Efficient Cryptographic Protocols Realizing E-Markets with Price Discrimination

Aggelos Kiayias[1] and Moti Yung[2]

[1] Computer Science and Engineering,
University of Connecticut Storrs, CT, USA
aggelos@cse.uconn.edu
[2] RSA Laboratories, Bedford, MA, USA and Computer Science,
Columbia University, New York, NY, USA
moti@cs.columbia.edu

Abstract. Perfect (or "first degree") Price Discrimination is a standard economic practice that is used to increase the pricing effectiveness over a diverse population of prospective buyers. It is done by selling to different buyers at different prices based on their respective willingness to pay. While the strategy achieves Pareto efficiency, there are a number of problems in realizing and giving incentive to buyers to participate (and stay) in a market with price discrimination. This is especially true in an open process (like Internet commerce), where parties may learn about their price's individual standing (within the group of buyers) and may withdraw due to being relatively "over-charged" or may "resell" due to getting the goods at a relatively low price. We investigate the difficulties of realizing perfect price discrimination markets when full information is available to the participants even under the assumption of using standard cryptographic techniques. We then propose a "fair solution" for price discrimination in e-markets: using efficient cryptographic protocols (much more efficient than secure function evaluation protocols) we give incentives to users to stay in a market that utilizes price discrimination. Our protocols assure that the seller obtains the total revenue it expects and no buyer learns the price of other buyers. In addition, each buyer gets a "fair" discount off the surplus (the accumulated suggested payments by buyers minus the seller's expected revenue) when applicable and the seller may get part of the surplus as well. Further, the seller gets to learn the market "willingness to pay" (for potential future use), while this knowledge does not affect the pricing of the current e-market instance. Along the way we investigate the cryptographic primitive of "robust distributed summation" that may be of independent interest as a protocol construction.

1 Introduction

Economics is a field where decision making is being studied and where methods, mechanisms and procedures are developed to solve market situations under rationality and other assumptions about agents. On the other hand computer science and cryptography in particular, study manipulation and exchanges of

G. Di Crescenzo and A. Rubin (Eds.): FC 2006, LNCS 4107, pp. 311–325, 2006.

information in the electronic world, based on the computational model and computational environment constraints. In this paper we investigate the concept of using cryptographic protocols to solve problems of economics: markets, exchanges and collaboration of agents, can be assisted in various environments where exchange and combination of information is done in a setting that due to partial-information constraints cryptography can help. We demonstrate the usefulness of what may be called a "Crypto-Economics" proposal by showing how under certain operational constraints we can employ efficient cryptographic protocols to realize e-markets with first order price discrimination.

Price Discrimination is a standard economic practice that can be used to increase the market efficiency in cases where there is a diverse population of prospective buyers of a certain good or service. We deal here with first-degree price discrimination, where users express their "willingness to pay" which is accumulated and if it is above or equal the revenue the seller wants to obtain (this is called "the surplus"), then the market transaction is performed. While having "Pareto efficiency" (no party's situation can be made better off without making someone else worse off), the practice generates "consumer discomfort" since users may realize that relative to other users they have paid too much (discouraging loyalty) or too little (encouraging re-sale). We refer to the works of Varian [Va96] and Odlyzko [Od02] for more information of such issues in the setting of Internet commerce. The present work focuses on the problem of how to incentivize agents to remain in such markets not using economic means and business tricks (versioning, bundling, etc. see below) but rather through cryptographic techniques.

Suppose that a seller S advertises a good and attracts a number of prospective buyers $B_1, \ldots, B_n$. Let ρ be the total revenue that S wishes to extract out of selling the good. Suppose additionally that buyer B_i is willing to pay an amount v_i for obtaining the advertised good. In the perfect price discrimination setting the good is sold to buyer B_i at price v_i resulting in a total revenue of $\sum_{i=1}^{n} v_i$. If the summation exceeds ρ the transaction can take place.

To illustrate the benefits of price discrimination consider the following scenario: the seller wishes to sell an advertised good expecting a total revenue of $\rho = \$1500$. Three buyers express interest for the product with respective prices $v_1 = \$400$, $v_2 = \$600$ and $v_3 = \$800$. Without price discrimination the seller can set an average price of $\$1500/3 = \$500$. This will result in a revenue of \$1000 that is below the expectations of the seller (this is because the product would be too expensive for the first buyer). In the perfect price discrimination setting the total revenue is \$1800 but it is quite likely that the third buyer will not be willing to pay \$500 more than the first buyer for the same product (and similarly for the second).

Indeed, this setting, although ideal from a simple economical point of view (the "name your price" practice as in priceline.com and other Internet selling sites follow this strategy), it is not practical in many cases, and it is hardly possible to enforce in cases where potential buyers have concerns about fairness. Indeed, economists have noticed that there are numerous problems that arise

in the employment of price-discrimination despite the fact that in most settings it is beneficial both for the buyers as well as for the seller. One of the most important problems is convincing the buyers to accept the price discrimination scheme. Possible techniques include versioning and bundling, nevertheless these do not apply to all settings and types of goods. When the same product or service is to be sold in different prices, buyers may worry about their relative price and the unfairness of the process. They may realize that they pay "too much" and hold it against the seller, or may realize that they can resell in case their price was low.

We would like a mechanism that employs price discrimination, yet assures some conditions of fairness. First we view a number of potential realizations of a market with perfect price discrimination. We explain the difficulties in these realizations. Then, a fair solution that we propose in this setting is to use cryptographic methods to enable the sale only under certain conditions and without leaking private information. In particular, the protocol allows the transaction to take place only if the seller gets the expected revenue he has expressed ρ. The solution then yields a discount to all buyers (this will be the incentive to remain in the price discriminated market). The price discrimination happens in an *oblivious manner* and is distributed among all buyers based on buyer i calling a price v_i that remains private. In this case all users will be willing to participate motivated by the potential discount. Moreover, they will be given a (cryptographic) guarantee that their price will be kept secret while computing the discount and they will not be treated unfairly, in case the seller's constraints are satisfied.

The total discount that can be applied is $\sum_{i=1}^{n} v_i - \rho$ (the slack), and in the particular example above it is \$300. This slack can be divided among the buyers using some method accepted by all parties. In the simplest example we consider, all buyers should get a $\$300/3 = \$100$ discount: in particular the buyers will pay $\$300, \$500, \$700$ respectively. Provided that no buyer learns the price paid by other buyers, all parties are motivated to participate in the protocol: the seller obtains the total revenue he expected; all buyers get a discounted price. In our second solution we consider weighted discounts where the slack will be divided according to the relative bids of each party; in this setting the buyers will pay $\$333, \$500, \$667$ respectively (i.e., the higher bidder receives more discount).

Let us complete the section by clarifying the distributed secure computation problem that we consider:

- $n+1$ active participants: S denotes the seller, and $B_1, \ldots, B_n$ the prospective buyers.
- The private inputs of the participants $S, B_1, \ldots, B_n$ are $\rho, v_1, \ldots, v_n$.
- The goal of the the protocol is the following: provided that $\sum_{i=1}^{n} v_i \geq \rho$, each buyer B_i privately computes a value v_i' so that (i) no other participant gets to learn the value v_i; (ii) the values $\sum_{i=1}^{n} v_i' = \rho$; (iii) $v_i' \leq v_i$.
- In case of a transaction, the seller gets the revenue he declared as its desire (this is an incentive for the seller to input its "real willingness for revenue."

- The seller gets a feedback about the "market willingness to pay as a whole" which does not influence the current transaction. On the other hand, the seller does not learn the individual price bids (v_i's).
- We consider two alternative methods for the calculation of the discounted values v'_k:
  (a) Same discount for all buyers: $v'_k = v_k - \frac{\sum_{i=1}^{n} v_i - \rho}{n}$.
  (b) Weighted Discount: $v'_k = v_k \frac{\rho}{\sum_{i=1}^{n} v_i}$.

The cryptographic protocol realizing this "oblivious price discrimination method" assures secrecy (no buyer or seller learns the initial price bid of another buyer), auditability (the buyers are sure the realization was calculated correctly) and robustness (the solution can be calculated distributedly with no disruptions).

We can also vary the treatment of the surplus. The parties may decide a-priori to split it between the buyers and the seller (in some chosen way that can be calibrated); this allows markets that support sellers based on global conditions. For example, in depressed markets, the buyers may be willing (or be regulated) to give half the surplus to the seller, in order to keep it in business.

1.1 Motivation: Problems with Potential Realizations

The motivation for our cryptographic protocol designs is illustrated by the following potential "solutions" to the above problem, that are not satisfactory:

1. Non-Cryptographic Solution #1: All buyers send their values v_i to the seller who, in turn, returns the values v'_i. **Problem:** This may result in over-pricing since there is no guarantee that the seller will not get for more revenue than it expects and the fairness-minded group of buyers do not like this situation. (This is the "name your price" mechanism).
2. Non-Cryptographic Solution #2: The seller publicizes the expected revenue ρ. **Problem:** It is hard to achieve price-discrimination in this setting, since the seller has "revealed its cards" and it is unreasonable to expect that buyers would be willing to pay more than ρ/n.
3. Commitment Based Setting: Seller commits to his value and buyers communicate their prices to him. **Problem:** Buyers only collectively can verify the commitment of the seller with respect to the slack. Without employing any other cryptographic techniques, this results in revealing all buyers' prices and this level of insecurity may allow buyers to learn prices of other buyers. Naturally a trusted third party may carry the checking but this would centralize too much trust on a single entity.
4. Cryptographic Setting: The seller and buyer may employ generic secure function evaluation [Ya86, GMW87] to compute the total discount $\sum_{i=1}^{n} v_i - \rho$ or the discount ratio $\rho / \sum_{i=1}^{n} v_i$. These general procedures, in fact, can solve any mechanism design problem as noted in [Ni99]. **Problem:** While these methods are "universal protocols" that solve any problem, these methods are computationally very demanding and serve only as plausibility results; instead here we seek more efficient solutions of a specific secure function evaluation problem, (cf. [G97]).

We assume that a seller and a buyer have signed a contract clarifying the rules of payments after the protocol is over and the confidentiality of payments; this can also use cryptography (digital signing). In addition, we need to assure the seller its revenue. The protocol has to be auditable by the participants assuming they are fairness-minded users who monitor the publicly available protocol transcript. Furthermore, the tool has to be robust against users who attempt to fail the system.

2 Tools

In this section we go over some basic cryptographic tools that we employ in our construction. In particular, Homomorphic Encryption Schemes, Paillier Encryption, Proofs of Knowledge, Interval Proofs, etc. Readers familiar with these tools may skim read this section and move on to section 3.

Homomorphic Encryption Schemes. An encryption scheme is a triple $\langle \mathcal{K}, \mathcal{E}, \mathcal{D} \rangle$. The key-generation $\mathcal{K}$ is a probabilistic TM which on input a parameter 1^w (which specifies the key-length) outputs a key-pair pk, sk (public-key and secret-key respectively).

The encryption function is a probabilistic TM $\mathcal{E}_{\mathsf{pk}} : \mathbb{R} \times \mathbb{P} \to \mathbb{C}$, where $\mathbb{R}$ is the randomness space, $\mathbb{P}$ is the plaintext space, and $\mathbb{C}$ the ciphertext space. When the $\mathbb{P} = \mathbb{Z}_a$ for some integer a, we will say that encryption function has "*additive capacity*" (or just capacity) a.

The basic property of the encryption scheme is that $\mathcal{D}_{\mathsf{sk}}(\mathcal{E}_{\mathsf{sk}}(\cdot, x)) = x$ for all x independently of the coin tosses of the encryption function $\mathcal{E}$. If we want to specify the coin tosses of $\mathcal{E}$ we will write $\mathcal{E}_{\mathsf{pk}}(r, x)$ to denote the ciphertext that corresponds to the plaintext x when the encryption function $\mathcal{E}_{\mathsf{pk}}$ makes the coin tosses r. Otherwise we will consider $\mathcal{E}_{\mathsf{pk}}(x)$ to be a random variable.

For homomorphic encryption, we assume additionally the operations $+, \oplus, \bigodot$ defined over the respective spaces $\mathbb{P}, \mathbb{R}, \mathbb{C}$, so that $\langle \mathbb{P}, + \rangle$, $\langle \mathbb{R}, \oplus \rangle$, $\langle \mathbb{C}, \bigodot \rangle$ are groups written additively (the first two) and multiplicatively respectively.

Definition 1. *An encryption function $\mathcal{E}$ is homomorphic if, for all $r_1, r_2 \in \mathbb{R}$ and all $x_1, x_2 \in \mathbb{P}$, it holds that $\mathcal{E}_{\mathsf{pk}}(r_1, x_1) \bigodot \mathcal{E}_{\mathsf{pk}}(r_2, x_2)$ equals $\mathcal{E}_{\mathsf{pk}}(r_1 \oplus r_2, x + y)$.*

Here we will employ a Homomorphic Encryption scheme due to Paillier, [Pai99], that is presented in the next section.

Paillier Encryption. We use the first encryption scheme presented in [Pai99]. It is a triple $\langle \mathcal{K}, \mathcal{E}, \mathcal{D} \rangle$, defined as follows: the key-generation $\mathcal{K}$ outputs an integer N, that is a product of two safe primes, and an element $g \in \mathbb{Z}^*_{N^2}$ of order a multiple of N. The public-key of the system pk is set to $\langle g, N \rangle$ and the secret-key sk is set to the factorization of N.

For a public-key $\langle g, N \rangle$, the encryption function $\mathcal{E}(r, x)$ equals the value $g^x r^N (\mathrm{mod} N^2)$ and the domains $\mathbb{P} := \mathbb{Z}_N$, $\mathbb{R} := \mathbb{Z}^*_N$, and $\mathbb{C} := \mathbb{Z}^*_{N^2}$. The operation $+$ is defined as addition modulo N, and the operations $\oplus, \bigodot$ are defined as multiplication modulo N^2. The decryption function $\mathcal{D}$ for a secret-key p, q it operates as follows: first it computes $\lambda := \lambda(N)$ the Carmichael function of

N, and given a ciphertext c, it returns $L(c^{\lambda}(\mathrm{mod}N^2))/L(g^{\lambda}(\mathrm{mod}N^2))$ where $L(u) = \frac{u-1}{N}$ and L is defined over the set of integers $\{u \mid u \equiv 1(\mathrm{mod}N)\}$.

Observe that $\langle \mathbb{P}, + \rangle$, $\langle \mathbb{R}, \oplus \rangle$ and $\mathbb{C}, \bigodot \rangle$ are all groups, and the encryption $\mathcal{E}$ is homomorphic with respect to these operations. Finally notice that the capacity of $\mathcal{E}$ is N.

Threshold Variant. A (t, m)-threshold homomorphic encryption scheme is a triple $\langle \mathcal{K}, \mathcal{E}, \mathcal{D} \rangle$ so that $\mathcal{K}$ is a protocol between a set of participants $A_1, \ldots, A_m$, that results in the publication of the public-key pk and the sharing of the secret-key sk so that any t of them can reconstruct it. Additionally, $\mathcal{D}$ is also a protocol between the participants $A_1, \ldots, A_m$ that results in the decryption of the given ciphertext in a publicly verifiable manner (i.e. each participant writes a proof that he follows the decryption protocol according to the specifications). Paillier encryption has a threshold variant see [FPS00, DJ00].

Proofs of Knowledge. Proofs of knowledge are protocols between two players, the Prover and the Verifier. In such protocols there is a publicly known predicate Q_y with parameter y for which the prover knows some witness x, i.e. $Q_y(x) = 1$. The goal of such protocols is for the prover to convince the verifier that he indeed knows such witness. The reader may refer to [DDPY94, CDS94] for descriptions of such protocols and their composition in AND-OR circuit fashion. We note that we will employ such protocols based on a computational soundness argument: i.e., soundness on the side of the prover will be ensured only under the presumed hardness of a certain computational problem. This technique was used in a number of previous works, notably in [FO97, DF02].

Interval Proof for Paillier Encryption. An interval proof shows that a committed integer value belongs to a certain interval. Such proofs have been used in a variety of settings, e.g. in group-signatures [ACJT00], or e-cash [CFT98]. In our protocol constructions we will need the design of an interval proof for an integer that is encrypted into a Paillier ciphertext. We start by presenting in figure 1 a basic interval proof for a commitment (for related previous work see also [CFT98, Bou00, KTY04]).

For the proof, we consider security parameters k, l and further let N' be a safe RSA Modulus with unknown factorization to the prover and verifier. Also let g_0 be an element that generates the quadratic residues in $\mathbb{Z}^*_{N'}$ and h a full order element inside $\langle g_0 \rangle$ with unknown discrete logarithm base g_0. Let $0, \ldots, B$ be the interval over which the prover will show that a committed value belongs to. The commitment scheme that is used is defined as follows: $V = g_0^x h^r (\mathrm{mod}N')$ where r is selected at random from the interval $\{0, \ldots, \lceil N'/4 \rceil\}$.

A crucial tool for the soundness of interval proofs is the Strong-RSA assumption, defined below:

Strong-RSA Assumption. Given N' and $v \in \mathbb{Z}^*_{N'}$ it is computationally hard to find $b \in \mathbb{Z}^*_{N'}$ and e a prime number such that $b^e = v(\mathrm{mod}N')$.

We remark that the proof of knowledge of figure 1 has the following properties: (1) If $x \in \{0, \ldots, B\}$ the honest prover always convinces the honest

Prover		Verifier
selects $\omega \in_R \{-2^{k+l}B, \ldots, 2^{k+l}B\}$		
$\eta \in_R \{-2^{k+l}\lceil N'/4\rceil, \ldots, 2^{k+l}\lceil N'/4\rceil\}$		
computes $W = g_0^{\omega} h^{\eta} (\mathrm{mod} N')$	$\xrightarrow{W}$	
	$\xleftarrow{c}$	$c \in_R \{0, \ldots, 2^k\}$
$d_1 = \omega - xc, d_2 = \eta - rc$ (in $\mathbb{Z}$)	$\xrightarrow{d_1, d_2}$	$d_1 \in_? \{-2^{k+l}B - (2^k - 1)B, \ldots, 2^{k+l}B\}$
		$g_0^{d_1} h^{d_2} V^c \stackrel{?}{=} W (\mathrm{mod} N')$

Fig. 1. Interval Proof $x \in \{0, \ldots, B\}$ for the commitment $V = g_0^x h^r$

verifier. (2) Conditional Soundness. A cheating prover using an integer $x \notin \{-2^{k+l+2}B, \ldots, B + 2^{k+l+2}B\}$ can succeed with probability less than 2^{-k}, under the Strong-RSA assumption. (3) The protocol is statistical zero-knowledge for a honest verifier with statistical distance negligible in the parameter l.

We can combine the above protocol using regular AND composition with a standard proof of knowledge of a Paillier encryption (e.g. as those presented in [DJ00]) in order to obtain a proof of knowledge that shows that a Paillier Encryption hides a value in the interval $\{-2^{k+l+2}B, \ldots, B + 2^{k+l+2}B\}$. Let g, N be a public-key for the Paillier encryption function with $N > N'$. This can be done as shown in figure 2.

Prover		Verifier
selects $r \in \{0, \ldots, \lceil N'/4\rceil\}$		
$\omega \in_R \{-2^{k+l}B, \ldots, 2^{k+l}B\}$		
$\eta \in_R \{-2^{k+l}\lceil N'/4\rceil, \ldots, 2^{k+l}\lceil N'/4\rceil\}$		
$u \in_R \mathbb{Z}_N^*$		
computes $W = g_0^{\omega} h^{\eta} (\mathrm{mod} N')$, $V = g_0^x h^r (\mathrm{mod} N')$,		
$U = g^{\omega} u^N (\mathrm{mod} N^2)$	$\xrightarrow{W,U,V}$	
	$\xleftarrow{c}$	$c \in_R \{0, \ldots, 2^k\}$
$d_1 = \omega - xc$, $d_2 = \eta - rc$ (in $\mathbb{Z}$)		
$z = uv^c (\mathrm{mod} N^2)$	$\xrightarrow{d_1, d_2, z}$	$d_1 \geq_? -2^{k+l}B - (2^k - 1)B$
		$d_1 \leq_? 2^{k+l}B$
		$g_0^{d_1} h^{d_2} V^c \stackrel{?}{=} W (\mathrm{mod} N')$
		$g^{d_1} z^N \stackrel{?}{=} U E^c (\mathrm{mod} N^2)$

Fig. 2. Interval Proof $x \in \{0, \ldots, B\}$ for the Paillier Encryption $E = g^x v^N (\mathrm{mod} N^2)$

Boudot, in [Bou00], improved interval proofs of the type described above so that they become tight (i.e. there does not exist a discrepancy between the interval used for completeness, and the interval used for soundness). We can combine Boudot's proof in a standard AND fashion with a proof of knowledge of a Paillier encryption as we did above in figure 2 in order to obtain a interval proof for a Paillier encryption.

Notation. In the sequel we will use the notation $Q_{\mathsf{interval}}^{E,[0..B]}$ for the predicate that is 1 for values x, v such that $E = g^x v^N (\mathrm{mod} N^2)$ and $x \in \{0, 1, \ldots, B\}$. We will

say that a player writes a proof for $Q^{E,[0..B]}_{\mathsf{interval}}$ when he executes an interval proof for the Paillier encryption E.

Proving Equality of Paillier Ciphertexts with Different Bases. Let g, N be a public-key for the Paillier encryption. Let g_0 be an additional value in $\mathbb{Z}^*_{N^2}$ with order a multiple of N. In this section we will show how the prover can show that two ciphertexts C, C' encrypted under the public-keys g, N and g_0, N can be shown to encrypt the same plaintext. We will denote the corresponding predicate by $Q^{C,C',g,g_0,N}_{\mathsf{equal}}$, i.e. $Q^{C,C',g,g_0,N}_{\mathsf{equal}}(x) = 1$ if and only if there exist $v, v' \in \mathbb{Z}^*_N$ such that $C = g^x v^N (\mathrm{mod} N^2)$ and $C' = g_0^x (v')^N (\mathrm{mod} N^2)$. The proof of knowledge is presented in figure 3.

Prover		Verifier
select $y, y' \in_R \mathbb{Z}^*_N$ and $r \in_R \mathbb{Z}_N$		
$A := g^r y^N (\mathrm{mod} N^2) \bmod N^2$, $A' = g_0^r (y')^N (\mathrm{mod} N^2)$	$\xrightarrow{A,A'}$	
	$\xleftarrow{c}$	$c \leftarrow_R \{0, 1, \ldots, N-1\}$
$z = yv^c$, $z' = y'(v')^c$, $s = r + cx (\mathrm{mod} N)$	$\xrightarrow{s}$	$g^s z^N \stackrel{?}{=} AC^c (\mathrm{mod} N^2)$
		$g_0^s (z')^N \stackrel{?}{=} A'(C')^c (\mathrm{mod} N^2)$

Fig. 3. Proof of knowledge for the predicate $Q^{C,C',g,g_0,N}_{\mathsf{equal}}$ where $C = g^x v^N (\mathrm{mod} N^2)$ and $C' = g_0^x (v')^N (\mathrm{mod} N^2)$

3 Price Discrimination Protocols

- Active Participants: The Seller (S), the prospective buyers ($B_1, \ldots, B_n$). All communication takes place through a "bulletin board," a model that abstracts away all lower level communication details [CF85].
- Inputs: the expected revenue $\rho \in \mathbb{Z}$ of the Seller. The maximum amount that player B_i is willing to spend $v_i < B$.
- Output: The seller computes the total contribution $\sum_{i=1}^{n} v_i$. Each buyer either,
  1. receives the discounted price v'_i, that has the properties (i) $v'_i \leq v_i$, (ii) $\sum v'_i = \rho$.
  2. receives a notification that the expected revenue of the Seller has not been met.
- Correctness. Each active participant computes the outputs as specified above.
- Security Specifications.
  1. Privacy. The initial amount that each Buyer is willing to spend is kept secret (modulo the information that is leaked by the results of the procedure). Formally, privacy is intended to be shown by comparison to the ideal implementation of the scheme using a trusted third party: All buyers and sellers transmit privately their values to the trusted third party who announces the output as defined above.

2. Robustness. No participant can prevent the procedure from terminating.
3. Verifiability. Participants' actions can be verified to follow the protocols' specifications.

As explained above, we will consider two discount schemes: (i) absolute discount where $v_i' = v_i - \frac{\sum_{i=1}^{n} v_i}{\rho}$, and (ii) weighted discount where $v_i' = v_i \frac{\rho}{\sum_{i=1}^{n} v_i}$.

We remark that in the absolute discount case, some buyers may compute a negative value as their final price v_i'. This is not inconsistent with the specifications of price discrimination with absolute discount (i.e. in this case these buyers may end up getting some credit for participating in the procedure).

Meeting the security specifications will rely on assuming the semantic security of Paillier scheme, and on the assumptions (and idealized model, if used) needed for the proofs of knowledge (as explained above). More detailed treatment will be given in the full version of this work.

3.1 Robust Private Distributed Summation

Our protocol constructions can be seen as modifications of a basic primitive which we call Robust Private Distributed Summation. The primitive may be of independent interest and as it is quite general we present it first. As a primitive it relates to homomorphic encryption based voting schemes, e.g. [CGS97, FPS00]; the goal of the primitive is to add a sequence of distributed numbers into their sum while at the same time avoiding wraparounds (as the calculation is performed in a finite ciphertext domain).

We stress that distributed summation is very different from e-voting. To begin with, the individual summation terms are not supposed to be revealed and as a result no e-voting procedure based on blind-signatures and/or anonymous channels is suitable. Further, in case of homomorphic encryption based voting the difference is that the range of each summation term may be exponentially large and thus standard OR proofs ranging through the entire allowed domain cannot be used to validate the encrypted bid. Moreover the range of the summation register is exponentially large itself. Finally, we only need to avoid wraparounds, and thus a tight range proof may even not be required. For the above reasons a new construction is in place to solve the distributed summation problem. Below we outline the protocol problem we intend to solve.

We describe the protocol below. Let $\langle \mathcal{K}, \mathcal{E}, \mathcal{D} \rangle$ be the (t, m)-threshold variant of Paillier encryption defined in section 2.

1. **Setup.** The authorities $A_1, \ldots, A_m$ execute the protocol $\mathcal{K}$ which results in the publication in the bulletin board of the public-key pk. At the same time the secret-key sk is shared amongst the authorities $A_1, \ldots, A_m$.
2. **Value Submission.** Each eligible player gets authorized to the bulletin board and reads the public-key pk of the system. The player P_i publishes the encryption $C[i] := \mathcal{E}_{\mathsf{pk}}(s_i)$, where $s_i \in \{0, \ldots, B\}$.
   At the same time the player must publish an interval proof to show that he is not exceeding the boundary B. So he writes a proof of knowledge for the predicate $Q_{\mathsf{interval}}^{C[i],[0..B]}$.

Parameters	$t, m, B \in \mathbb{Z}$. Number of players is n.
Tools	Paillier Encryption $\langle \mathcal{K}, \mathcal{E}, \mathcal{D} \rangle$ with capacity N such that $N > nB$.
Participants	A set of players $P_1, \ldots, P_n$. We also assume a set of authorities $A_1, \ldots, A_m$ (which may coincide or overlap with the set of players).
Input	Each player has a private input, an integer $s_i \in \{0, 1, \ldots, B\}$.
Output	The sum $\sum_{i=1}^{n} s_i$.
	Properties
Security	Any adversary that controls a number of participants so that less than t Authorities are controlled by the adversary is incapable of computing the private input of any of the participants, prior to the announcement of the sum, the output of the protocol.
Robustness	Any adversary that controls a number of participants so that less than $m-t$ Authorities are controlled by the adversary is incapable of preventing the publication of the output sum of the protocol.
Verifiability	Any third party can verify that the participants are following the protocol according to the specifications.

Fig. 4. Specifications of the Secure Distributed Summation Protocol

3. **Aggregation.** The bulletin board authority terminates the value submission phase, and it collects the encrypted ballots $C[1], \ldots, C[n]$ to compute the "summation ciphertext" $C_{\mathsf{sum}} = C[1] \bigodot \ldots \bigodot C[n]$. Observe that due to the homomorphic property of the Paillier encryption scheme it holds that C_{sum} is indistinguishable from encryptions of the value $T := \sum_{i=1}^{n} s_i$.
4. **Decryption and Announcement of the Sum.** The authorities $A_1, \ldots, A_m$ execute the protocol $\mathcal{D}$ to reveal the the value T. Note that due to the capacity assumption there will be no wrap-arounds during the computation of the summation ciphertext C_{sum}.

Based on the properties of the cryptographic tools that are employed in our scheme, one can formulate and prove a theorem as follows:

Theorem 1. *The Distributed Summation Protocol defined above satisfies Security, Robustness, and Verifiability, under the assumptions (i) Semantic Security of Paillier encryption (ii) the strong-RSA assumption to show the soundness of the necessary interval proofs.*

Remark 1. A summation protocol for encrypted values appeared as part of a different application scenario in [C01]; the approach taken there was based on cut and choose techniques for ensuring proper value selection and thus compared to the solution presented here is much less efficient.

Remark 2. By calibrating the capacity of the encryption function to be $N > nB2^t$ where t is an appropriately chosen security parameter one can use more efficient proofs of knowledge compared to the ones of Boutot [Bou00] as the one in figure 2. We omit further details for the full version.

3.2 The Absolute Discount Protocol

Let $\langle \mathcal{K}, \mathcal{E}, \mathcal{D} \rangle$ be the (t, m)-threshold version of the Paillier homomorphic encryption as defined in section 2. We assume a set of "authorities" $A_1, \ldots, A_m$ that may either overlap with some of the active participants $S, B_1, \ldots, B_n$ or they may be third parties that participate in the procedure. We will denote by B an upper bound on the maximum price that a certain buyer is willing to pay to the seller.

Initialization. The Authorities $A_1, \ldots, A_m$ execute the key-generation protocol $\mathcal{K}$. This results in writing the public-key g, N in the bulletin board of the system. We assume that $N > 2Bn$ where B is the bound to the input of each buyer[1].

Depositing the Price-bids. Each buyer B_i selects his maximum price bid v_i and publishes $C[i] = \mathcal{E}(v_i) = g^{v_i} x_i^N (\mathrm{mod} N^2)$ where $v_i \in \{0, \ldots, B\}$ and $x_i \in_R \mathbb{Z}_N^*$.

In addition B_i writes a proof for the predicate $Q_{\mathsf{interval}}^{C[i],[0..B]}$, to ensure that the bid v_i is in the range $\{0, \ldots, B\}$.

The seller writes the encryption $C = \mathcal{E}(\rho) = g^\rho x^N (\mathrm{mod} N^2)$, together with an interval proof for the predicate $Q_{\mathsf{interval}}^{C,[0..\frac{N}{2}]}$.

Closing the Deposition Phase. The bulletin board server closes the deposition phase by signing and time-stamping the contents of the bulletin board. All proofs of knowledge are checked to ensure that all buyers have conformed to the interval requirement.

Computation of the Total Discount. The bulletin board server computes $C_{\mathsf{t-disc}} = \prod_{i=1}^n C[i]/C$. Observe that due to the capacity of the encryption function $\mathcal{E}$ and the homomorphic properties of the Paillier encryption function it follows that $C_{\mathsf{t-disc}}$ is a valid Paillier ciphertext that hides the integer $D = \sum_{i=1}^n v_i - \rho \in \{-\lfloor \frac{N}{2} \rfloor, \ldots, 0, \ldots, \lfloor \frac{N}{2} \rfloor\}$.

The authorities $A_1, \ldots, A_m$ execute the decryption protocol on the ciphertext $C_{\mathsf{t-disc}}$ to reveal the value D. Note that if $D < 0$ all parties conclude that the market cannot be realized (too high revenue expected / too little market interest).

Computation of Individual Prices. Provided that $D \geq 0$ each buyer B_i computes his discounted value as follows: $v_i' := v_i - \frac{D}{n}$. The Seller can also compute the gross value that its offer raised by calculating the sum $\sum_{i=1}^n v_i = D + \rho$.

Observe that the publication of the total discount $D = \sum_{i=1}^n v_i - \rho$ is not inconsistent with the security properties dictated by an ideal implementation of the absolute discount protocol since the value D is accessible also in the ideal implementation by each one of the buyers: indeed given v_i, v_i' in the ideal implementation one can compute D as follows: $D = n(v_i - v_i')$ (recall that we assume that the total number of buyers n is common knowledge).

[1] Note that the bound B will be chosen to be substantially larger from the actual bid and will be the same for all buyers. The only purpose of the bound is to ensure that no wraparound occurs during the modular addition of the buyers' bids. It will be easy to select B high enough so that no information about the buyers' prices leaks from the disclosure of B.

Under our cryptographic assumptions, one can argue that the absolute discount protocol above satisfies privacy, robustness and verifiability.

3.3 The Weighted Discount Protocol

The main technical issue for the weighted discount protocol compared to the absolute discount protocol is that the computation of the summation of the bid-prices appears in the denominator of the discounted final price values.

As in the case of the absolute discount protocol, let $\langle \mathcal{K}, \mathcal{E}, \mathcal{D} \rangle$ be the (t, m)-threshold version of the Paillier homomorphic encryption as defined in section 2. As before, we also assume a set of "authorities" $A_1, \ldots, A_m$ that may either overlap with some of the active participants $S, B_1, \ldots, B_n$ or they may be third parties that participate in the procedure.

Initialization. The Authorities $A_1, \ldots, A_m$ execute the key-generation protocol $\mathcal{K}$. This results in writing the public-key g, N in the bulletin board of the system. We assume that the capacity of the encryption N satisfies $N > ABn$ where B is the bound to the input of each buyer, and A is an integer parameter.

Seller-Initialization. The seller computes $\rho' := \lceil 10^e \frac{1}{\rho} \rceil$ where e is a public parameter so that $\rho' < A$. The seller publishes the encryption $C = \mathcal{E}(\rho') = g^{\rho'} x^N (\mathrm{mod} N^2)$. together with an interval proof for the predicate $Q^{C,[0..A]}_{\mathsf{interval}}$.

Depositing the Price-bids. Each buyer B_i selects his maximum price bid v_i and publishes $C[i] = C^{v_i} x_i^N (\mathrm{mod} N^2)$ and $C'[i] = g^{v_i} (x'_i)^N (\mathrm{mod} N)$ where $v_i \in \{0, \ldots, B\}$ and $x_i \in_R \mathbb{Z}^*_{N^2}$.

In addition B_i writes a proof for the predicate $Q^{C'[i],[0..B]}_{\mathsf{interval}}$, to ensure that the bid v_i is in the range $\{0, \ldots, B\}$, and additionally it writes a proof for the predicate $Q^{C'[i],C[i],g,C,N}_{\mathsf{equal}}$ to ensure that the two Paillier ciphertexts $C[i], C'[i]$ that have different bases hide the same value v_i.

Closing the Deposition Phase. The bulletin board server closes the deposition phase by signing and time-stamping the contents of the bulletin board. All proofs of knowledge are checked to ensure that all participants have conformed to the interval requirements.

Computation of the Discounts. The bulletin board server (or any observer) computes $C_{\mathsf{factor}} = \prod_{i=1}^n C[i]$. Observe that due to the capacity of the encryption function $\mathcal{E}$ and the homomorphic properties of the Paillier encryption function it follows that C_{factor} is a valid Paillier ciphertext that hides the integer $F = \rho' \sum_{i=1}^n v_i$.

The authorities $A_1, \ldots, A_m$ execute the decryption protocol on the ciphertext C_{factor} to reveal the value F. Note that if $F < 10^e$ all parties conclude that the market cannot be realized (too high revenue expected / too little market interest).

Otherwise, each buyer B_i computes his discounted price bid by as follows

$$v'_i := v_i \frac{10^e}{F} = v_i \frac{10^e}{\lceil \frac{10^e}{\rho} \rceil \sum_{i=1}^n v_i} \approx v_i \frac{\rho}{\sum_{i=1}^n v_i}$$

The Seller can also compute the total contribution $\sum_{i=1}^n v_i = F/\rho'$.

As an example of the above computation consider the case where $\rho = 500$ and there are three prospective buyers with initial price bids $v_1 = 60, v_2 = 300, v_3 = 400$. Let $e = 4$. In this case $\rho' = 20$ and $F = \rho' \sum_{i=1}^{n} v_i = 15200$. It follows that $\frac{10^e}{F} = 0.657$ i.e. the discounted prices will be $v_1' = 39.5$, $v_2' = 197.2$, $v_3' = 262.9$ (note that round-up can be used to ensure that the discounted values are not below the expected revenue).

Observe that the publication of the factor F, as it was the case with the total discount D for the absolute discount protocol, is not inconsistent with the security properties dictated by an ideal implementation of the weighted discount protocol since the value F is accessible also in the ideal implementation by each one of the buyers: indeed given v_i, v_i' in the ideal implementation one can compute F as follows: $F \approx v_i / v_i'$.

Under our cryptographic assumptions, one can prove that the absolute discount protocol above satisfies privacy, robustness and verifiability.

3.4 Variations on Dealing with the Surplus

In our two price-discrimination protocols, the surplus was divided among the buyers so that they could obtain a discount on their initial price bid. The seller on the other hand obtained exactly the price he named.

This is not satisfactory in some settings and for the purpose to increase the incentive of the seller, one can have some of this surplus actually be returned to the seller. We examine how this simple modification can be implemented in both of our protocols to give half the slack to the seller (other fractions are also easily implementable):

1. For the Absolute Discount Protocol: in the computation of the final prices v_i' each buyer divides the total discount D by 2, and computes he discounted price as $v_i' = v_i - \frac{\sum_{i=1}^{n} v_i - \rho}{2n}$. This will give half of the "slack" $\sum_{i=1}^{n} v_i - \rho$ back to the seller.
2. For the Weighted Discount Protocol: the multiplier $mult = \frac{10^e}{F}$ that satisfies $0 < mult < 1$ that is computed at the end of the protocol is modified to multiplier $mult' := mult + \frac{1 - mult}{2}$. Subsequently all buyers are using $mult'$ in order to compute their final discounted value.

4 Cryptographic Infrastructure for Transaction Support

A major advantage of the procedure of computing the discounted prices and the market be means of cryptographic protocol is that it is possible to use further Cryptographic tools to assist in the continuation of the transaction, after prices have been determined. The additional support can help maintaining certain privacy and enforcement properties that markets need. We will only briefly survey the methods; a more detailed description will be given in the next version.

4.1 Payment Services

To actually complete the transaction, the buyers should present a proof that they actually were assigned a certain price discount. While the total discount is publicly available on the bulletin board, this does not indicate the exact amount that a certain buyer is supposed to pay. In fact, it may be the case that an agent cheats and claims that he was actually awarded a larger discount than the one that was actually assigned by the protocol. The discrepancy would only be revealed after the last participating buyer submits his payment. As a result it is crucial during payment to produce some *proof* that a certain price bid was made. This is possible in our setting since the bid of the user appears encrypted on the bulletin-board, and acts as a public commitment to his original bid. As a result after the termination of the protocol the buyer can open the random pad that he used in the encryption of his bid and thus convince a third party (e.g. a payment server) of the real value of his discount; note that the total discount is publicly available information, and thus once the payment server is convinced that the original price bid of the user is v_i, he can verify whether the claimed discounted price by re-computing v_i' (depending on the protocol used weighted or absolute discount). Convincing can also be done in a zero-knowledge fashion. The payment server can serve as a trusted interface providing anonymity to users or a combination of anonymity and affinity programs based on user accounts.

4.2 Reducing Reselling Potential Using "Receipt-Freeness"

The ability of the buyers to present proofs of the original bids is useful on the one hand for the implementation of the final monetary transaction part, however it also raises some concerns. In particular, buyers are capable of proving to other potential buyers their price which will encourage "resell market." To prevent this we can have a private re-randomizing server (which is used in "voting protocols" to provide receipt-freeness, [Poi00]), and this server makes sure that the buyer cannot convince a third party what his price is.

Further, the same server can be the payment server above, thus it is possible for a user to be able to convince the server and only the server itself of his/her price.

References

[ACJT00] Giuseppe Ateniese, Jan Camenisch, Marc Joye and Gene Tsudik, *A Practical and Provably Secure Coalition-Resistant Group Signature Scheme*, CRYPTO 2000.

[Ben87] Josh Benaloh, *Verifiable Secret-Ballot Elections*, PhD Thesis, Yale University, 1987.

[Bou00] Fabrice Boudot, *Efficient Proofs that a Committed Number Lies in an Interval*, Eurocrypt 2000.

[CFT98] Agnes Hui Chan, Yair Frankel, Yiannis Tsiounis, *Easy Come - Easy Go Divisible Cash*, EUROCRYPT 1998.

[CF85] Josh D. Cohen (Benaloh) and Michael J. Fischer, *A Robust and Verifiable Cryptographically Secure Election Scheme*, FOCS 1985.

[CGS97] Ronald Cramer, Rosario Gennaro and Berry Schoenmakers, *A Secure and Optimally Efficient Multi-Authority Election Scheme*, EUROCRYPT 1997.

[CDS94] Ronald Cramer, Ivan Damgård and Berry Schoenmakers, *Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols*, CRYPTO 1994.(a personal generator for G).

[DF02] Ivan Damgård and Eiichiro Fujisaki, *A Statistically-Hiding Integer Commitment Scheme Based on Groups with Hidden Order* ASIACRYPT 2002, pp. 125-142

[C01] Giovanni Di Crescenzo, *Privacy for the Stock Market*, Financial Cryptography 2001, pp. 269-288

[DJ00] Ivan Damgård and Mats Jurik, *A Generalisation, a Simplification and Some Applications of Paillier's Probabilistic Public-Key System*, Public Key Cryptography 2001, pp. 119-136.

[DDPY94] Alfredo De Santis, Giovanni Di Crescenzo, Giuseppe Persiano, Moti Yung, *On Monotone Formula Closure of SZK*, FOCS 1994.

[FS87] Amos Fiat and Adi Shamir, *How to Prove Yourself: Practical Solutions to Identification and Signature Problems*, CRYPTO 1986.

[FPS00] Pierre-Alain Fouque, Guillaume Poupard and Jacques Stern, *Sharing Decryption in the Context of Voting or Lotteries*, In the Proceedings of Financial Cryptography 2000.

[FO97] E. Fujisaki, T. Okamoto, Statistical Zero Knowledge Protocols to Prove Modular Polynomial Relations, Crypto 97, LNCS 1294, pp. 16-30, 1997.

[G97] S. Goldwasser. *Multi-party computations: Past and present.* (invited talk), PODC'97, pages 1–6.

[GMW87] O. Goldreich, S. Micali, and A. Wigderson, *How to play any mental game*, Proceedings of the Nineteenth annual ACM Symp. Theory of Computing, 1987.

[KTY04] Aggelos Kiayias, Yiannis Tsiounis, Moti Yung, *Traceable Signatures*, EUROCRYPT 2004, pp. 571-589.

[Ni99] N. Nisan, *Algorithms for Selfish Agents: mechanism design for distributed computation*, STACS 99.

[Od02] A. Odlyzko, *Privacy, Economics, and Price Discrimination on the Internet*, First Workshop on Economics and Information Security, Berkeley, 2002.

[Pai99] Pascal Paillier, *Public-Key Cryptosystems Based on Composite Degree Residuosity Classes*, EUROCRYPT 1999.

[Poi00] David Pointcheval, *Self-Scrambling Anonymizers*, Financial Cryptography 2000, pp. 259-275.

[Rab83] Michael Rabin, *Transactions protected by beacons*, Journal of Computer and System Sciences, Vol. 27, pp 256-267, 1983.

[Va96] Hal R. Varian, *Differential Pricing and Efficiency*, First Monday, peer-reviewed journal on the Internet. 1996. http://www.firstmonday.dk/issues/issue2/different/

[Ya86] A. Yao, *How to generate and exchange secrets*, IEEE FOCS, pages 162–167, 1986.