EasyChair Preprint

№ 9387

Privacy-Preserving and Truthful Auction for Task Assignment in Outsourced Cloud Environments

Xufeng Jiang and Lu Li

November 29, 2022

Privacy-Preserving and Truthful Auction for Task Assignment in Outsourced Cloud Environments

Xufeng Jiang[1,2] and Lu Li[2,3*]

[1] Nanjing Tech University, Nanjing, China
[2] Yancheng Teachers University, Yancheng, China
[3] University of Science and Technology of China, Suzhou, China

Abstract. Due to high fairness and allocation efficiency, the task assignment problem of mobile applications via auctions has become a promising approach to motivate bidders to provide their mobile device resources effectively. However, most of existing works focus on the auction mechanism under the plaintexts, and ignore the problems caused by information leakage. In this paper, we study the problem of the privacy-preserving auction for task assignment in outsourced cloud environments without leaking any private information to anyone. Specifically, we use Yao's garbled circuits and homomorphic encryption system as underlying tools. Along with several elaborately designed secure arithmetic subroutines, we propose a privacy-preserving and truthful auction framework for task assignment in outsourced cloud environments. Theoretically, we analyze the complexity of our scheme in detail and prove the security in the presence of semi-honest adversaries. Finally, we evaluate the performance and feasibility of our scheme through a large number of simulation experiments.

Keywords: Privacy-preserving · Auction · Task assignment · Yao's garbled circuits.

1 Introduction

With the development of mobile applications, a single mobile device can no longer meet the resource requirements of mobile application tasks. On the one hand, due to expensive mobile devices, enterprises that need devices are reluctant to buy large amounts of devices to support the development or operation of mobile applications. On the other hand, it is unrealistic for manufacturers with idle mobile devices to share their device resources or perform tasks for others. In recent years, due to high fairness and allocation efficiency, cloud auctions for task assignment have become a promising approach to motivate bidders to provide their mobile device resources effectively. However, most of existing works [1–4] focus on the truthfulness, personal rationality and computational efficiency of auctions, but ignore the security problems caused by information leakage in outsourced cloud environments. For example, bidders may eavesdrop on other bidders' bid information to modify their actual bids, which will win the

auction with a higher probability; the cloud auctioneer may know the bidder's identity and bid information, which will tamper with the pricing strategy to obtain additional profits. The above problems will break the truthfulness and fairness of the auction. Therefore, sensitive data should be encrypted before uploading to the cloud auctioneer, who requires to perform the auction process on the encrypted data and output the same auction results as the original auction mechanism without leaking any intermediate results to anyone. In addition, it is also necessary to ensure the high efficiency of the system when dealing with large amounts of users in real-life applications. The above requirements make the privacy-preserving auction for task assignment a challenging task.

We focus on the problem of a privacy-preserving auction framework for task assignment in outsourced cloud environments. In this paper, we select the recent work [1] as the underlying auction mechanism. There are two reasons for our choice. First, this work designs an optimal winning bids determination algorithm and employs a one-to-many matching manner. However, the other works [2, 3] limit the auction mechanism in a one-to-one matching manner, which omits the fact that the resource-rich devices can support the resource requirements of multiple buyers in a practical system. Second, this scheme has proven the properties of truthfulness, individual rationality, and system efficiency. Some works like [4] do not provide complete proof for these properties and have high system overhead. Recently, there exist lots of privacy-preserving solutions to tackle various cloud auction problems [5–13], which are introduced in Related Work. Nevertheless, none of the above works can directly deal with our problem. Specifically, since our auction process involves a lot of nonlinear arithmetic operations, which is hard to guarantee security throughout the whole auction process. For example, Jiang *et al.* [7] propose a secure auction scheme for task assignment, but this scheme do not consider the privacy of the number of resources required, which is critical data during the auction process. Wang *et al.* [11] propose a secure and truthful double auction scheme for heterogeneous spectrum allocation, but this scheme discloses the number of candidates, which leaks the privacy of data access patterns. In addition to security, ensuring the system efficiency of our privacy-preserving auction is still a challenging task. These recent works [5, 8, 9, 12] design a serious of secure auction schemes to provide a strong security guarantee for bidders. However, these schemes involve large amounts of public-key encryption operations, which leads to huge computation and communication costs.

In this paper, we propose a Privacy-preserving and Truthful Auction scheme for Task Assignment (PTATA) based on a novel composite method of combining Paillier homomorphic cryptosystem [16] with Yao's garbled circuits [17], which fully protects the privacy information for each participant in the presence of semi-honest adversaries. Our contributions are as follows:

1. Based on Paillier homomorphic cryptosystem and Yao's garbled circuits, we propose a privacy-preserving and truthful auction scheme for task assignment in outsourced cloud environments without leaking any actual intermediate results to anyone.

2. We design two secure arithmetic subroutines over the encrypted data, which can be the critical building blocks in other applications.
3. We prove that our scheme can guarantee a strong security under the semi-honest model and analyze the system complexity. Based on extensive experiments, we evaluate the performance and feasibility of our scheme.

The rest of our paper is organized as follows. In Section 2, we present problem formulation and primitives. Our scheme PTATA is presented in Section 3. In Section 4, we present our simulation experiments. Related works are discussed in Section 5. Finally, the conclusion is made in Section 6.

2 Problem Formulation and Primitives

2.1 System Framework

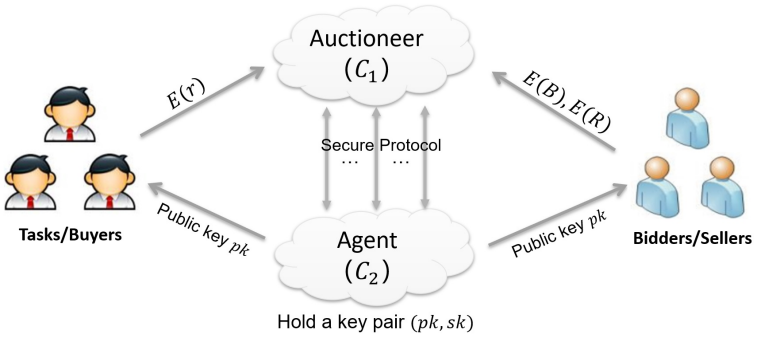


Fig. 1: System framework

As shown in Fig. 1, we construct the system framework of our problem under the semi-honest model [14]. Specifically, *Buyers* (or task demanders) submit the encrypted resource requirements and identity information to the *Cloud Auctioneer* (C_1), who performs the auction process over the encrypted data. *Sellers* (or bidders) have idle mobile devices and bid for each task. They encrypt these bids and the number of resources provided before uploading to the cloud auctioneer. The *Cloud Agent* (C_2) provides cryptographic services and helps the cloud auctioneer to execute the secure auction protocol. Note that the cloud auctioneer and cloud agent are competitive companies, that are highly improbable to conspire with each other, such as *Amazon* and *Google* [15]. Such a system framework is widely used in various related domains [5–9].

The main goals of our privacy-preserving auction scheme are as follows:

- **Correctness:** The results of the privacy-preserving auction scheme are consistent with the original auction mechanism.
- **Security:** Except for the auction results, all participants will not learn anything about the actual information during the auction process.
- **Efficiency:** In practical applications, it is important to ensure system efficiency when dealing with large users.

2.2 Auction Mechanism

In this paper, we consider a truthful auction for task assignment, where n sellers want to compete for homogeneous tasks of m buyers. Let $t_j(1 \leq j \leq m)$ denote the set of buyers, each of whom has one task that requires the same amount of resources required r. Let $d_i(1 \leq i \leq n)$ denote the set of sellers, each of whom provides a certain number of resources R_i and bids $b_{i,j}$ for each task. Then, each buyer t_j submits r to the auctioneer while each seller d_i submits (B_i, R_i), where $B_i \in B$ is the set of bids by d_i. Note that each seller can meet the resource requirements of multiple buyers.

We review a truthful auction mechanism for task assignment [1]. The following is a brief description of this scheme.

Step1: Winning Bids Determination For each seller d_i, the auctioneer first calculates the number of tasks that d_i can accept, which is constrained as follows:

$$K_i = min\{\lfloor \frac{R_i}{r} \rfloor, m\}, (1 \leq i \leq n). \tag{1}$$

After obtaining $K = \{K_i\}_{i=1}^{n}$, the auctioneer selects the least cost in B, i.e., $b_{i_1,j_1} = min\{b_{i,j} | b_{i,j} \in B\}$, which is a winning bid. We set $l_{i_1,j_1} = 1$ and $W = W \cup \{d_{i_1}, t_{j_1}, b_{i_1,j_1}\}$. Then, the auctioneer removes this winning task t_{j_1} and all the bids in B for t_{j_1}, and updates $K_{i_1} = K_{i_1} - 1$. When K_{i_1} is 0, the seller d_{i_1} and its bids should be removed. After that, the auctioneer continues the above process until all the tasks are allocated. Finally, based on all the winning bids, the auctioneer calculates the minimum overall cost, as follows:

$$\mathbb{C}_B = \sum_{k=1}^{m} b_{i_k,j_k}, (1 \leq k \leq m). \tag{2}$$

Step2: The Payments of Winning Bids The auctioneer initially restores all data to original values. For each winning bid $b_{i_k,j_k} \in W(1 \leq k \leq m)$, the auctioneer first removes this winning bid b_{i_k,j_k} from B, and re-executes Step1 to output the minimum overall cost $\mathbb{C}_{B \setminus \{b_{i_k,j_k}\}}$ without the presence of b_{i_k,j_k}. The payment of b_{i_k,j_k} is denoted by p_{i_k,j_k}, calculated as follows:

$$p_{i_k,j_k} = \mathbb{C}_{B \setminus \{b_{i_k,j_k}\}} - (\mathbb{C}_B - b_{i_k,j_k}). \tag{3}$$

2.3 Cryptographic Tools

Paillier Cryptosystem. To protect the sensitive information of buyers and sellers, we adopt Paillier homomorphic encryption scheme [16] to encrypt the sensitive data before uploading to the cloud auctioneer. A pair of key (public key pk and privacy key sk) of this system is generated by the cloud agent. Buyers and sellers encrypt data by $E_{pk}(\cdot)$, and the agent uses $D_{sk}(\cdot)$ decrypt the ciphertext. Paillier cryptosystem has the following excellent properties: 1) *Homomorphic addition:* $D_{sk}(E_{pk}(m_1) * E_{pk}(m_2)) = m_1 + m_2$ and $D_{sk}(E_{pk}(m_1)^{m_2}) = m_1 * m_2$, where m_1, $m_2 \in \mathbb{Z}_n^*$, n is a product of two large primes. 2) *Indistinguishability:* the same plaintext m is encrypted by pk multiple times, the obtained ciphertexts are different, i.e., $E_{pk}(m)_1 \neq E_{pk}(m)_2$ and $D_{sk}(m)_1 = D_{sk}(m)_2$.

Yao's garbled circuits. Yao's garbled circuits (a.k.a Yao's protocol) [17] is a general solution for secure two-party computation. The main idea is that two parties C_1 and C_2, who respectively hold their own private inputs m_1 and m_2, calculate an arbitrary function $f(m_1, m_2)$ without leaking their inputs. The main method of Yao's protocol is that C_1 (*circuit generator*) transforms the function f into an encrypted boolean circuit (*garbled circuit*) and generates the inner circuit labels (*garbled values*) of own input m_1, denoted as $\widetilde{m_1}$, and then sends this garbled circuit and garbled values to the C_2 (*circuit evaluator*). To secretly obtain the garbled values of C_2's input value m_2, C_1 and C_2 cooperate to execute *1-out-of-2 oblivious transfer* (OT) protocol [18]. Finally, with inputting garbled values $\widetilde{m_1}$ and $\widetilde{m_2}$, C_1 and C_2 execute the garbled circuit $f(\widetilde{m_1}, \widetilde{m_2})$, and output the result.

We briefly introduce the following garbled circuits, which have been constructed in [19]. Note that all the inputs and outputs are inner circuit labels, and the cloud servers do not learn any information from these labels.

- **XOR/AND:** The two circuits take as input an array $\{\widetilde{a_1}, \widetilde{a_2}, ..., \widetilde{a_n}\}$, where $\widetilde{a_i}$ is a l-bit binary, and return a l-bit value $\widetilde{z} = \widetilde{a_1} \oplus / \wedge \widetilde{a_2} \oplus / \wedge, ..., \oplus / \wedge \widetilde{a_n}$.
- **ADD/SUB:** The ADD/SUB circuit outputs an unsigned value of the addition/subtraction of two numbers $\widetilde{a_1}$ and $\widetilde{a_2}$, i.e., $\widetilde{z} = |\widetilde{a_1} + / - \widetilde{a_2}|$.
- **CMP:** To secretly compare the values of two numbers, we use CMP circuit input two l-bit binary numbers $\widetilde{a_1}$ and $\widetilde{a_2}$ to return a one-bit compared result $\widetilde{z}$. If $a \leq b$, then $z = 1$; otherwise, $z = 0$.
- **MUX:** The MUX circuit is a multiplexer that has three inputs $\widetilde{a_1}$, $\widetilde{a_2}$, and an extra bit $\widetilde{\sigma}$. If $\sigma = 0$, the MUX circuit outputs a_1; otherwise, outputs a_2. In this paper, we usually use this circuit to remove the invalid bids by setting 1^l. That is, we input $\widetilde{b_{i,j}}$, $\widetilde{1^l}$, and an extra bit $\widetilde{\sigma}$, i.e., if $\sigma = 1$, we set the bid $b_{i,j} = 1^l$.

3 Our protocol

3.1 Overview

The overview of PTATA is proposed in Algorithm 1. Specifically, the cloud agent C_2 generates a key pair (pk, sk) of Paillier cryptosystem, and publishes pk. Sellers and buyers submit the encrypted data $(E(B), E(R), E(r))$ to the cloud auctioneer C_1. After that, C_1 secretly shares these encrypted data with C_2 via the property of homomorphic addition, denoted as $(\langle B\rangle, \langle R\rangle, \langle r\rangle)$, in which the secret-shared value $\langle r\rangle$ is $\langle r\rangle^{C_1} = s \ mod \ 2^l$ and $\langle r\rangle^{C_2} = (r + s) \ mod \ 2^l$, $s \in \mathbb{Z}_{2^l}$ is a random number generated by C_1. To run the auction process in a oblivious way, C_1 constructs the garbled circuits of the original auction mechanism and generates the inner circuit labels of all the inputs. Finally, C_1 and C_2 cooperate to execute garbled circuits to output the actual auction results. The main algorithm of auction circuit construction will be presented later.

Algorithm 1 The overview of PTATA

Input: Sellers: the amount of resources provided R and the set of bids B.
Buyers: the amount of resources required r.
Output: C_1 and C_2: the actual auction results.

Phase 1: Encrypted Private Data

1: **C_2:** generates a key pair (pk, sk) of Paillier cryptosystem.
2: **Each Seller d_i:** encrypts its bids B_i and available resources R_i by pk, i.e., $E(B_i)$ and $E(R_i)$, and sends them to C_1.
3: **Buyers:** encrypt resources required r by pk, and send $E(r)$ to C_1.

Phase 2: Computing Garbled Circuits

4: **C_1 and C_2:** generates the secret-shared values of all received data.
5: **C_1:** converts the original auction mechanism into garbled circuits, generates the inner circuit labels of its inputs, and then sends garbled circuits and garbled values to C_2.
6: **C_1 and C_2:** compute the garbled values of C_2's inputs via OT protocol, run garbled circuits of secure winning bids determination (Section 3.2) and secure payments computation (Section 3.3), and output the actual results.

3.2 Secure Winning Bids Determination

After receiving the encrypted data, C_1 and C_2 execute secure winning bids determination protocol to secretly obtain all the winning bids and the overall cost. Specifically, as shown in Algorithm 2, C_1 first calculates the amount of tasks that each seller d_i can accept with C_2, i.e., $K_i = min\{\lfloor \frac{R_i}{r} \rfloor, m\}(1 \leq i \leq n)$, in which $\lfloor \frac{R_i}{r} \rfloor$ can be computed via secure division computation protocol (SDC) [20] based on the property of Paillier homomorphic addition. The inputs of SDC are $E(R_i)$ and $E(r)$, and the output is the secret-shared value $\langle Rr_i \rangle$. Based on OT protocol, C_1 and C_2 obtain the garbled values $(\langle \widetilde{B} \rangle, \langle \widetilde{R} \rangle, \langle \widetilde{Rr} \rangle)$. To secretly calculate acceptable task amount K_i, C_1 and C_2 invoke the following TwoSMIN circuit. As shown in Fig. 2, we combine two SUB circuits and a MIN circuit to realize the desired functionality.

TwoSMIN Circuit. Since the secret-shared values has been transformed into garbled values, the complete values can be obtained by the SUB circuit, e.g., $\widetilde{a} = \text{SUB}(\widetilde{a+s}, \widetilde{s})$, s is a random number, and the MIN circuit is used to output the minimum value between two numbers. e.g., if $a \geq b$, then $\text{MIN}(\widetilde{a}, \widetilde{b})$ outputs $\widetilde{z} = \widetilde{b}$; otherwise, outputs $\widetilde{z} = \widetilde{a}$. Note that, the MIN circuit has been proposed in [19], where σ^1 is a one-bit comparison result. Based on the MIN circuit, we construct the CMIN circuit to output this comparison result $\widetilde{\sigma^1}$, which can be used to determine the index of minimum value.

Based on TwoSMIN circuit, C_1 and C_2 can compute the acceptable task amount $\widetilde{K_i} = \text{TwoSMIN}(\widetilde{Rr_i}, \widetilde{m})$ of each seller b_i. Next, the main process is to secretly determine the minimum bid and its index from B. To realize the above functionality, we build an efficient FILMIN circuit, as shown in Fig 3. Compared with the recent work [7], the computational overhead is reduced by 50%.

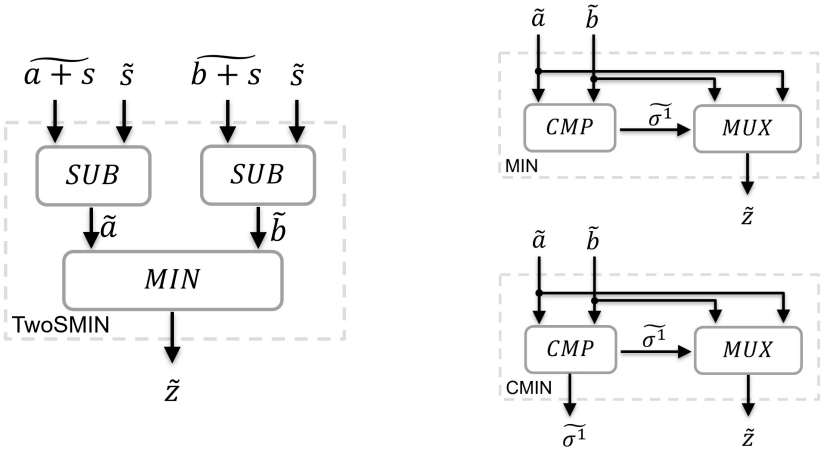


Fig. 2: The structure of TwoSMIN, MIN, and CMIN circuits

FILMIN Circuit. As shown in Fig. 3, we combine CMIN and FILTER circuits to get the minimum value and its index from an array, e.g., if an array $\{a_1, a_2, a_3, a_4\}$ is $\{4, 3, 3, 9\}$, then FILMIN$(\widetilde{a_1}, \widetilde{a_2}, \widetilde{a_3}, \widetilde{a_4})$ outputs the minimum value 3 and its index $\{0, 0, 1, 0\}$. The CMIN circuit that we constructed in Fig. 2 outputs the minimum value $\widetilde{z}$ and the comparison result $\widetilde{\sigma^1}$, and the FILTER circuit [21] is used to filter binary numbers, e.g, FILTER(1110) outputs {0,0,1,0}.

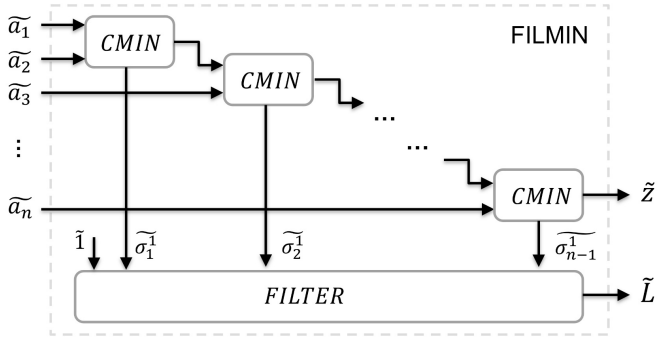


Fig. 3: The structure of the FILMIN circuit

FILMIN circuit is executed by C_1 and C_2 to get the winning bid $\widetilde{b_{i_k,j_k}}$ and its index set $\widetilde{L_k}$. After that, C_1 and C_2 need to secretly update the acceptable task amount K_i and remove the invalid bids in B, which are used to determine the next winning bid. The above operations are presented in detail as follows.

1 Based on index set $\widetilde{L_k}$, the indexes $(\widetilde{x_i}, \widetilde{y_j})$ of the winning seller d_{i_k} and the winning task t_{j_k} can be secretly calculated via XOR circuit (line 9-10), in which only $y_{j_k} = 1$ and $x_{i_k} = 1$, and others are 0.

2 After obtaining the seller d_{i_k}'s index set $\widetilde{x_i}$, C_1 and C_2 use SUB circuit to update the acceptable task amount $\widetilde{K_i}$, i.e., $K_i = K_i - x_i$. To secretly determine which K_i is 0, they evaluate the EQ circuit to output a one-bit value $\widetilde{e_i}$. That is, if K_i is 0, then $e_i = 1$; otherwise, $e_i = 0$.

3 Since all the bids of the seller d_i, whose K_i is 0, are invalid, and all the bids for the task t_{j_k} are invalid, C_1 and C_2 according to $\widetilde{e_i}$ and $\widetilde{y_j}$ can determine the invalid bids, denoted as the flag set $\widetilde{\sigma_{i,j}}$. That is, if the bid $b_{i,j}$ is invalid, then $\sigma_{i,j} = 1$; otherwise, $\sigma_{i,j} = 0$. Next, MUX circuit is used to remove all the invalid bids, denoted as $\widetilde{1^l}$, i.e., if $\sigma_{i,j} = 1$, then $b_{i,j} = 1^l$; otherwise, $b_{i,j} = b_{i,j}$.

Algorithm 2 Secure Winning Bids Determination

Input: C_1: Encrypted $E(B)$, $E(R)$, and $E(r)$.
C_2: A key pair (pk, sk).

Output: C_1 and C_2: $\widetilde{W} = \{\widetilde{b_{i_k,j_k}}\}_{k=1}^m$, $\widetilde{L} = \{\widetilde{L_k}\}_{k=1}^m$, and the overall cost $\widetilde{\mathbb{C}_B}$.

C_1 **and** C_2**:**

$\widetilde{\langle Rr_i \rangle} = \widetilde{SDC}(E(R_i), E(r)), (\forall i \in [1, n])$.
Convert $(E(B), E(R))$ into secret-shared values $(\langle B \rangle, \langle R \rangle)$.
Generate the garbled values $(\widetilde{B}, \widetilde{R}, \widetilde{Rr})$ via OT protocol and SUB circuit.
$\widetilde{K_i} = \text{TwoSMIN}(\widetilde{Rr_i}, \widetilde{m}), (\forall i \in [1, n])$.
Initialize $\widetilde{W} = \varnothing$, $\widetilde{L} = \varnothing$, and $\widetilde{\mathbb{C}_B} = 0$.
for $k = 1$ to m **do**
  FILMIN$(\widetilde{b_{1,1}}, \widetilde{b_{1,2}}, ..., \widetilde{b_{n,m}})$, get the least value $\widetilde{b_{i_k,j_k}}$ and its index $\widetilde{L_k}$.
  Set $\widetilde{b_{i_k,j_k}}$ as one of the winning bids.
  Task t_{j_k}'s index: $\widetilde{y_j} = \text{XOR}(\widetilde{l_{1,j}}, ..., \widetilde{l_{n,j}}), (\forall j \in [1, m])$, where $\widetilde{l_{i,j}} \in \widetilde{L_k}$.
  Seller d_{i_k}'s index: $\widetilde{x_i} = \text{XOR}(\widetilde{l_{i,1}}, ..., \widetilde{l_{i,m}}), (\forall i \in [1, n])$.
  Update the acceptable task amount $\widetilde{K_i} = \text{SUB}(\widetilde{K_i}, \widetilde{x_i})$.
  $\widetilde{e_i} = \text{EQ}(\widetilde{K_i}, 0)$, if K_i is 0 that the seller d_i should be remove, set $e_i = 1$.
  Find the invalid bid indexes in $\widetilde{B}$ via $\widetilde{y_j}$ and $\widetilde{e_i}$, denoted as $\widetilde{\sigma_{i,j}}$.
  Remove all the invalid bids $\widetilde{b_{i,j}} = \text{MUX}((\widetilde{b_{i,j}}, \widetilde{1^l}), \widetilde{\sigma_{i,j}})$.
  $\widetilde{L} = \widetilde{L} \cup \widetilde{L_k}$ and $\widetilde{W} = \widetilde{W} \cup \{\widetilde{b_{i_k,j_k}}\}$.
end
Calculate the overall cost $\widetilde{\mathbb{C}_B} = \text{ADD}(\widetilde{b_{i_1,j_1}}, \widetilde{b_{i_2,j_2}}, ..., \widetilde{b_{i_m,j_m}})$.
return $\widetilde{W} = \{\widetilde{b_{i_k,j_k}}\}_{k=1}^m$, $\widetilde{L} = \{\widetilde{L_k}\}_{k=1}^m$, and $\widetilde{\mathbb{C}_B}$.

After that, C_1 and C_2 put this winning bid value $\widetilde{b_{i_k,j_k}}$ and its index set $\widetilde{L_k}$ into the set $\widetilde{W}$ and $\widetilde{L}$, respectively. They continue the above process until all the tasks are allocated. Finally, they use a ADD circuit to calculate the overall cost $\widetilde{\mathbb{C}_B} = \text{ADD}(\widetilde{b_{i_1,j_1}}, \widetilde{b_{i_2,j_2}}, ..., \widetilde{b_{i_m,j_m}})$, and then output $\widetilde{W}$, $\widetilde{L}$, and $\widetilde{\mathbb{C}_B}$.

3.3 Secure Payments Computation

Based on the winning bids determined in the above subsection, C_1 and C_2 need to secretly calculate the payment for each winning bid. Specifically, as shown in

Algorithm 3, they initially restore $(\widetilde{B}, \widetilde{R}, \widetilde{K})$ to the initial garbled values. For each winning bid $\widetilde{b_{i_k,j_k}}$ in $\widetilde{W}$, they first use MUX circuit to remove $\widetilde{b_{i_k,j_k}}$ from $\widetilde{B}$ and then re-execute Algorithm 2 to get the another overall cost $\widetilde{\mathbb{C}}_{B\setminus\{b_{i_k,j_k}\}}$. Based on Eq. (2), two SUB circuits are used to obtain the payment $\widetilde{p_{i_k,j_k}}$. After that, C_1 and C_2 continue the above process until all the payments are calculated. Finally, they reveal all the winning sellers and buyers (d_{i_k}, t_{j_k}) and the payments p_k. The payments for other bids that do not win are 0.

Algorithm 3 Secure Payments Computation

Input: C_1 and C_2: $\widetilde{W} = \{\widetilde{b_{i_k,j_k}}\}_{k=1}^{m}$, $\widetilde{L} = \{\widetilde{L_k}\}_{k=1}^{m}$, and $\widetilde{\mathbb{C}_B}$.
Output: C_1 and C_2: The actual auction results $(d_{i_k}, t_{j_k}, p_{i_k,j_k})$.

C_1 **and** C_2**:**

1: Restore $(\widetilde{B}, \widetilde{R}, \widetilde{K})$ to the initial values.
2: **while** $\forall \widetilde{b_{i_k,j_k}} \in \widetilde{W}$ **do**
3: Remove the winning bid $\widetilde{b_{i,j}} = \text{MUX}((\widetilde{b_{i,j}}, 1^l), \widetilde{l_{i,j}})$, where $\widetilde{l_{i,j}} \in \widetilde{L_k}$.
4: Execute Alg. 2(line 5-18) to output $\widetilde{\mathbb{C}}_{B\setminus\{b_{i_k,j_k}\}}$ without $\widetilde{b_{i_k,j_k}}$.
5: \\ Calculate $p_{i_k,j_k} = \mathbb{C}_{B\setminus\{b_{i_k,j_k}\}} - (\mathbb{C}_B - b_{i_k,j_k})$.
6: $\widetilde{p_{i_k,j_k}} = \text{SUB}(\widetilde{\mathbb{C}}_{B\setminus\{b_{i_k,j_k}\}}, \text{SUB}(\widetilde{\mathbb{C}_B}, \widetilde{b_{i_k,j_k}}))$.
7: **end**
8: Reveal the results of winners (d_{i_k}, t_{j_k}) and the payments p_{i_k,j_k}.
9: The payments for other bids that do not win are 0.

3.4 Security and Efficiency Analysis

Security Analysis. Based on composition theory [15], we prove the cryptography security of PTATA under the semi-honest model [14].

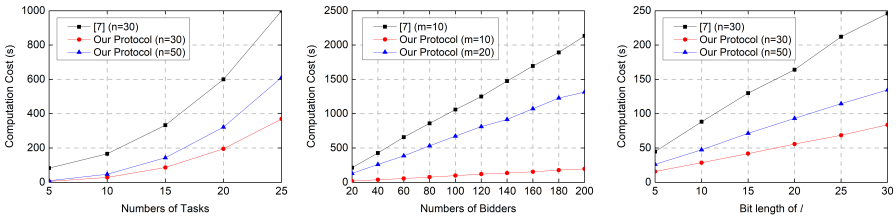
Theorem 1 *As long as Paillier cryptosystem and various circuits are secure under the semi-honest model, PTATA is secure under the semi-honest model.*

Proof. On the one hand, Since C_2 is responsible for generating the key pair (pk, sk) of Paillier cryptosystem, C_1 cannot decrypt the encrypted data. Before obtaining the secret-shared values, C_1 uses homomorphic addition to randomize these encrypted data, which sent to C_2. Since Paillier cryptosystem has been proved to be semantically secure, C_1 and C_2 cannot learn anything from these encrypted data and secret-shared values. On the other hand, various circuits including XOR, EQ, MIN, SUB, CMP, MUX and TwoSMIN, and FILMIN, are both applied in Yao's garbled circuits, and all intermediate values are inner circuit labels. Since Yao's garbled circuits have been proved to be secure under the semi-honest model [22], PTATA is secure under the semi-honest model.

Efficiency Analysis. The main cost is garbled circuits for execution. Fortunately, the XOR gate has almost no overhead with "free XOR" technique [23], and the efficiency of our system depends on the amount of non-XOR gates. In

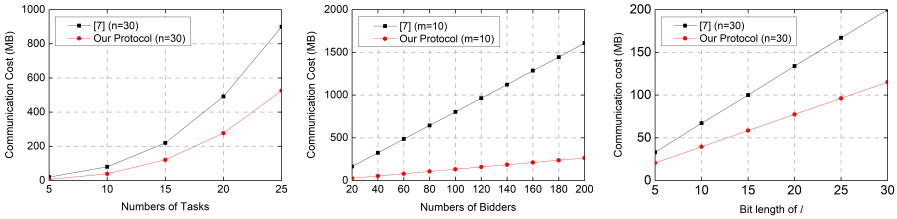
Algorithm 2, the main process is to determine the winning bids and remove the invalid bids, and the efficiency of this process is $O(nm^2l)$, where l is the bit length of each bid. In Algorithm 3, the main process is to calculate the overall cost without the winning bid, and the efficiency of this process is $O(nm^3l)$.

4 Experiments



(a) Numbers of Tasks m (b) Numbers of Bidders n (c) Bit length of l

Fig. 4: Computation cost induced by PTATA.



(a) Numbers of Tasks m (b) Numbers of Bidders n (c) Bit length of l

Fig. 5: Communication cost induced by PTATA.

We implement our scheme in FastGC [21], which is a Java-based framework. The cloud servers C_1 and C_2 are both simulated on an Intel i5-11600H CPU, 3.90GHz, and 16GB RAM computer. The security level of inner circuit labels for garbled circuits is 80-bit and the security modulus of Paillier cryptosystem is 1024-bit. In our experiments, we mainly evaluate the computation and communication costs for different bidder numbers n, task numbers m, and bit lengths l. The default settings of n, m, and l are 30, 10, and 10, respectively.

As shown in Fig. 4 and Fig. 5, we compare the computation and communication costs of our protocol with the recent work [7]. We can see that the increasing trends of the costs for different n, m, and l are consistent with our analysis $O(nm^3l)$. Note that, our protocol is more efficient than this recent work. Specifically, in Fig. 4(a) and Fig. 5(a), when $n = 30$ and m changes from 5 to 25, the costs of our protocol increase from (5.7s, 6.1MB) to (369.5s, 526.1MB), but the costs of the recent work increase from about (80s, 20MB) to (990s, 900MB). In Fig. 4(b) and Fig. 5(b), when $m = 10$ and $n = 200$, our protocol (195.8s, 263.6MB) only costs about 9.1% computation overhead and 16.3%

communication overhead of the recent work, respectively. In summary, the experiment results show that the costs of our protocol are acceptable in practical applications.

5 Related Work

Recently, there are various works to deal with privacy-preserving cloud auction, Specifically, Chen *et al.* [5] first design a privacy-preserving cloud auction for Virtual Machines (VMs) allocation based on a data-oblivious way, which protects the privacy of bidders. However, this scheme does not consider available resources privacy. Then, Cheng *et al.* [6] propose an efficient and secure double cloud auction scheme, which can protect the privacy of all users. Nevertheless, the challenges of this solution (such as secure compare-and-swap and secure sorting) are different from our problem. Besides, the recent work [7] proposes a secure auction scheme for heterogeneous task assignment, but this scheme does not consider the number of task required privacy.

Another related research topic is privacy-preserving auction for spectrum allocation. Chen *et al.* [8, 9] propose two privacy-preserving double auction schemes for homogenous spectrum allocation based on a series of secure arithmetic public-key operations. To improve the system efficiency, the work [10] designs a secure spectrum auction scheme via public-key encryption system, but this scheme does not consider bidders' location information privacy. To protect location information, Wang *et al.* [11, 12] propose a series of privacy-preserving and truthful auction schemes for double spectrum auction. Unfortunately, these schemes disclose the access patterns privacy. Recently, Cheng *et al.* [13] propose a lightweight framework, which ensures high efficiency while providing a strong security guarantee. However, this solution involves large amounts of pre-computed multiplication triplets between two cloud servers, which is unrealistic.

6 Conclusion

In this paper, we have proposed PTATA, a privacy-preserving and truthful auction scheme for task assignment in outsourced cloud environments. Moreover, we have proved that PTATA protocol is secure under the semi-honest model and have analyzed the system efficiency. Based on extensive experiments, our solution is acceptable in real-life applications.

Acknowledgement

This work was partially supported by Natural Science Foundation of China (Grant No. 61602400) and Jiangsu Provincial Department of Education (Grant NO. 16KJB520043).

References

1. Wang, X., Chen, X. Wu, W.: Towards truthful auction mechanisms for task assignment in mobile device clouds. In: Proceedings of Conference on Computer Communications (INFOCOM), pp. 1-9. IEEE, Atlanta (2017)

2. Jin, A.-L., Song, W. Zhuang, W.: Auction-based resource allocation for sharing cloudlets in mobile cloud computing. Transactions on Emerging Topics in Computing **6**(1), 1-12 (2015)
3. Wang, A.-L., Song, W., Wang, P., *et al.*: Auction mechanisms toward efficient resource sharing for cloudlets in mobile cloud computing. Transactions on Services Computing **9**(6), 1-14 (2015)
4. Shi, J., Yang, Z., Zhu, J.: An auction-based rescue task allocation approach for heterogeneous multi-robot system. Multimedia Tools and Applications **79**(21-22), 14529-14538 (2018)
5. Chen, Z., Chen ,L., Huang, L., *et al.*: On privacy-preserving cloud auction. In: Symposium on Reliable Distributed Systems, pp. 279-288. IEEE, Budapest (2016)
6. Cheng, K., Shen, Y., Zhang, Y., *et al.*: Towards efficient privacy-preserving auction mechanism for two-sided cloud markets. In: ICC, pp. 1-6. IEEE (2019)
7. Jiang, X., Pei, X., Tian, D., *et al.*: Privacy-Preserving Auction for Heterogeneous Task Assignment in Mobile Device Clouds. In: WASA, pp. 345-358. Springer (2021)
8. Chen, Z., Huang, L., Li, L., *et al.*: Ps-trust: Provably secure solution for truthful double spectrum auctions. In: INFOCOM, pp. 1249-1257. IEEE, Toronto (2014)
9. Chen, Z., Wei, X., Zhong, H., *et al.*: Secure, efficient and practical double spectrum auction. In: IWQoS, pp. 1-6. IEEE, Vilanova (2017)
10. Wang, J., Lu, N., Cheng, Q., *et al.*: A secure spectrum auction scheme without the trusted party based on the smart contract. Digital Communications and Networks **7**(2), 223-234 (2021)
11. Wang, Q., Huang, J., Chen, Y., *et al.*: Privacy-preserving and truthful double auction for heterogeneous spectrum. TON **27**(2), 848-861 (2019)
12. Wang, Q., Huang, J., Chen, Y., *et al.*: Prost: Privacy-preserving and truthful online double auction for spectrum allocation. Transactions on Information Forensics and Security **14**(2), 374-386 (2019)
13. K. Cheng, L. Wang, Y. Shen, *et al.*: A Lightweight Auction Framework for Spectrum Allocation with Strong Security Guarantees. In: INFOCOM, pp. 1708-1717. IEEE Toronto (2020)
14. Goldreich, O.: Foundations of Cryptography. Cambridge University Press, Volume 2-Basic Applications (2004)
15. Liu, A., Zheng, K., Li, L., *et al.*: Efficient secure similarity computation on encrypted trajectory data. In: ICDE, pp. 66-77. IEEE (2015)
16. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: EUROCRYPT, pp. 223-238. Springer (1999)
17. Yao, A. C.: How to generate and exchange secrets. In: FOCS, pp. 162-167. (1986)
18. Ishai, Y., Kilian, J., Nissim, K., *et al.*: Extending oblivious transfers efficiently. In: CRYPTO, pp. 145-161. Springer (2003)
19. Kolesnikov, V., Sadeghi, A.-R., Schneider, T.: Improved garbled circuit building blocks and applications to auctions and computing minima. In: Cryptology and Network Security (CANS), pp. 1-20. Springer (2009)
20. Cui, N., Yang, X., Wang, B., *et al.*: SVknn: Efficient secure and verifiable k-nearest neighbor query on the cloud platform, in: ICDE, pp. 253-264. IEEE (2020)
21. Huang, Y., Evans, D., Katz, J., *et al.*: Faster secure two-party computation using garbled circuits. In: USENIX Security, San Francisco (2011)
22. Lindell, Y. Pinkas, B.: A proof of security of Yao's protocol for two-party computation. Journal of Cryptology **22**(2), 161-188 (2009)
23. Kolesnikov, V. Schneider, T.: Improved Garbled Circuit: Free XOR Gates and Applications. In: ICALP, pp. 486-498. Springer (2008)