Research Article

PP-VCA: A Privacy-Preserving and Verifiable Combinatorial Auction Mechanism

Mingwu Zhang1,2,3 and Bingruolan Zhou2,3

1School of Computer Science and Information Security, Guilin University of Electronic Technology, China
2State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, China
3School of Computers, Hubei University of Technology, China

Correspondence should be addressed to Mingwu Zhang; csmwzhang@gmail.com

Received 24 June 2020; Revised 22 August 2020; Accepted 18 September 2020; Published 20 October 2020

Academic Editor: Weizhi Meng

Copyright © 2020 Mingwu Zhang and Bingruolan Zhou. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Combinatorial auctions can be employed in the fields such as spectrum auction, network routing, railroad segment, and energy auction, which allow multiple goods to be sold simultaneously and any combination of goods to be bid and the maximum sum of combinations of bidding prices to be calculated. However, in traditional combinatorial auction mechanisms, data concerning bidders’ price and bundle might reveal sensitive information, such as personal preference and competitive relation since the winner determination problem needs to be resolved in terms of sensitive data as above. In order to solve this issue, this paper exploits a privacy-preserving and verifiable combinatorial auction protocol (PP-VCA) to protect bidders’ privacy and ensure the correct auction price in a secure manner, in which we design a one-way and monotonically increasing function to protect a bidder’s bid to enable the auctioneer to pick out the largest bid without revealing any information about bids. Moreover, we design and employ three subprotocols, namely, privacy-preserving winner determination protocol, privacy-preserving scalar protocol, and privacy-preserving verifiable payment determination protocol, to implement the combinatorial auction with bidder privacy and payment verifiability. The results of comprehensive experimental evaluations indicate that our proposed scheme provides a better efficiency and flexibility to meet different types of data volume in terms of the number of goods and bidders.

1. Introduction

1.1. Backgrounds. Combinatorial auctions allow multiple goods to be sold simultaneously and any combination of goods to be bid, which provides a vivid and wide auction application on the Internet with the online e-commerce enabling consumers to complete a variety of complex activities, such as bank account deposit withdrawal, commodity trading service, and transaction information inquiry [1]. The auction is gradually changing from traditional auction to electronic auction and becoming an important part of e-commerce. For example, spectrum [2, 3] and energy [4] can be auctioned through the networks. The electronic auction system generally consists of an auctioneer, several sellers, and bidders. The seller entrusts the auctioneer to arrange the auction, accept the bids, and declare the winner [6]. Combinatorial auction is an important part of electronic auction, which is more scalable and can adapt to more complex demands. In a single auctioneer combinatorial auction, the auctioneer sells multiple heterogeneous goods simultaneously and bidders bid on any combination of the goods (called bundle or set) instead of just ones [7], which have been researched extensively because of the generality and scalability of on-growing applications [8].

Privacy-preserving combinatorial auction protocols usually employ the cryptographic technique to protect bidders’ private information. When the auction terminates, only the auction outcomes, i.e., who are winners and the corresponding payments, are revealed. In the auction process, the losers’ bids and bundles are kept private since the auctioneers might use losers’ bids to maximize their revenues in future auctions [6]. For example, the average value of losers’ bids can motivate auctioneers to increase the starting price in future auction of similar goods. In addition, private information of
bidders, such as bundle and bids, can be used to disclose personal preference and competitive relationship. In an auction system, there is serious competition between bidders, and this information is vital and needs to be protected.

1.2. Scenario and Application. Assume that an auctioneer publishes the information of some goods simultaneously on the Internet. Product numbers are labelled from #1 to #10. Every bidder chooses the sequence of the good number that he wants to own (i.e., bundle) and then provides the price that he is willing to pay (i.e., bid). The chosen list is described in Table 1.

Every bidder computes \( \text{average value} = \text{Bid}/(\text{Bundle} \mid ) \), where |Bundle| is the number of products in the bundle. The auctioneer picks out the largest average value 7750 and finds that #4 and #7 are still available, which means \( b_3 \) is the winner of the first round. In the second round, the auctioneer finds that \( b_2 \)’ average value is the largest. However, \( b_2 \)’ bundle contains one good that is already auctioned (#4), which means \( b_2 \) cannot be a winner. The auctioneer will choose all the winners in this way.

In private-preserving combinatorial auction, a crucial issue to be solved is how to pick out a set of disjoint goods under the price value of which is the maximized. Actually, this problem can be classified as an optimization problem. In [9], Zhang et al. proposes a privacy-preserving optimization for distributed fractional knapsack, which uses the greedy algorithm to find an optimal solution. Suzuki and Yokoo [10, 11] introduce dynamic programming to solve the winner determination problem on finding the shortest path of the directed graph [12]. However, the schemes in [10–12] may lead to a superpolynomial run time when the combinatorial auction parameters, i.e., the number of bidders and the number of goods, increase rapidly [13].

Threshold secret sharing schemes can also be used to solve the privacy-preserving problem in combinatorial auctions. For example, Kikuchi and Thorpe [14] proposed a privacy-preserving combinatorial auction protocol which employed a Shamir secret sharing scheme to share bids between multiple auctioneers, which allows any entity to detect misbehavior of bidders and auctioneers. Considering the high communication cost in [14], Hu et al. [15] provided an authentication property without increasing the communication cost in combinatorial auctions. Homomorphic encryption provides an available approach to protect each bidding value with a vector of ciphertext and then guarantees the auctioneer to figure out the maximum value securely [16–20]. In order to improve the performance, Xu et al. [21] give the comparison of different sorting algorithms and show that different sorting algorithms may have great effect on the performance of the protocol.

1.3. Organization. The remainder of this paper is organized as follows: We provide an overview of related work and background in Section 2. In Section 3, we introduce some terms used in the paper and provide the system framework, adversary model, and security requirement. In Section 4, we introduce the technology used in the paper. We provide our concrete scheme in Section 5 and give the security analysis in Section 6. The feature comparison and performance analysis are presented in Section 7. Finally, we draw our conclusion in Section 8.

2. Background and Related Work

2.1. Backgrounds of Combinatorial Auction. The traditional combinatorial auction includes one auctioneer and N bidders, as shown in Figure 1. The auctioneer is responsible for arranging the auction, accepting the bids, and declaring the winner. This process consists of two steps. Firstly, the auctioneer will send the information of goods to be auctioned to N bidders. Bidders give the sequence of goods that they want to obtain (called bundle) and quotation for bundle (called bid). The auctioneer selects the winner and announces the results according to some mechanism. Then, the auctioneer determines the price that the winner should pay. Notice that the winner’s bid and payment are not necessarily equal.

When the auctioneer picks out the winners, the main goal is to maximize social welfare, which is the sum of the winners’ bids. In this process, we should ensure that no information about the others’ bundles and bids are released. Also, the winner’s payment determination should be verifiable [22].

In order to reduce the losses caused by collusion and cheating among bidders, the famous Generalized Vickrey Auction (GVA) strikes a balance between risk and profit, in which the GVA is a sealed bid auction where auction goods are sold to bidders at the second highest price, which guarantees the authenticity of the auction while maximizing the interests of the auctioneer and bidders. However, the implementation of GVA is NP-hard even under the assumption of single-minded bidders. Zhang et al. [23] investigated the impact on such mechanisms of replacing exact solutions by approximate ones and proposed a particular greedy optimization method, which could guarantee the truthfulness of the auction.

2.2. Related Work. Currently, there exist several approaches to achieve privacy-preserving secure combinatorial auction, such as dynamic programming, Shamir’s threshold secret sharing scheme, homomorphic encryption, and secure multi-party computation.

| ID   | Bundle          | Bid     | Bidding price |
|------|-----------------|---------|---------------|
| b_1  | (#1, #7, #8)    | 10000   | 3333          |
| b_2  | (#2, #4, #10)   | 15000   | 5000          |
| b_3  | (#4, #7)        | 15500   | 7750          |
| b_4  | (#1, #5, #9, #10) | 14500   | 3625          |
| b_5  | (#3, #6, #7, #9, #10) | 17000 | 3400          |
| b_6  | (#2, #8, #9)    | 13000   | 4333          |
| b_7  | (#5, #9)        | 9000    | 4500          |
| b_8  | (#6, #8, #9)    | 14000   | 4667          |
Sakurai et al. [24] and Sandholm [25] employed dynamic programming to combinatorial auction. However, with the increase of the number of bidders and goods, dynamic programming will lead to nonpolynomial computation cost. Kikuchi and Thorpe [14] proposed a privacy-preserving combinatorial auction using Shamir’s threshold secret sharing scheme, and through further improvements, Hu et al. [15] presented a method to reduce the communication cost and that could resist the collusion attack and passive attack.

Some combinatorial auction protocols are based on homomorphic encryption technique in ciphertext fields [16–20]. However, these protocols need a high computational cost. Palmer et al. [26] employed the technique of secure multiparty computation to implement privacy-preserving combinatorial auction, where the protocol is not scalable since the inputs of combinatorial auction cannot be predetermined. In [9], Zhang et al. employed the inner product of matrix and cancellation with invertible matrix to achieve asymmetric scalar product preserving encryption. Instead of homomorphic encryption, Li et al. [27] used random noise to mask the bid values. By using a masking approach, the server only knows the noise, and the auctioneer only knows the auction results, which will decrease computational complexity in the combinatorial auction.

As an emerging decentralized security data management system, blockchain has gained much popularity recently and has been applied in electronic auction. As the participants in the conventional auction-based trading may collude or take selfish actions, [28] employed the Ethereum framework for trustless, secure, and distributed auctioning. [29] proposed a decentralized electricity transaction mode for microgrids based on blockchain and continuous double auction (CDA) mechanism, which could solve problems in traditional management, such as high operation cost and low transparency.

3. Model of Privacy-Preserving Combinatorial Auction

3.1. System Model. We first present our system model for privacy-preserving combinatorial auction, in which there are three kinds of participants, i.e., an auctioneer who wants to sell several products \( G = \{g_1, \ldots, g_m\} \) simultaneously, \( N \) bidders \( B = \{B_1, \ldots, B_n\} \) who want to succeed in the auction, and a crypto service provider who is responsible for key distribution and collaborative computation. In the privacy-preserving combinatorial auction model, we suppose that there is a classical channel between any two participants.

As shown in Figure 2, during the auction, every bidder \( B_i \) has his own bundle \( S_i \in G \) that he expects to obtain and his bid \( b_i \), i.e., the price \( B_i \) he is willing to pay on his bundle \( S_i \). During the auction, one product can only be auctioned to one bidder, and the auctioneer’s goal is to maximize social welfare. So, the winners are chosen by the auctioneer as follows:

\[
W = \left\{ B_i \in B \mid \arg\max_{B_i} \sum_{S_i} b_i(S_i) \text{ s.t. } \bigcap_{B_i} S_i = \emptyset \right\},
\]  

Figure 1: Traditional combinatorial auction.

Table 2: Main notations.

| Notations | Terms |
|-----------|-------|
| \( i \)th bidder | \( B_i \) |
| \( m \)th goods | \( g_m \) |
| \( B_i \)'s bundle | \( S_i \) |
| \( B_i \)'s bid | \( b_i \) |
| \( k \) | System security parameter |
| \((G, p, g)\) | Finite group \( G \) of order \( p \), generator \( g \) |
| \( \mathbb{Z}_p, \mathbb{Z}_p^* \) | \( \mathbb{Z}_p = \{0, 1, \ldots, p - 1\}, \mathbb{Z}_p^* = \mathbb{Z}_p \setminus \{0\} \) |
| \( [m] \) | Ciphertext of message \( m \) |
| \( \mathcal{A} \) | Adversary algorithm |
| \( \text{negl}(k) \) | Negligible function in parameter \( k \) |
| CSP | Crypto service provider |
| AUCT | Auctioneer |
i.e., a set of conflict-free bidders whose total bid is maximized, and \( A = \bigcup_{i=1}^{N} S_i \) is winners’ bundle. After that, the auctioneer will determine the price that the winner should pay according to some mechanism. Besides, CSP will generate a blind signature for bidders’ bid and bundle, which will be used to verify the correctness of the result later.

3.2. Attack Models and Security Requirements. Different from a previous work that assumes CSP is trustworthy, in this paper, we assume that the crypto service provider is semihonest. That is, CSP will follow the protocol steps honestly but tries to learn the bidders’ bundles and bids, i.e., "curious." But CSP cannot collude with the auctioneer, i.e., noncooperative. Because CSP and the auctioneer are usually service providers with industry certification standards, if either party has any collusion or deception, it will greatly damage its reputation and interests.

In the semihonest adversary model, the main idea is to limit the information exposed to the auctioneer and CSP. When the allocation terminates, the auctioneer is supposed to only know the winners, their bundles, and payments. Each bidder only knows whether he is a winner. The bidder will also be informed the price he should pay, if he is the winner. Each bidder does not know anything about others’ bundle or bid. CSP will help auctioneer to decrypt but know nothing about auction results.

Also, the auctioneer is assumed to be curious, malicious, and ignorant, which is interested in bidders’ bundles and bids because this information will enable the auctioneer to have more advantage in future auction of similar goods, i.e., "curious." Besides, bidders’ preferences and competitive relationship will be disclosed according to the bundles and bids. The auctioneer may also try to obtain secret key \( msk \) to decrypt bids or report a fake payment to the winners (i.e., "malicious"), but he is not aware of bidders’ bid for a specific product or preference on these goods (i.e., "ignorant"). The auctioneer may also report a fake payment to the winners, i.e., "malicious," but he is not aware of bidders’ bid for a specific product or preference on these goods, i.e., "ignorant." In our system, bidders are assumed to be noncooperative and curious. They will follow the scheme honestly but want to know others’ bundles and bids to help them make decision, i.e., "curious." However, they will not collude with each other, i.e., "noncooperative."

In our scheme, the following security goals should be achieved:

(i) Privacy preservation: no one can obtain the others’ bundle and bid. Winner determination and payment determination should not arrive at the expense of revealing the losing bids and bundles

(ii) Verifiability and integrity: the winner should be able to verify whether the auctioneer gives a wrong payment to maximize social welfare

Our scheme focuses on the confidentiality of losers’ bundle and bid since winners’ bundles and payments might be learned from the valid output of the auction.

3.3. Design Goal. Our design goal is to develop an efficient, verifiable, and privacy-preserving combinatorial auction scheme. In particular, the following four desirable objectives need to be considered:

(i) Fairness: all bidders should have the same advantage to win the auction

(ii) Security: the proposed scheme should meet the security requirements as above

(iii) Anonymity: the protocol should not reveal any indications about bidder-bid relation. In other words, the auctioneer cannot get bidder’s identity information from bid

(iv) Scalability: when the combinatorial auction parameters, such as the number of bidders and goods, increase rapidly, the protocol is still efficient in terms of both computation and communication cost
4. Preliminaries

We first introduce the primitives and terms that will be used in our scheme.

4.1. ElGamal Cryptosystem. The ElGamal encryption scheme provides a multiplicative homomorphic encryption that comprises the algorithms as key generation, encryption algorithm, and decryption algorithm that are described as follows.

(i) ElGamal.KeyGen: randomly select a large prime number $p$ and at random select a generator $g \in \mathbb{Z}_p^*$. At random, select a number $x \in \mathbb{Z}_p^*$. Calculate $y = g^x \pmod{p}$. The public key is $pk = (y, g, p)$, and the private key is $sk = x$

(ii) ElGamal.Encrypt: to encrypt a message $m \in \mathbb{G}$, at first, select a random number $k$, which is relatively prime with $(p - 1)$, and then calculate $C_1 = g^k \pmod{p}$, $C_2 = m \cdot y^k \pmod{p}$. The ciphertext is set as $ct = (C_1, C_2)$

(iii) ElGamal.Decrypt: on input a ciphertext $ct = (C_1, C_2)$ and a private key $sk = x$, output the plaintext $m$ by computing

$$m = \frac{C_2}{(C_1)^x} = \frac{y^k \cdot m}{g^x} = \frac{y^k \cdot m}{y^k} \pmod{p} \quad (2)$$

Homomorphic multiplication: let $[[m]]$ be the ciphertext of plaintext $m$. We have

$$[[m_1 \cdot m_2]] = [[m_1]] \cdot [[m_2]]. \quad (3)$$

4.2. The Monotonically Increasing and One-Way Function. In this section, we give the notation of monotonically increasing and one-way function [30], which will serve as the building block of combinatorial auction with privacy preservation in our scheme.

Suppose that $\mathcal{D} = \{x_1, x_2, \ldots, x_l\}$, where $x_i \in \mathbb{Z}_q^+$ and $x_i \leq U$ for $i = 1, 2, \ldots, l$, where $\mathcal{D}$ is an $l$-dimensional dataset and $U$ is the upper bound of all data values in $\mathcal{D}$. Meanwhile, we denote a set of Euclidean distance by ED, where

$$ED = \left\{\text{dist}^2(x, y) = \sum_{i=1}^{l} (x_i - y_i)^2 | x, y \in \mathcal{D}\right\}. \quad (4)$$

Then, we construct a function $f$, which maps each element $d^2 \in ED$ to $f(d^2)$. In particular, for each $d^2 \in ED$, $f(d^2) = a_1 (d^2 \pmod{\Delta}) + a_2 (d^2 \pmod{\Delta})^2 + \cdots + a_n (d^2 \pmod{\Delta})^n + e$, where $\Delta = 1 \cdot U^2$, each coefficient $a_i$ is an integer, and $a_i > \Delta$ for $i = 1, 2, \ldots, n$. In addition, $e$ is a noise and randomly chosen from $[\Delta, a_1 + a_2 + \cdots + a_n]$.

Obviously, the function $f$ is a monotonically increasing function, that is,

$$f(d_1^2) > f(d_2^2) \text{ for } \forall d_1^2, d_2^2 \in ED \land d_1^2 > d_2^2. \quad (5)$$

Moreover, the function $f$ is also a one-way function. That is, it is infeasible to recover $d^2$ from $f(d^2)$ for any $d^2 \in ED$.

Both security and computation overhead need to be considered to determine the degree of function $f$. With the increasing of $n$, the computation overhead of function $f$ will be increasing. Thus, an optimal value $n$ should be chosen according to the balance of security and efficiency. In our protocol, we set the degree of $f$ to be $N$, which is equal to the number of bidders.

4.3. Blind Signature. In the PP-VCA scheme, we employ a blind signature to guarantee that a signer can create a signature for bidder’s bid and bundle without knowing the real bid price. Concretely, in the blind signature scheme, the signer can generate the signature of bidding price $m$ without knowing $m$. In our scheme, we utilize a blind signature to ensure the authenticity and reliability of the combinatorial auction and verify whether the payment price is correctly calculated. By analyzing the inherent disadvantages of the blinded Nyberg-Rueppel scheme, Qi et al. [31] gave an improved scheme by adding hash function in the signature, which enables the signature scheme to be against changing agreed information attack. We give the concrete blinded Nyberg-Rueppel scheme in Scheme 1.

**Scheme 1.** Blinded Nyberg-Rueppel scheme (BNR).BNR.SyPara: at random, select a multiplicative group $\mathbb{G} \in \mathbb{Z}_p^*$ of prime order $q$ and its generator $g$, where $q$ is a prime factor of prime number $p$. Select a hashing function $h : \{0, 1\}^* \rightarrow \mathbb{Z}_q^*$.

BNR.KeyGen: let $c$ be information agreed by the signer and the signee in advance. Compute $T(c) = 2h^{-1} + 2h_1(c) + 1$, where $h_1(\cdot)$ is a one-way function. The signer picks a random number $x \in \mathbb{Z}_q^*$ and keeps $x \cdot T(c)$ secret and publishes the public parameters as $g$ and $g^{2 \cdot T(c)} \pmod{p}$.

BNR.Signing: signer blindly signs signee’s message $m$.

1. The signer randomly selects $\tilde{k} \in \mathbb{Z}_q^*$ and sends $\tilde{r} = g^{k \tilde{h}} \pmod{p}$ to the signee.

2. The signee randomly selects $\alpha \in \mathbb{Z}_q^*$, $\beta \in \mathbb{Z}_q^*$, computes $r = h(m, c)g^{\alpha \cdot \tilde{r} \cdot \beta} \pmod{p}$ and $\tilde{m} = r \beta^{-1} \pmod{p}$ until $\tilde{m} \in \mathbb{Z}_p^*$. Then, he sends $\tilde{m}$ to the signer.

3. The signer computes $\tilde{s} = \tilde{m} \cdot T(c) + \tilde{k} \pmod{p}$ and sends $\tilde{s}$ to the signee.

4. The signee computes $s = \tilde{s} \beta + \alpha \pmod{q}$

5. Set the signature as $\sigma = (r, s, c)$

BNR.Verify: the verifier checks whether $h(m, c) = g^s \cdot \tilde{r} \pmod{p}$ and accepts the signature if the equation holds.
5. Our Proposed Scheme

Before submitting the combinatorial auction, all bidders blind sign their bundle $S_i$ and average value $\varphi_i$ through the crypto service provider. As we deploy the blind signature scheme, CSP will not attain any relevant information about the real message $S_i$ and $\varphi_i$. Also, we can combine the auction scheme with anonymization techniques to protect bidders’ identity information [32]. In our protocol, bidders’ personally identifiable information will be protected by anonymous techniques, which keeps the bidder-bid relation private. Our framework of proposed PP-VCA is described in Figure 3, in which we employ three subprotocols, namely, privacy-preserving winner determination protocol (PPWD), privacy-preserving scalar protocol (PPSP), and privacy-preserving verifiable payment determination protocol (PPVPD), to implement the combinatorial auction with bidder privacy and payment verifiability.

5.1. Privacy-Preserving Winner Determination. At first, we give a greedy winner determination protocol in Algorithm 1. Note that in order to protect the privacy information $S_i$ and $b_i$ of the bidder, AUCT cannot directly sort $B_i$ on the plaintext and select the winner (see Step 2), because the comparison and sorting will reveal the private information $S_i$ and $b_i$ of the bidders. So, we use a monotonically increasing and one-way function to protect the bidder’s $b_i$, which enables the auctioneer to pick out the largest one without knowing any information about $b_i$.

The above GWD algorithm needs to check whether $B_i$’s bundle $S_i$ does not contain the goods that has already been auctioned, which can be solved by privacy-preserving scalar product. We utilize $m$-dimensional binary vector $\vec{A}$ to represent the auction status of $m$ goods, where the $k$th bit $a_k = 1$ if the $k$th goods $g_k$ have already been auctioned and $a_k = 0$ otherwise. Similarly, we utilize another $m$-dimensional binary vector $\vec{S}_i$ to represent $B_i$’s bundle $S_i$, where $k$th bit $s_{i,k} = 1$ if the $k$th goods $g_k \in S_i$ and $s_{i,k} = 0$ if the $k$th goods $g_k \notin S_i$.

If $B_i$’s bundle $S_i$ does not contain the goods that has already been auctioned, then

$$\vec{A} \cdot \vec{S}_i = 0 \Longleftrightarrow \sum_{k=1}^{m} a_k \cdot s_{i,k} = 0.$$  \hspace{1cm} (6)

If the scalar product is $\theta$, that means $B_i$’s bundle $S_i$ includes $\theta$ already-auctioned goods. During this process, both $\vec{S}_i$ and $\vec{A}$ are private information and have to be protected. Besides, the auctioneer $E$ will obtain side information about $\vec{S}_i$, because $E$ can guess which $\theta$ goods $B_i$ wants to get according to $\vec{A}$. Similarly, $B_i$ is able to gain some side information about $A$ from $\theta$. During this process, it is easy to see that $\theta$ and $\vec{S}_i$ are $B_i$’s privacy information, which should be kept from the auctioneer AUCT. Besides, $\theta$ and $A$ are AUCT’s privacy information, which should be kept private from $B_i$. We design Algorithm 2 to solve the product calculation of two vectors while protecting the privacy and check whether the result is equal to 0.

If $g^{\vec{S}_i} = 1$, AUCT will explicitly know that $B_i$’s bundle $S_i$ does not contain the goods that have already been auctioned, and otherwise, the final output is indistinguishable from a random number in $\mathbb{Z}_n$ from the auctioneer’s perspective. Combining Algorithms 1 and 2, we give a privacy-preserving winner determination model (Algorithm 3), which can be regarded as a black-box algorithm and only outputs the winner and the corresponding bundle.

In Algorithm 3, $B_i (i = 1, \cdots, n)$ computes the average value $\varphi_i = b_i / |S_i|$ and calculates $f(\varphi_i)$ using the parameters provided by CSP. Because $f(\varphi_i)$ is a one-way increasing function, the auctioneer AUCT is able to pick out the largest $f(\varphi_i)$ by comparing the value of $f(\varphi_i)$, which is equivalent to picking the largest $\varphi_i$. Furthermore, AUCT asks the corresponding $B_i$ to execute Algorithm 2 together, in which $B_i$ will not reveal any information about $S_i$. AUCT
Input: each \( B_i (i = 1, \ldots, n) \) has bundle \( S_i \) and bid \( b_i \).

Output: AUCT obtains the winner set \( W \) and bundle set \( A \).

1: \( B_i \)
   (a) Compute average value \( \varphi_i = b_i / |S_i| \)
   (b) Send \( \varphi_i \) to AUCT
2: AUCT:
   (a) Initialize \( A = \emptyset \), \( W = \emptyset \)
   (b) Sort \( B_i \) in a nonincreasing order according to the value of \( \varphi_i \). That is, the bigger the \( \varphi_i \), the former the \( B_i \). The sorted sequence is called \( L \).
   (c) Check \( B_i \) in \( L \) and test whether \( A \cap S_i = \emptyset \). If true, update \( A \) with \( A \cup S_i \), \( W \) with \( W \cup B_i \).

---

**Algorithm 1:** Greedy winner determination (GWD).

Input: CSP has a pair of Elgamal key: \( \text{mpk} = (h_1 = g^{s_1}, h_2 = g^{s_2}, \ldots, h_m = g^{s_m}) \), \( msk = S = (s_1, s_2, \ldots, s_m) \).

The auctioneer AUCT has \( A = (a_1, a_2, \ldots, a_m) \); \( B_i (i = 1, \ldots, n) \) has \( S_i = (s_{i1}, s_{i2}, \ldots, s_{im}) \).

Output: AUCT obtains \( g^{\overline{s}_{iA}} \).

1: AUCT: send \( \overline{A} = (a_1, a_2, \ldots, a_m) \) to CSP
2: CSP:
   (a) Compute \( sk_j = S \cdot \overline{A} = (s_1 a_1 + s_2 a_2 + \cdots + s_m a_m) \)
   (b) Send \( sk_j \) to AUCT
3: For each \( B_j \):
   (a) Pick a random number \( r_j \) and encrypt \( \overline{S_j} = (s_{j1}, s_{j2}, \ldots, s_{jm}) \) to compute \( c_{j1} = h_1^{r_j} g^{s_1}, c_{j2} = h_2^{r_j} g^{s_2}, \ldots, c_{jm} = h_m^{r_j} g^{s_m}, c_{j,m+1} = g^{r_j} \)
   (b) Send \((c_{j1}, c_{j2}, \ldots, c_{jm}, c_{j.m+1})\) to the auctioneer AUCT
4: AUCT: compute \( \prod_{j=1}^{m+1} (c_j)^{s_i} / (c_{i,m+1})^{sk_j} = g^{\overline{s}_{iA}} \)

---

**Algorithm 2:** Privacy-preserving scalar product (PPSP).

Input: CSP has a pair of ElGamal key: \( \text{mpk} = (h_1 = g^{s_1}, h_2 = g^{s_2}, \ldots, h_m = g^{s_m}) \), \( msk = \overline{S} = (s_1, s_2, \ldots, s_m) \); the auctioneer AUCT has \( \overline{A} = (a_1, a_2, \ldots, a_m) \); \( B_i (i = 1, \ldots, n) \) has \( \overline{S_i} = (s_{i1}, s_{i2}, \ldots, s_{im}) \) and \( b_i \).

Output: AUCT obtains the winner set \( W \) and corresponding bundle set \( A \).

1: CSP:
   (a) Select a large number \( U \), calculate \( \Delta = l \cdot U^2 \), and select \( a_1, a_2, \ldots, a_n \) s.t. \( a_i > \Delta \) \((i = 1, 2, \ldots, n)\)
   (b) Randomly choose noise \( e \) from \((\Delta, a_1 + a_2 + \cdots + a_n)\)
   (c) Send \( a_1, a_2, \ldots, a_n, e \) and \( \Delta \) to \( B_i \)
2: \( B_i \):
   (a) Compute the average value \( \varphi_i = b_i / |S_i| \)
   (b) Compute \( f(\varphi_i) = a_1 (\varphi_i \mod \Delta) + a_2 (\varphi_i \mod \Delta)^2 + \cdots + a_n (\varphi_i \mod \Delta)^n + e \)
   (c) Send \( f(\varphi_i) \) to AUCT
3: AUCT and \( B_i \) jointly perform:
   (a) AUCT picks the largest \( f(\varphi_i) \) and records the corresponding bidder as \( B_i \). \( \overline{S}_i \) is the bundle of \( B_i \)
   (b) On input \( \text{mpk}, msk, \overline{A}, \overline{S_i} \), perform privacy-preserving scalar product protocol (PPSP) to obtain \( g^{\overline{s}_{iA}} \)
   (c) AUCT receives \( g^{\overline{s}_{iA}} \)
4: AUCT:
   (a) If \( g^{\overline{s}_{iA}} = 1 \), \( B_i \) is the winner. Inform \( B_i \) to send \( f(\varphi_i) \), bundle \( S_i \) and \( \text{Sign}(S_i) \) and put \( B_i \) into the winner set \( W \) and mark \( B_i \)’s bundle as auctioned in \( A \)
   (b) Otherwise, remove \( B_i \) from bidders

---

**Algorithm 3:** Privacy-preserving winner determination (PPWD).
Input: the auctioneer AUCT has $A$ and the winner’s $S_j$ and Sign($S_j$).
Output: $B_i$ obtains the payment $p_i$.
1: AUCT removes the winner $B_i$ from bidders and modifies $A$ to $(A - S_j)$, where $A$ is the set of auctioned goods and $S_j$ is the bundle of $B_i$. Then, through Algorithm 3, AUCT chooses a freshful winner $B_j$, who is the candidate of $B_i$. AUCT notifies $B_j$ to send average value $q_j = b_j/|S_j|$ and Sign($q_j$) to AUCT.
2: If the candidate of $B_i$ can be successfully found, AUCT computes $p_i = (b_i/|S_i|)|S_i|$ and sends $p_i$ and Sign($q_j$) to $B_j$. If no candidate is found, AUCT sets $p_i$ as the agreed default value and notifies $B_i$ that $p_i$ is the default value.
3: If $p_i$ is not the default value, $B_i$ can recover $q_j$ from $p_i/|S_i|$ and verify whether $q_j$ is correct through Sign($q_j$). If they are not equal to each other, $B_i$ knows that the payment is not correct.

Algorithm 4: Privacy-preserving and verifiable payment determination (PPVPD).

Verifies whether $B_i$’s bundle contains the goods that have already been auctioned through judging whether $g_j = 1$, that means compared with other bidders, the average value of $B_i$ is the largest one, and the corresponding bundle is also available, which means $B_i$ is the winner of this round. The auctioneer will inform $B_i$ to submit $f(q_j)$, bundle $S_j$ and Sign$(S_j)$, and then update $A$ and $W$ to continue the search for the next winner. $f(q_j)$ can prove the identity of $B_j$, and the signature Sign$(S_j)$ can guarantee the integrity of $S_j$. If $g_j = 1$, that means the bundle of $B_i$ contains at least one good that has been auctioned, so AUCT will remove $B_i$ from bidders and enter the next round of selection.

5.2. Privacy-Preserving Verifiable Payment Determination. We propose a privacy-preserving verifiable payment determination protocol that is shown in Algorithm 4. AUCT determines the payment that the winner $B_i$ should pay by the following algorithm: Among the bidders whose bundle would have been allocated if $B_i$ were not the winner, AUCT finds out $B_j$ whose average value is maximum, i.e., the candidate of $B_i$. Then, $B_i$’s payment is $p_i = (b_j/|S_j|)|S_j|$, where $b_j/|S_j|$ is the average value of $B_j$.

In our scheme, the winner $B_i$’s payment is determined by his candidate $B_j$’s average value $b_j/|S_j|$. In Algorithm 2, AUCT cannot know any information about the bundle of $B_j$. As a result, AUCT also cannot know any information about $b_j$ from $b_j/|S_j|$. Similarly, the winner $B_i$ cannot obtain any information about $B_j$’s bundle $S_j$ and $b_j$, and $B_i$ even does not know who is $B_j$.

6. Security Analysis
6.1. Bidder’s Privacy Preservation. In the PP-VCA protocol, neither the crypto service provider CSP nor the auctioneer AUCT can learn the full information of bidders. CSP is only responsible for key distribution and blind signature, so it cannot obtain any information about bidders’ private data. The auctioneer only knows the winners and their bundles and payments. As to auction losers, we give Theorems 1-4 to prove that the auctioneer and other bidders cannot obtain any information about losers’ bundles and bids, even their real identity.

$$adv_{msk} = Pr [msk | S', Output \leftarrow A_{out} (1^k)] - Pr [msk | Output \leftarrow A_{black}] \negl(k).$$

(7)

$$adv_B = Pr [b_j | S', Output \leftarrow A_{out} (1^k)] - Pr [b_j | Output \leftarrow A_{black}] \negl(k).$$

(8)

Theorem 1. An adversarial auctioneer $E$’s advantage $adv_{msk}$ is negligible.

Proof. If the auctioneer $E$ wants to construct $A$ skillfully to obtain $msk$, for example, let $A = (1, 0, \cdots, 0)$, and $E$ will obtain $sk_j = s_1 + s_{m+1}$. Due to the discrete logarithms, an adversarial $E$ cannot obtain $s_j$ or $s_{m+1}$. Similarly, an adversarial $E$ cannot obtain any information about $s_j(i = 1, 2, \cdots, m + 1)$. Therefore, we have

Theorem 2. An adversarial auctioneer $AUCT$’s advantage $adv_{S_j}$ is negligible for all losers.

Proof. Every winner’s bundle $S_j$ is given to AUCT; therefore, we have $adv_{S_j} = Pr [S_j | S', Output \leftarrow A_{out} (1^k)] - Pr [S_j | Output \leftarrow A_{black}] = 0$ if $B_j$ is a winner of the auction. Further, we assume that the ElGamal encryption algorithm is semantically secure, during the privacy-preserving scalar product PPSP protocol (see Algorithm 2), an adversarial AUCT learns whether there exists a feasible bundle which is negligible, and this reveals nothing about losers’ $S_j$; therefore the adversary’s view on losers’ bundle in our PP-VCA is the same as the one in an ideal black-box algorithm. Therefore, $adv_{S_j} = Pr [S_j | S', Output \leftarrow A_{out} (1^k)] - Pr [S_j | Output \leftarrow A_{black}] \negl(k)$ is negligible in security parameter $k$, where $B_j$ is a loser and $\negl(\cdot)$ is a negligible function.

Theorem 3. An adversarial auctioneer $AUCT$’s advantage $adv_{B_j}$ is negligible for all losers.
Proof. In the payment determination model of the winner \( B_i \), the candidate \( B_i \)'s average value \( \phi_j \) is disclosed to AUCT. Because of the privacy-preserving scalar product PPSP, AUCT knows nothing about \( S_j \), so he does not learn \( b_j \) from \( \phi_j = b_j / S_j \). We have

**Theorem 4.** An adversarial bidder \( B_k \)'s advantage \( \text{adv}_{S_k} \) and \( \text{adv}_{b_k} \) are negligible for all \( i \neq k \).

Proof. For all adversarial bidders, no matter he is a winner or not, all he learns from the PP-VCA protocol is a valid auction output Output. We have demonstrated in Section 5.2, and then, the winner cannot obtain any information about the candidate’s \( S_j \) and \( b_j \).

As a result, in a collusion-free case, our proposed combinatorial auction scheme can protect the information of bidders.

### 6.2. Payment Verification

In Algorithm 4, the winner’s payment is determined by his candidate’s average value. Since AUCT and \( B_i \) use a blind signature \( \text{Sign}(\phi_j) \) generated by CSP, AUCT to convince that \( B_i \) provides the correct \( \phi_j \), and \( B_i \) can easily verify whether AUCT the data are modified \( p_i \) to maximize social welfare, while protecting the plaintext itself in the signature.

### 7. Performance and Evaluation

We give the performance analysis and evaluation of our combinatorial auction scheme PP-VCA in terms of communication overhead and computation overhead.

#### 7.1. Communication Cost

In our PP-VCA combinatorial auction scheme, each bidder needs to transfer \( (m + 1) \) ciphertext, so \( N \) bidders need to transfer a total of \( N \cdot (m + 1) \) ciphertext, and the auctioneer needs to return the result. The security parameter used in our scheme is \( k \), and the length of the ciphertext of Elgamal is \( 2k \). Because the length of the result is relatively small compared to \( k \), so it can be ignored. Therefore, in our combinatorial auction scheme, the communication overhead is \( N \cdot (m + 1) \cdot 2k = 2kN(m + 1) \).
7.2. Benchmark and Computational Overhead. To evaluate the computation overhead, we conducted an experiment, which was in Windows 8 with a 64-bit operating system, RAM 4 G, Intel® Core™ i5-4210U CPU @ 1.70 GHz. In order to exclude the communication I/O during the simulation, we generated all strings in the communication and conducted the computation in the local instance. Security parameter $k$ is 128-bit, and every operation is run 1000 times to evaluate the average running time.

In the winner determination protocol, $T_{\text{enc}}$ is the time which the bidder spends on encrypt $S_i$ using CSP’s $pk$, and $T_{\phi}$ is the time which the bidders take to calculate $f(\phi_i)$ using the parameters provided by CSP. $T_{\text{auc}}$ is the total time that the auctioneer spends on the decryption of the ciphertext, selection of the winner, and update of $A$ and $W$. In terms of different goods and bidders, we give the performance and analysis of computational cost in the winner determination protocol that is shown in Figures 4–6, respectively.

By Figures 4 and 5, it is easy to see that the auctioneer’s computation overhead will increase logarithmically with the increasing of the value of max bid and will increase linearly with the increasing of the amount of total bidders and total goods. Firstly, the larger the value of max bid, the larger the average value $\phi_i$. So, $f(\phi_i)$ obtained by one-way and monotonically increasing function is larger, which will increase the auctioneer’s computation overhead to select the largest $f(\phi_i)$. Secondly, the increase of the amount of total bidders and total goods will inevitably increase the auctioneer’s computation overhead. Figures 5 and 6 demonstrate that, in our protocol, the auctioneer’s computation overhead grows with small constant factors linearly.

Meanwhile, Figures 4 and 5 indicate that the value of max bid and the amount of total bidders do not have a big impact on bidder’s computation overhead, since each bidder calculates the average values $\phi_i$, $f(\phi_i)$ and encrypts the bundle locally. The increase of the number of total goods will increase the bidder’s encryption time $T_{\text{enc}}$, but Figure 6 illustrates that the bidder’s computation overhead grows with small constant factors linearly as well.

7.3. Comparison with Peer Works. We compare scalability of our PP-VCA protocol with peer works in Table 3. Considering the actual running time of our protocol with peer works, we notice that our protocol’s run time increases logarithmically with the increasing of the max bid and increases linearly with the increasing of total bidders and total goods. We improve the performance to a linear growth and logarithmic growth, which illustrates that our PP-VCA protocol provides a better scalability in the practice.

| Variable      | Our PP-VCA | [26]  | [10]  | [11]  | [16]  |
|---------------|------------|-------|-------|-------|-------|
| Max bid       | Logarithmic| Logarithmic | Linear | Linear | Linear |
| # of bidders  | Linear     | Linear | Linear | Linear | Linear |
| # of goods    | Linear     | Exponential | Logarithmic | Logarithmic | Logarithmic |

8. Conclusion

In this work, we proposed an effective, scalable, and flexible privacy-preserving combinatorial auction scheme to protect bidder’s privacy and ensure the correctness and verifiability of the bidding price. We employed a monotonically increasing one-way function to ensure the auctioneer to pick out the largest bid without disclosing the bidding price. In addition, we put forward a privacy-preserving verifiable payment determination protocol to confirm the payment that the winner should pay. Furthermore, we used a blind signature scheme to succeed in allowing all bidders to verify the payment without knowing the real sensitive bidding price. Performance analysis and experimental results indicate that our scheme provides a better performance and scalability in combinatorial auction systems.

Data Availability

Data is available on request.

Additional Points

This is the extended and full version of [5].

Conflicts of Interest

The authors declare no conflict of interest regarding this publication.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under grant 62072134 and 61672010, the Open Research Project of State Key Laboratory of Cryptology of China, the Key projects of Guangxi Natural Science Foundation under grant 2019JJJD170020, and the Open Fund Program for State Key Laboratory of Information Security of China under Grant 2020-MS-05.

References

[1] M. Zhang, Y. Yao, B. Li, and C. Tang, “Accountable mobile e-commerce scheme in intelligent cloud system transactions,” Journal of Ambient Intelligence and Humanized Computing, vol. 9, no. 6, pp. 1889–1899, 2018.
[2] Y. Chen, Z. Ma, Q. Wang, J. Huang, X. Tian, and Q. Zhang, “Privacy-preserving spectrum auction design: challenges, solutions, and research directions,” IEEE Wireless Communications, vol. 5, pp. 142–150, 2019.
[3] Q. Wang, J. Huang, Y. Chen, X. Tian, and Q. Zhang, “Privacy-preserving and truthful double auction for heterogeneous...
spectrum,” *IEEE/ACM Transactions on Networking*, vol. 27, no. 2, pp. 848–861, 2019.

[4] J. Lin, M. Pipattanasomporn, and S. Rahman, “Comparative analysis of auction mechanisms and bidding strategies for P2P solar transactive energy markets,” *Applied energy*, vol. 255, p. 113687, 2019.

[5] M. Zhang and B. Zhou, *A verifiable combinatorial auction scheme with bidder’s privacy protection*, 2020.

[6] R. Alvare and M. Nojoumian, “Comprehensive survey on privacy-preserving protocols for sealed-bid auctions,” *Computers & Security*, vol. 88, 2020.

[7] T. Jung and X. Li, “Enabling privacy-preserving auctions in big data,” *Journal of Parallel and Distributed Computing*, vol. 73, no. 4, pp. 495–508, 2013.

[8] M. Zhou, C. Niu, Z. Zheng, F. Wu, and G. Chen, “An efficient, privacy-preserving, and verifiable online auction mechanism for Ad exchanges,” in *Globecom, IEEE Global Communications Conference*, San Diego, CA, USA, 2015.

[9] M. Zhang, Y. Chen, Z. Xia, J. Du, and W. Susilo, “PPO-DFK: a privacy-preserving optimization of distributed fractional knapsack with application in secure footballer configurations,” *IEEE Systems Journal*, vol. 2020, pp. 1–12, 2020.

[10] K. Suzuki and M. Yokoo, “Secure combinatorial auctions by dynamic programming with polynomial secret sharing,” in *Lecture Notes in Computer Science*, vol 2357, Springer, Berlin, Heidelberg, 2002.

[11] M. Yokoo and K. Suzuki, “Secure multi-agent dynamic programming based on homomorphic encryption and its application to combinatorial auctions,” *Proceedings of the first international joint conference on Autonomous agents and multiagent systems*, pp. 112–119, 2002.

[12] D. C. Parkes, M. O. Rabin, and C. Thorpe, “Cryptographic combinatorial clock-proxy auctions,” in *Lecture Notes in Computer Science*, vol 5628, Springer, Berlin, Heidelberg.

[13] M. Zhang, S. Zhang, and L. Harn, “An efficient and adaptive data-hiding scheme based on secure random matrix,” *PLOS One*, vol. 14, no. 10, article e0222892, 2019.

[14] H. Kikuchi and C. Thorpe, 

(Proceedings of the first international joint conference on Autonomous agents and multiagent systems), pp. 112–119, 2002.

[15] C. Hu, R. Li, B. Mei, W. Li, A. Alrawais, and R. F. Bie, “Privacy-preserving combinatorial auction without an auctioneer,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, no. 1, 38 pages, 2018.

[16] M. Pan, X. Zhu, and Y. Fang, “Using homomorphic encryption to secure the combinatorial spectrum auction without the trustworthy auctioneer,” *Wireless Networks*, vol. 18, no. 2, pp. 113–128, 2012.

[17] M. Pan, J. Sun, and Y. Fang, “Purging the back-room dealing: secure spectrum auction leveraging Paillier cryptosystem,” *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 866–876, 2011.

[18] M. Larson, R. Li, C. Hu, W. Li, X. Cheng, and R. Bie, “A bidder-oriented privacy-preserving VCG auction scheme,” in *International Conference on Wireless Algorithms, Systems, and Applications*, pp. 284–294, Springer, Cham.

[19] W. Gao, W. Yu, F. Liang, W. G. Hatcher, and C. Lu, “Privacy-preserving auction for big data trading using homomorphic encryption,” *IEEE Transactions on Network Science Engineering*, vol. 7, 2018.

[20] K. Xing, C. Hu, J. Yu, X. Cheng, and F. Zhang, “Mutual privacy preserving $k$-means clustering in social participatory sens-