Auctions*

Felix Brandt

October 1, 2009

1 Introduction

Auctions are key mechanisms for allocating scarce resources among multiple parties. While traditionally auctions have mainly been applied to the selling of physical goods, they are becoming increasingly popular as mechanisms for such diverse tasks as procurement, bandwidth allocation, or selling online ad space. At the same time, privacy is a crucial issue in electronic commerce. A major reason why people may be hesitant to use software agents, or to participate in Internet commerce themselves, is the worry that too much of their private information is revealed. Furthermore, in the modern electronic society, the information might get propagated to large numbers of parties, stored in permanent databases, and automatically used in undesirable ways. This chapter studies the possibility of executing the most common types of sealed-bid auctions in a way that preserves the bidders' privacy.

1.1 A Very Short Introduction to Auction Theory

Auctions can be used in a variety of resource allocation settings differing in the number of sellers, buyers, and goods for sale [see, e.g., 18, for an excellent overview of auction theory]. Here we will focus on the most basic setting consisting of one seller, n buyers, and a single good. By symmetry, our results will also apply to so-called *reverse auctions* (as used for

*Chapter draft for the upcoming CRC Handbook on Financial Cryptography procurement) where there is one buyer and multiple sellers. An auction is simply a protocol that yields the winner of the item and information on the exchange of payments (typically only the winning bidder is charged). The prototypical auction types in the basic setting are the *English auction*, the *Dutch auction*, the *first-price sealed-bid* auction, and the *second-price sealed-bid* auction. In an English or "ascending open-cry" auction, the auctioneer (a trusted party who may or may not be the seller) continuously raises the selling price until only one bidder is willing to pay. This bidder is awarded the item and pays the current price. In a Dutch auction, the auctioneer starts at a high price and continuously *reduces* it until the first bidder expresses his willingness to pay. Again, this bidder is awarded the item and pays the current price. In both types of sealed-bid auctions, each bidder submits a sealed bid to the auctioneer and the bidder who submitted the highest bid is awarded the item. In the first-price auction, the winning bidder pays the amount he bid, whereas in the second-price auction, he has to pay the amount of the second highest bid. The second-price auction is often called *Vickrey auction* in memory of Nobel Laureate William Vickrey who first proposed it [33].

Despite their different appearance, some of these auction types have very strong similarities. For instance, the Dutch auction and the first-price auction are known to be strategically equivalent, which essentially means that they will always yield the same result (this was first observed by Vickrey [33]). A similar equivalence holds for the English auction and the second-price auction when bidders have independent valuations of the good to be sold. Since our main interest is privacy, this leaves us with first-price and second-price auctions. And even between these two there are some similarities. The revenue equivalence theorem—one of the most celebrated results of auction theory—states that almost all reasonable types of auctions (including the first-price and second-price auction) yield the same revenue when valuations are independent and bidders are risk-neutral. This seemingly paradoxical equivalence is due to the different behavior of rational bidders in different auctions, e.g., identical bidders bid less in first-price auctions than they do in second-price auctions. Despite this equivalence, both auction formats have their individual strengths and weaknesses. For example, the first-price auction yields more revenue when bidders are risk-averse (which is often the case). The second-price auction, on the other hand, is strategy-proof, which means that bidders are best off bidding their true valuation of the good to be sold, no matter what the other bidders do. Thus, in contrast to the first-price auction, bidders need not estimate other bidders' valuations. Interestingly, the side-effects of this striking advantage are said to contribute to the fact that second-price auctions are not commonly used in practice, for two reasons [27, 26, 29]:

1. Bidders are reluctant to reveal their true valuations to the auctioneer since the auctioneer can exploit this information during and after the auction, or spread it to others in ways that adversely affect the bidder.
2. Bidders doubt the correctness of the result as they do not pay what they bid. For example, the auctioneer might create a fake second highest bid slightly below the highest bid in order to increase his revenue

Both issues mentioned above are rooted in a lack of trust in the auctioneer. For this reason, it would be desirable to somehow "force" the auctioneer to always select the right outcome (*correctness*) and "prohibit" the propagation of private bid information (*privacy*).

1.2 Cryptographic Auction Protocols

Inspired by early work of Nurmi and Salomaa [22] and Franklin and Reiter [12], various cryptographic protocols for achieving privacy and correctness (in first-price as well as second-price auctions) have been proposed in recent years. Most of the protocols for second-price auctions are also applicable to a generalization of second-price auctions known as $(M+1)$st-price auctions. In an $(M+1)$st-price auction—which is also due to Vickrey [33]—the seller offers M indistinguishable units of the same item ($1 \leq M < n$) and each bidder is assumed to be interested in at most one unit (in auction theory this is called *single-unit demand*). After all bidders submitted their bids, each of the M highest bidders receives one unit, all of which are sold for the same price given by the $(M+1)$st highest bid. Clearly, the second-price auction is just the special case for $M = 1$. Besides these protocols, protocols for more general types of auctions such as multi-unit auctions, combinatorial auctions, or double auctions have been proposed in the literature [see, e.g., 30, 31, 36, 8, 4, 24, 32]. However, we will focus on the single-unit demand case in this chapter.

Cryptographic auction protocols are one of the main applications of *secure multiparty computation* as first suggested by Yao [35]. Secure multiparty computation studies how a group of agents can jointly evaluate a function of privately held inputs such that only the function value, but not the individual inputs, are revealed. While it is known that, in principle, any function can be computed privately when making certain assumptions on the number of corrupted parties and their computational abilities [13, 3, 10], protocols for general multiparty computation are still very inefficient and unpractical. For this reason, the development of efficient special-purpose protocols for auctions has attracted the interest of many researchers. The asymptotic complexity of the protocols considered in this chapter depends on two parameters: the number of bidders n and the number of possible prices or bids k. Since prices can be encoded in binary, one would hope for a logarithmic dependence on k. However, as it turns out, representing bids in unary (resulting in a linear dependence on k) sometimes allows to reduce other important complexity measures such as the number of rounds.

In the next section, we will discuss the possibility of unconditionally fully private auction protocols, i.e., auction protocols whose security does not rely on computationally intractability assumptions. More precisely, we study the existence of protocols that enable bidders to jointly compute the auction outcome without revealing any other information in an information-theoretic sense. Results in this setting are rather negative, which motivates the study of computationally private protocols in Section 3. Here, we consider bidder-resolved protocols (as in Section 2), protocols with a single auctioneer, and protocols with two or more auctioneers, respectively.

2 Unconditional Privacy

In this section, we consider protocols where the auctioneer is emulated by the bidders, i.e., the computation of the auction outcome is distributed onto the bidders. Let $x_1, \dots, x_n$ be the inputs of the individual bidders, i.e., their bids, and $f(x_1, \dots, x_n)$ the output of the protocol, i.e., the outcome of the auction. The formal model we employ is the standard information-theoretic private-channels model introduced independently by Ben-Or et al. [3] and Chaum et al. [10], inspired by earlier work of Yao [34]. Thus, function $f(x_1, \ldots, x_n)$ is jointly computed by n parties using a distributed, randomized protocol consisting of several rounds. In order to enable the secure exchange of messages, we assume the existence of a complete synchronous network of private channels between the parties. In each round, each party may send a message to any other party. Each message a party sends is a function of his input x_i, his independent random input r_i, the messages he received so far, and the recipient. When the protocol is finished, all parties know the value of $f(x_1, \ldots, x_n)$.

Typically, when talking about the security of a distributed protocol one thinks of an *adversary* who may corrupt parties. In this section no restrictive assumptions as to the computational power of the adversary are made. A distributed protocol for computing function $f(x_1, \ldots, x_n)$ is *fully private* if an adversary who can corrupt any number of parties is incapable of revealing any information besides what can be inferred from the output $f(x_1, \ldots, x_n)$ and the corrupted parties' inputs.

It is known that only a restricted class of functions can be computed while maintaining unconditional full privacy [3].[1] However, a complete characterization of this class is not yet known [see 19, 11, for characterizations of special cases]. As it turns out, the outcome function of the first-price auction belongs to the class of unconditionally fully privately computable functions.

Theorem 1 (Brandt and Sandholm [9]). *The first-price auction can be emulated by an unconditionally fully private k-round protocol. There is no more efficient protocol.*

Interestingly, the above mentioned protocol is essentially a *Dutch auction* and is sometimes used in the real world for selling flowers or fish. Recall that in a Dutch auction the auctioneer starts by offering a high price, which is then continuously reduced until the first bidder expresses his willingness to buy. Obviously, no information except the auction outcome is revealed. An attractive property of the Dutch auction is that it only requires a broadcast channel rather than a complete a network of private channels. On the other hand, it is not very efficient as it has to iterate through every possible price in the worst case.

[1] When assuming that a majority of the agents is trustworthy, *all* functions can be jointly computed in the unconditional passive adversary model [3, 10].

It has been shown that there exists no such protocol for the second-price auction.

Theorem 2 (Brandt and Sandholm [9])**.** *The second-price auction cannot be emulated by an unconditionally fully private protocol (when there are more than two bidders).*

The previous impossibility is very robust in the sense that it even holds when only protecting a single losing bid or revealing the second-highest bidder's identity.

3 Computational Privacy

It has become common practice in cryptography to assume that the adversary is limited in its computational abilities. This is usually implemented by surmising the existence of one-way functions, i.e., functions that are easy to compute but hard to invert. Two popular candidates for one-way functions are multiplication and exponentiation in certain finite groups. A plethora of cryptographic auction protocols that rely on various variants of these intractability assumptions (such as the Decisional Diffie-Hellman Assumption (DDH) or the Decisional Composite Residuosity Assumption (DCR)) have been proposed. In this section, we will informally discuss a small selection of the proposed protocols.

The protocols essentially fall into four categories depending on their underlying security model. First, there are fully private protocols where the auction outcome is jointly computed by the bidders as in the previous section. Then there are protocols that retain the traditional model of a single auctioneer and can therefore only provide limited privacy guarantees. In most protocols the trust is distributed on two parties (e.g., the auctioneer and an "auction issuer" or the auctioneer and an "auction authority"). These protocols use asymmetric multiparty computation such as Yao's garbled circuit technique. Finally, there are protocols where the trust is distributed on multiple, symmetric auctioneers who jointly determine the outcome using some form of threshold multiparty computation.

Table 1 highlights some of the differences between the proposed protocols. A protocol is *verifiable* if the correctness of the auction outcome can be verified by bidders and external parties. A protocol satisfies *non-repudiation* if winning bidders cannot deny having won the auction.

3.1 No Auctioneers

Brandt [6, 7] has put forward protocols for first-price and $(M+1)$st-price auctions that are executed by the bidders themselves without the help of any third-party. The protocols are based on El Gamal encryption and require three rounds of interaction in the random oracle model. Communication complexity, however, is linear in the number of possible bids k. The protocol for $(M+1)$st-price auctions is significantly more complex than the one for first-price auctions. The main advantage of these protocols is that they are *fully private*, i.e.,—based on certain intractability assumptions—*no* coalition of parties is capable of breaching privacy. The drawbacks implied by such a model are low robustness and relatively high computational and communication complexity (although round complexity is low and constant). As a tradeoff between the unconditional and computational model, Brandt [5] proposed a second-price auction protocol that is unconditionally anonymous and computationally private. The joint computation of social outcomes without third-parties has also been suggested in the context of secure voting [see, e.g., 17].

3.2 One Auctioneer

In this section, we outline two protocols that are based on the traditional model of a single auctioneer.

Baudron and Stern 2001 The protocol by Baudron et al. [2] relies on a semi-trusted third-party that does not learn any information unless it colludes with a bidder. The protocol is is based on the joint evaluation of a special-purpose Boolean circuit using Paillier encryption. The communication complexity is $O\left(n(\log k)^{n-1}\right)$ and thus exponential in n, which makes the scheme only applicable to a very limited number of bidders (five to six according to the authors). Bidders encrypt each bit of the binary representations of their bids n times with each bidder's public key. 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 the bidders. After the result is broadcasted, the winner is required to claim that he won (violating non-repudiation). This is a disadvantage because the winner is able to back out of the protocol if he is not satisfied with the selling price. When computing the outcome of a second-price auction, additional interaction is required to compute the second highest bid (while also revealing the identity of the second highest bidder). Bidders' actions are verifiable. However, it is not possible to verify whether the third-party behaves correctly.

Parkes, Rabin, Shieber, and Thorpe 2008 Parkes et al. [23] proposed auction protocols for all common types of sealed-bid auctions with the primary goal of practicality rather than complete privacy. The system is based on a single auctioneer and Paillier encryption. The bidders send commitments to their bids to the auctioneer until the submission deadline is over. After the deadline, bidders publish their encrypted bids, which verifiably agree with their earlier commitments. The auctioneer decrypts the bids, publishes the auction outcome, and proves its correctness using elaborate zero-knowledge proofs, which are based on cut-and-choose techniques. This protocol differs from the other protocols considered in this chapter in that complete information on all bids is revealed to the auctioneer after the submission deadline.

3.3 Two Auctioneers

Most of the auction protocols suggested in the literature are based on a pair of auctioneers and the assumption that the auctioneers will not collude.

Naor, Pinkas, and Sumner 1999 The scheme by Naor et al [21] is based on Yao's garbled circuit technique and thus requires two parties, the auctioneer and the "auction issuer." The auction issuer, who "is typically an established party such as a financial institution or large company, which supplies services to numerous auctioneers" [21], constructs an obfuscated Boolean circuit that outputs the auction outcome for any given set of bids. After the bidders submitted their encrypted bids, the auction issuer generates garbled inputs for the circuit from the bids and sends them to the auctioneer who obliviously evaluates the circuit and publishes the result. This protocol is very efficient both in terms of round complexity ($O(1)$) and communication complexity ($O(n \log k)$). However, Yao's protocol was originally conceived for a model with passive adversaries. If malicious deviations by either one of the two parties are taken into account, costly verification techniques such as cut-and-choose, consistency proofs, and the additional evaluation of a majority circuit need to be implemented [25]. Cut-and-choose, for example, requires that 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 method can provide an exponentially large probability of correctness of the circuit. Juels and Szydlo [16] 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 more complexity for the auction servers. Due to a lack of verifiability, a coalition of both auctioneers can not only reveal all private information but also claim an arbitrary auction outcome.

Lipmaa, Asokan, and Niemi 2002 The protocol by Lipmaa et al. [20] requires a single semi-trusted third-party, the auction authority, in addition to the seller. 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 an efficient, special-purpose zero-knowledge proof. Winning bidders are required to claim that they won (violating non-repudiation). The protocol scales very well with respect to the number of bidders, but only provides limited privacy as the auction authority learns all bid amounts. The only information hidden from the authority is the connection between bidders and bids. Neither the seller nor the auction authority can manipulate the outcome without being detected.

Abe & Suzuki 2002 Like the protocols described in Section 3.1, the protocol by Abe et al. [1] is based on a unary representation of bids and homomorphic encryption such as El Gamal or Paillier. However, in contrast to bidder-resolved protocols, the position of the $(M + 1)$st-highest bid is jointly determined by the auctioneer and an "authority" using a binary search subprotocol. More specifically, the auctioneer releases mixed vector components to the authority who decrypts them to detect if there are either more than M bidders or less than $M + 1$ bidders willing to pay. The entire process takes $\log k$ rounds. The protocol is based on Jakobsson et al.'s mix-and-match technique [15] and is publicly verifiable.

3.4 m Auctioneers

The remaining protocols are based on secure multiparty computation where a certain threshold of auctioneers (typically a majority or two thirds) is assumed to be trustworthy. The round complexity of such protocols is generally not constant.

Harkavy, Tygar, and Kikuchi 1998 The protocol by Harkavy et al. [14] was probably the first auction protocol that guarantees complete privacy of all bids, even after the auction terminated. It relies on verifiable secret sharing as described by Ben-Or et al. [3]. Bids are distributed on m auctioneers, $\lfloor \frac{m-1}{3} \rfloor$ of which may be corrupted. In the following, the auction outcome is determined bit by bit using techniques for secure multiparty computation that haven been proposed by Ben-Or et al. [3]. In particular, the second-highest bid is found by checking whether the set of bids can be partitioned into two subsets such that each subset contains a bid that is greater than a test value. The protocol iterates over the possible test values using binary search and therefore requires a number of rounds that is logarithmic in the number of possible prices k.

Sako 2000 Sako's first-price auction protocol [28] is based on a probabilistic encryption scheme. 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 El Gamal 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 important properties of RSA are unknown and the joint generation of RSA keys is very cumbersome. The protocol 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 first-price auctions with very little hope of a possible generalization for other auction types like Vickrey auctions. Additionally, the auctioneers need $O(k)$ rounds to determine the highest bid.

References

[1] 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 (LNCS)*, pages 115–224. Springer-Verlag, 2002.

[2] O. Baudron and J. Stern. Non-interactive private auctions. In *Proceedings of the 5th Annual Conference on Financial Cryptography (FC)*, volume 2339 of *Lecture Notes in Computer Science (LNCS)*, pages 300–313. Springer-Verlag, 2001.

[3] 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.

[4] P. Bogetoft, D. L. Christensen, I. Damgård, M. Geisler, T. Jakobsen, M. Krøigaard, J. D.

Protocol	Price	Auctioneers	Rounds	Computation[a]	V[b]	NR[c]
Brandt [7]	1st	0	$O(1)$	$O(k)$	✓	✓
Brandt [6, 7]	$(M+1)$st	0	$O(1)$	$O(nk)$	✓	✓
Baudron and Stern [2]	1st	1	$O(1)$	$O(n(\log k)^{n-1})$	–	–
Parkes et al. [23][d]	$(M+1)$st	1	$O(1)$	$O(n)$	✓	✓
Naor et al. [21][e]	$(M+1)$st	2	$O(1)$	$O(n \log k)$	–	✓
Lipmaa et al. [20][f]	$(M+1)$st	2	$O(1)$	$O(k)$	✓	–
Abe and Suzuki [1]	$(M+1)$st	2	$O(\log k)$	$O(k)$	✓	✓
Harkavy et al. [14]	2nd	m	$O(\log k)$	$O(n \log k)$	✓	✓
Sako [28]	1st	m	$O(k)$	$O(nk)$	✓	✓

[a] Number of modular exponentiations per auctioneer (or bidder in case there is no auctioneer)

[b] Verifiability

[c] Non-Repudiation

[d] The protocol by Parkes et al. [23] reveals complete bid information to the auctioneer.

[e] The table entries for the improved version suggested by Juels and Szydlo [16] are identical.

[f] The protocol by Lipmaa et al. [20] reveals complete bid statistics to one auctioneer.

Table 1: Overview of selected cryptographic auction protocols (n is the number of bidders and k the number of possible prices or bids). In order to enable a fair comparison, we assume that M and m are constant.

Nielsen, J. B. Nielsen, K. Nielsen, J. Pagter, M. I. Schwartzbach, and T. Toft. Secure multiparty computation goes live. In *Proceedings of the 13th International Conference on Financial Cryptography and Data Security (FC)*, volume 5628 of *Lecture Notes in Computer Science (LNCS)*, pages 325–343. Springer-Verlag, 2009.

[5] F. Brandt. A verifiable, bidder-resolved auction protocol. In R. Falcone, S. Barber, L. Korba, and M. Singh, editors, *Proceedings of the 5th AAMAS Workshop on Deception, Fraud and Trust in Agent Societies (Special Track on Privacy and Protection with Multi-Agent Systems)*, 2002.

[6] F. Brandt. Fully private auctions in a constant number of rounds. In R. N. Wright, editor, *Proceedings of the 7th Annual Conference on Financial Cryptography (FC)*, volume 2742 of *Lecture Notes in Computer Science (LNCS)*, pages 223–238. Springer-Verlag, 2003.

[7] F. Brandt. How to obtain full privacy in auctions. *International Journal of Information Security*, 5(4):201–216, 2006.

[8] F. Brandt and T. Sandholm. Efficient privacy-preserving protocols for multi-unit auctions. In A. Patrick and M. Yung, editors, *Proceedings of the 9th International Conference on Financial Cryptography and Data Security (FC)*, volume 3570 of *Lecture Notes in Computer Science (LNCS)*, pages 298–312. Springer-Verlag, 2005.

[9] F. Brandt and T. Sandholm. On the existence of unconditionally privacy-preserving auction protocols. *ACM Transactions on Information and System Security*, 11(2), 2008.

[10] 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.

[11] B. Chor and E. Kushilevitz. A zero-one law for Boolean privacy. In *Proceedings of the 21st Annual ACM Symposium on the Theory of Computing (STOC)*, pages 62–72. ACM Press, 1989.

[12] 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.

[13] 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.

[14] 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.

[15] 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 (LNCS)*, pages 162–177. Springer-Verlag, 2000.

[16] 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 (LNCS)*, pages 72–86. Springer-Verlag, 2002.

[17] A. Kiayias and M. Yung. Self-tallying elections and perfect ballot secrecy. In *Proceedings of the 5th International Workshop on Practice and Theory in Public Key Cryptography*

(PKC), number 2274 in Lecture Notes in Computer Science (LNCS), pages 141–158. Springer-Verlag, 2002.

[18] V. Krishna. *Auction Theory.* Academic Press, 2002.

[19] 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.

[20] 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 (LNCS)*, pages 87–101. Springer-Verlag, 2002.

[21] M. Naor, B. Pinkas, and R. Sumner. Privacy preserving auctions and mechanism design. In *Proceedings of the 1st ACM Conference on Electronic Commerce (ACM-EC)*, pages 129–139. ACM Press, 1999.

[22] H. Nurmi and A. Salomaa. Cryptographic protocols for Vickrey auctions. *Group Decision and Negotiation*, 2:363–373, 1993.

[23] D. C. Parkes, M. O. Rabin, S. M. Shieber, and C. A. Thorpe. Practical secrecy-preserving, verifiably correct and trustworthy auctions. *Electronic Commerce Research and Applications*, 7(3):294–312, 2008.

[24] D. C. Parkes, M. O. Rabin, and C. Thorpe. Cryptographic combinatorial clock-proxy auctions. In *Proceedings of the 13th International Conference on Financial Cryptography and Data Security (FC)*, volume 5628 of *Lecture Notes in Computer Science (LNCS)*, pages 305–324. Springer-Verlag, 2009.

[25] B. Pinkas. Fair secure two-party computation. In *Proceedings of the 20th Eurocrypt Conference*, volume 2656 of *Lecture Notes in Computer Science (LNCS)*, pages 87–105. Springer-Verlag, 2003.

[26] M. H. Rothkopf and R. M. Harstad. Two models of bid-taker cheating in Vickrey auctions. *Journal of Business*, 68(2):257–267, 1995.

[27] M. H. Rothkopf, T. J. Teisberg, and E. P. Kahn. Why are Vickrey auctions rare? *Journal of Political Economy*, 98(1):94–109, 1990.

[28] 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 (LNCS)*, pages 422–432. Springer-Verlag, 2000.

[29] 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.

[30] K. Suzuki and M. Yokoo. Secure combinatorial auctions by dynamic programming with polynomial secret sharing. In *Proceedings of the 6th Annual Conference on Financial Cryptography (FC)*, volume 2357 of *Lecture Notes in Computer Science (LNCS)*, pages 44–56. Springer-Verlag, 2002.

[31] K. Suzuki and M. Yokoo. Secure generalized Vickrey auction using homomorphic encryption. In *Proceedings of the 7th Annual Conference on Financial Cryptography (FC)*, volume 2742 of *Lecture Notes in Computer Science (LNCS)*, pages 239–249. Springer-Verlag, 2003.

[32] C. Thorpe and D. C. Parkes. Cryptographic combinatorial securities exchanges. In *Proceedings of the 13th International Conference on Financial Cryptography and Data Security (FC)*, volume 5628 of *Lecture Notes in Computer Science (LNCS)*, pages 285–304. Springer-Verlag, 2009.

[33] W. Vickrey. Counter speculation, auctions, and competitive sealed tenders. *Journal of Finance*, 16(1):8–37, 1961.

[34] A. C. Yao. Some complexity questions related to distributed computing. In *Proceedings of the 11th Annual ACM Symposium on the Theory of Computing (STOC)*, pages 209–213. ACM Press, 1979.

[35] 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.

[36] M. Yokoo and K. Suzuki. Secure generalized Vickrey auction without third-party servers. In *Proceedings of the 8th Annual Conference on Financial Cryptography (FC)*, volume 3110 of *Lecture Notes in Computer Science (LNCS)*, pages 132–146. Springer-Verlag, 2004.