Efficient and Private Divisible Double Auction in Trusted Execution Environment

Bingyu Liu, Shangyu Xie, and Yuan Hong(✉)

Department of Computer Science, Illinois Institute of Technology, Chicago, USA
{bliu40,sxie14}@hawk.iit.edu, yuan.hong@iit.edu

Abstract. Auction mechanisms for exchanging divisible resources (e.g., electricity, cloud resources, and network bandwidth) among distributed agents have been extensively studied. In particular, divisible double auction allows both buyers and sellers to dynamically submit their prices until convergence. However, severe privacy concerns may arise in the double auctions since all the agents may have to disclose their sensitive data such as the bid profiles (i.e., bid amounts and prices in different iterations) to other agents for resource allocation. To address such concerns, we propose an efficient and private auction system ETA by co-designing the divisible double auction mechanism with the Intel SGX, which executes the computation for auction while ensuring confidentiality and integrity for the buyers/sellers' sensitive data. Furthermore, ETA seals the bid profiles to achieve a Progressive Second Price (PSP) auction for optimally allocating divisible resources while ensuring truthfulness with a Nash Equilibrium. Finally, we conduct experiments to validate the performance of private auction system ETA.

Keywords: Secure computation · Auction mechanism design · TEE

1 Introduction

In the past decade, divisible resources (e.g., electricity, computation and storage resources in the cloud, stock shares, and network bandwidth) have been frequently exchanged or allocated in a peer-to-peer mode. All the buyers and sellers can trade any amount of the resources in such markets. In such markets, all the buyers/sellers generally compete with each other to maximize their payoffs. Then, divisible double auction mechanisms [48] have been extensively studied to allow both buyers and sellers to dynamically submit their prices until convergence. For instance, all the participants will converge to achieve a Nash Equilibrium (NE) [23,31] in a game.

However, severe privacy concerns may arise in double auctions [4]. For instance, if the bid profiles (i.e., amounts and prices) are disclosed, rival agents may be able to win more payoffs by reporting untruthful bids. Such violation of truthfulness would explicitly deviate the market of exchanging divisible resources

B. Chen and X. Huang (Eds.): AC3 2021, LNICST 386, pp. 75–92, 2021.
https://doi.org/10.1007/978-3-030-80851-8_6

[27]. Even worse, agents (aka. buyers or sellers) may collect such information from their competitors, and misuse such private data, e.g., reselling the data [4]. Thus, it is desirable to explore a divisible double auction mechanism while preserving all the agents' privacy (e.g., sealing all the bid profiles).

To this end, we propose an efficient and truthful auction system ETA by co-designing the divisible double auction mechanism with the Intel SGX, which is a Trusted Execution Environment (TEE) supported by an architecture extension of Intel [16]. Then, ETA is designed in two folds. On one hand, the divisible double auction mechanism will be designed based on the Progressive Second Price (PSP) [28] auction, which is extended from the Vickrey-Clarke-Groves (VCG) [38] auction and ensures truthfulness for all the buyers/sellers. No buyers or sellers can gain any additional payoff by changing their strategies since the best responses result in an equilibrium. On the other hand, Intel SGX allows applications to execute within a protected environment called an *enclave* [37] that ensures the confidentiality and integrity for all the buyers and sellers' sensitive data. In summary, we make the following major contributions in this paper:

- We propose an efficient and private auction mechanism ETA that executes truthful divisible double auction without disclosing private information.
- The auction mechanism (based on the PSP auction) in ETA ensures Individual Rationality, Incentive Compatibility and Pareto Efficiency. The privacy of all the buyers and sellers (i.e., bid profiles, and valuation function) can also be guaranteed in ETA. We formally analyze such properties for ETA.
- We design and implement the system prototype for ETA with the Intel SGX, and conduct experiments to validate the performance using real datasets.

2 Divisible Double Auction

In the divisible double auction, we denote the sets of buyers and sellers as $\mathcal{B}$ and $\mathcal{S}$, respectively. b_m and s_n are defined as two-dimensional bid profiles (bid price, and the maximum amount to buy or sell) as follows: (1) buyer $m \in \mathcal{B}$: $b_m = (\alpha_m, d_m)$ with bid price α_m and amount d_m to buy, and (2) seller $n \in \mathcal{S}$: $s_n = (\beta_n, h_n)$ with bid price β_n and amount h_n to sell. Each buyer or seller is required to submit bid b_m or s_n before the auction starts. $b = (b_m, m \in \mathcal{B})$ represents the bid profiles of all the buyers while $s = (s_n, n \in \mathcal{S})$ denotes the bid profiles of all the sellers. $r = (b, s)$ is defined as the bid profiles for all the agents. These are private information, supposed to be sealed among all the participants.

2.1 Models

We represent the strategy of each agent with a non-negative valuation function $\widehat{V}_m(\cdot)$ for buyers and negative cost function $\widehat{C}_n(\cdot)$ for sellers. These functions are also considered as private information, which quantifies the value or cost of the resources by each buyer or seller.

In the mechanism design, we adopt generic assumptions [28,38] for the valuation function $\widehat{V}_m(\cdot)$: (1) $\widehat{V}_m$ is differentiable and $\widehat{V}_m(0) = 0$, and (2) $\widehat{V}'_m(\cdot)$ is

non-increasing and continuous. A_m and A_n represent the allocation of buyer m and seller n, respectively. In the k-th iteration of the double auction, $A_m^{(k)}, A_n^{(k)}$ represent the allocation of buyer m and seller n, respectively. For all agents, we assume that $\widehat{V}_m(A_m^k) > \widehat{V}_m(A_m^{k+1})$ where $\forall m \in \mathcal{B}$ and $A_m^k < A_m^{k+1}$ while $\widehat{C}_n(A_n^k) < \widehat{C}_n(A_n^{k+1})$ where $\forall n \in \mathcal{S}$ and $A_n^k < A_n^{k+1}$, since buyers have diminishing marginal utility while sellers have increasing marginal cost.

We also denote the payoff function for buyer m and seller n as $f_m(r)$ and $f_n(r)$, respectively, for representing their payoffs w.r.t. the bid profiles of all the buyers/sellers r. In addition, ρ_m is the payment made by buyer m while ρ_n is the payment received by seller n. Also, $\rho(r_i, r_{-i})$ is defined as the difference between all the buyers' aggregated valuation if any buyer i is not in the auction minus the aggregated valuation if i is in the auction [25,28,48]. Similarly, $\rho(r_j, r_{-j})$ is defined as the difference between all the sellers' aggregated cost if any seller j is not in the auction minus the aggregated cost if j is in the auction. Thus, we have:

$$\rho(r_i, r_{-i}) = \sum_{m \neq i} \alpha_m [A_m(0; r_{-i}) - A_m(r_i; r_{-i})]$$

$$\rho(r_j, r_{-j}) = \sum_{n \neq j} \beta_n [A_n(0; r_{-j}) - A_n(r_j; r_{-j})] \tag{1}$$

Then, given the optimal allocation profile for buyer $m \in \mathcal{B}$ and seller $n \in \mathcal{S}$ as A_m^* and A_n^*, we can define the payoff of the buyer m and seller n as:

$$f_m(r) = \widehat{V}_m(A_m^*) - \rho(r_i, r_{-i}), \forall m \in \mathcal{B}$$

$$f_n(r) = \rho(r_j, r_{-j}) - \widehat{C}_n(A_n^*), \forall n \in \mathcal{S} \tag{2}$$

Definition 1 (Incentive Compatibility). *The divisible double auction is incentive compatible [25] if the following conditions hold:*

$$\sum_{m \in \mathcal{B}} \widehat{V}_m(r) - \rho_m \geq \sum_{m \notin \mathcal{B}} \widehat{V}_m(r) - \rho(r_i, r_{-i}), \forall m \in \mathcal{B}$$

$$\rho_n - \sum_{n \in \mathcal{S}} \widehat{C}_n(r) \geq \rho(r_j, r_{-j}) - \sum_{n \notin \mathcal{S}} \widehat{C}_n(r), \forall n \in \mathcal{S} \tag{3}$$

Incentive compatibility guarantees that every buyer/seller cannot make better payoff with untruthful bid if other buyers/sellers report the true bids.

Definition 2 (Feasible). *The divisible double auction mechanism is feasible, we have:*

$$\sum_{\forall m \in \mathcal{B}} d_m \leq \sum_{\forall n \in \mathcal{S}} h_n \tag{4}$$

This mechanism is *feasible* requires that the amount of resources that sold by sellers is weakly larger than the amount that buyers intend to purchase. In other words, it ensures that the overall supply satisfies the demand.

Definition 3 (Clearing Price). *If there exists a feasible and efficient allocation, such that, $A^*(\cdot)$ at the price θ, the social welfare (defined as $F(\cdot) = \sum_{m\in\mathcal{B}} V_m(A_m) - \sum_{n\in\mathcal{S}} C_n(A_n)$) is maximized to achieve the best response. Then, price θ is defined as the* clearing price.

We say that the clearing price θ [5] supports the optimal allocation $A^*(\cdot)$ with the maximum social welfare.

Definition 4 (Nash Equilibrium). *Given the bid profiles r^*, a Nash Equilibrium (NE) holds for double auction such that:*

$$f_m(b_m^*, r_{-m}^*) \geq f_m(b_m, r_{-m}^*), \forall m \in \mathcal{B}$$

$$f_n(s_n^*, r_{-n}^*) \geq f_n(s_n, r_{-n}^*), \forall n \in \mathcal{S} \tag{5}$$

where $r_{-m} = r \setminus b_m$ is a bid profile for all the buyers except buyer m and $r_{-n} = r \setminus s_n$ is a bid profile for all the sellers except seller n.

ETA aims at achieving the best response (approximate allocation efficiency) to maximize social welfare of allocation to all the buyers and sellers. It also guarantees that the truthfulness of bid profiles is the best response for all the agents. More importantly, each buyer or seller's submitted bid profiles (amounts, prices) and its valuation/cost function are protected in ETA.

2.2 Auction Properties

Recall that ETA will be designed based on the Progressive Second Price (PSP) [28] auction. Thus, it has the following properties.

- ***Weakly dominant strategy*** [28]. Each buyer/seller truthfully participates in the auction would gain more payoff than the untruthful response. The auction mechanism in ETA pursues weakly dominant strategies since it is extended from the PSP auction mechanism (second price).
- ***Budget balanced***. We assume "no budget deficit", the total budget of the buyers is no less than the total payment requested by the sellers: $\sum_{\forall m\in\mathcal{B}}(\alpha_m \cdot d_m) \geq \sum_{\forall n\in\mathcal{S}}(\beta_n \cdot h_n)$.
- ***Individual rationality***. All the buyers and sellers will have non-negative payoffs in the auction.
- ***Pareto efficiency*** [9,43]. The divisible resources are supposed to be sold to buyers with the highest valuation.
- ***Privacy***. Buyers/sellers' bid profiles (bid prices and amounts) and valuation/cost functions are kept private; every pair of potential buyer and seller only know their transaction amount and the clearing price.

3 Mechanism Design

In this section, we design the divisible double auction mechanism and its program $\mathsf{Prog}_{\mathsf{x}}$ in Intel SGX. The program ensures that all the bid profiles achieve the best response in multiple iterations for the Nash Equilibrium and eventually converge with a termination condition.

Initialization. While executing $\mathsf{Prog}_{\mathsf{x}}$, the bid profiles of all the buyers and sellers will be checked to guarantee that $(\alpha_i)_{\max} \geq (\beta_j)_{\min}$. Otherwise, it requests them to update the bid profiles. Meanwhile, it ensures that the potential amount of the resources in the auction C is smaller than the overall demand and supply. The auction will start once the above conditions are satisfied.

Divisible Double Auction $\mathrm{Prog}_x(\mathcal{B}, \mathcal{S}, r)$

Input: $\forall m \in \mathcal{B}, \forall n \in \mathcal{S}, r = (b, s)$

Output: $b_m^*, s_n^*, \forall m \in \mathcal{B}, \forall n \in \mathcal{S}$

set iteration $k := 1$

initialize $(\alpha_i)_{\max} \geq (\beta_j)_{\min}$ and $C < \min\{\sum_{i \in B} d_i, \sum_{j \in S} h_j\}$

while *true* **do**

$A_m^*(b, C) := \min\{d_m, \{[C - \sum_{i \in \mathcal{B}_m(b)} d_i], 0\}_{\max}\}$

$A_n^*(s, C) := \min\{h_n, \{[C - \sum_{i \in \mathcal{S}_n(s)} h_j], 0\}_{\max}\}$

$Q(r, C) := \min\{\sum_{i \in \mathcal{B}} A_i^*(r, C), \sum_{j \in \mathcal{S}} A_j^*(r, C)\}$

$\widehat{\mathcal{P}} := \dfrac{p_b(r, C) - p_s(r, C)}{\omega_{\max} + \sigma_{\max}}$

$\widetilde{C}(r, C) := Q(r, C) + \widehat{\mathcal{P}}$

$b_m^* = \arg\max\{f_m(b_m, b_{-m})\}, m \in \mathcal{B}$

$s_n^* = \arg\max\{f_n(s_n, s_{-n})\}, n \in \mathcal{S}$

terminate until convergence

iteration $k := k + 1$

endwhile

Fig. 1. Divisible double auction

Iteration. Given the total amount for the auction (potential amount) C, it will be updated as below:

$$\widetilde{C}(r, C) = Q(r, C) + \frac{p_b(r, C) - p_s(r, C)}{\omega_{\max} + \sigma_{\max}} \tag{6}$$

where $Q(r, c) = \min\{\sum_{m \in \mathcal{B}} A_m^*, \sum_{n \in \mathcal{S}} A_n^*\}$, $p_b(r, C) = \min\{\alpha_i, A_i \geq 0\}$ and $p_s(r, C) = \max\{\beta_j, A_j \geq 0\}$. And $\widehat{\mathcal{P}} = \frac{p_b(r,C) - p_s(r,C)}{\omega_{\max} + \sigma_{\max}}$. Specifically, $Q(r, c)$ is the

smaller one of the total demand and total supply; $\widehat{\mathcal{P}}$ is a coefficient for the gradients of marginal valuations (costs); $p_b(r, C)$ and $p_s(r, C)$ are two variables defined to help converge much faster in iterations via updating the potential amount; $\omega_{\max}$ is an upper bound for buyers' valuations $\omega_{\max} \geq \max \sup_{A_m} \{|\widehat{V}'_m(A_m)|\}$ while $\sigma_{\max}$ is a upper bound for sellers' costs $\sigma_{\max} \geq \max \sup_{A_n} \{|\widehat{C}'_n(A_n)|\}$. Using the gradients of marginal valuations/costs, the potential amount can efficiently reach a Nash Equilibrium.

In each iteration, A_m^* and A_n^* are the optimal allocation of buyers and sellers, respectively. Each agent updates its best response. Given (r, C), we derive the optimal allocation for each buyer $A_m^* = \min\{d_m, \{[C - \sum_{i \in B_m(b)} d_i], 0\}_{\max}\}$ and seller $A_n^* = \min\{h_n, \{0, [C - \sum_{j \in \mathcal{S}_n(s)} h_j]\}_{\max}\}$, where $B_m(b) = \{i \in \mathcal{B} | \alpha_i > \alpha_m\} \cup \{\alpha_i = \alpha_n \text{ and } i < m\}$ and $S_n(s) = \{j \in \mathcal{S} | \beta_j > \beta_n\} \cup \{\beta_i = \beta_m \text{ and } j < n\}$. Given the potential amount C obtained from initialization, the updated potential amount $\widetilde{C}(r, C)$ can be iteratively derived.

Best Response. We denote the best response of any buyer m as b_m^* and any seller n as and s_n^*. After updating the potential amount, we have the bid profiles $r = (b, s)$ and a pair of potential amount $(C, \widehat{C})$. Then, we define the best response as follows.

$$b_m^*(r, C, \widehat{C}) = \arg\max\{f_m(b_m, b_{-m})\}$$

$$s_n^*(r, C, \widehat{C}) = \arg\max\{f_n(s_n, s_{-n})\} \tag{7}$$

The auction program $\mathsf{Prog}_{\mathsf{x}}$ will find the best responses in each iteration and finally converges to a Nash Equilibrium. Note that the valuation and cost functions might be different. In this dynamic auction game, all the buyers/sellers recompute their best response to the current strategy (bid profiles) of other agents. We now study the game of the divisible double auction mechanism as follows (all of them are proven in the Appendix).

Theorem 1. *The divisible double auction in* ETA *ensures (1) Weakly dominant strategy, (2) Individual rationality, (3) Pareto efficiency, (4) Incentive compatibility, and (5) Feasibility.*

4 ETA System Design

In this section, we design the ETA system for deploying divisible double auction with the Intel SGX [16], which guarantees the confidentiality and integrity for all the computation.

4.1 SGX Formalization

Intel SGX provides a solution to run programs in a secure container, which is referred as an *enclave*, on an untrusted OS. Sensitive data and codes within the

protected memory regions can be isolated by the *enclave*, and can be protected against powerful attackers (e.g., controlling the OS). We use relay R to represent the physical SGX host, which is the interface of the *enclave*. Other components cannot directly access to the *enclave*, unless it relies on R.

Our program Prog_x (shown in Fig. 1) will be executed in the *enclave*, which is trusted by all the buyers and sellers. In this paper, we design the ETA system with the formalization of Intel SGX in [33].

SGX Functions $\mathcal{F}_{SGX}[\mathsf{Prog}_\mathsf{x}, R]$

Initialize() :

1 : upon receiving (Init) from R :

2 : output := Prog_x.**initialize()**

/ generate an output with attestation

3 : $\psi_{sgx} := \sum_{sgx} \cdot \mathsf{Sig}(\mathsf{sk}_{sgx}, (\mathsf{Prog}_\mathsf{x}, \mathsf{output}))$

4 : return $(\mathsf{output}, \psi_{sgx})$

Resume() :

5 : upon receiving Auth(meg) from R :

6 : output := Prog_x.**resume**()

7 : return output

Fig. 2. SGX functions

In order to model the ideal functionality channel with some proprieties such as confidentiality and authenticity, we use a global universal composability (UC) functionality [6] to instantiate the SGX Functions as $\mathcal{F}_{SGX}(\sum_{sgx})[\mathsf{Prog}_\mathsf{x}, R]$ parameterized by a group signature scheme $\sum_{sgx}$. Our program Prog_x is loaded into *enclave* via the "init" call. When it calls "resume", the program is executed based on the given incoming requests (or inputs, denoted as inp), and generates the output with an attestation $\psi_{att} := \sum_{sgx} \cdot \mathsf{Sig}(\mathsf{sk}_{sgx}, (\mathsf{Prog}_\mathsf{x}, \mathsf{output}))$. The signature under TEE hardware key sk_{sgx} and pk_{sgx} could be obtained from the SGX Functions ($\mathcal{F}_{SGX}$). The details are given in Fig. 2.

4.2 ETA System

As shown in Fig. 3, there are four main components in ETA: (1) **Enclave** is used for protecting program codes and data, which guarantees the confidentiality and integrity of computations. Remote attestation allows remote users to establish encrypted and authenticated channels to *enclave*; (2) **Relay** R is the interface of Intel SGX. It can be used as a physical SGX host, and other components cannot directly access to the *enclave* without connecting with the R; (3) **Key Management Committee** generates key pairs $(\mathsf{pk}, \mathsf{sk})$ for buyers/sellers' input

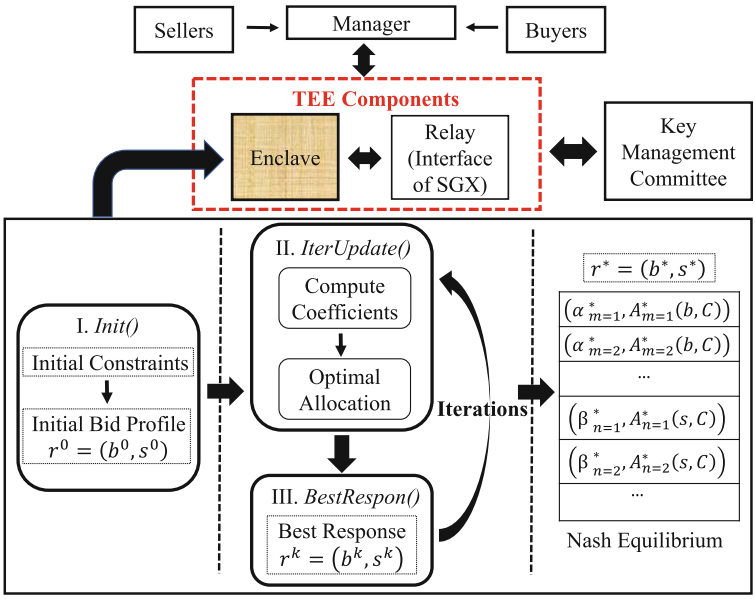


Fig. 3. ETA system (Intel SGX-based)

encryption (encrypted by the public key pk). Private key sk is used inside the TEE components for decryption and preparing for the further computation; (4) **All the Agents** ($\forall m \in \mathcal{B}$, $\forall n \in \mathcal{S}$ and manager $\mathcal{P}$). All the agents participates in the auction while the manager handles all the incoming requests and delivers results as the administrator. Also, TEE components will be triggered by the manager.

Note that the manager is not a trusted third party (TTP). All the all the buyers/sellers' inputs loaded via a secure channel are not visible to the manager. Moreover, it can even deviate arbitrarily from program or collude with other agents. However, manager cannot affect the correct computation/execution of the program and will be penalized for program interrupts or aborts.

4.3 ETA Execution Program

ETA is executed in 3 phases: (1) Setup, (2) Compute, and (3) Delivery, as illustrated in Fig. 4. Assume that secure channels are established between the TEE components and all the buyers and sellers.

(1) Setup. The TEE initializes the system with the "init" call and prepares for loading our auction program $\mathsf{Prog}_{\mathsf{x}}$. Once it receives the "init" request, the key pair $(\mathsf{pk}_{sgx}, \mathsf{sk}_{sgx})$ is created, and then pk_{sgx} which is bound to the TEE code (Program) by the initial attestation will be distributed. Meanwhile, the TEE will obtain the key pairs $(\mathsf{pk}, \mathsf{sk})$ from the Key Management Committee and distribute the public key pk to buyers and sellers for encrypting their inputs.

ETA Execution Program $(\mathcal{B}, \mathcal{S}, R, E)$

$InitEnclave()$:

1 : receive(init(), $request$) to load Prog_x inside E
2 : boost *enclave* with Prog_x.**initialize()**
3 : call Prog_x.**resume()** to handle($\mathsf{Enc}_{\mathsf{pk}}(\mathsf{inp}), l_{id}$)
4 : distribute $(\mathsf{pk}, \psi_{sgx})$ for attestation
/ generate key pairs for inputs encryption
5 : $(\mathsf{pk}, \mathsf{sk}) \leftarrow\!\!\$\ \mathsf{KGen}(1^n)$

$InitAgents()$:

6 : upon receiving init() to obtain pk for encryption
7 : buyers/sellers make deposits ξ_{b_m} or ξ_{s_n}
8 : $\mathcal{P}$ make deposits $\xi_{\mathcal{P}}$

$Compute()$:

9 : boost *enclave* with Prog_x.**resume()**
10 : decrypt $(\mathsf{Enc}_{\mathsf{pk}}(\mathsf{inp}), l_{id})$ with sk
11 : execute and compute Prog_x

$Delivery()$:

12 : fetch(output, ψ_{sgx}, l_{id})
13 : forward output to the honest buyers/sellers by $\mathcal{P}$

Fig. 4. ETA execution program

Next, all the buyers and sellers encrypt their input data (denoted as inp overall) with pk, prepare the deposits ξ_{b_m} or ξ_{s_n}) and send meg := ($\mathsf{Enc}_{\mathsf{pk}}(\mathsf{inp})$, l_{id}, ξ_{b_m}, ξ_{s_n}) to manager $\mathcal{P}$ where inp denotes the inputs of all the buyers/sellers. Note that l_{id} represents a fresh and unique identifier (ID). In practice, it will use 128-bit MAC of the AES-GCM encryptions as ID.

Note that all the buyers and sellers send the messages through secure authenticated channels. The manager is responsible for batching transactions of all the inputs from all the buyers and sellers. Also, validation of messages will be checked by the manager. If it is valid, manager will forward all the requests to the TEE Components. Otherwise, once malicious behaviors are detected, the deposits will be stored into the TEE for next computation instead of refund.

(2) Compute. TEE components handle the incoming requests (inputs) in parallel. TEE components retrieve the program Prog_x with the "resume" call and decrypt the inputs with the private key sk. The execution of the auction program Prog_x is conducted in a sandboxed environment (*enclave*). Then, a software adversary and/or a physical adversary cannot interrupt the execution or inspect (monitor) data that lives inside the *enclave*. The result of the Prog_x is a secret output which will be securely returned to the manager with a black-box program execution.

(3) Delivery. Once the execution has been completed inside the TEE, the output (of all the buyers/sellers in the divisible double auction) is generated and

signature is provided to prove the computation correctness. As a result, the output message (output, $\psi_{sgx}, l_{id}, \xi_{b_m}, \xi_{s_n}$) will be delivered to the corresponding honest buyers/sellers by the manager.

4.4 Threat Model

To ensure confidentiality and integrity for computation, we use the TEE's attestation [44]. In particular, the computation is executed correctly inside the Intel SGX (all the agents trust the *enclave*). However, the remaining software stack outside the *enclave* and the hardware is not trusted. The adversary may corrupt any number of agents, assuming that honest agents will trust their own codes and platform (leakage resulted from its software bugs are out of the scope).

Furthermore, we assume that multiple agents do not trust each other mutually. They can be potentially malicious such as stealing the bid profiles information and modifying the execution flow. During the execution, each buyer or seller may send, drop, modify and record arbitrary messages. Even worse, any buyer or seller may crash and stop responding. Note that the side-channel attacks against *enclave* and DoS attacks are not considered in this paper.

4.5 Security Analysis

Enclave Isolation. Intel SGX enables the program (data) to be executed inside the secure container (*enclave*) for confidentiality and integrity. The adversary cannot interrupt the computation executed in a sandboxed environment (*enclave*). Note that *enclave* is created in its virtual address space by an untrusted hosting application with OS support. Once *enclave* starts initialization, data and codes inside it will be isolated from the rest of the system and secured.

Also, the encrypted data are sent from buyers/sellers to *enclave* through secure channels. However, other malicious servers cannot eavesdrop on the encrypted data, and cannot interrupt the communication.

Data Sealing. Intel SGX reads the encrypted data as inputs and decrypts the data with its private key inside the *enclave*. Once the computation is completed within the *enclave*, the results output will be encrypted again before distribution. Due to the data sealing, the vulnerability of data leakage can be addressed.

Malicious Manager. The manager handles all the incoming requests and delivers the results as the administrator. If the untrusted manager interrupts the delivery rule, delays or tampers with the communication among all the components, he/she will be punished by losing all the initial deposits.

5 Experiments

In this section, we evaluate the performance for ETA in Graphene[1] on the Microsoft Azure.[2] Specifically, considering energy trading as an experimental application, we conduct experiments on real power consumption and generation data (available at the UMASS Trace Repository [2]) which can be used for the application of energy trading in a peer-to-peer mode amongst up to 200 agents, each of which is either a buyer or seller in the divisible double auction.

The auction mechanisms have been designed for many different divisible resources, such as electricity [18,39], cloud resources [13,22], and wireless spectrum [24]. Note that different valuation/cost functions are defined in different applications. We take energy trading application [17,41] as an example. Entities on the power grid may trade their excessive locally generated energy, e.g., the renewable energy resources [1,11]. Since electricity is divisible, ETA can establish a privacy preserving divisible double auction for energy trading. The valuation/-cost functions are defined as $V_m(x_m) = \zeta_m \log(x_m+1)$ and $C_n(y_n) = a_n y_n^2 + b_n y_n$ [3], where ζ_m is a parameter determined by the behavior preference of buyer, a_n and b_n are the parameters for measuring how much the sellers incline to sell. The valuation/cost functions follow the general assumption in Sect. 2. Finally, ETA only outputs the *clearing price* for the auction to all the agents, and each pair of buyer and seller only receive the amount traded between them.

We conduct experiments using the energy trading application [41], and set $\zeta_m = 50$, $a_n = 30$ and $b_n = 0$ in the valuation function $V_m(x_m) = \zeta_m \log(x_m+1)$ and cost function $C_n(y_n)=a_n y_n^2 + b_n y_n$ (adopting the same parameters as [48]).

5.1 ETA Performance Evaluation

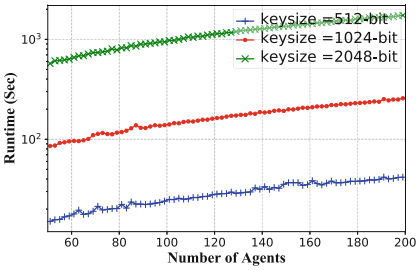
We evaluate the system performance (up to 200 agents) for ETA and demonstrate the results for both auction performance and system efficiency.

Figure 5(a) shows the total runtime on a varying number of agents (from 50 to 200) in an auction. The auction includes both secure computation for the optimal allocation and the peer-to-peer trading in the TEE. It takes up to only 10–15 min for 200 agents even if the 2048-bit key size is adopted for very strong security. Moreover, Fig. 5(b) illustrates the relationship between the number of agents and the throughput (bits/sec). As the number of agents increases, throughput increases linearly as shown in Fig. 5(b).

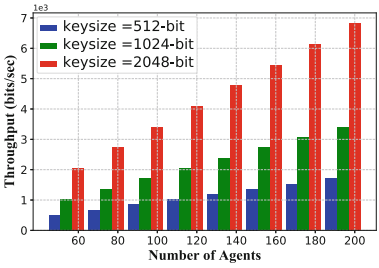
For evaluating the efficiency in multiple auctions, we use the key size as 1024 bit and 20 agents in ETA for continuously executing 720 auctions. Figure 5(c) shows that the runtime for each auction lies close to 60 s, which would not result in much latency if we execute multiple auctions in real time.

[1] Graphene [36] is a lightweight guest OS, designed for minimal host requirements. Applications can be protected in a hardware-encrypted memory region.

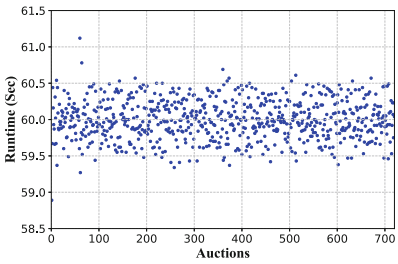
[2] https://azure.microsoft.com/en-us/solutions/confidential-compute/.



(a) Runtime vs. Number of Agents



(b) Throughput vs. Number of Agents



(c) Runtime vs. Auctions (1024-bits)

Fig. 5. ETA performance evaluation

Table 1. Executing ETA for divisible double auction among 12 buyers and 8 sellers; allocation (MWh), potential amount (MWh); social welfare ($)

Criteria	Iterations				
Allocation $(A^*_{m=1}(b,C))$	0.23	0.24	0.24	0.44	0.57
Allocation $(A^*_{m=2}(b,C))$	0.16	0.55	0.57	0.57	0.59
Allocation $(A^*_{m=3}(b,C))$	0.14	0.14	0.24	0.55	0.55
Allocation $(A^*_{n=1}(s,C))$	0.10	0.55	0.74	0.90	0.90
Allocation $(A^*_{n=2}(s,C))$	0.41	0.43	0.64	0.64	0.90
Potential amount (C)	4.38	4.37	2.64	2.64	2.64
Social welfare $(F(\cdot))$	62.9	68.2	70.7	74.0	74.0

5.2 Case Study

Furthermore, we perform a case study by applying ETA for energy trading, and present more detailed results in a divisible double auction. Specifically, 20 agents include 12 buyers and 8 sellers. We will present the fine-grained results in multiple iterations of the auction with respect to the following criteria: (1) allocation, (2) potential amount, and (3) social welfare.

Table 1 shows the detailed results. First, it presents that 5 groups of the allocation for both buyers and sellers ($A^*_m(b,C)$ or $A^*_n(s,C)$), which are generated

from multiple iterations (finally converged to the Nash Equilibrium). Due to space limit, we illustrate the results of 5 representative agents (3 buyers and 2 sellers). The allocation of all the participants will increase, and finally they all converge to the final allocated amounts after several iterations. Second, potential amount (C) decreases until convergence. 2.64 MWh will be the total selling/buying amount in the auction. Third, the social welfare ($F(\cdot)$) is derived based on the equation $F(\cdot) = \sum_{m \in \mathcal{B}} V_m(A_m) - \sum_{n \in \mathcal{S}} C_n(A_n)$. It has an increasing trend in multiple iterations and converges to the maximized social welfare \$74.

6 Related Work

Auction mechanisms for divisible resources were widely studied for spectrum allocation [10,40,42]. Spectrum allocation problem was discussed in cognitive radio networks with the combinatorial auctions [10]. Wu and Vaidya [40] modeled the radio spectrum allocation problem as a sealed-bid reserve auction, and investigated strategy-proof mechanism for multi-radio spectrum buyers. Hoefer et al. [15] proposed an approximation algorithm (LP formulation) for combinatorial auctions with a conflict graph. Other similar studies with respect to divisible auctions focused on the revenue maximization [21] or social efficiency maximization [10]. Yu et al. [45] proposed the auction model to handle the uncertain demand conditions. In order to promote high efficiency for the auctions, Lorenzo et al. [30] designed the matching game mechanism in the auction based on the game theory.

The privacy concerns have been raised in auctions for divisible resources [7,19]. To address them, cryptographic techniques [29,32,34] were proposed to achieve both privacy preservation and incentive compatibility for the auctions. Huang et al. [20] proposed a cryptographic scheme for one-side strategy-proof auctions. Some other existing works [8,35] apply the homomorphic encryption to address privacy issues in the auctions. Furthermore, some deployed systems can also be utilized to address the privacy concerns for auctions. In [14], the involved third-party auction platform was designed to preserve privacy for all the participants. The `HAWK` system [26] was deployed as a decentralized smart contract that privately processes transactions with the private and public portions for the sealed-bid auctions.

However, such techniques are not directly applicable to privacy preserving double auction for divisible resources. Besides, they may require high computational overheads due to heavy cryptographic primitives. Instead, our `ETA` can efficiently perform secure computation for divisible double auction while ensuring truthfulness.

7 Conclusion

We design an efficient and private system `ETA` which securely execute double auction for allocating divisible resources among distributed agents within the

Intel SGX. The auction mechanism in ETA ensures individual rationality, incentive compatibility and Pareto efficiency. The input data of both buyers and sellers in the auction can also be protected in ETA. The experimental results have demonstrated a high efficiency for ETA to privately execute the divisible double auction. In the future, blockchain-based auction mechanisms will be investigated by executing the secure computation inside the Intel SGX for trading divisible resources among distributed agents with the cryptocurrencies (e.g., bitcoins).

Acknowledgments. This work is partially supported by the National Science Foundation (NSF) under Grant No. CNS-1745894. The authors would like to thank the anonymous reviewers for their constructive comments.

Appendix

Proof of Theorem 1

Proof. (1) Per the payment rule of ETA (extended from the VCG auction) in Eq. 1, buyer $m \in \mathcal{B}$ will change its own strategies based on other buyers' strategies (so does seller $n \in \mathcal{S}$). As defined in Sect. 2.1, ρ_m is the payment made by buyer m while ρ_n is the payment received by seller n. Also, $\rho(r_i, r_{-i})$ is defined as the difference between all the buyers' aggregated valuation if any other buyer i is not in the auction and the aggregated valuation if buyer i is in the auction. Then, $\rho(r_i, r_{-i})$ can be transformed into the difference between two payoff functions:

$$\begin{aligned}\rho(r_i, r_{-i}) &= \sum_{m \neq i} \alpha_m [A_m(0; r_{-i}) - A_m(r_i; r_{-i})] \\ &= \underbrace{(\max \sum_{m \neq i} f_m(r))}_{\text{without m}} - \underbrace{\sum_{m \neq i} f_m(r^*)}_{\text{with m}} \end{aligned} \tag{8}$$

In addition, as defined in Sect. 2.1, the payoff function for buyer m is $\widehat{V}_m(A_m^*(r)) - \rho(r_i, r_{-i})$. The payoff function is supposed to be maximized if there exists the optimal bid profile r^*, including the optimal allocation profiles for buyers and sellers: $A_m^*(r)$ and $A_n^*(r)$. After integrating Eq. 8, we have the payoff function w.r.t. the buyer m as below:

$$\begin{aligned}&\widehat{V}_m(A_m^*(r)) - \rho(r_i, r_{-i}) \\ &= \left[\widehat{V}_m(A_m^*(r^*)) + \sum_{m \neq i} f_m(r^*)\right] - \left[(\max \sum_{m \neq i} f_m(r))\right] \end{aligned} \tag{9}$$

In Eq. 9, the $\left[(\max \sum_{m \neq i} f_m(r))\right]$ is the same for all the buyers ($\forall m \in \mathcal{B}$). Then, the problem of maximizing buyer m's payoff is reduced to the problem of

maximizing $\left[\widehat{V}_m(A_m^*(r^*)) + \sum_{m \neq i} f_m(r^*)\right]$. Intuitively, buyer m would choose the strategy to maximize $\left[\widehat{V}_m(A_m^*(r^*)) + \sum_{m \neq i} f_m(r^*)\right]$. Per Definition 4 and incentive compatibility proven in (4), if each agent responds untruthfully, it would not obtain a higher payoff than truthful response. If buyer m bids truthfully and the objective (to maximize) for double auction mechanism becomes identical to the $\left[\widehat{V}_m(A_m^*(r^*)) + \sum_{m \neq i} f_m(r^*)\right]$. The payoff will be maximized if buyer m bids truthfully. Therefore, the truthful responses in the double auction mechanism are the best strategies for all the buyer $\forall m \in \mathcal{B}$.

Similarly, for any seller $n \in \mathcal{S}$, its payoff function $f_n(r) = \rho(r_j, r_{-j}) - \widehat{C}_n(A_n^*)$ as defined in Sect. 2.1 can also be proven in the same way. Thus, the divisible double auction mechanism in ETA ensures weakly dominant strategy.

(2) If buyer $m \in \mathcal{B}$ provides truthful bid profile, then it has a non-negative payoff function $f_m(r) = \widehat{V}_m(A_m^*) - \rho(r_i, r_{-i}), \forall m \in \mathcal{B}$. Similarly, seller $n \in \mathcal{S}$ can also obtain a non-negative payoff function: $f_n(r) = \rho(r_j, r_{-j}) - \widehat{C}_n(A_n^*), \forall n \in \mathcal{S}$ with truthful bid profile. Thus, the double auction satisfies individual rationality, which indicates that all the agents have non-negative payoffs by participating in the double auction of ETA.

(3) Allocation (A_m, ρ_m) satisfies Pareto efficiency within the budget φ_m ($\rho_m < \varphi_m$) in the divisible double auction ETA if there does not exist a better allocation $(A_m^{'}, \rho_m^{'})$: $f_m(A_m, \rho_m) > f_m(A_m^{'}, p_m^{'})$. Suppose that buyer $m \in \mathcal{B}$ is allocated with amount A_m in bid profile r (satisfying individual rationality as above and incentive compatibility as proven in (4)). We now prove the Pareto efficiency (optimality). Given $f_m^* = \max_{A_m} f_m(A_m, \rho(A_m, (r_{-m}))$, buyer m's payoff is upper bounded by f_m^*. If m would like to gain more payoff, then it needs to pay $\rho(A_m, (r_m, r_{-m}))$. Thus, the payoff is supposed to be lowered bounded by f_m^*. Thus, buyer m's payoff is exactly f_m^* for the optimality. Similarly, $f_n^* = \max_{A_n} f_n(A_n, \rho(A_n, (r_{-n}))$ can be proven for sellers. Therefore, the Pareto efficiency is verified in ETA.

(4) Denote the allocation of buyer $m \in \mathcal{B}$ as the A_m, and also denote the allocation in the k-th iteration as A_m^k. To show the incentive compatibility for any buyer $m \in \mathcal{B}$, we verify that for any bid profile $b = (b_m, m \in \mathcal{B})$. Given r_{-m}, there exists a truthful bid profile $b_m = (\alpha_m, d_m^k)$ where $\alpha_m = \widehat{V}_m^{'}(d_m^k)$, such that $f_m(b_m^k, r_{-m}) \geq f_m(b_m, r_{-m}), \forall m \in \mathcal{B}$:

- Case 1: if $\alpha_m < \widehat{V}_m^{'}(d_m)$. Consider a bid b_m^k, such that $d_m^k = A_m \leq d_m$. Based on the diminishing marginal utility of the valuation function for buyers, we have $\alpha_m^k \geq \widehat{V}_m^{'}(d_m) > \alpha_m$. Since we get the maximum social welfare, we have $A_m^k \geq A_m$. Thus, we have $f_m(b_m^k, r_{-m}) \geq f_m(b_m, r_{-m}), \forall m \in \mathcal{B}$.
- Case 2: if $\alpha_m > \widehat{V}_m^{'}(d_m)$. Considering bid b_m^k, such that $d_m^k = d_m$, we have $\alpha_m > \widehat{V}_m^{'}(d_m) = \widehat{V}_m^{'}(d_m^k) = \alpha_m^k$. Also, $A_m^k \leq A_m$ holds for the maximum social welfare. When $A_m^k = A_m$, we have $f_m(b_m^k, r_{-m}) = f_m(b_m, r_{-m}), \forall m \in \mathcal{B}$. When $A_m^k < A_m$, we have:

$$\begin{aligned} & f_m(b_m, r_{-m}) - f_m(b_m^k, r_{-m}) \\ = \; & \widehat{V}_m(A_m) - \widehat{V}_m(A_m^k) + \rho(A_m^k, r_{-m}) - \rho(A_m, r_{-m}) \\ \leq \; & \alpha_m^k(A_m - A_m^k) + F(r) - \alpha_m A_m - F(r^k) + \alpha_m^k A_m^k \\ \leq \; & \alpha_m^k(A_m - A_m^k) - \alpha_m^k(A_m - A_m^k) = 0 \end{aligned} \tag{10}$$

Given Case 1 and 2, we have $f_m(b_m^k, r_{-m}) \geq f_m(b_m, r_{-m}), \forall m \in \mathcal{B}$. Similarly, incentive compatibility can also be proven for all the sellers $\forall n \in \mathcal{S}$.

(5) Assuming that $\sum_{m\in\mathcal{B}} d_m \geq \sum_{n\in\mathcal{S}} h_n$ holds for the initialization, then the potential amount for all divisible resources $C = \min\{\sum_{m\in\mathcal{B}} d_m, \sum_{n\in\mathcal{S}} h_n\}$ holds for the iterative computation in the ETA. Thus, we have $\sum_{m\in\mathcal{B}} d_m = \sum_{n\in\mathcal{S}} h_n$. Furthermore, compared with the other case: $\sum d_m \leq h_n$, the divisible double auction mechanism in ETA satisfies the feasibility. In summary, these complete the proof. □

References

1. Aliabadi, D.E., Kaya, M., Şahin, G.: An agent-based simulation of power generation company behavior in electricity markets under different market-clearing mechanisms. Energy Policy **100**, 191–205 (2017)
2. Barker, S., Mishra, A., Irwin, D., Shenoy, P., Albrecht, J.: SmartCap: flattening peak electricity demand in smart homes. In: IEEE PerCom, pp. 67–75 (2012)
3. Bompard, E., Ma, Y., Napoli, R., Abrate, G.: The demand elasticity impacts on the strategic bidding behavior of the electricity producers. IEEE Trans. Power Syst. **22**(1), 188–197 (2007)
4. Brandt, F., Sandholm, T., Shoham, Y.: Spiteful bidding in sealed-bid auctions. In: Veloso, M.M. (ed.) IJCAI, pp. 1207–1214 (2007)
5. Brero, G., Lahaie, S., Seuken, S.: Fast iterative combinatorial auctions via bayesian learning. In: AAAI, pp. 1820–1828 (2019)
6. Canetti, R.: Universally composable security: a new paradigm for cryptographic protocols. In: Annual Symposium on Foundations of Computer Science, FOCS 2001, pp. 136–145. IEEE Computer Society (2001)
7. Chen, Z., Huang, L., Li, L., Yang, W., Miao, H., Tian, M., Wang, F.: PS-TRUST: provably secure solution for truthful double spectrum auctions. In: 2014 IEEE Conference on Computer Communications, INFOCOM, pp. 1249–1257 (2014)
8. Chen, Z., Chen, L., Huang, L., Zhong, H.: On privacy-preserving cloud auction. In: 35th IEEE SRDS, pp. 279–288 (2016)
9. Dobzinski, S., Lavi, R., Nisan, N.: Multi-unit auctions with budget limits. Games Econ. Behav. **74**(2), 486–503 (2012)
10. Dong, M., Sun, G., Wang, X., Zhang, Q.: Combinatorial auction with time-frequency flexibility in cognitive radio networks. In: IEEE INFOCOM (2012)
11. Faqiry, M.N., Das, S.: Double-sided energy auction in microgrid: equilibrium under price anticipation. IEEE Access **4**, 3794–3805 (2016)
12. Feng, Z., Qiu, C., Feng, Z., Wei, Z., Li, W., Zhang, P.: An effective approach to 5g: wireless network virtualization. IEEE Commun. Mag. **53**(12), 53–59 (2015)
13. Fujiwara, I., Aida, K., Ono, I.: Applying double-sided combinational auctions to resource allocation in cloud computing. In: SAINT, pp. 7–14 (2010)

14. Gao, W., Yu, W., Liang, F., Hatcher, W.G., Lu, C.: Privacy-preserving auction for big data trading using homomorphic encryption. IEEE Trans. Netw. Sci. Eng. (2020)
15. Hoefer, M., Kesselheim, T., Vöcking, B.: Approximation algorithms for secondary spectrum auctions. ACM Trans. Internet Techn. **14**(2–3), 16:1–16:24 (2014)
16. Hoekstra, M., Lal, R., Pappachan, P., Phegade, V., del Cuvillo, J.: Using innovative instructions to create trustworthy software solutions. In: HASP@ISCA, p. 11 (2013)
17. Hong, Y., Wang, H., Xie, S., Liu, B.: Privacy preserving and collusion resistant energy sharing. In: 2018 IEEE ICASSP, pp. 6941–6945 (2018)
18. Hong, Y., Goel, S., Liu, W.: An efficient and privacy-preserving scheme for P2P energy exchange among smart microgrids. Int. J. Energy Res. **40**(3), 313–331 (2016)
19. Huang, H., Li, X., Sun, Y., Xu, H., Huang, L.: PPS: privacy-preserving strategyproof social-efficient spectrum auction mechanisms. IEEE Trans. Parallel Distrib. Syst. **26**(5), 1393–1404 (2015)
20. Huang, Q., Tao, Y., Wu, F.: SPRING: a strategy-proof and privacy preserving spectrum auction mechanism. In: Proceedings of the IEEE INFOCOM, pp. 827–835 (2013)
21. Jia, J., Zhang, Q., Zhang, Q., Liu, M.: Revenue generation for truthful spectrum auction in dynamic spectrum access. In: ACM MobiHoc, pp. 3–12 (2009)
22. Jin, A., Song, W., Zhuang, W.: Auction-based resource allocation for sharing cloudlets in mobile cloud computing. IEEE Trans. Emerg. Topics Comput. **6**, 45–57 (2018)
23. Johari, R., Tsitsiklis, J.N.: Efficiency loss in a network resource allocation game. Math. Oper. Res. **29**(3), 407–435 (2004)
24. Kebriaei, H., Maham, B., Niyato, D.: Double-sided bandwidth-auction game for cognitive device-to-device communication in cellular networks. IEEE Trans. Vehicular Technol. **65**(9), 7476–7487 (2016)
25. Kojima, F., Yamashita, T.: Double auction with interdependent values: incentives and efficiency. Theor. Econ. **12**(3), 1393–1438 (2017)
26. Kosba, A.E., Miller, A., Shi, E., Wen, Z., Papamanthou, C.: Hawk: the blockchain model of cryptography and privacy-preserving smart contracts. In: IEEE Symposium on Security and Privacy, pp. 839–858 (2016)
27. Krishna, V.: Auction Theory. Academic Press, Boston (2009)
28. Lazar, A.A., Semret, N.: Design and analysis of the progressive second price auction for network bandwidth sharing. Telecommun. Syst. **13** (2001)
29. Liu, B., Xie, S., Hong, Y.: PANDA: privacy-aware double auction for divisible resources without a mediator. In: AAMAS, pp. 1904–1906 (2020)
30. Lorenzo, B., González-Castaño, F.J.: A matching game for data trading in operator-supervised user-provided networks. In: IEEE ICC, pp. 1–7 (2016)
31. Maheswaran, R.T., Başar, T.: Nash equilibrium and decentralized negotiation in auctioning divisible resources. Group Decis. Negot. **12**(5), 361–395 (2003)
32. Peng, K., Boyd, C., Dawson, E., Viswanathan, K.: Robust, privacy protecting and publicly verifiable sealed-bid auction. In: International Conference, ICICS, pp. 147–159 (2002)
33. Shi, E., Zhang, F., Pass, R., Devadas, S., Song, D., Liu, C.: Trusted hardware: life, the composable universe, and everything. Manuscript (2015)
34. Suzuki, K., Yokoo, M.: Secure generalized Vickrey auction using homomorphic encryption. In: Wright, R.N. (ed.) FC 2003. LNCS, vol. 2742, pp. 239–249. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-45126-6_17

35. Suzuki, K., Yokoo, M.: Secure combinatorial auctions by dynamic programming with polynomial secret sharing. In: Blaze, M. (ed.) FC 2002. LNCS, vol. 2357, pp. 44–56. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-36504-4_4
36. Tsai, C., et al.: Cooperation and security isolation of library OSES for multi-process applications. In: EuroSys, pp. 9:1–9:14. ACM (2014)
37. Tsai, C., Porter, D.E., Vij, M.: Graphene-SGX: a practical library OS for unmodified applications on SGX. In: Silva, D.D., Ford, B. (eds.) USENIX, pp. 645–658 (2017)
38. Tuffin, B.: Revisited progressive second price auction for charging telecommunication networks. Telecommun. Syst. **20**(3–4), 255–263 (2002)
39. Wang, Y., Saad, W., Han, Z., Poor, H.V., Basar, T.: A game-theoretic approach to energy trading in the smart grid. IEEE Trans. Smart Grid **5**(3), 1439–1450 (2014)
40. Wu, F., Vaidya, N.H.: SMALL: a strategy-proof mechanism for radio spectrum allocation. In: IEEE INFOCOM, pp. 81–85 (2011)
41. Xie, S., Wang, H., Hong, Y., Thai, M.: Privacy preserving distributed energy trading. In: IEEE ICDCS (2020)
42. Xu, P., Xu, X., Tang, S., Li, X.: Truthful online spectrum allocation and auction in multi-channel wireless networks. In: IEEE INFOCOM, pp. 26–30 (2011)
43. Yokoo, M., Sakurai, Y., Matsubara, S.: The effect of false-name bids in combinatorial auctions: new fraud in internet auctions. Games Econ. Behav. **46**, 174–188 (2004)
44. Yuan, R., Xia, Y., Chen, H., Zang, B., Xie, J.: Shadoweth: private smart contract on public blockchain. J. Comput. Sci. Technol. **33**(3), 542–556 (2018)
45. Yu, J., Cheung, M.H., Huang, J., Poor, H.V.: Mobile data trading: a behavioral economics perspective. In: IEEE WiOpt, pp. 363–370 (2015)
46. Zhang, D., Chang, Z., Yu, F.R., Chen, X., Hämäläinen, T.: A double auction mechanism for virtual resource allocation in SDN-based cellular network. In: IEEE International Symposium on PIMRC, pp. 1–6 (2016)
47. Zhang, F., Cecchetti, E., Croman, K., Juels, A., Shi, E.: Town crier: an authenticated data feed for smart contracts. In: ACM Conference on CCS, pp. 270–282 (2016)
48. Zou, S., Ma, Z., Liu, X.: Resource allocation game under double-sided auction mechanism: efficiency and convergence. IEEE Trans. Automat. Contr. **63**, 1273–1287 (2018)