij}^{\times h}$:
   If $r = 0$, set $K_{aj}^{\times h} = {}_0K_{ij}^{\times h}$.
   Else, compute ${}_{r-1}K_{ij}^{\times h} = ({}_rK_{ij}^{\times h})^{m_j^{\times a}}$ and send it to bidder $S(a)$ if $r > 2$ or send a commitment to ${}_0K_{ij}^{\times h}$ to bidder i if $r = 1$.
   Repeat this step until all key shares (for each i and h) have been exponentiated.

4. After all commitments have been revealed, send ${}_0K_{ij}^{\times h}$ to bidder i for each $i \neq a$ and h.

5. Compute $K_{aj} = \prod_{i=1}^{n} K_{aj}^{\times i}$ for each j.

De-share codes / Share bids

6. Send $y_i^{\times a}$ to bidder i for each $i \neq a$.

7. Compute $Y_a = \prod_{i=1}^{n} y_a^{\times i}$.

8. Choose $b_{aj}^{\times i}$ for each j and i, so that $\prod_{i=1}^{n} b_{aj}^{\times i} = \begin{cases} (Y_a)^2 & \text{if } j \leq b_a \\ Y_a & \text{else} \end{cases}$.

9. Send $b_{aj}^{\times i}$ for each j to bidder i for each $i \neq a$.

10. Receive $b_{ij}^{\times a}$ for each $i \neq a$ and j.

Ring exponentiation

11. Compute ${}_{n}B_j^{\times a} = \left(\prod_{h=1}^{n} b_{hj}^{\times a}\right)^{m_j^{\times a}}$ for each j and send them to bidder $S(a)$.

12. When receiving ${}_{r}B_j^{\times i}$, compute ${}_{r-1}B_j^{\times i} = ({}_{r}B_j^{\times i})^{m_j^{\times a}}$. If $r > 1$, send it to bidder $S(a)$; else, publish a commitment to $B_j^{\times i} = {}_{0}B_j^{\times i}$.
    Repeat this step until all bid shares (for each i) have been exponentiated.

13. After all commitments have been revealed, publish $B_j^{\times i}$ for each i.

Outcome determination

14. Compute $B_j = \prod_{i=1}^{n} B_j^{\times i}$ for each j.

15. If $B_j = K_{aj}$ for any j, then bidder a won the auction. He then contacts the seller and authenticates by supplying the signed messages containing $K_{aw}^{\times i}$ for each i and $w = \min\{j \mid B_j = K_{aj}\}$. The selling price is p_{w-1}.

6.5.2 Analysis

The computation of personalized keys for each bidder does not increase the asymptotical number of messages or rounds, but results in a high demand for bandwidth ($\mathcal{O}(n^2k)$) when compared to the previous protocols. The huge amount of numbers to exponentiate can be reduced by substituting ${}_{r-1}K_{ij}^{\times h}$ with an arbitrary random number when $j \leq b_a$ because it is unnecessary to compute keys for a price at which bidder i can not win as his bid is lower than bidder a's.

Correctness

The protocol has been carefully designed to make it impossible to deliberately change the outcome by sending faulty messages. In other words, YMB-SHARE is *non-manipulable.* It yields no result when a malicious bidder sent faulty messages or when the two highest bids are equal. Distributed random generation of Y_i ensures that $Y_i \neq 1$ and $Y_i \neq Y_h$ for any $i, h \in [n]$ (with exponentially small error probability). The codes for "yes" and "no" cannot be equal because $Y_i \neq (Y_i)^2$ in any but the trivial singleton group. Y_i values are not revealed to the corresponding bidders until the computation of keys is finished. This is of particular importance because the creation of keys only consisting of no-codes ($(\prod_{i=1}^{n} Y_i)^{M_j}$) has to be prevented. Such a key would reveal the amount of the highest bid.

Privacy

Many efforts have been made to guarantee privacy while designing this protocol. As a matter of fact, YMB-SHARE provides full privacy in the computational model. This is mainly due to the computational immutability of B_j and K_{ij}. Even in the unconditional model, some level of privacy can be maintained. A single passive unbounded adversary a can determine M_j for any j and thus read the highest bid by testing $B_j \cdot y_a^{M_j} = K_{aj}$. All other bids, however, cannot be opened by this bidder because all Y_i codes, except his own, are unknown to him. It takes $n-2$ unbounded adversaries to reveal the remaining bids.

In contrast to the previous two protocols, the selling price is only visible to the winning bidder and the seller. This is achieved by issuing individual keys for each bidder. However, as correct behaviour of bidders can not be guaranteed, the winning bidder might decide not to contact the seller (if he is unhappy with the selling price for example). In order to resolve such conflicts, intricate methods involving the revelation of all messages (as described in Section 6.3.2) and the fining of agents have to be applied. It is not possible to send copies of all individual keys to the seller, because this would reveal the highest bid to the seller.

6.6 Protocol vMB-SHARE

YMB-SHARE, as well as B-SHARE and MB-SHARE, suffer from the fact that malicious behaviour by active adversaries can not be prevented because these protocols are only non-manipulable but not weakly robust. Weak robustness could be reached if each of the protocol steps were universally verifiable.

Moreover, verifiability of protocol steps enables more general protocols because certain misbehaviour can be ruled out. The following auction protocol is applicable to a generalized version of the Vickrey auction, the $(M+1)$st-price auction (see Section 3.3.2). In an $(M+1)$st-price auction, the seller offers M identical items and each bidder intends to buy *one* of them. It is a strategy-proof mechanism to sell those items to the M highest bidders for the uniform price given by the $(M+1)$st highest bid.

6.6.1 Building Blocks

In order to gain public verifiability, we need cryptographic primitives like verifiable secret sharing and various interactive and non-interactive proofs of correctness. These proofs are "zero-knowledge" [GMR85] which, loosely speaking, means that someone is able to prove knowledge of a certain secret, e.g. a discrete logarithm, without revealing any information about that secret. The most important result regarding zero-knowledge proofs is that any statement in $\mathcal{NP}$ can be proven by a zero-knowledge proof [GMW86a].

Please note that interactive proofs can be made non-interactive by using the Fiat-Shamir heuristic in which the random challenge (c) is derived from the first message of the proof, e.g., by applying a suitable hash function on the message and the sender's id (to avoid proof duplication). This common speed-up of interactive zero-knowledge proofs relies on the so-called *random oracle model* in which it is assumed that agents have access to an oracle that generates true random numbers. In the case of hash functions like MD5 or SHA-1, it is assumed that a cryptographic secure hash functions yields unpredictable numbers.

p and q are large primes, so that q divides $p-1$. $\mathbb{G}_q$ is the unique multiplicative subgroup of $\mathbb{Z}_p^*$ with order q (see Appendix A.1). g_1 and g_2 are random elements of $\mathbb{G}_q$, so that no participant knows $\log_{g_1} g_2$.

Verifiable secret sharing

So far, we shared secrets by simply dividing them into additive or multiplicative shares in a finite Abelian group. Linear combinations of secrets could easily be computed by applying the group operation on these shares. However, there was no possibility to verify the correctness of the resulting shares, i.e. to check if each participant truthfully applied the prescribed calculation rules.

In Shamir's secret sharing scheme [Sha79], a secret is shared among n participants as n points $f(i)$ $(1 \leq i \leq n)$ of an arbitrary, degree $n-1$ polynomial $f(x)$ with $f(0) = s$. A shared secret (SS) can be retrieved by computing $f(0)$ with Lagrange interpolation from these n points.

$$f(0) = \sum_{i=1}^{n} \gamma_i f(i) \quad \text{with} \quad \gamma_i = \prod_{j=1\, j\neq i}^{n} \frac{j}{j-i} \qquad \text{(Lagrange)}$$

Figure 6.4 illustrates the modus operandi. $n-1$ points of the polynomial yield absolutely no information about the secret value. As we want to achieve full privacy, we will not make use of the threshold capabilities of this scheme and always use degree $n-1$ polynomials.

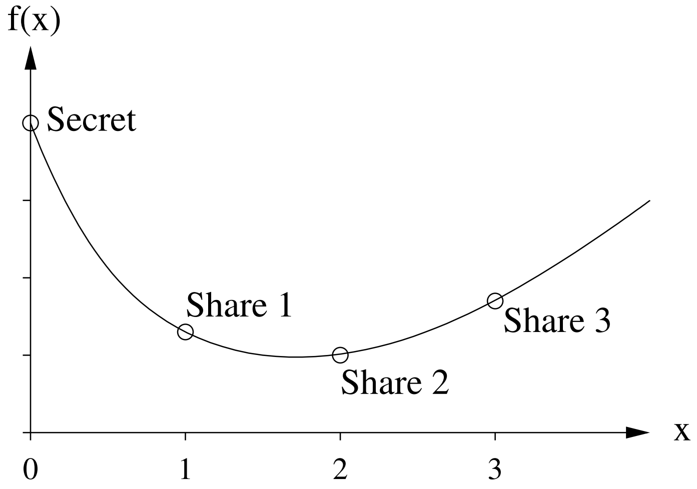


Figure 6.4: Shamir's polynomial secret sharing scheme

Lagrange interpolation can also be applied when shares are only available as exponentiated values $\hat{f}(i) = g^{f(i)}$ to compute the exponentiated shared secret (ESS) $\hat{f}(0) = g^s$.

$$\hat{f}(0) = \prod_{i=1}^{n} (\hat{f}(i))^{\gamma_i} \tag{6.6}$$

The correctness of shares can be proven by using Pedersen's commitment scheme [Ped91]. It provides non-interactive verification of shares and their linear combinations as the commitment scheme is homomorphic. The dealer, who distributes s, chooses two polynomials

$$\begin{aligned} f(x) &= s + F_1 x + F_2 x^2 + \cdots + F_{n-1} x^{n-1} \quad \text{and} \\ h(x) &= H_0 + H_1 x + \cdots + H_{n-1} x^{n-1} \end{aligned}$$

and publishes

$$E_0 = g_1^s g_2^{H_0} \quad \text{and} \quad \forall l \in \{1, 2, \ldots, n-1\} : \; E_l = g_1^{F_l} g_2^{H_l} \quad .$$

He sends shares $f(i)$ and $h(i)$ to participant i. Participant i can verify the correctness of the share by testing

$$g_1^{f(i)} g_2^{h(i)} = \prod_{l=0}^{n} (E_l)^{i^l}.$$

The commitment values E_l reveal absolutely no information about the secret s. The secret is unconditionally secure. However, the holder of the secret can distribute incorrect shares if he is able to solve the discrete logarithm problem. This is inevitable because unconditional security of the secret *and* the unconditional correctness of shares is impossible in $(n-1)$-private verifiable secret sharing [Ped91].

Proof of equality of two SSs

After distributing two SSs with commitment values $E'_0 = g_1^{G'_0} g_2^{H'_0}$ and $E''_0 = g_1^{G''_0} g_2^{H''_0}$, Alice[10] is capable of proving the equality of those secrets by sending $t = H'_0 - H''_0$ to Bob who then verifies that $\frac{E'_0}{E''_0} = g_2^t$. No information on any of the secrets is revealed [Ped91]. Please note that this proof is non-interactive.

Verifiable linear combination computation

Correctness of any linear combination of SSs can be proven without interaction [Ped91].

Two secrets s' and s'' are verifiably distributed. E'_l and E''_l are the corresponding commitment values, $f'(i)$ and $f''(i)$ the secret shares. The participants want to compute s's shares with $s = s' + s''$. Any observer can verify that $(f(i) = f'(i) + f''(i), h(i) = h'(i) + h''(i))$ is a correct share of $s' + s''$ by testing whether

$$g_1^{f(i)} g_2^{h(i)} = \prod_{l=0}^{n} (E'_l E''_l)^{i^l} \quad .$$

When computing $s = as'$ for any $a \in \mathbb{Z}_q^*$, a share $(f(i) = af'(i), h(i) = ah'(i))$ can be proven correct by testing

$$g_1^{f(i)} g_2^{h(i)} = \prod_{l=0}^{n} \left((E'_l)^a\right)^{i^l} \quad .$$

A publicly known summand a can simply be added to each f-share $(f(i) = f'(i) + a)$. The share is correct when $g_1^{f(i)} g_2^{h(i)} = \prod_{l=0}^{n} (g_1^a E'_l)^{i^l}$.

Proof of knowledge of a discrete logarithm

This is a classic, interactive, three-step, zero-knowledge proof by Schnorr [Sch91]. Alice and Bob know v and g, but only Alice knows x, so that $v = g^x$.

[10] The cryptographic literature usually denotes two parties by "Alice" and "Bob".

1. Alice chooses z at random and sends $a = g^z$ to Bob.

2. Bob chooses a challenge c at random and sends it to Alice.

3. Alice sends $r = (z + cx) \mod q$ to Bob

4. Bob checks that $g^r = av^c$.

Proof of equality of two discrete logarithms

When executing the previous protocol in parallel, the equality of two discrete logarithms can be proven [CP92]. Alice and Bob know v, w, g_1, and g_2, but only Alice knows x, so that $v = g_1^x$ and $w = g_2^x$.

1. Alice chooses z at random and sends $a = g_1^z$ and $b = g_2^z$ to Bob.

2. Bob chooses a challenge c at random and sends it to Alice.

3. Alice sends $r = (z + cx) \mod q$ to Bob

4. Bob checks that $g_1^r = av^c$ and that $g_2^r = bw^c$.

Proof that a SS is one out of two values

We designed the following protocol according to the results of Cramer et al [CDS94, CGS97]. Alice shows that a SS is either z or 0 by proving that the corresponding commitment value $x = E_0$ is either $g_1^z g_2^t$ or g_2^t.

1. If $x = g_1^z g_2^t$, Alice chooses r_1, d_1, and w at random and sends x, $a_1 = g_2^{r_1} x^{d_1}$, and $a_2 = g_2^w$ to Bob.
   If $x = g_2^t$, Alice chooses r_2, d_2, and w at random and sends x, $a_1 = g_2^w$, and $a_2 = g_2^{r_2}(xg_1^{-z})^{d_2}$ to Bob.

2. Bob chooses a challenge c at random and sends it to Alice.

3. If $x = g_1^z g_2^t$, Alice sends d_1, $d_2 = c - d_1 \mod q$, r_1, and $r_2 = w - d_2 t \mod q$ to Bob.
   If $x = g_2^t$, Alice sends $d_1 = c - d_2 \mod q$, d_2, $r_1 = w - d_1 t \mod q$, and r_2 to Bob.

4. Bob checks that $c = d_1 + d_2 \mod q$, $a_1 = g_2^{r_1} x^{d_1}$, and $a_2 = g_2^{r_2}(xg_1^{-z})^{d_2}$.

Verifiable random multiplication of an eSS

In contrast to addition, multiplication of shared secrets is hard and all existing techniques require a threshold secret sharing scheme because the pointwise multiplication of polynomials generally results in a higher degree polynomial. However, for the auction protocol, we only need to multiply a SS with a jointly created random number ($s = s'M$) that is unknown to all participants ($M = \prod_{i=1}^{n} m_i$). This is obtained by raising each exponentiated share $\hat{f}(i) = g^{f(i)}$ to the power of each participant's multiplier factor m_i until $g^{f(i)M}$ is computed. It must be impossible for a bounded adversary to reveal s', $g^{s'}$, or g^M and the protocol must be completely verifiable. For this reason, participants are required to spread their shares as exponentiated shares. A combination of Schnorr's and Pedersen's proofs are then used to guarantee the correctness of the first step of ring exponentiation. The details of this novel technique can be seen in the formal specification of vMB-share (Section 6.6.3).

6.6.2 General Description

With all the described cryptographic means at hand, we are able to design a protocol that has several advantages over YMB-share and is applicable to $(M+1)$st-price auctions. In the following abstract description, computation takes place in a finite additive group with neutral element 0.

Each bidder sets the bid vector

$$\vec{b_i} = (b_{i1}, b_{i2}, \dots, b_{ik}) = (\underbrace{0, \dots, 0}_{b_i - 1}, Y, \underbrace{0, \dots, 0}_{k - b_i})$$

according to his bid $b_i \in [k]$, distributes its shares, and shows its correctness by proving $\forall j \in [k] : \;\; b_{ij} \in \{0, Y\}$ and $\sum_{j=1}^{k} b_{ij} = Y$ in zero-knowledge manner [AS02a]. $Y \neq 0$ is a publicly known element, e.g. 1.

Verifiable secret sharing allows verifiable computation of linear combinations of SSs in a single round. When computing on vectors of SSs (like $\vec{b_i}$), this means that besides addition and subtraction of shared vectors, multiplication with (known) matrices is feasible. For example, the "integrated" bid vector (as introduced in [AS02a])

$$\vec{b_i'} = (\underbrace{Y, \dots, Y}_{b_i}, \underbrace{0, \dots, 0}_{k - b_i}) = (b_{i1} + b'_{i2}, b_{i2} + b'_{i3}, \dots, b_{ik})$$

can be derived by multiplying the bid vector with the $k \times k$ lower triangular matrix L ($\vec{b}'_i = \mathsf{L}\vec{b}_i$).

$$\mathsf{L} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ 1 & \cdots & \cdots & 1 \end{pmatrix} \qquad \text{(lower triangular matrix)}$$

Multiplying a vector with $\mathsf{L} - \mathsf{I}$, where I is the $k \times k$ identity matrix, yields $\vec{b}'_i$ shifted down by one component.

$$\mathsf{I} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{pmatrix} \qquad \text{(identity matrix)}$$

If we sum up all integrated bid vectors and down-shifted integrated bid vectors, we obtain a vector that has the following structure (let us for now disregard the possibility of equal bids, we will refer to this case in Section 6.7.3).

$$(2\mathsf{L} - \mathsf{I}) \sum_{i=1}^{n} \vec{b}_i = (\ldots, 6Y, \ldots, 6Y, 5Y, 4Y, \ldots, 4Y, 3Y, 2Y, \ldots, 2Y, Y, 0, \ldots, 0)$$

The position of the (single) component that equals $3Y$ denotes the second highest bid, $5Y$ the third highest bid, and so forth. Subtracting $(2M+1)Y\vec{e}$ with $\vec{e} = (1, \ldots, 1)$, thus yields a vector in which the component, that refers to the amount of the $(M+1)$st highest bid, is zero. All other components are not zero.

As we intend to create personal indicators for each bidder, we mask the resulting vector so that only winning bidders can read the selling price. This is achieved by adding $\mathsf{U}\vec{b}_i$.

$$\mathsf{U} = \begin{pmatrix} 1 & \cdots & \cdots & 1 \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 \end{pmatrix} \qquad \text{(upper triangular matrix)}$$

For an arbitrary bidder a, the vector

$$(2\mathsf{L} - \mathsf{I}) \sum_{i=1}^{n} \vec{b}_i - (2M+1)Y\vec{e} + (2M+2)\mathsf{U}\vec{b}_a$$

only contains a component equal to zero, when a qualifies as a winner of the auction. The position of this component then indicates the selling price.
In order to get rid of all information besides the selling price, each component is multiplied with a different random multiplier M_{ij} that is jointly created and unknown to any subset of bidders. Finally, each bidder's personal indication vector is computed according to the following equation.

$$\vec{v}_a = \left((2\mathsf{L} - \mathsf{I}) \sum_{i=1}^{n} \vec{b}_i - (2M+1)Y\vec{e} + (2M+2)\mathsf{U}\vec{b}_a \right) \mathsf{R}^*_a \tag{6.7}$$

$$\mathsf{R}^*_i = \begin{pmatrix} M_{ik} & 0 & \cdots & 0 \\ 0 & M_{i,k-1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & M_{i1} \end{pmatrix} \qquad \text{(random multiplication matrix)}$$

The components M_{ij} are unknown to bidders.

The invariant of this "blinding" transformation are components that equal zero[11]. Only bidder i and the seller get to know $\vec{v}_i$.

$$v_{ij} = 0 \quad \Longleftrightarrow \quad \text{Bidder } i \text{ won and has to pay } p_j$$

The following simple example for two bidders illustrates the functionality of the protocol.

Example: The computations take place in $\mathbb{Z}_{11}$ and the auction to be conducted is a Vickrey auction ($M = 1$). Let the vector of possible prices be $\vec{p} = (10, 20, 30, 40, 50, 60)$. The two bids are 20 ($b_1 = 2$) and 50 ($b_2 = 5$): $\vec{b}_1 = (0,1,0,0,0,0)$, $\vec{b}_2 = (0,0,0,0,1,0)$. The selling price can be determined

[11] As described in the previous section, random exponentiation works on exponentiated shares, resulting in exponentiated vector $\hat{\vec{v}}_i$ and 1s instead of 0s.

by computing

$$
\begin{aligned}
&(2\mathsf{L}-\mathsf{I})\sum_{i=1}^{n}\vec{b}_i-(2M+1)Y\vec{e}= \\
&=\begin{pmatrix}1&0&0&0&0&0\\2&1&0&0&0&0\\2&2&1&0&0&0\\2&2&2&1&0&0\\2&2&2&2&1&0\\2&2&2&2&2&1\end{pmatrix}\left(\begin{pmatrix}0\\0\\0\\0\\1\\0\end{pmatrix}+\begin{pmatrix}0\\1\\0\\0\\0\\0\end{pmatrix}\right)-\begin{pmatrix}3\\3\\3\\3\\3\\3\end{pmatrix}= \\
&=\begin{pmatrix}0\\1\\2\\2\\3\\4\end{pmatrix}-\begin{pmatrix}3\\3\\3\\3\\3\\3\end{pmatrix}=\begin{pmatrix}8\\9\\10\\10\\0\\1\end{pmatrix}=\vec{x} \quad .
\end{aligned}
$$

Now, the selling price has to be masked to losing bidders. Bidder 1 is unable to identify the selling price. His indication vector ($\vec{v}_1$) contains random numbers.

$$
\begin{aligned}
&\vec{v}_1=\vec{x}+(2M+2)\mathsf{U}\vec{b}_1= \\
&=\begin{pmatrix}8\\9\\10\\10\\0\\1\end{pmatrix}+4\begin{pmatrix}1&1&1&1&1&1\\0&1&1&1&1&1\\0&0&1&1&1&1\\0&0&0&1&1&1\\0&0&0&0&1&1\\0&0&0&0&0&1\end{pmatrix}\begin{pmatrix}0\\0\\0\\0\\1\\0\end{pmatrix}=\begin{pmatrix}8\\9\\10\\10\\0\\1\end{pmatrix}+\begin{pmatrix}4\\4\\4\\4\\4\\0\end{pmatrix}= \\
&=\begin{pmatrix}1\\2\\3\\3\\4\\1\end{pmatrix}\xrightarrow{\times R_1^*}\begin{pmatrix}.\\.\\.\\.\\.\\.\end{pmatrix}
\end{aligned}
$$

Bidder 2's indication vector $\vec{v}_2$, however, indicates the selling price at the second component (bottom-up).

$$\vec{v}_2 = \vec{x} + (2M+2)\mathsf{U}\vec{b}_2 =$$

$$= \begin{pmatrix} 8 \\ 9 \\ 10 \\ 10 \\ 0 \\ 1 \end{pmatrix} + 4 \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 8 \\ 9 \\ 10 \\ 10 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 4 \\ 4 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} =$$

$$= \begin{pmatrix} 1 \\ 2 \\ 10 \\ 10 \\ 0 \\ 1 \end{pmatrix} \xrightarrow{\times R_2^*} \begin{pmatrix} . \\ . \\ . \\ . \\ 0 \\ . \end{pmatrix}$$

6.6.3 Formal Description

In this section, the abstract protocol of the previous section is implemented using verifiable secret sharing. The following protocol rules are specified for bidder a. $i, h \in [n]$, $j, b_a \in [k]$, and $l \in \{0, 1, \dots, n-1\}$ unless otherwise noted. All calculations are done in the multiplicative group $\mathbb{G}_q$. $Y \in \mathbb{Z}_q^*$ is known to all bidders, e.g. $Y = 1$. $g_1, g_2 \in \mathbb{G}_q$.

Like in MB-SHARE and YMB-SHARE, an ordering on bidders is needed for ring exponentiation. $S(i)$ and $P(i)$ return the successor and predecessor to bidder i, respectively.

$$S(i) = ((i+1) \mod n) + 1, \quad P(i) = ((i-1) \mod n) + 1 \tag{6.8}$$

Verifiably share bid

1. Choose random multipliers $m_{aij} \in \mathbb{Z}_q^*$ for each i and j, and $2j$ polynomials with random coefficients in $\mathbb{Z}_q$ and $F_{aj0} = \begin{cases} Y & \text{if } j = b_a \\ 0 & \text{else} \end{cases}$.

$$f_{aj}(x) = F_{aj0} + F_{aj1}x + \cdots + F_{aj,n-1}x^{n-1}$$
$$h_{aj}(x) = H_{aj0} + H_{aj1}x + \cdots + H_{aj,n-1}x^{n-1}$$

2. Publish $E_{ajl} = g_1^{F_{ajl}} g_2^{H_{ajl}}$ for each j and l.

3. Prove that $E_{aj0} \in \{g_1^Y g_2^t, g_2^t\}$ for each j, and $\displaystyle\prod_{j=1}^{k} E_{aj0} = g_1^Y g_2^{H_a}$ by publishing $H_a = \sum_{j=1}^{k} H_{aj0}$.

4. Publish $$E^*_{ijl} = \left(\frac{\left(\prod_{h=1}^{n} \prod_{d=j}^{k} E_{hdl} E_{h,d+1,l}\right) \left(\prod_{d=1}^{j} E_{idl}\right)^{2M+2}}{g_1^{(2M+1)Y}} \right)^{m_{aij}}$$ for each i, j, and l, and prove the discrete logarithm knowledge.

5. Send $\hat{f}_{aj}(i) = g_1^{f_{aj}(i)}$ and $\hat{h}_{aj}(i) = g_2^{h_{aj}(i)}$ to bidder i for each $i \neq a$ and j.

6. Publish $2nk$ exponentiated random numbers $\hat{y}_{aij} = g_1^{y_{aij}}$ and $\hat{z}_{aij} = g_2^{z_{aij}}$.

7. Choose $c^{+a} \in \mathbb{Z}_q$ at random and publicly commit to it.

8. Publish c^{+a} after all bidders published their commitments.

9. Compute $c = \sum_{i=1}^{n} c_j^{+i} \mod q$ and send $r_{aij} = y_{aij} + c f_{aj}(i) \mod q$ and $s_{aij} = z_{aij} + c h_{aj}(i) \mod q$ to bidder i for each $i \neq a$ and each j.

10. Verify $$g_1^{r_{iaj}} = \hat{y}_{iaj} + \left(\hat{f}_{ij}(a)\right)^c \quad , \quad g_2^{s_{iaj}} = \hat{z}_{iaj} + \left(\hat{h}_{ij}(a)\right)^c \quad \text{, and}$$ $$\hat{f}_{ij}(a)\hat{h}_{ij}(a) = \prod_{l=0}^{n-1} (E_{ijl})^{a^l} \quad \text{for each } i \neq a,\ j.$$

Ring exponentiation

11. Publish for each i and j:

$$
\begin{aligned}
{}_a\hat{v}_{ij}(a) &= \left(\frac{\left(\prod_{h=1}^{n} \prod_{d=j}^{k} \hat{f}_{hd}(a)\hat{f}_{h,d+1}(a)\right)\left(\prod_{d=1}^{j} \hat{f}_{id}(a)\right)^{2M+2}}{g_1^{(2M+1)Y}} \right)^{m_{aij}} ,\\
{}_a\hat{w}_{ij}(a) &= \left(\left(\prod_{h=1}^{n} \prod_{d=j}^{k} \hat{h}_{hd}(a)\hat{h}_{h,d+1}(a)\right)\left(\prod_{d=1}^{j} \hat{h}_{id}(a)\right)^{2M+2} \right)^{m_{aij}} ,\\
\hat{y}^*_{iaj} &= \left(\left(\prod_{h=1}^{n} \prod_{d=j}^{k} \hat{y}_{had}\hat{y}_{ha,d+1}\right)\left(\prod_{d=1}^{j} \hat{y}_{iad}\right)^{2M+2} \right)^{m_{aij}} , \text{ and}\\
\hat{z}^*_{iaj} &= \left(\left(\prod_{h=1}^{n} \prod_{d=j}^{k} \hat{z}_{had}\hat{z}_{ha,d+1}\right)\left(\prod_{d=1}^{j} \hat{z}_{iad}\right)^{2M+2} \right)^{m_{aij}}
\end{aligned}
$$

and prove the equality of the discrete logarithms of $\hat{y}^*_{iaj}$ and $\hat{z}^*_{haj}$, and the ones used in step 4.

12. Send

$$
\begin{aligned}
r^*_{iaj} &= \left(\left(\sum_{h=1}^{n} \sum_{d=j}^{k} r_{had} r_{ha,d+1} \right) + \sum_{d=1}^{j} (2M+2) r_{iad} - (2M+1)Y \right) m_{aij} \\
&\quad \bmod q \quad \text{and} \\
s^*_{iaj} &= \left(\left(\sum_{h=1}^{n} \sum_{d=j}^{k} s_{had} s_{ha,d+1} \right) + \sum_{d=1}^{j} (2M+2) s_{iad} \right) m_{aij} \mod q
\end{aligned}
$$

to bidder i for each $i \neq a$ and each j.

13. Verify

$$
g_1^{r^*_{ihj}} = \hat{y}^*_{ihj} + (\hat{v}_{ij}(h))^c \quad , \quad g_2^{s^*_{ihj}} = \hat{z}_{ihj} + \left(\hat{h}_{ij}(h)\right)^c \quad \text{, and}
$$

$$
\hat{v}_{ij}(h)\hat{w}_{ij}(h) = \prod_{l=0}^{n-1} (E^*_{ijl})^{h^l} \quad \text{for each } i,\ j, \text{ and } h \neq a.
$$

14. Compute and publish $\forall i, j, h :\ {}_a\hat{v}_{ij}(h) = \left({}_{P(a)}\hat{v}_{ij}(h)\right)^{m_{aij}}$ and prove its correctness by showing the equality of logarithms. Repeat this step until all ${}_{P(P(h))}\hat{v}_{ij}(h)$ are computed.

15. Compute $\forall i, j: \ {}_a\hat{v}_{ij}(S(a)) = \left({}_{P(a)}\hat{v}_{ij}(S(a))\right)^{m_{aij}}$ and privately send it and a proof of its correctness to the seller who publishes all ${}_h\hat{v}_{ij}(S(h))$ and the corresponding proofs of correctness for each $i, j, h \neq i$ after having received all of them.

Outcome determination

16. Compute $v_{aj} = \prod_{i=1}^{n} \left({}_{P(i)}\hat{v}_{aj}(i)\right)^{\gamma_i}$ for each j.

17. If $v_{aw} = 1$ for any w, then bidder a is a winner of the auction. p_w is the selling price.

6.6.4 Analysis

Like in YMB-SHARE, the computation of personalized indicators for each bidder results in a high demand for bandwidth and computation ($\mathcal{O}(n^2k)$ for each bidder). To give an example, in an auction with hundred bidders ($n = 100$) and 200 possible prices ($k = 200$)[12], each bidder has to compute and publish hundreds of megabytes of data when p and q are 1024-bit primes.

Correctness

vMB-SHARE computes the outcome of an $(M + 1)$st-price auction. As the protocol is publicly verifiable, malicious bidders that do not follow the protocol will be detected immediately and can be excluded from the set of bidders which makes the protocol *weakly robust*, except in the case of ties. When two or more bidders have the $(M + 1)$st highest bid in common, the protocol yields no winners at all. There is no information revelation in this case, except that there has been a tie. Items could be re-auctioned in another auction, in which bidders slightly change their bids to avoid ties. Nevertheless, tieing bidders remain an important problem and Section 6.7.3 proposes three methods to resolve ties.

Privacy

The final ring exponentiation steps are conducted in a way that allows the seller to see all indication vectors before bidders can compute them. This prevents a bidder from aborting the protocol after having learned the auction result. vMB-SHARE is fully private in the computational model due to

[12] Usually the number of different prices or valuations is much lower than one would expect, e.g., Lipmaa et al argue that $k \leq 500$ is sufficient for most auctions [LAN02].

exponentiated vector components. An unbounded adversary, however, can reveal all bids.

When the selling price does not need to be protected, the computational complexity can be reduced to $\mathcal{O}(nk)$ by just computing *one* indication vector that indicates the selling price w to all bidders (*public price mode*).

$$\vec{v} = \left((2\mathsf{L} - \mathsf{I}) \sum_{i=1}^{n} \vec{b}_i - (2M+1)Y\vec{e} \right) \mathsf{R}^* \tag{6.9}$$

Winning bidders can prove their claims to the seller by providing t, so that $E_{i,w+1,0} = g_1^Y g_2^t$. However, winning bidders are able to remain silent if they dislike the selling price like in YMB-SHARE. In public price mode, a single unbounded adversary can only read bid statistics, i.e. all bid amounts, but no relation on who bid which amount. $n-1$ unbounded bidders can reveal all information.

6.7 Protocol vX-share

vMB-share fulfills our demands for correctness and privacy: It is weakly robust and fully private. However, it is not very efficient as it takes n rounds of interaction and because the computational amount per bidder is quadratic in n. The following protocol is based on the same ideas presented in Section 6.6.2. The major difference is that it is based on distributed homomorphic probabilistic encryption (ElGamal) instead of verifiable secret sharing. Homomorphic encryption gained much attention in the context of voting and recently in general multiparty computation [CDN01].

6.7.1 ElGamal Cipher

Instead of computing on distributed shares of secret values, it would be nice to be able to compute on encrypted values directly, thus drastically reducing complexity as this would result in only two operations that require the cooperation of all participants:

- the joint generation of a generally known public key and a shared private key, and

- the joint decryption of the result.

In order to be able to compute on encrypted values like on shared secrets, a cryptosystem needs to be homomorphic.

Definition 6.6 (Homomorphic Encryption)
A cryptosystem is called *homomorphic* if there is an efficient (polynomial-time) algorithm that computes an encryption of $a \circ b$ from encryptions of a and b without revealing a or b. $\circ$ denotes an arbitrary algebraic operation, e.g. addition, in the plaintext space.

A problem that arises when multiparty computation is based on homomorphic encryption is that adversaries can try to "guess" secret values by encrypting likely values with the public key and comparing these ciphertexts with the secret's ciphertext. As a matter of fact, *deterministic* public-key cryptosystems always leak information. In particular, these cryptosystems are vulnerable to chosen-plaintext attacks. Let us consider a homomorphic encryption based version of the previous protocol vMB-share. Bid values could easily be identified because each encryption of Y in the bid vector has the same ciphertext. In a *probabilistic* cryptosystem, the encryption algorithm is probabilistic rather than deterministic. A large number of different ciphertexts will decrypt to the same plaintext. As a consequence, the ciphertext space has to be larger than the plaintext space. The most important property of a probabilistic encryption scheme is semantical security.

Definition 6.7 (Semantically secure probabilistic encryption)
A probabilistic cryptosystem is called *semantically secure* if it is impossible to distinguish between the encryptions of any two given messages in polynomial time.

ElGamal cipher [ElG85] is a probabilistic public-key cryptosystem. p and q are large primes so that q divides $p-1$. $\mathbb{G}_q$ denotes $\mathbb{Z}_p^*$'s unique multiplicative subgroup of order q. The *private key* is $x \in \mathbb{Z}_q$, the *public key* $y = g^x$ ($g \in \mathbb{G}_q$). A message $m \in \mathbb{G}_q$ is *encrypted* by computing the ciphertext tuple

$$(\alpha, \beta) = (my^r, g^r) \tag{6.10}$$

where r is an arbitrary number in $\mathbb{Z}_q$. A message is *decrypted* by computing

$$\frac{\alpha}{\beta^x} = \frac{my^a}{(g^a)^x} = m \quad . \tag{6.11}$$

Proposition 6.3 (ElGamal homomorphicity)
The ElGamal cryptosystem is homomorphic.

Proof: The component-wise product of two ciphertexts $(\alpha\alpha', \beta\beta') = (mm'y^{r+r'}, g^{r+r'})$ represents an encryption of the plaintexts' product mm'. □

Theorem 6.1 (ElGamal semantical security)
The ElGamal cryptosystem is semantically secure (assuming the intractability of the decisional Diffie-Hellman problem, see Appendix A.1).

Proof: See [TY98] □

We will now describe how to apply the ElGamal cryptosystem as a fully private, i.e. non-threshold, multiparty computation scheme.

Distributed key generation: Each participant chooses x_{+i} at random and publishes $y_{\times i} = g^{x_{+i}}$ along with a zero-knowledge proof of knowledge of $y_{\times i}$'s discrete logarithm. The public key is $y = \prod_{i=1}^{n} y_{\times i}$, the private key is $x = \sum_{i=1}^{n} x_{+i}$. The broadcast round complexity and the computational complexity of the key generation are $\mathcal{O}(1)$.

Distributed decryption: Given an encrypted message (α, β), each participant publishes $\beta_{\times i} = \beta^{x_{+i}}$ and proves its correctness. The plaintext can be derived by computing $\frac{\alpha}{\prod_{i=1}^{n} \beta_{\times i}}$. Like the key generation, the decryption can be performed in constant time.

Random Exponentiation: A given encrypted value (α, β) can easily be raised to the power of an unknown random number $M = \sum_{i=1}^{n} m_{+i}$ whose addends can be freely chosen by the participants if each bidder publishes $(\alpha^{m_{+i}}, \beta^{m_{+i}})$ and proves the equality of logarithms. The product of published ciphertexts yields (α^{M}, β^{M}). Random Exponentiation can thus be executed simultaneously with distributed decryption in a single step. Random exponentiation was the bottleneck of VMB-SHARE (ring exponentiation).

6.7.2 Formal Description

What follows is the step-by-step protocol specification for bidder a. $i, h \in [n]$, and $j, b_a \in [k]$. $Y \in \mathbb{G}_q \backslash \{1\}$ is known to all bidders.

Share key / Publish encrypted bid

1. Choose $x_{+a} \in \mathbb{Z}_q$ and $m_{ij}^{+a}, r_{aj} \in \mathbb{Z}_q$ for each i and j at random.
2. Publish $y_{\times a} = g^{x_{+a}}$ along with a zero-knowledge proof of knowledge of $y_{\times a}$'s discrete logarithm. Compute $y = \prod_{i=1}^{n} y_{\times i}$.
3. Set $b_{aj} = \begin{cases} Y & \text{if } j = b_a \\ 1 & \text{else} \end{cases}$ and publish $\alpha_{aj} = b_{aj} y^{r_{aj}}$ and $\beta_{aj} = g^{r_{aj}}$ for each j.
4. Prove that $\alpha_{aj} \in \{Y y^{r_{aj}}, y^{r_{aj}}\}$ for each j and $\displaystyle\prod_{j=1}^{k} \alpha_{aj} = Y y^{r_a}$.

Compute and decrypt outcome

5. Compute $\gamma_{ij} = \dfrac{\prod_{h=1}^{n} \prod_{d=j}^{k} (\alpha_{hd}\alpha_{h,d+1}) \left(\prod_{d=1}^{j} \alpha_{id}\right)^{2M+2}}{(2M+1)Y}$ and $\delta_{ij} = \displaystyle\prod_{h=1}^{n} \prod_{d=j}^{k} (\beta_{hd}\beta_{h,d+1}) \left(\prod_{d=1}^{j} \beta_{id}\right)^{2M+2}$ for each i and j.
6. Send $\gamma_{ij}^{\times a} = (\gamma_{ij})^{m_{ij}^{+a}}$ and $\delta_{ij}^{\times a} = (\delta_{ij})^{m_{ij}^{+a} x_{+a}}$ for each i and j with a proof of their correctness to the seller who publishes all $\gamma_{ij}^{\times h}$ and $\delta_{ij}^{\times h}$ and the corresponding proofs of correctness for each i, j, and $h \neq i$ after having received all of them.

Outcome determination

7. Compute $v_{aj} = \dfrac{\prod_{i=1}^{n} \gamma_{aj}^{\times i}}{\prod_{i=1}^{n} \delta_{aj}^{\times i}}$ for each j.
8. If $v_{aw} = 1$ for any w, then bidder a is a winner of the auction. p_w is the selling price.

Like in vMB-SHARE, the final steps are conducted in a way that allows the seller to assemble all decrypted indication vectors before the bidders can compute them. This prevents a bidder from aborting the protocol after having learned the auction result. Alternatively, a sub-protocol that enables *fair exchange of secrets* (see e.g. [Yao82]) could be used while including the seller into the secret sharing process.

6.7.3 The Problem of Equal Bids

When two or more bidders have the $(M+1)$st highest bid in common, vX-share (and vMB-share) yield no winners. There is no information revelation in this case, except that there has been a tie on the $(M+1)$st highest bid. However, this might be used by a group of malicious bidders who submit equal bids on purpose to learn about the selling price. If the tie is undetected, their bids were lower than the selling price. If the protocol fails, their bids were at least as high as the selling price would have been (without their participation). Besides, ties can be used to impair the protocol's robustness, as tieing bidders can anonymously disrupt the auction. In the following, we will discuss three different methods to circumvent the tie problem. The first two avoid ties while the last one identifies ties.

"Interlacing" Vector Components (Int)

A straightforward way to avoid the problem is to increase the number of components in $\vec{v}_i$ from k to nk and insert bidder i's bid in row $nj+i-1$. This increases the computational complexity to $\mathcal{O}(n^2k)$. Unfortunately, this method reveals the identity of one of the $(M+1)$st highest bidders to the winners.

Preventing Equal Bids (Pre)

Bid amount b_i can be computed from bid vector $\vec{b}_i$ by summing up the components of $\mathsf{L}\vec{b}_i$. The equality of bids can be detected by computing $(b_i - b_h)M_{ih}$ for each pair of bids, requiring $\frac{n^2-n}{2}$ comparisons. When equal bids have been detected, k extra rows might be inserted similar to the previous technique. As $n<k$ in most reasonable auction settings, the computational complexity per bidder remains $\mathcal{O}(nk)$ when bids are pairwise different. The exact complexity is $\mathcal{O}(nkT)$, where T is the maximum number of equal bids. This technique is usually less complex than the previous one (they are equally complex for the extreme case when all bids are equal). Due to the revelation of equal bids, there is no incentive for malicious bidders to use ties on purpose anymore. However, malicious bidders can try to "guess" bids, i.e., they submit numerous different bids and hope for ties, because ties reveal opponents'

bids. Moreover, bidders are able to leave the protocol *after* having learned some information (about equal bids).

Determining Ties (Det)

Instead of trying to avoid ties, we can locate the position of ties. As mentioned before, ties only inhibit the protocol when they occur at the $(M+1)$st-highest bid. For this reason, "bad" ties always indicate the selling price. The following method marks ties if they prevent the regular protocol from working. $\sum_{i=1}^{n}\vec{b}_i - t\vec{e}$ is a vector that contains zeros if t bidders share the same bid at the corresponding position $(1 < t \leq n)$. "Good" ties can be masked by adding $(n+1)\left(\mathsf{L}\sum_{i=1}^{n}\vec{b}_i - (t+u)\vec{e}\right)$ where $0 \leq u \leq M$ and $M+1 \leq t+u \leq n$. The resulting vector contains a zero when t bids are equal and there are u bids higher than the tie. The preceding factor $(n+1)$ is large enough to ensure that both addends do not add up to zero. Finally, the position of the tie (which is the selling price) has to be made invisible to losing bidders like in Section 6.6.2. This can be done by adding $(n^2+2n+1)(\mathsf{U}-\mathsf{I})\vec{b}_a$. Concluding, this method requires the computation of additional indication vectors

$$\begin{aligned}\vec{v}'_{atu} &= \left(\sum_{i=1}^{n}\vec{b}_i - t\vec{e} + (n+1)\left(\mathsf{L}\sum_{i=1}^{n}\vec{b}_i - (t+u)\vec{e}\right)\right.\\ &\quad \left. +(n^2+2n+1)(\mathsf{U}-\mathsf{I})\vec{b}_a\right)\mathsf{R}^*_{atu} = \\ &= \left((\mathsf{L}+(n+1)\mathsf{I})\sum_{i=1}^{n}\vec{b}_i - \left(nt+nu+2t+u\right)\vec{e}\right.\\ &\quad \left. +(n^2+2n+1)(\mathsf{U}-\mathsf{I})\vec{b}_a\right)\mathsf{R}^*_{atu} \quad ,\end{aligned} \tag{6.12}$$

which increases the overall computational complexity to $\mathcal{O}(n^2kM)$. Information revelation is low compared to the previous two methods. Winning bidders learn that the selling price was shared by t bidders and that there were u higher bids. Losing bidders do not learn anything. In contrast to the previous two methods not a single bid origin, i.e. a bidder's identity, is uncovered.

Example: Suppose we have the following compilation of bids ($M = 1$, computation takes place in $\mathbb{Z}_{11}$):

$$\vec{b}_1 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \vec{b}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \vec{b}_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad \text{and} \quad \vec{b}_4 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad .$$

The first two ($t = 2, u \in \{0, 1\}$) indication vectors look like this (before being masked for each bidder):

$$\begin{pmatrix} 0 \\ 2 \\ 0 \\ 2 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \end{pmatrix} + 5 \left(\begin{pmatrix} 0 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \end{pmatrix} - \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \end{pmatrix} \right) = \begin{pmatrix} 10 \\ 0 \\ 9 \\ 10 \\ 8 \\ 8 \end{pmatrix} \xrightarrow{\times R^*_{1,2,0}} \begin{pmatrix} \cdot \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}$$

$$\begin{pmatrix} 0 \\ 2 \\ 0 \\ 2 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 2 \end{pmatrix} + 5 \left(\begin{pmatrix} 0 \\ 2 \\ 2 \\ 4 \\ 4 \\ 4 \end{pmatrix} - \begin{pmatrix} 3 \\ 3 \\ 3 \\ 3 \\ 3 \\ 3 \end{pmatrix} \right) = \begin{pmatrix} 5 \\ 6 \\ 4 \\ 5 \\ 3 \\ 3 \end{pmatrix} \xrightarrow{\times R^*_{1,2,1}} \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}$$

For $t > 2$ the first difference contains no zeros, leading to random vectors.

6.7.4 Analysis

Like vMB-share, vX-share is *weakly robust* and fully private in the computational model. However, vX-share fulfills these demands in a constant number of rounds and consumes much less resources (bandwidth and computation). Like described in Section 6.6.1, the interactive proofs of knowledge can be made non-interactive. As a consequence, the entire execution of vX-share needs just three rounds of interaction. Figure 6.5 illustrates the modus operandi of the protocol (garbled characters on the blackboard represent encrypted information). The seller broadcasts the type of good to be sold, the number of units, a deadline, and the bid function. Interested bidders then have the chance to publish their id's before the deadline expires.

In the following, each bidder broadcasts two encrypted messages and sends one encrypted message to the seller who broadcasts the encrypted result, so that only winning bidders are able to read it.

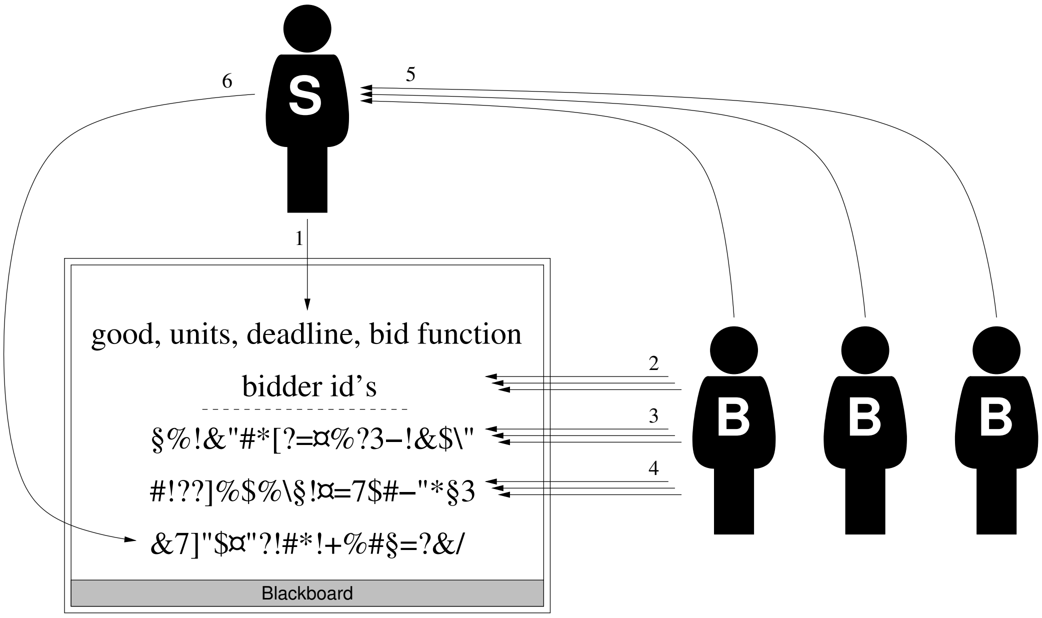


Figure 6.5: vX-share

Three methods to provide robustness in the case of ties, that can also be used in vMB-share, have been presented. In a nutshell, vX-share achieves the lowest round complexity and the highest level of privacy[13] and robustness possible in bidder-resolved auctions.

Computational complexity can be reduced by computing just one indication vector for all bidders and publicly announcing the selling price. In the previous protocol vMB-share, the main problem in public price mode is that the seller is unable to identify who won the auction. This allows a winning bidder to repudiate the auction outcome. The encryption-based approach of vX-share enables the computation of an additional outcome vector that indicates the selling price *and* the winners' identities (to the seller). Let Y_i be an arbitrary, publicly known id code of bidder i, so that $\forall i, h \in [n] : \; Y_i \neq Y_h$ and

$$\sum_{1 \leq m_1 < m_2 < \cdots < m_i < \cdots < m_M \leq n} Y_{m_i} \neq \sum_{1 \leq n_1 < n_2 < \cdots < n_i < \cdots < n_M \leq n} Y_{n_i}$$

[13] when disregarding the negligible tie information revelation

(using the additive notation of Section 6.6.2).

Besides vector $\vec{v}$ specified in Equation 6.9 on page 125, bidders jointly compute vector $\vec{w}$.

$$\vec{w} = \left((2\mathsf{L} - \mathsf{I}) \sum_{i=1}^{n} \vec{b}_i - (2M+1) Y \vec{e} \right) \mathsf{R}'^{*} + (\mathsf{L} - \mathsf{I}) \sum_{i=1}^{n} Y_i \vec{b}_i \qquad (6.13)$$

Vector $\vec{w}$ contains the sum of all winners' id's at the position of the selling price with negligibly small error probability (because $\binom{n}{M}$ is much smaller than the (exponential) number of group elements). It is essential that the seller broadcasts a proof of the selling price ($\vec{v}$) because this proves to losing bidders that they lost. The identity of the winner, however, remains confidential.

6.8 Other Protocols

It was mentioned in Section 5.1 that parallel to our work, numerous cryptographic auction protocols have been proposed in the literature. In the following, a selection of these protocols will be briefly discussed. The list of protocols is certainly not complete. They have merely been selected to illustrate the variety of solutions to a common problem. The main difference between these protocols and the ones presented in this thesis is that privacy can be breached by a collusion of trusted third-parties in all of them. For this reason, we specify the type of collusion that leads to information revelation for each protocol.

6.8.1 Naor, Pinkas, & Sumner 1999

The scheme by Naor et al [NPS99] is based on two servers (the auctioneer and the "auction issuer") and is applicable to general secure Boolean two-party computation. The auction issuer generates a "garbled" Boolean circuit that computes the auction outcome for any given set of bids. The bidders then submit their encrypted bids. The auction issuer produces garbled inputs to the circuit from the bids and sends them to the auctioneer who then evaluates the circuit and publishes the result.

The basic scheme is very efficient as it works on binary representations of bids. The size of a Boolean circuit that computes the outcome of an $(M+1)$st-price auction is $\mathcal{O}((n+M)\log k)$. However, the auctioneer needs to verify that the auction issuer's garbled circuit is correct. This is solved by the "cut-and-choose" technique in which the auction issuer provides several copies of the garbled circuit out of which the auctioneer chooses some to be opened and verified. The remaining circuits are used to resolve the auction and it is checked whether they produce the same output. This expensive method can never guarantee the correctness of the circuit, but it can provide exponentially small error probability. Although this scheme is based on a sophisticated trust model (e.g., "the auction issuer is typically an established party such as a financial institution or large company, which supplies services to numerous auctioneers" [NPS99]), privacy vanishes if the auction issuer and the auctioneer collude.

Juels and Szydlo [JS02] removed a critical security flaw in the original protocol and based their version on RSA which results in less computational complexity for the bidders but even more complexity for the auction servers.

Privacy can be invaded by: auctioneer and auction issuer collusion

6.8.2 Sako 2000

Sako's protocol [Sak00] uses a probabilistic encryption scheme and a list of k possible bids to implement a 1^{st}-price sealed-bid auction. There is a number of auctioneers that generate k values M_i and k public/private key pairs E_i and D_i. The public keys and all M_i are published. In the bidding phase each bidder publishes M_{b_i} encrypted with public key E_{b_i} where b_i denotes bidder i's bid. Thus, even though the scheme works on linear lists of valuations, each bidder only needs to submit a single encrypted value. The auctioneers then jointly decrypt all bids with the private key belonging to the highest valuation D_k. If none of the values decrypts to M_k, the auctioneers try the key belonging to the next valuation. This step is repeated until one of the bids correctly decrypts to M_i. The corresponding bidder is the winner and i refers to the selling price. The author gives two examples of the proposed scheme based on ElGamal and RSA encryption, respectively. Basing the scheme on RSA has the advantage that no list containing M_i, E_i, and D_i needs to be published as those values can be derived from i. On the other hand, semantical security and other required properties are not proven for RSA and the joint generation of RSA keys is very complicated.

Clearly, the scheme has the strong advantage of minimal bidder effort. Bidders just submit one encrypted value and do not need to participate any further. However, the "Dutch auction style" approach makes it only applicable to 1^{st}-price auctions with very little hope of a possible generalization for other auction types like Vickrey auctions. Additionally, the auctioneers need $\mathcal{O}(k)$ rounds to determine the highest bid and this bid is publicly revealed.

Privacy can be invaded by: auctioneer collusion

6.8.3 Baudron & Stern 2001

The Baudron & Stern scheme [BS01] uses a semi-trusted third-party that does not learn any information if it does not collude with a bidder. The scheme is quite complex and is based on the joint evaluation of a special-purpose Boolean circuit with the help of a third-party. Bidders encrypt each bit of the binary representations of their bids n times with each bidder's public key of a homomorphic cryptosystem. In the following, each logical gate of a Boolean circuit that computes the auction outcome is blindly evaluated by the third-party with assistance by bidders. This process is optimized by taking into account the level of Boolean gates in the special-purpose circuit. The result's size is exponential in the number of bidders ($\mathcal{O}\left(n(\log k)^{n-1}\right)$ where n is the number of bidders and k the number of possible prices) which makes the scheme only applicable to a very limited number of bidders (four to five, as stated in [BS01]). After the result is broadcasted, the winner is required to claim that he won (violating non-repudiation). When computing the outcome of a Vickrey auction, additional interaction is required to compute the second highest bid. The bidders' actions are verifiable. However, it is not possible to verify if the third-party behaves correctly.

Privacy can be invaded by: third-party and bidder collusion

6.8.4 Kikuchi 2001

The protocols by Kikuchi [Kik02, Kik01] make use of a clever idea. They are based on polynomial secret sharing as explained in Section 6.6.1. However, the secret (one out of k bids) is hidden in the *degree* of a polynomial. Computation is distributed among m auctioneers out of which less than c can not disclose any information. As the bids are hidden in the degree of shared polynomials, this implies that $m > k + c$. The author proposes a protocol for 1^{st}-price auctions and for the more general $(M + 1)$-st price auctions. The 1^{st}-price protocols exploit the fact that the sum of two polynomials results in a polynomial whose degree is the maximum degree of the input polynomials. This enables quite efficient auction protocols. The computational complexity of bidders in the 1^{st}-price auction protocol is only $\mathcal{O}(m)$ and auctioneers need just $\mathcal{O}(n)$ rounds to determine the winner. Bidders' actions are publicly verifiable. However, this does not hold for auctioneers. They can manipulate the auction outcome without being detected. This becomes even worse in the $(M+1)$-st price auction protocols. Either the actions of auctioneers *and* bidders are not verifiable (and thus not robust against active adversaries), or the bids of all M winners are publicly revealed. Moreover, one of the protocols can not handle ties and, generally speaking, the encoding of secrets in the degree of a polynomial poses several problems. The number of auctioneers m is required to be greater than the number of possible prices k. As a result, the complexity of the protocols is higher than it seems because there have to be umpteen auctioneers. Even though the proposed protocols are not based on any threshold assumptions, the underlying technique can not be used for bidder-resolved ($m = n$) auction protocols because full privacy implies $c = n$ and this yields $k < 0$.

Privacy can be invaded by: auctioneer collusion

6.8.5 Abe & Suzuki 2002

Abe and Suzuki's scheme [AS02a] is similar to vX-SHARE, but differs in using a technique called "mix-and-match" [JJ00] and in a trust model with two third-parties: the auctioneer and a "trusted authority" (which can be distributed to achieve threshold security). It is based on a homomorphic cryptosystem like ElGamal [ElG85] or Paillier [Pai99]. Bidders encrypt bidding vectors that are defined as in Equation 6.6.2 on page 117 with a public key of the trusted authority, send them to the auctioneer, and prove their correctness. The auctioneer computes the sum of integrated bid vectors based on the homomorphic property of the cryptosystem. The resulting vector's components denote how many bidders are willing to pay a given price. In the following, the position of the $(M + 1)$st highest bid is determined by binary searching the lowest price that exactly M bidders are willing to pay. This is achieved by gradually releasing vector components to the authority who decrypts them and proves in zero-knowledge (using mix-and-match) that there were either more than M bidders or less than $M + 1$ bidders willing to pay. The whole process takes $\log k$ rounds where k is the number of possible bids. Obviously, the authority learns statistical information during this process. The entire protocol is publicly verifiable and thus achieves robustness. Apparently, a collusion of the auctioneer and the trusted authority can learn complete information. Furthermore, the selling price must be publicly announced to convince losing bidders of their failure[14].

Privacy can be invaded by: auctioneer and trusted authority collusion

[14] When distributing the mix-and-match technique on bidders in order to realize a bidder-resolved protocol with a private-key shared among bidders and discarding binary search to minimize the round complexity, mixing [Abe99] would require n rounds. The computational and message complexity per bidder would be $\mathcal{O}(kM \log(M))$ with no need for further efforts to resolve ties. The drawbacks of such a scheme would be the public announcement of the selling price and n rounds of bidder interaction.

6.8.6 Lipmaa, Asokan, & Niemi 2002

The protocol by Lipmaa et al [LAN02] requires a single semi-trusted third-party: the auction authority. Bidders encrypt their bids using the auction authority's public key and send them to the seller who checks accompanying signatures, sorts the encrypted bids according to a pre-determined scheme (e.g. in lexicographic ciphertext order), and broadcasts them. The auction authority then opens all bids, determines the selling price (e.g. the second highest bid), sends it to the seller, and proves its correctness by applying a novel, sophisticated zero-knowledge proof. Winning bidders are required to claim that they won (violating non-repudiation). The protocol is very efficient, but only provides limited privacy as the selling price is published and the auction authority learns *all* bids. The only information hidden from the authority is the connection between bidders and bids. The scheme easily scales to a high number of bidders as $k \log n < \log |\mathcal{M}|$ where n is the number of bidders, k the number of possible bids, and $\mathcal{M}$ the message space of the applied cryptosystem. However, the number of possible bids k is severely limited. Neither the seller nor the auction authority can manipulate the outcome without being detected. A collusion of both instances uncovers complete information.

Privacy can be invaded by: seller and auction authority collusion

Chapter 7

Conclusion

The contribution of this thesis is two-fold. In the first part, a particular novel form of strategic bidding behaviour was introduced and analyzed, whereas the main part identified privacy problems in auctions (and mechanisms in general) and presented solutions to these problems.

There are several known types of deceptive behaviour in auctions, such as bidder collusion, shills, or sniping. We extended this list by introducing so-called antisocial agents, i.e. agents who intend to maximize the difference of their profit and the profit of other agents. When bidding in Vickrey auctions, antisocial agents are *not* best off bidding their true valuation of the good to be sold. The truth-revealing dominant-strategy equilibrium does not hold anymore and there is no other dominant-strategy equilibrium, except for purely destructive agents. An antisocial agent's optimal strategy depends on the highest and second highest private value of participating agents. As other bidders' private values are unknown to an antisocial bidder, we proposed a Bayesian Nash equilibrium strategy that assumes that values are uniformly distributed in a given interval, and a strategy that learns the desired private values in a multiagent task-allocation scenario where identical tasks are auctioned off in sequential reverse auctions. The latter strategy was successfully evaluated in an experimental implementation. A third possibility to acquire private values would be to buy this information from the auctioneer. However, the cryptographic protocols presented in the second part of this dissertation allow the execution of Vickrey auctions without revealing confidential information to an auctioneer.

In the following, we investigated how the presence of antisocial agents can be incorporated in the design of an auction mechanism. More precisely, there might be a mechanism in which bidders submit their valuations and their degree of antisociality, i.e. their derogation rate, in equilibrium and that yields a socially desirable outcome. As a matter of fact, the aggregation of *conflicting* preferences is the purpose of mechanism design. However, we have proven the impossibility of such a mechanism under quite reasonable assumptions. For the future, it might be interesting to analyze the concept of antisocial agents in other application scenarios than auctions.

In the second part, novel cryptographic auction protocols where bidders jointly compute the outcome were presented. In contrast to all existing auction protocols, distributing the outcome determination on bidders enables what we call full privacy, i.e. bids can not be revealed despite any collusion of participants. The protocols make differing tradeoffs across (unconditional) privacy, robustness and efficiency (see Table 7.1). The most advanced protocol, vX-SHARE, requires just a constant number of broadcasting rounds. The price we pay for low round complexity is computational complexity that is linear in the number of possible bids. However, experimental results (see Appendix B.2) indicate that the computational amount and message sizes are manageable in many realistic settings, despite its linearity in k.

It is possible to apply variations of the proposed protocols to securely emulate ascending auctions, e.g. English auctions in which the identity of the current highest bidder is hidden. This might be useful to generate more revenue than in sealed-bid auctions when the common- or correlated-value model is appropriate. If desired, e.g. to achieve *strong* robustness, the presented protocols can be distributed on two (or more) auctioneers instead on bidders. The advantages would be minimal bidder effort and robustness. On the other hand, privacy could only be guaranteed if the auctioneers do not collude.

From a more abstract point of view, vX-SHARE is an interaction protocol that provides a solution to the problem stated at the beginning of Chapter 1: An agent possessing an object[1] of undetermined value is willing to sell this object. There is a group of agents that have differing valuations of the good. The problem is: To whom does he sell the good, for what price, and how can agents determine this information without disclosing their valuations? vX-SHARE provides various desirable properties[2].

Social-welfare-maximization The sum of all agents' utilities is maximized. As a consequence, the good is sold to the agent that values it the most.

[1] As a matter of fact, vX-SHARE can also be applied to scenarios in which the seller has several identical units for sale.

[2] We assume quasilinear utility functions, the private value model, and the existence of trapdoor one-way permutations. Obviously, some properties hold because vX-SHARE emulates the Vickrey auction [Vic61].

Individual rationality No agent will have a disadvantage from participating in the protocol. This now holds even with respect to information secrecy (the revelation of private information might be a disadvantage).

Strategy-proofness There is no need for strategic behaviour by any of the agents. They are best off by simply submitting their true valuations.

Full privacy No unnecessary information on any of the agents' valuations is revealed. The seller learns the identity of the buyer and both get to know the selling price[3]. There is *no* third-party that learns any information.

Correctness Every participant can verify that the outcome is determined correctly. There is no risk of an insincere auctioneer. Together with full privacy, this removes the two, generally accepted, major drawbacks of Vickrey auctions (see Chapter 5).

Efficiency Each agent just needs to compute and broadcast *three* messages, independently of the number of participants or possible prices (see Figure 6.5).

Besides secure auction protocols, the most far-reaching contribution of this thesis is the establishment of a general connection between preference aggregation and secure multiparty computation (MPC). Starting from the three-layer model depicted in Figure 1.2, we analyzed the feasibility of executing mechanisms *without* a trusted mechanism infrastructure. We have shown that this is generally possible when allowing intractability assumptions (such as that a discrete logarithm is hard to compute). In the unconditional model, when intractability assumptions can *not* be made, only very limited mechanisms can be emulated by cryptographic protocols. Furthermore, the unconditional model requires a broadcast channel and allows manipulation by agents that have unlimited computing power. Of course, such agents are not likely to exist. But nevertheless the unconditional model is important, because it ensures that preferences of participants will *never* be revealed, something that can not be guaranteed in the cryptographic model due to constantly increasing computing power[4].

On the basis of these observations and taking into account that there is (yet) no efficient *general* MPC protocol, there is a broad spectrum of future work. In particular, it would be worthwhile to construct protocols

[3] Negligible information is revealed to the winner if there are certain ties.

[4] It is debatable whether the revelation of votes of an election that happened decades ago can pose a problem.

		Revelation to Adversaries		weakly			Bandw./
Protocol	Price	bounded	unbounded[1]	robust	Rounds	Msgs.	Comp.
Dutch	1st	price	—	yes	$\mathcal{O}(k)$	$\mathcal{O}(1)$	$\mathcal{O}(1)$
`dbs`	2nd	$b_{h_1(b)}$	—	yes	$\mathcal{O}(nk)$	$\mathcal{O}(k)$	$\mathcal{O}(k)$
`ubs`	2nd	part. info.	—	yes	$\mathcal{O}(n+k)$	$\mathcal{O}(k)$	$\mathcal{O}(k)$
`bbs`	2nd	part. info.	—	yes	$\mathcal{O}(nk)$	$\mathcal{O}(k)$	$\mathcal{O}(k)$
B-share	1st	$b_{h_{c+1}(b)}$	—	no	$\mathcal{O}(1)$	$\mathcal{O}(n)$	$\mathcal{O}(nk)$
MB-share	1st	price	$b_{h_{c+1}(b)}$	no	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(nk)$
YMB-share	2nd	—	$b_{h_1(b)}$	no	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n^2k)$
vMB-share$^{\text{INT}}$	uniform	[a]	everything	yes	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n^3k)$
vMB-share$^{\text{PRE}}$	uniform	[ab]	everything	yes	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n^2kT)$
vMB-share$^{\text{DET}}$	uniform	[c]	everything	yes	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n^3kM)$
vMB-share$^{\text{INT}}_{\text{pp}}$	uniform	price, [d]	bid stats.[g]	yes	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n^2k)$
vMB-share$^{\text{PRE}}_{\text{pp}}$	uniform	price, [db]	bid stats.[g]	yes	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(nkT)$
vMB-share$^{\text{DET}}_{\text{pp}}$	uniform	price, [e]	bid stats.[g]	yes	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$\mathcal{O}(n^2kM)$
vX-share$^{\text{INT}}$	uniform	[a]	everything	yes	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(n^2k)$
vX-share$^{\text{PRE}}$	uniform	[ab]	everything	yes	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(nkT)$
vX-share$^{\text{DET}}$	uniform	[c]	everything	yes	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(n^2kM)$
vX-share$^{\text{INT}}_{\text{pp}}$	uniform	price, [d]	everything	yes	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(nk)$
vX-share$^{\text{PRE}}_{\text{pp}}$	uniform	price, [db]	everything	yes	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(kT)$[f]
vX-share$^{\text{DET}}_{\text{pp}}$	uniform	price, [e]	everything	yes	$\mathcal{O}(1)$	$\mathcal{O}(1)$	$\mathcal{O}(nkM)$

Messages and Bandwidth/Computation are measured per bidder.
n: bidders, k: prices/possible bids, M: units to be sold, T: maximum number of equal bids
Index "pp" denotes public price mode. Int, Pre, and Det are three methods to resolve ties (see Section 6.7.3).

[1] besides the information already revealed to bounded adversaries
[a] tie at $(M+1)$st highest bid: reveals identity of one of $(M+1)$st highest bidders *to winners*
[b] reveals equals bids to corresponding bidders
[c] tie at $(M+1)$st highest bid: reveals number of tieing bidders and number of bidders with higher bids *to winners*
[d] tie at $(M+1)$st highest bid: reveals identity of one of $(M+1)$st highest bidders
[e] tie at $(M+1)$st highest bid: reveals number of tieing bidders and number of bidders with higher bids
[f] more precisely, the complexity is $\mathcal{O}(\max(kT, n^2))$
[g] all bid amounts, but no information on *who* bid which amount

Table 7.1: Overview of proposed protocols

that emulate the following example mechanisms *fully privately*, i.e. without any trusted infrastructure.

Combinatorial auctions Since we found protocols that emulate $(M+1)$st-price auctions, tractable instances of combinatorial auctions (e.g. multi-unit auctions or linear-good auctions, see Section 3.3.3) are the next logical step. There were recent efforts to design cryptographic protocols for general combinatorial auctions [YS02, SY02]. However, these approaches are not fully private and very inefficient.

Clarke tax mechanism The Clarke tax mechanism is applicable whenever a group of agents has to make a global decision and side-payments are possible. A classic example is the joint acquisition of a good such as a street-light for neighbouring residents, a shared TV set in an apartment, or the acquisition of a subsidiary by an association of companies. The first protocol to emulate the Clarke tax mechanism was proposed in [Bra03b].

Veto voting Such a protocol would only disclose *if* more than a fixed number of agents do not agree with a decision, but the identities of these agents would remain secret. Clearly, this can be very desirable in some scenarios.

Voting There are numerous voting procedures, such as majority voting, Borda count, or vote by approval, that would benefit when votes would not have to be revealed to a central institution. Moreover, voting protocols could only yield the final result, if wished. E.g., a majority voting protocol could only reveal *who* received the most votes but not disclose each candidate's vote percentage (or only the winner's if this is desirable).

Due to the universality of mechanism design, it was recently applied to algorithmic problems in a distributed setting, like finding the shortest path in a computer network, task scheduling, or determining a maximum independent set in a linear processor array [NR99]. *Decentralized* mechanisms that solve these problems fully privately are certainly desirable. Concluding, we believe that the proposed combination of cryptography and preference aggregation will lead to more preferable decision protocols in a variety of application fields.

Appendix A

Cryptographic Background

Since 1976, cryptography has undergone a major revolution. After the seminal paper by Diffie and Hellman [DH76], cryptography has grown from the science of encrypting messages to the field now known as *modern cryptography* [Gol97, BSW01, Sch96, MvOV96]. Modern cryptography is closely related to complexity theory and includes techniques such as public-key encryption, digital signatures, secret sharing, zero-knowledge proofs, multiparty computation, commitment schemes, and many more. This appendix will just briefly explain two concepts: discrete exponentiation as a candidate for a one-way function and commitment schemes.

A.1 The Discrete Logarithm Problem

One of the most vital concepts of cryptography is that of *one-way functions*. A one-way function is easy to compute, but hard to invert. In [Sch96], breaking a plate is given as an example of a one-way function in real life. It is easy to smash a plate into thousands of tiny pieces. However, it is much harder to put all pieces back together into a plate. Formally, f is a one-way function if there is a deterministic polynomial-time algorithm that computes $f(x)$ whereas there is no such algorithm to compute x from $f(x)$. As the existence of one-way functions would imply $\mathcal{P} \neq \mathcal{NP}$, no function has been proven to be one-way.

Besides the multiplication of large primes, exponentiation in certain groups is a common (conjectured) one-way function. Multiplicative (sub-) groups of finite fields[1] or elliptic curve groups over finite fields are candidates where the computation of the discrete logarithm is considered to be hard.

[1] Interestingly computing discrete logarithms in the Galois field $GF(2^n)$ seems to be substantially easier than in residue class groups $\mathbb{Z}_n$.

Figure A.1 visualizes that inverting the discrete exponential function might be hard.
In the multiplicative group of a finite prime field $\mathbb{Z}_p$, the discrete logarithm of $a^x \mod p$ can be easily computed when a has low order. We could require a to be a generator or at least an element with very high order, but this leaves the problem of finding such elements. Furthermore, it might be a problem that not all discrete logarithms in this group have a solution and that the product of two generators is not a generator as well. As a solution to these problems, it has become common practice to work in a multiplicative subgroup $\mathbb{G}_q$ of order q, where q is a large prime that divides $p-1$. Every element in $\mathbb{G}_q$ besides 1 generates the group, i.e., all elements have order q. $\mathbb{G}_q$ is called a "simple group". Elements of $\mathbb{G}_q$ can easily be found by computing $a^{(p-1)/q}$ for an arbitrary $a \in \mathbb{Z}_p^*$. A given element a is in $\mathbb{G}_q$ if $a^q \equiv 1 \pmod{p}$. When $p-1 = 2q$, then $\mathbb{G}_q$ is the group of quadratic residues in $\mathbb{Z}_p^*$.

Example: Let $p = 11$ and $q = 5$ (of course, in reality, the discrete logarithm problem is only hard for very large primes). An arbitrary element of $\mathbb{G}_q$ can be found by computing a^2 for an arbitrary $a \in \mathbb{Z}_p^*$, e.g. $2^2 = 4$. As 4 generates $\mathbb{G}_q$, $\mathbb{G}_q = \{4^0, 4^1, 4^2, 4^3, 4^4\} = \{1, 4, 5, 9, 3\}$

The discrete exponential function $f(n) = g^n$ can be efficiently computed by applying the *square-and-multiply* algorithm. This technique builds upon the fact that instead of multiplying g by itself n times, g^n can be computed by repeatedly squaring g and then multiplying g to the result. For example, the computation of $g^{10} = \left((g^2)^2\right)^2 \cdot g \cdot g$ needs five modular multiplications instead of ten when using the naive approach. Generally, the algorithm requires $\mathcal{O}(\log n)$ multiplications to compute g^n.

Numerous algorithms to compute discrete logarithms in finite groups have been proposed [Sch96]. The most important, besides the well-known *baby-step giant-step* algorithm (that requires $\sqrt{p}$ operations), is the *Silver-Pohlig-Hellman* algorithm. This algorithm's complexity is polynomial in q ($p-1$'s largest divisor). This is one of the reasons why we required q to be a *large* prime above.

Another reason is that the so-called *decisional Diffie-Hellman problem* is *not* intractable if the group order has small prime divisors. Hardness of the decisional Diffie-Hellman problem is essential for many security proofs (e.g. Theorem 6.1). The problem is to distinguish between the two distributions $\langle g^a, g^b, g^{ab} \rangle$ and $\langle g^a, g^b, g^c \rangle$ where a, b, c are elements of a cyclic group generated by g.

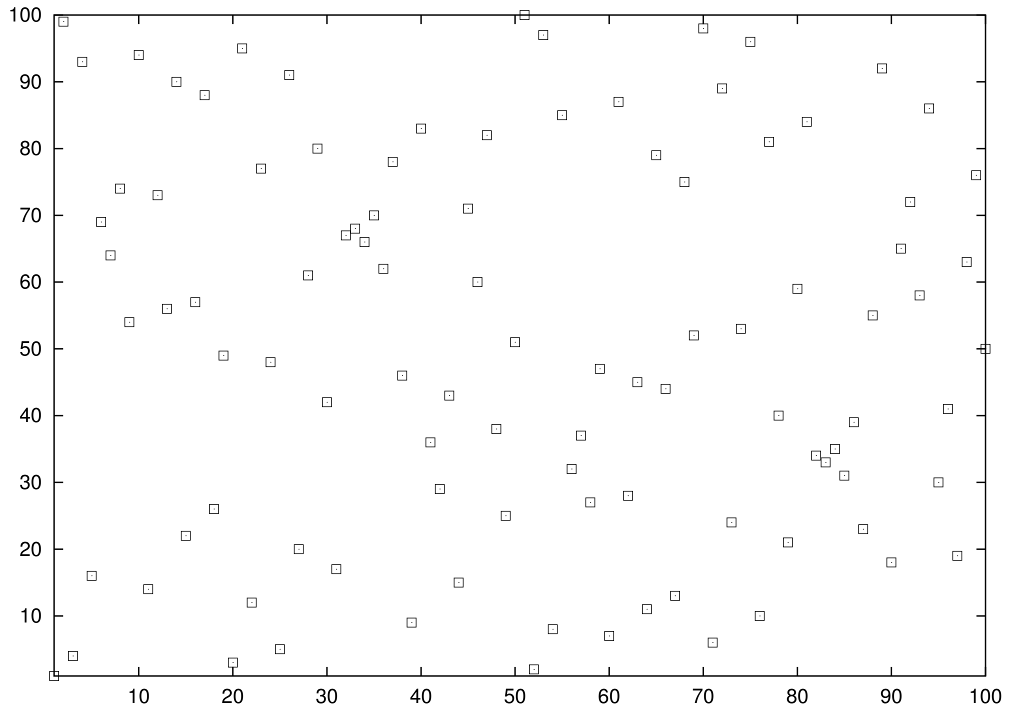


Figure A.1: Discrete exponential function: $f(n) = 99^n \mod 101$

A.2 Commitment Schemes

A Commitment scheme can be described as a sealable opaque envelope. The envelope's content cannot be changed once it has been sealed (even not by the owner) and cannot be read until the envelope is opened by the owner. Please note that conventional encryption techniques do *not* necessarily fulfill these demands because different keys could decrypt a given ciphertext into different plaintexts.

Commitment schemes are usually based on one-way functions like modular exponentiation or integer multiplication. An efficient way to implement *bit* commitment, i.e. commitment to a single bit, is to use a cryptographic one-way hash function $f(\cdot)$ (e.g. SHA-1 or MD5), compute and send $f(b|r)$ where b is the committed bit and r a random value. The commitment can be opened by sending $(b|r)$. Collision-freeness prevents a (computationally bounded) agent from changing his commitment.

There are commitment schemes that provide unconditional privacy of the committed value, i.e., even an computationally unbounded adversary is incapable of revealing the secret. However, the committed value can be forged by the committer if he possesses unbounded computational power. Generally, it has been shown that commitment schemes that are unconditionally binding *and* unconditionally private are impossible.

A well-known commitment scheme works as follows. Let $\mathbb{G}_q$ be a multiplicative group as described in the previous section. Let g be an arbitrary element of $\mathbb{G}_q$. An agent can commit to a secret $s \in \mathbb{G}_q$ by choosing $t \in \mathbb{Z}_q$ and publishing $E = sg^t$. The commitment value E reveals absolutely no information about the secret s (it is uniformly distributed in $\mathbb{G}_q$ for randomly uniformly chosen t). The agent can open the commitment by providing s and t. He is incapable of forging the commitment unless he can compute $\log_g s$ which is intractable for sufficiently large primes p and q.

Appendix B

Implementation

During the work on this dissertation, two software systems were implemented. They roughly relate to both parts of this thesis as the first system was implemented to investigate strategic behaviour in various auction types whereas the second system is an implementation of early cryptographic auction protocols presented in Chapter 6.

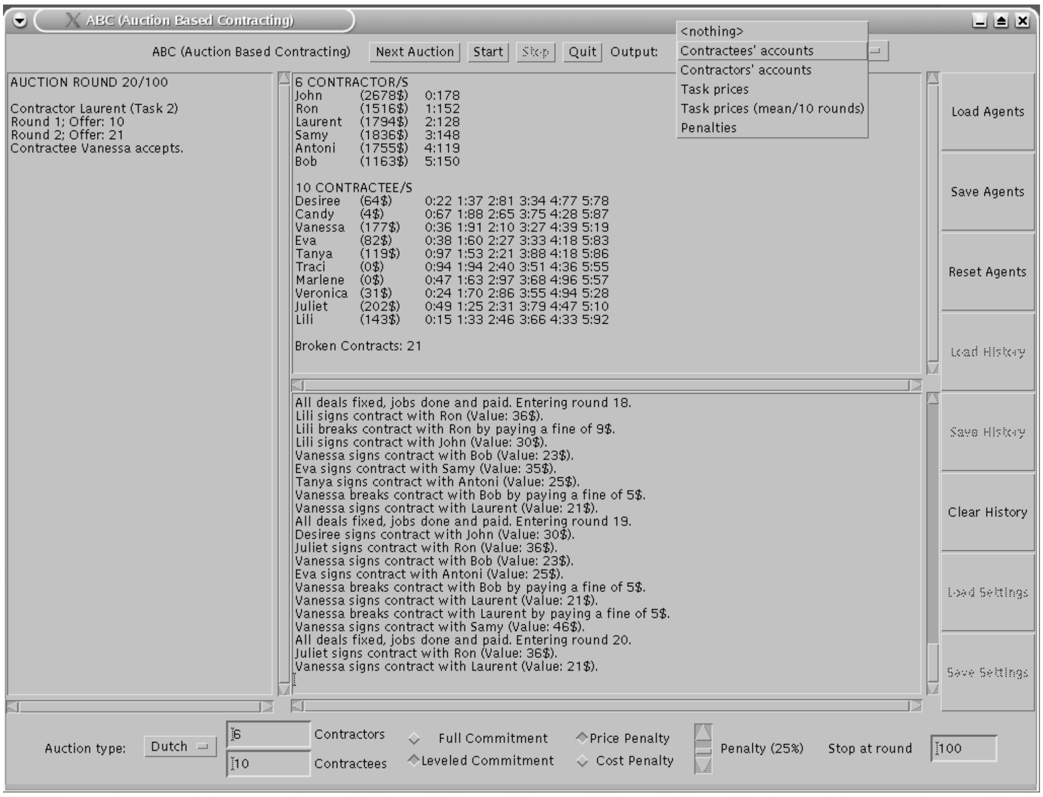
B.1 ABC (Auction-based Contracting)

ABC (see Figure B.1) is a multiagent task-assignment environment that has been extensively studied in [BW99, BW00a, BW00b, BBW00] and later has been used to evaluate antisocial agents [Bra00, BW01b, BW01a]. The basic framework works as follows. We consider a group of agents that contains two different types of business partners. *Contractors* CR_i $(i = 1, 2, \ldots, m)$ who offer a unique task i and *Contractees* CE_j $(j = 1, 2, \ldots, n)$ who are willing to execute tasks. A contractor CR_i is capable of executing task i by himself for his prime costs $C[CR_i]$. A contractee CE_j is able to do each task i for $C[CE_j, i]$. We assume that contractees can accomplish tasks cheaper than contractors by defining two intervals.

$$\begin{aligned} \forall i : C[CR_i] &\in [cr_{min}, cr_{max}] \\ \forall j, i : C[CE_j, i] &\in [ce_{min}, ce_{max}] \\ ce_{max} &\leq cr_{min} \end{aligned}$$

This ensures that both, contractors and contractees, are interested in signing contracts with each other. Pursuing conflicting goals, both types of agents are “true capitalists”: Contractors intend to pay the lowest feasible price for a task, while contractees try to earn as much money as possible.

An experiment consists of a fixed number of rounds. During each round, each contractor offers his task one after another, where the contractor sequence randomly varies from round to round. Applying an auction mechanism (English, Dutch, or Vickrey) the agents then come to an agreement which contractee will execute the task. A contractee is only able to accept *one* task per round. For this reason, two basic types of contract obligation are considered: *full commitment* (a contractee has to stay with the first deal he made) and *leveled commitment* (contractors can breach contracts by paying a fine to the concerning contractor CR_i). Different types of fines, e.g. a fraction of the contract value, are possible. Simple, adaptive strategies for different types of auctions were evaluated in this scenario.



```
AUCTION ROUND 20/100

Contractor Laurent (Task 2)
Round 1; Offer: 10
Round 2; Offer: 21
Contractee Vanessa accepts.
```

```
6 CONTRACTOR/S
John      (2678$)   0:178
Ron       (1516$)   1:152
Laurent   (1794$)   2:128
Samy      (1836$)   3:148
Antoni    (1755$)   4:119
Bob       (1163$)   5:150

10 CONTRACTEE/S
Desiree   (64$)     0:22 1:37 2:81 3:34 4:77 5:78
Candy     (4$)      0:67 1:88 2:65 3:75 4:28 5:87
Vanessa   (177$)    0:36 1:91 2:10 3:27 4:39 5:19
Eva       (82$)     0:38 1:60 2:27 3:33 4:18 5:83
Tanya     (119$)    0:97 1:53 2:21 3:88 4:18 5:86
Traci     (0$)      0:94 1:94 2:40 3:51 4:36 5:55
Marlene   (0$)      0:47 1:63 2:97 3:68 4:96 5:57
Veronica  (31$)     0:24 1:70 2:86 3:55 4:94 5:28
Juliet    (202$)    0:49 1:25 2:31 3:79 4:47 5:10
Lili      (143$)    0:15 1:33 2:46 3:66 4:33 5:92

Broken Contracts: 21
```

```
All deals fixed, jobs done and paid. Entering round 18.
Lili signs contract with Ron (Value: 36$).
Lili breaks contract with Ron by paying a fine of 9$.
Lili signs contract with John (Value: 30$).
Vanessa signs contract with Bob (Value: 23$).
Eva signs contract with Samy (Value: 35$).
Tanya signs contract with Antoni (Value: 25$).
Vanessa breaks contract with Bob by paying a fine of 5$.
Vanessa signs contract with Laurent (Value: 21$).
All deals fixed, jobs done and paid. Entering round 19.
Desiree signs contract with John (Value: 30$).
Juliet signs contract with Ron (Value: 36$).
Vanessa signs contract with Bob (Value: 23$).
Eva signs contract with Antoni (Value: 25$).
Vanessa breaks contract with Bob by paying a fine of 5$.
Vanessa signs contract with Laurent (Value: 21$).
Vanessa breaks contract with Laurent by paying a fine of 5$.
Vanessa signs contract with Samy (Value: 46$).
All deals fixed, jobs done and paid. Entering round 20.
Juliet signs contract with Ron (Value: 36$).
Vanessa signs contract with Laurent (Value: 21$).
```

Auction type: Dutch

6 Contractors

10 Contractees

○ Full Commitment

● Leveled Commitment

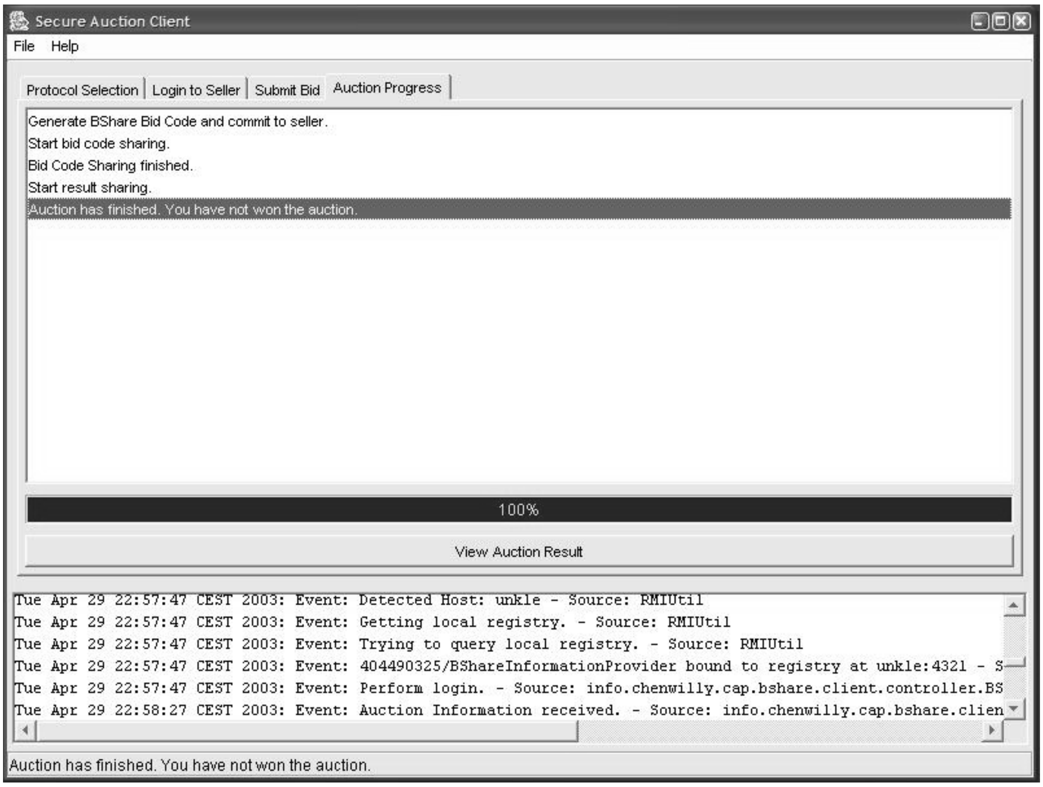
● Price Penalty

○ Cost Penalty

Penalty (25%)

Stop at round 100

Figure B.1: Screenshot of ABC



Generate BShare Bid Code and commit to seller.
Start bid code sharing.
Bid Code Sharing finished.
Start result sharing.
Auction has finished. You have not won the auction.

```
Tue Apr 29 22:57:47 CEST 2003: Event: Detected Host: unkle - Source: RMIUtil
Tue Apr 29 22:57:47 CEST 2003: Event: Getting local registry. - Source: RMIUtil
Tue Apr 29 22:57:47 CEST 2003: Event: Trying to query local registry. - Source: RMIUtil
Tue Apr 29 22:57:47 CEST 2003: Event: 404490325/BShareInformationProvider bound to registry at unkle:4321 - S
Tue Apr 29 22:57:47 CEST 2003: Event: Perform login. - Source: info.chenwilly.cap.bshare.client.controller.BS
Tue Apr 29 22:58:27 CEST 2003: Event: Auction Information received. - Source: info.chenwilly.cap.bshare.clien
```

Auction has finished. You have not won the auction.

Figure B.2: Screenshot of CAP

B.2 CAP (Cryptographic Auction Protocol)

CAP (see Figure B.2) is an implementation by Willy Chen that includes auction protocols B-share, MB-share, and YMB-share (Sections 6.3 – 6.5) and a graphical user interface. The reason for implementing these protocols was to measure their practical performance in a realistic network environment. It turned out that despite a computational/bandwidth complexity which is quadratic in the number of bidders, YMB-share is feasible for a moderate number of bidders. In one of the experiments, a Vickrey auction with 20 bidders and 200 possible bids was resolved in less than eight hours in a network of consumer computers with mediocre computing power. This result is of particular convenience as the newer protocol vX-share is significantly faster. For further information regarding the implementation, consult [Che02].

Bibliography

[Abe99] M. Abe. Mix-networks on permutation networks. In *Proceedings of the 5th Asiacrypt Conference*, volume 1716 of *Lecture Notes in Computer Science*, pages 258–273. Springer, 1999.

[ACS02] J. Algesheimer, J. Camenisch, and V. Shoup. Efficient computation modulo a shared secret with application to the generation of shared safe-prime products. In *Advances in Cryptology - Proceedings of the 22th Annual International Cryptology Conference (CRYPTO)*, volume 2442 of *Lecture Notes in Computer Science*, pages 417–432. Springer, 2002.

[AS98] M. R. Andersson and T. Sandholm. Leveled commitment contracts with myopic and strategic agents. In *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI)*, pages 38–45, 1998.

[AS02a] M. Abe and K. Suzuki. M+1-st price auction using homomorphic encryption. In *Proceedings of the 5th International Conference on Public Key Cryptography (PKC)*, volume 2274 of *Lecture Notes in Computer Science*, pages 115–224. Springer, 2002.

[AS02b] M. Abe and K. Suzuki. Receipt-free sealed-bid auction. In *Proceedings of the 1st Information Security Conference (ISC)*, volume 2433 of *Lecture Notes in Computer Science*, pages 191–199, 2002.

[Bau00] F. L. Bauer. *Entzifferte Geheimnisse - Methoden und Maximen der Kryptologie.* Springer, 3rd edition, 2000.

[BBW00] F. Brandt, W. Brauer, and G. Weiß. Task assignment in multiagent systems based on Vickrey-type auctioning and leveled commitment contracting. In M. Klusch and L. Kerschberg, editors, *Cooperative Information Agents IV*, volume 1860 of *Lecture Notes in Artificial Intelligence*, pages 95–106. Springer, 2000.

[BE00] S. Bach and G. Erber. Die UMTS-Lizenzvergabe in Deutschland - Auktionsverfahren unbefriedigend. *DIW-Wochenberichte*, 20, 2000.

[Ben01] M. Benning. Schwindel unterm Hammer - Betrug auf Internet-Versteigerungen. *c't (Magazin für Computertechnik)*, 8:108–111, 2001.

[BF97] D. Boneh and M. Franklin. Efficient generation of shared RSA keys. In *Advances in Cryptology - Proceedings of the 17th Annual International Cryptology Conference (CRYPTO)*, volume 1294, pages 425–439. Springer, 1997.

[BGW88] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the 20th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 1–10. ACM Press, 1988.

[Bra00] F. Brandt. Antisocial Bidding in Repeated Vickrey Auctions. Technical Report FKI-241-00, Department for Computer Science, Technical University of Munich, 2000. ISSN 0941-6358.

[Bra01] F. Brandt. Cryptographic protocols for secure second-price auctions. In M. Klusch and F. Zambonelli, editors, *Cooperative Information Agents V*, volume 2182 of *Lecture Notes in Artificial Intelligence*, pages 154–165. Springer, 2001.

[Bra02a] F. Brandt. Secure and private auctions without auctioneers. Technical Report FKI-245-02, Department for Computer Science, Technical University of Munich, 2002. ISSN 0941-6358.

[Bra02b] F. Brandt. A verifiable, bidder-resolved auction protocol. In R. Falcone, S. Barber, L. Korba, and M. Singh, editors, *Proceedings of the 5th International Workshop on Deception, Fraud and Trust in Agent Societies (Special Track on Privacy and Protection with Multi-Agent Systems)*, pages 18–25, 2002.

[Bra03a] F. Brandt. Fully private auctions in a constant number of rounds. In *Proceedings of the 7th Annual Conference on Financial Cryptography (FC)*, Lecture Notes in Computer Science. Springer, 2003. to appear.

[Bra03b] F. Brandt. Private public choice. Technical Report FKI-247-03, Department for Computer Science, Technical University of Munich, 2003. ISSN 0941-6358.

[Bra03c] F. Brandt. Social choice and preference protection - Towards fully private mechanism design. In *Proceedings of the 4th ACM Conference on Electronic Commerce*. ACM Press, 2003. to appear.

[BS83] M. Bazerman and W. Samuelson. I won the auction but don't want the prize. *Journal of Conflict Resolution*, 27:618–634, 1983.

[BS01] O. Baudron and J. Stern. Non-interactive private auctions. In *Proceedings of the 5th Annual Conference on Financial Cryptography (FC)*, pages 300–313, 2001.

[BSW01] A. Beutelspacher, J. Schwenk, and K.-D. Wolfenstetter. *Moderne Verfahren der Kryptographie - Von RSA zu Zero-Knowledge.* Vieweg, 2001.

[BW99] F. Brandt and G. Weiß. Exploring auction-based leveled commitment contracting. Part I: English-type auctioning. Technical Report FKI-234-99, Department for Computer Science, Technical University of Munich, 1999. ISSN 0941-6358.

[BW00a] F. Brandt and G. Weiß. Exploring auction-based leveled commitment contracting. Part II: Dutch-type auctioning. Technical Report FKI-237-00, Department for Computer Science, Technical University of Munich, 2000. ISSN 0941-6358.

[BW00b] F. Brandt and G. Weiß. Exploring auction-based leveled commitment contracting. Part III: Vickrey-type auctioning. Technical Report FKI-238-00, Department for Computer Science, Technical University of Munich, 2000. ISSN 0941-6358.

[BW01a] F. Brandt and G. Weiß. Antisocial agents and Vickrey auctions. In J.-J. Ch. Meyer and M. Tambe, editors, *Intelligent Agents VIII*, volume 2333 of *Lecture Notes in Artificial Intelligence*, pages 335–347. Springer, 2001. Revised papers from the 8th Workshop on Agent Theories, Architectures and Languages.

[BW01b] F. Brandt and G. Weiß. Vicious strategies for Vickrey auctions. In J.P. Müller, E. Andre, S. Sen, and C. Frasson, editors, *Proceedings of the 5th International Conference on Autonomous Agents*, pages 71–72. ACM Press, 2001.

[Cac99] C. Cachin. Efficient private bidding and auctions with an oblivious third party. In *Proceedings of the 6th ACM Conference on Computer and Communications Security*, pages 120–127, 1999.

[Cas67] R. Cassady. *Auctions and Auctioneering.* California University Press, Berkeley, 1967.

[CCD88] D. Chaum, C. Crépeau, and I. Damgård. Multi-party unconditionally secure protocols. In *Proceedings of the 20th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 11–19. ACM Press, 1988.

[CD02] R. Cramer and I. Damgård. Multiparty computation - An introduction. Lecture Notes, University of Aarhus, Department for Computer Science, 2002.

[CDN01] R. Cramer, I. Damgård, and J. B. Nielsen. Multiparty computation from threshold homomorphic encryption. In *Advances in Cryptology - Proceedings of the 18th Eurocrypt Conference*, volume 2045 of *Lecture Notes in Computer Science*, pages 280–300. Springer, 2001.

[CDS94] R. Cramer, I. Damgård, and B. Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols. In *Advances in Cryptology - Proceedings of the 14th Annual International Cryptology Conference (CRYPTO)*, volume 893 of *Lecture Notes in Computer Science*, pages 174–187. Springer, 1994.

[CGJ$^+$99] R. Canetti, R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin. Adaptive security for threshold cryptosystems. In *Advances in Cryptology - Proceedings of the 19th Annual International Cryptology Conference (CRYPTO)*, volume 1666 of *Lecture Notes in Computer Science*, pages 98–115. Springer, 1999.

[CGS97] R. Cramer, R. Gennaro, and B. Schoenmakers. A secure and optimally efficient multi-authority election scheme. In *Advances in Cryptology - Proceedings of the 14th Eurocrypt Conference*, volume 1233 of *Lecture Notes in Computer Science*, pages 103–118. Springer, 1997.

[Cha88] D. Chaum. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Journal of Cryptology*, 1(1):65–75, 1988.

[Che02] W. Chen. Kryptographische Auktionsprotokolle - Implementierung und Analyse. Systementwicklungsprojekt, Department for Computer Science, Technical University of Munich, available at `http://www.chenwilly.info`, 2002.

[CP92] D. Chaum and T. P. Pedersen. Wallet databases with observers. In *Advances in Cryptology - Proceedings of the 12th Annual International Cryptology Conference (CRYPTO)*, volume 740 of *Lecture Notes in Computer Science*, pages 3.1–3.6. Springer, 1992.

[Cra00] R. Cramer. Introduction to secure computation. Lecture Notes, University of Aarhus, Department for Computer Science, 2000.

[CS02a] W. Conen and T. Sandholm. Partial-revelation VCG mechanism for combinatorial auctions. In *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI)*, pages 367–372. AAAI Press, 2002.

[CS02b] V. Conitzer and T. Sandholm. Complexity of manipulating elections with few candidates. In *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI)*, pages 314–319. AAAI Press, 2002.

[CS02c] V. Conitzer and T. Sandholm. Complexity of mechanism design. In *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 103–110, 2002.

[CvdGT95] C. Crépeau, J. van de Graaf, and A. Tapp. Comitted oblivious transfer and private multiparty commputation. In *Advances in Cryptology - Proceedings of the 15th Annual International Cryptology Conference (CRYPTO)*, volume 963 of *Lecture Notes in Computer Science*, pages 110–123, 1995.

[DH76] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, 1976.

[DK01] I. Damgård and M. Koprowski. Practical threshold RSA signatures without a trusted dealer. In *Advances in Cryptology -*

Proceedings of the 18th Eurocrypt Conference, volume 2045 of *Lecture Notes in Computer Science*, pages 152–165. Springer, 2001.

[dNPv01] B. de Decker, G. Neven, F. Piessens, and E. van Hoeymissen. Second price auctions - A case study of secure distributed computing. In K. Zielinski, K. Geihs, and A. Laurentowski, editors, *New Developments in Distributed Applications and Interoperable Systems*, pages 217–228. Kluwer Academic Publishers, 2001.

[dVV01] S. de Vries and R. Vohra. Auctions and the German UMTS-auction. *DMV-Mitteilungen*, pages 31–38, 2001.

[eba03] ebay, Inc. Corporate Presentation. Available at `http://www.ebay.com`, February 2003.

[Eck01] C. Eckert. *IT-Sicherheit - Konzepte, Verfahren, Protokolle.* Oldenbourg, 2001.

[ElG85] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31:469–472, 1985.

[FBI02] Internet Fraud Complaint Center - 2001 Internet Fraud Report. `http://www.ifccfbi.gov`, 2002. National White Collar Crime Center and Federal Bureau of Investigation.

[Fer99] J. Ferber. *Multi-Agent Systems. An Introduction to Distributed Artificial Intelligence.* John Wiley & Sons Inc., New York, 1999.

[FGY92] M. Franklin, Z. Galil, and M. Yung. An overview of secure distributed computing. Technical Report TR CUCS-008-92, Columbia University, 1992.

[FR96] M. K. Franklin and M. K. Reiter. The design and implementation of a secure auction service. *IEEE Transactions on Software Engineering*, 22(5):302–312, 1996.

[GIKR01] R. Gennaro, Y. Ishai, E. Kushilevitz, and T. Rabin. The round complexity of verifiable secret sharing and secure multicast. In *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing (STOC)*, pages 580–589. ACM Press, 2001.

[GJKR99] R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin. Secure distributed key generation for discrete-log based cryptosystems. In *Advances in Cryptology - Proceedings of the 16th Eurocrypt Conference*, volume 1592 of *Lecture Notes in Computer Science*, pages 295–310. Springer, 1999.

[GM87] D. A. Graham and R. C. Marshall. Collusive bidder behaviour at single-object second-price and English auctions. *Journal of Political Economy*, 95(6), 1987.

[GMR85] S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof systems. In *Proceedings of the 17th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 291–304. ACM Press, 1985.

[GMW86a] O. Goldreich, S. Micali, and A. Wigderson. How to prove all NP-statements in zero-knowledge, and a methodology of cryptographic protocol design. In *Advances in Cryptology - Proceedings of the 12th Annual International Cryptology Conference (CRYPTO)*, volume 263 of *Lecture Notes in Computer Science*, pages 171–185. Springer, 1986.

[GMW86b] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity and a methodology of cryptographic protocol design. In *Proceedings of the 27th Symposium on Foundations of Computer Science (FOCS)*, pages 174–187. IEEE Computer Society Press, 1986.

[GMW87] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game or a completeness theorem for protocols with honest majority. In *Proceedings of the 19th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 218–229. ACM Press, 1987.

[Gol97] O. Goldreich. On the foundations of modern cryptography. In *Advances in Cryptology - Proceedings of the 17th Annual International Cryptology Conference (CRYPTO)*, volume 1294 of *Lecture Notes in Computer Science*, pages 46–74. Springer, 1997.

[GRR98] R. Gennaro, M. Rabin, and T. Rabin. Simplified VSS and fast-track multiparty computations with applications to threshold cryptography. In *Proceedings of the 17th annual ACM Symposium on Principles of Distributed Computing (PODC)*, pages 101–111. ACM Press, 1998.

[GRW01] V. Grimm, F. Riedel, and E. Wolfstetter. Low price equilibrium in multi-unit auctions: The GSM spectrum auction in Germany. Working Paper, 2001.

[GRW02] V. Grimm, F. Riedel, and E. Wolfstetter. The third generation (UMTS) spectrum auction in Germany. *ifo Studien*, 48(1), 2002.

[HKI03] W. Ham, K. Kim, and H. Imai. Yet another strong sealed-bid auctions. In *Proceedings of the Symposium on Cryptography and Information Security (SCIS)*, pages 11–16, 2003.

[HTK98] M. Harkavy, J. D. Tygar, and H. Kikuchi. Electronic auctions with private bids. In *Proceedings of the 3rd USENIX Workshop on Electronic Commerce*, pages 61–74, 1998.

[JJ00] M. Jakobsson and A. Juels. Mix and match: Secure function evaluation via ciphertexts. In *Proceedings of the 6th Asiacrypt Conference*, volume 1976 of *Lecture Notes in Computer Science*, pages 162–177. Springer, 2000.

[JMS96] P. Jehiel, B. Moldovanu, and E. Stacchetti. How (not) to sell nuclear weapons. *American Economic Review*, 86:814–829, 1996.

[JS02] A. Juels and M. Szydlo. A two-server, sealed-bid auction protocol. In M. Blaze, editor, *Proceedings of the 6th Annual Conference on Financial Cryptography (FC)*, volume 2357 of *Lecture Notes in Computer Science*. Springer, 2002. to appear.

[KHAN00] H. Kikuchi, S. Hotta, K. Abe, and S. Nakanishi. Resolving winner and winning bid without revealing privacy of bids. In *Proceedings of the International Workshop on Next Generation Internet (NGITA)*, pages 307–312, 2000.

[KHT98] H. Kikuchi, M. Harkavy, and J. D. Tygar. Multi-round anonymous auction protocols. In *Proceedings of the 1st IEEE Workshop on Dependable and Real-Time E-Commerce Systems*, pages 62–69, 1998.

[Kik01] H. Kikuchi. (M+1)st-price auction protocol. In *Proceedings of the 5th Annual Conference on Financial Cryptography (FC)*, volume 2339 of *Lecture Notes in Computer Science*, pages 351–363. Springer, 2001.

[Kik02] H. Kikuchi. (M+1)st-price auction protcol. *IEICE Transaction Fundamentals*, E85(A):676–683, 2002.

[Kil88] J. Kilian. Founding cryptography on oblivious transfer. In *Proceedings of the 20th ACM Symposium on Theory of Computing*, pages 20–31. ACM Press, 1988.

[Kle99] P. Klemperer. Auction theory: A guide to the literature. *Journal of Economic Surveys*, 13(3):227–286, 1999.

[Kle02a] P. Klemperer. How (not) to run auctions: the European 3G telecom auctions. *European Economic Review*, 46:829–845, 2002.

[Kle02b] P. Klemperer. What really matters in auction design. *Journal of Economic Perspectives*, 16(1):169–189, 2002.

[KO02] K. Kurosawa and W. Ogata. Bit-slice auction circuit. In *Proceedings of the 7th European Symposium on Research in Computer Security (ESORICS)*, volume 2502 of *Lecture Notes in Computer Science*, pages 24–38. Springer, 2002.

[KT01] P. Klemperer and P. Temin. An early example of the "winner's curse" in an auction. *Journal of Political Economy*, 109(6), 2001. Lagniappe.

[Kud98] M. Kudo. Secure electronic sealed-bid auction protocol with public key cryptography. *IEICE Transaction Fundamentals*, E81-A(1), 1998.

[Kus89] E. Kushilevitz. Privacy and communication complexity. In *Proceedings of the 30th Symposium on Foundations of Computer Science (FOCS)*, pages 416–421. IEEE Computer Society Press, 1989.

[LAN02] H. Lipmaa, N. Asokan, and V. Niemi. Secure Vickrey auctions without threshold trust. In M. Blaze, editor, *Proceedings of the 6th Annual Conference on Financial Cryptography (FC)*, volume 2357 of *Lecture Notes in Computer Science*. Springer, 2002. to appear.

[LBST00] K. Leyton-Brown, Y. Shoham, and M. Tennenholtz. Bidding clubs: institutionalized collusion in auctions. In *Proceedings of the 2nd ACM Conference on Electronic Commerce*, pages 253–259. ACM Press, 2000.

[Leu96] A. Leutbecher. *Zahlentheorie - Eine Einführung in die Algebra.* Springer, 1996.

[LK02] D. Levin and J. H. Kagel. *Common Value Auctions and the Winner's Curse.* Princeton University Press, 2002.

[LS01] K. Larson and T. Sandholm. Computationally limited agents in auctions. In *Theoretical Aspects of Reasoning about Knowledge (TARK)*, pages 169–182, 2001.

[LSP82] L. Lamport, R. Shostak, and M. Pease. The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, 1982.

[McM91] J. McMillan. Dango: Japan's price-fixing conspiracies. *Economics and Politics*, 3:201–218, 1991.

[McM94] J. McMillan. Selling spectrum rights. *Journal of Economic Perspectives*, 8(3):145–162, 1994.

[Mea87] W.J. Mead. Natural resource disposal policy: Oral auction versus sealed bids. *Natural Resources Journal*, 7:195–224, 1987.

[Mil89] P. Milgrom. Auctions and bidding: A primer. *Journal of Economic Perspectives*, 3(3):3–22, 1989.

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

[MM92] R. P. McAfee and J. McMillan. Bidding rings. *Amercican Economic Review*, 82(3):579–599, 1992.

[MvOV96] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, editors. *Handbook of Applied Cryptography.* CRC Press, 1996.

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

[MWG95] A. Mas-Colell, M. D. Whinston, and J. R. Green. *Microeconomic Theory.* Oxford University Press, Inc., 1995.

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

[MZ91] G. Mailath and P. Zemsky. Collusion in second price auctions with heterogenous bidders. *Games and Economic Behaviour*, 3:467–486, 1991.

[NPS99] M. Naor, B. Pinkas, and R. Sumner. Privacy preserving auctions and mechanism design. In *Proceedings of the 1st ACM Conference on Electronic Commerce*, pages 129–139. ACM Press, 1999.

[NR99] N. Nisan and A. Ronen. Algorithmic mechanism design. In *Proceedings of the 31st ACM Symposium on Theory of Computing*, pages 129–140. ACM Press, 1999.

[NS93] H. Nurmi and A. Salomaa. Cryptographic protocols for Vickrey auctions. *Group Decision and Negotiation*, 2:363–373, 1993.

[OJ96] G. M. P. O'Hare and N. R. Jennings, editors. *Foundations of Distributed Artificial Intelligence.* John Wiley & Sons Inc., New York, 1996.

[OR02] A. Ockenfels and A. E. Roth. The timing of bids in internet auctions: Market design, bidder behavior, and artificial agents. *Artificial Intelligence Magazine*, pages 79–87, 2002.

[Pai99] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology - Proceedings of the 16th Eurocrypt Conference*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238. Springer, 1999.

[Par01] D. Parkes. *Iterative Combinatorial Auctions: Achieving Economic and Computational Efficiency.* PhD thesis, Department of Computer and Information Science, University of Pennsylvania, 2001.

[Ped91] T. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In J. Feigenbaum, editor, *Advances in Cryptology - Proceedings of the 11th Annual International Cryptology Conference (CRYPTO)*, volume 576 of *Lecture Notes in Computer Science*, pages 129–140. Springer, 1991.

[Rai82] H. Raiffa. *The Art and Science of Negotiation.* Harvard University Press, Cambridge, Mass., 1982.

[Ras95] E. Rasmusen. *Games and Information.* Basil Blackwell, 1995.

[RH95] M. H. Rothkopf and R. M. Harstad. Two models of bid-taker cheating in Vickrey auctions. *Journal of Business*, 68(2):257–267, 1995.

[RO02] A. E. Roth and A. Ockenfels. Last-minute bidding and the rules for ending second-price auctions: Evidence from ebay and amazon on the internet. *American Economic Review*, 92(4):1093–1103, 2002.

[Rob85] M. S. Robinson. Collusion and the choice of auction. *RAND Journal of Economics*, 16:141–145, 1985.

[RTK90] M. H. Rothkopf, T. J. Teisberg, and E. P. Kahn. Why are Vickrey auctions rare? *Journal of Political Economy*, 98(1):94–109, 1990.

[RZ94] J. Rosenschein and G. Zlotkin. *Rules of Encounter*. The MIT Press, Cambridge, Mass., 1994.

[SA99] F. Stajano and R. J. Anderson. The cocaine auction protocol: On the power of anonymous broadcast. In *Information Hiding*, pages 434–447, 1999.

[Sak00] K. Sako. An auction protocol which hides bids of losers. In *Proceedings of the 3rd International Conference on Public Key Cryptography (PKC)*, volume 1751 of *Lecture Notes in Computer Science*, pages 422–432. Springer, 2000.

[San93] T. Sandholm. An implementation of the contract net protocol based on marginal cost calculations. In *Proceedings of the 10th National Conference on Artificial Intelligence (AAAI)*, pages 256–262, 1993.

[San96] T. Sandholm. Limitations of the Vickrey auction in computational multiagent systems. In *Proceedings of the 2nd International Conference on Multiagent Systems (ICMAS)*, pages 299–306, Menlo Park, CA, 1996. AAAI Press.

[San99] T. Sandholm. An algorithm for optimal winner determination in combinatorial auctions. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 542–547, 1999.

[San00] T. Sandholm. Issues in computational Vickrey auctions. *International Journal of Electronic Commerce, Special issue on Intelligent Agents for Electronic Commerce*, 4(3):107–129, 2000.

[Sch91] C. P. Schnorr. Efficient signature generation by smart cards. *Journal of Cryptology*, 4(3):161–174, 1991.

[Sch96] B. Schneier. *Applied Cryptography.* John Wiley and Sons, Inc., 2nd edition, 1996.

[Sha79] A. Shamir. How to share a secret. *Communications of the ACM*, 22:612–613, 1979.

[SL95] T. Sandholm and V. R. Lesser. Issues in automated negotiation and electronic commerce: Extending the contract net framework. In *Proceedings of the 1st International Conference on Multi-Agent Systems (ICMAS)*, pages 328–335, 1995.

[SL96] T. Sandholm and V. R. Lesser. Advantages of a leveled commitment contracting protocol. In *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI)*, pages 126–133, 1996.

[SL97] T. Sandholm and V. R. Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94:99–134, 1997.

[SLA$^+$98] T. Sandholm, K. Larson, M. Andersson, O. Shehory, and F. Tohmé. Anytime coalition structure generation with worst case guarantees. In *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI)*, pages 46–53, 1998.

[SM99] K. Sakurai and S. Miyazaki. A bulletin-board based digital auction scheme with bidding down strategy - Towards anonymous electronic bidding without anonymous channels nor trusted centers. In *Proceedings of the International Workshop on Cryptographic Techniques and E-Commerce*, pages 180–187, 1999.

[SM00a] K. Sakurai and S. Miyazaki. An anonymous electronic bidding protocol based on a new convertible group signature scheme. In *Proceedings of the 5th Australasian Conference on Information Security and Privacy (ACISP2000)*, volume 1841 of *Lecture Notes in Computer Science*, pages 385–399. Springer, 2000.

[SM00b] D. X. Song and J. K. Millen. Secure auctions in a publish/subscribe system. Available at http://www.csl.sri.com/users/millen/, 2000.

[Smi80] R. G. Smith. The contract-net protocol: High-level communication and control in a distributed problem solver. *IEEE Transactions on Computers*, C-29(12):1104–1113, 1980.

[SS99] S. G. Stubblebine and P. F. Syverson. Fair on-line auctions without special trusted parties. In M. Franklin, editor, *Proceedings of the 3rd Annual Conference on Financial Cryptography (FC)*, volume 1648 of *Lecture Notes in Computer Science*, pages 230–240. Springer, 1999.

[SSGL01] T. Sandholm, S. Suri, A. Gilpin, and D. Levine. CABOB: A fast optimal algorithm for combinatorial auctions. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1102–1108, 2001.

[SSN99] T. Sandholm, S. Sikka, and S. Norden. Algorithms for optimizing leveled commitment contracts. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 535–541, 1999.

[SY02] K. Suzuki and M. Yokoo. Secure combinatorial auctions by dynamic programming with polynomial secret sharing. In M. Blaze, editor, *Proceedings of the 6th Annual Conference on Financial Cryptography (FC)*, volume 2357 of *Lecture Notes in Computer Science*. Springer, 2002. to appear.

[Ten00] M. Tennenholtz. Some tractable combinatorial auctions. In *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI)*, pages 98–103. AAAI Press / The MIT Press, 2000.

[TY98] Y. Tsiounis and M. Yung. On the security of ElGamal-based encryption. In *Proceedings of the 1st International Workshop on Practice and Theory in Public Key Cryptography (PKC)*, volume 1431 of *Lecture Notes in Computer Science*, pages 117–134. Springer, 1998.

[Var95] H. R. Varian. Economic mechanism design for computerized agents. In *Proceedings of the 1st Usenix Workshop on Electronic Commerce*, pages 13–21, 1995.

[Var99] H. R. Varian. *Intermediate Microeconomics - A Modern Approach.* W. W. Norton & Company, 5th edition, 1999.

[VBD00] K. Viswanathan, C. Boyd, and E. Dawson. A three phased schema for sealed bid auction system design. In *Proceedings of the Australasian Conference for Information Security and Privacy (ACISP)*, Lecture Notes in Computer Science, pages 412–426, 2000.

[Vic61] W. Vickrey. Counter speculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16(1):8–37, 1961.

[Wei99] G. Weiß, editor. *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence.* The MIT Press, Cambridge, MA, 1999.

[WI00] Y. Watanabe and H. Imai. Reducing the round complexity of a sealed-bid auction protocol with an off-line TTP. In *Proceedings of the 7th ACM Conference on Computer and Communications Security*, pages 80–86. ACM Press, 2000.

[Wol96] E. Wolfstetter. Auctions: An introduction. *Journal of Economic Surveys*, pages 367–420, 1996.

[WWW98] P. Wurman, W. Walsh, and M. Wellman. Flexible double auctions for electronic commerce: Theory and implementation. *Decision Support Systems*, 24:17–27, 1998.

[Yao82] A. C. Yao. Protocols for secure computation. In *Proceedings of the 23th Symposium on Foundations of Computer Science (FOCS)*, pages 160–164. IEEE Computer Society Press, 1982.

[Yao86] A. C. Yao. How to generate and exchange secrets. In *Proceedings of the 27th Symposium on Foundations of Computer Science (FOCS)*, pages 162–167. IEEE Computer Society Press, 1986.

[YS02] M. Yokoo and K. Suzuki. Secure multi-agent dynamic programming based on homomorphic encryption and its application to combinatorial auctions. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 112–119. ACM Press, 2002.

[YSM03] M. Yokoo, Y. Sakurai, and S. Matsubara. The effect of false-name bids in combinatorial auctions: New fraud in internet auctions. *Games and Economic Behaviour*, 2003. to appear.

Index