This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2818147, IEEE Transactions on Services Computing



# A Distributed Truthful Auction Mechanism for Task Allocation in Mobile Cloud Computing

Xiumin Wang, Yang Sui, Jianping Wang, Chau Yuen and Weiwei Wu

***Abstract*—In mobile cloud computing, offloading resource-demanded applications from mobile devices to remote cloud servers can alleviate the resource scarcity of mobile devices, whereas long distance communication may incur high communication latency and energy consumption. As an alternative, fortunately, recent studies show that exploiting the unused resources of the nearby mobile devices for task execution can reduce the energy consumption and communication latency. Nevertheless, it is non-trivial to encourage mobile devices to share their resources or execute tasks for others. To address this issue, we construct an auction model to facilitate the resource trading between the owner of the tasks and the mobile devices participating in task execution. Specifically, the owners of the tasks act as bidders by submitting bids to compete for the resources available at mobile devices. We design a distributed auction mechanism to fairly allocate the tasks, and determine the trading prices of the resources. Moreover, an efficient payment evaluation process is proposed to prevent against the possible dishonest activity of the seller on the payment decision, through the collaboration of the buyers. We prove that the proposed auction mechanism can achieve certain desirable properties, such as computational efficiency, individual rationality, truthfulness guarantee of the bidders, and budget balance. Simulation results validate the performance of the proposed auction mechanism.**

***Index Terms*—Mobile cloud computing, incentive mechanism, auction, truthfulness, budget balance.**

## I. Introduction

While mobile devices have become increasingly ubiquitous in our daily life, they are seriously constrained by limited battery capacities [1]–[3] and computation capabilities [4]. To alleviate resource scarcity of mobile devices, one effective way is to offload their complex or resource-demanded tasks to remote cloud [5]–[8] through *pay as you go*. However, such an approach may suffer large internet delay and high energy consumption, due to long distance communication [9]–[11]. To address this issue, recent work proposes that utilizing the unused resources of the mobile devices in the proximity can achieve better system performance. For example, communications among the nearby mobile devices through WLAN/WiFi can significantly reduce the communication latency and network congestion [12]–[18]. Nevertheless, it is non-trivial to encourage mobile devices to share their mobile resources or execute tasks for others, as these actions may incur non-negligible inconvenience to themselves, e.g., performance degradation and battery outage.

In the literature, only a few works [17]–[21] have been published on designing incentive schemes to incentivize mobile devices to provide their unused resources or execute tasks offloaded from others. Specifically, [17] proposes a reputation-based economic incentive model to stimulate the mobile devices to provide services with others. Based on the rating points recorded for the devices, the winning bids can then be selected. The work in [18] imposes a bill backlog on each mobile device. When a device's bill backlog is larger than a threshold, it will be unable to get extra services from others. Although the fairness of mobile devices can be achieved, the mobile devices are always forced to provide services to others, which might not be appealing to many mobile devices. [19] constructs a Stackelberg game model to capture the interaction between the owner of the tasks (i.e., buyer) and the mobile devices that participate in task execution (i.e., sellers). By appropriately allocating the tasks and determining the payments of task executions, the Stackelberg equilibrium of the game can be achieved. Although this approach can benefit both the buyer and the sellers of mobile resources, it does not capture the preference of task execution. It is worth noting that tasks executed on different mobile devices may achieve different system performances, e.g., different completion time and communication latency. Under such a circumstance, different mobile devices should get different payments for their task executions. Moreover, the model in [19] considers only a single buyer of mobile resources, which omits the competition among multiple buyers in a general model. [21] proposes two truthful incentive mechanisms with auctions, which however requires a centralized auctioneer to make the auction decisions. It is particularly noted that the centralized auctioneer must hold the global knowledge of the mobile cloud system, which firstly is prone to expose the privacy of mobile users, and secondly incurs high update cost because of the dynamic nature of the smartphones.

Auction [22]–[26], a major trading scheme in economics, has been widely used in many areas. By viewing the resource trading system as an ecosystem, the auction mechanism aims to appropriately address the conflicts between the buyer's and the seller's interests, and the internal competitions among themselves. Generally, the designed auction mechanism should be able to fairly allocate the trading resources, determine the price/payment of the buyer/seller, and guarantee the truthfulness of the information submitted by the bidders. In mobile

Xiumin Wang is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. Email: xmwang@scut.edu.cn.

Yang Sui is with the School of Computer and Information, Hefei University of Technology, Hefei, China. Email: yangsui@mail.hfut.edu.cn.

Jianping Wang is with the Department of Computer Science, City University of Hong Kong, Hong Kong. Email: jianwang@cityu.edu.hk.

Chau Yuen is with Singapore University of Technology and Design, Singapore. Email: yuenchau@sutd.edu.sg.

Weiwei Wu is with School of Computer Science and Engineering, Southeast University, Nanjing, China. Email: weiweiwu@seu.edu.cn.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2818147, IEEE Transactions on Services Computing



cloud computing, task execution among mobile devices also involves a resource trading between the owners of the tasks and the mobile devices participating in task execution. To capture the competition and conflict of task owners and mobile devices, in this paper, we aim to design an auction mechanism to solve the task assignment problem in mobile cloud computing.

A lot of research efforts have been made on developing the auction mechanisms in the literature [23]–[29]. For example, Vickrey-Clarke-Groves (VCG) auction [24], [25] is a well-known auction scheme to achieve the truthfulness of the information submitted by the bidders. However, its truthfulness is based on the optimal allocation of the resources, which might be impossible since there is no computationally efficient algorithms to solve NP-hard problems. The work in [27] proposes an auction mechanism for crowdsourcing. However, it assumes that each mobile device holds a set of predefined services, and if one of its bids wins, all its predefined services will be traded, even if some of them might be useless to the tasks. Moreover, it does not consider the resource requirement of running each task. [28], [29] study resource sharing for cloudlet in mobile cloud computing, and design efficient auction mechanisms to guarantee the truthfulness of the bidders. The proposed schemes conduct well in homogeneous systems, where the amount of resources required by each buyer is exactly the same as the amount of resources available at each seller. Moreover, [28], [29] restrict the one-to-one resource trading mode, which omits the fact that the resource-rich seller (i.e., mobile device) can support multiple buyers of resources in a practical mobile cloud system.

Compared with conventional schemes, the auction problem designed in this paper is rather challenging, and has the following differences: (1) the tasks to be considered require different amounts of resources, and have heterogeneous preferences on their executors; (2) each resource-rich mobile device can work on more than one task as long as it has enough resources, which is the major difference from the existing one-to-one resource trading schemes; (3) the auction decision is locally made at each mobile device, which does not require a centralized third-party auctioneer as in [28], [29]. Specifically, we aim to design a distributed auction mechanism, which efficiently captures the interaction between the owners of the tasks and the mobile devices that can participate in task executions. To ease presentation, we will use buyers and sellers interchangeably with the owners of the tasks and the mobile devices respectively for the rest of the paper. The main contributions of this paper can be summarized as follows.

- We construct a distributed auction model, where the buyers submit their bids to compete for the resources of the sellers, while each seller, acted as auctioneer, locally makes their auction decisions.
- We propose an auction mechanism to allocate the tasks to the appropriate mobile devices, determine the price paid for each task and the payment earned at each mobile device. An efficient payment evaluation process is also proposed to detect the possible dishonest activity of the sellers on their decided payments.
- We prove that the proposed auction mechanism holds certain desirable properties, such as computational efficiency, individual rationality, truthfulness, and budget balance.

TABLE I
MAIN NOTATIONS AND THEIR DESCRIPTIONS

| | |
|---|---|
| $B$ | The set of mobile tasks to be considered, i.e., buyers |
| $b_i$ | The $i$-th mobile task (i.e., buyer) in $B$ |
| $B_j^c$ | The set of the winning buyer candidates of seller $s_j$ |
| $B_j^w$ | The set of final buyers that win bids at seller $s_j$ |
| $C_i$ | The bid vector submitted by buyer $b_i$ (or the owner of task) |
| $c_{i,j}$ | The bid of buyer $b_i$ to seller $s_j$ for unit resource |
| $m$ | The total number of tasks in $B$, i.e., $m = \|B\|$ |
| $n$ | The total number of mobile devices that can participate in task execution, i.e., $n = \|S\|$ |
| $p_i^b$ | The final price per unit resource, paid by $b_i$ |
| $p_j^s$ | The payment gained by providing unit resource at $s_j$ |
| $\widetilde{p}_j$ | The minimum price per unit resource asked by $s_j$ |
| $r_i$ | The amount of resources required to run task $b_i$ |
| $R_j$ | The total amount of resources available at device $s_j$ |
| $S$ | The set of mobile devices (i.e., sellers) that are willing to participate in task execution |
| $s_j$ | The $j$-th mobile device in $S$ |
| $T_j$ | The least staying time of mobile device $s_j$ |
| $v_{i,j}$ | The true value per unit resource that $b_i$ can get if it wins bid from $s_j$ |
| $V_i$ | $V_i = (v_{i,1}, v_{i,2}, \cdots, v_{i,n})$, the truth valuation vector of $b_i$ |
| $x_{i,j}$ | The variant to denote whether $b_i$ finally wins its bid at $s_j$ |
| $x'_{i,j}$ | The variant to denote whether $b_i$ is selected as one of the winning buyer candidates of $s_j$ |
| $U_i^b$ | The utility achieved by buyer $b_i$ |
| $U_j^s$ | The utility achieved by seller $s_j$ |

The rest of this paper is organized as follows. We describe the system model and formulate the problem in Section II. In Section III, we present and analyze the detailed algorithm to solve the problem. Simulation results are given in Section IV. Finally, we conclude the paper in Section V.

## II. Problem Formulation

In this section, we first introduce the auction model. Then, we formulate the problem and present its desirable properties. For easy understanding, the main notations used in this paper are listed in Table II.

### A. Auction Model

Consider a set of mobile applications/tasks, denoted as $B = \{b_1, b_2, \cdots, b_m\}$, to be executed in a mobile cloud system. Generally, running each mobile application requires a certain amount of resources (CPU, memory, etc). We use $r_i$ to denote the amount of resources required to run task $b_i$, where $b_i \in B$. Let $S = \{s_1, s_2, \cdots, s_n\}$ be the set of resource-rich mobile devices in a mobile cloud system, which are willing to participate in task execution for others if getting satisfied payments. Specifically, we use $R_j$ to denote the total amount of resources available at $s_j$. Suppose that $\widetilde{p}$ is the minimum price that must be paid if mobile device consumes a unit resource for task execution. Such information can be estimated as the minimum consumption cost of mobile device, and provided by the system to avoid unfair competition, which is shared by all the mobile devices and the owners of the tasks. In other words, the overall amount of resources required by

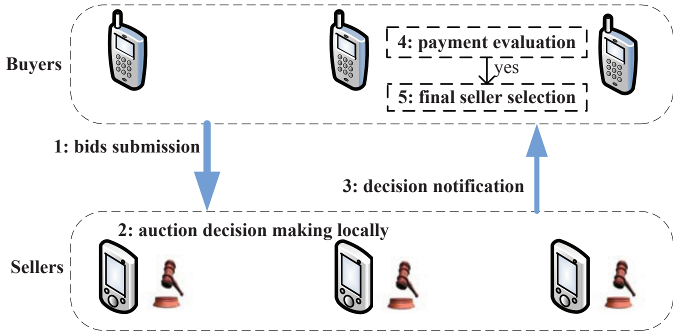


Fig. 1. A distributed auction model

the tasks executed at $s_j$ cannot exceed $R_j$, and the payment per unit resource must be no less than $\widetilde{p}$. Moreover, due to the mobility nature, a mobile device may leave the system or move to other places dynamically, which will significantly affect the quality of services, such as task completion time. As the staying time can usually be estimated by the owners of the mobile devices, we use $T_j$ to denote the least staying time before device $s_j$ will leave the system, which is assumed to be local information to device $s_j$.

To capture the resource trading between the owners of the tasks and the mobile devices, we construct a distributed auction model, where the owners of tasks are the *buyers* of resources, and the mobile devices in $S$ are the *sellers* of resources. Each buyer $b_i$ submits its bid privately to their interested sellers, so that nobody has any knowledge of others. Particularly, for each buyer, the bids to different sellers might not be the same, i.e., the buyers may have different preferences over the sellers. This is because, the tasks executed on different mobile devices may achieve different system performances, e.g., different execution time, communication latency [13]. As such, for each buyer $b_i \in B$, we use $C_i = (c_{i,1}, c_{i,2}, \cdots, c_{i,n})$ to denote the bid vector submitted to the sellers, where $c_{i,j}$ represents the price per unit resource that $b_i$ is willing to pay to seller $s_j \in S$. Moreover, each task gives the information of execution time it expects, such that the selected executor can complete it before leaving the system. Let $t_i$ be the minimum time that task $b_i$ expects its executor to stay, which is sent along the bid. It is also possible that a mobile device may have more than one task for execution. In this context, multiple virtual "buyers" can be added in the above distributed auction model, where the number of virtual buyers equals the number of tasks of the device. Each virtual buyer then represents a task owning by that device, and makes trade with the sellers.

In our auction model, the mobile devices, which are the sellers owning the mobile resources, also act as the *auctioneers*. Fig. 1 illustrates the details of the auction process, which includes the following five procedures:

- Each buyer first submits its bid to its preference sellers, which specifies the amount of resources required, the maximum price willing to pay per unit resource, and the staying time requirement of its executor.
- After collecting the bids from the buyers, each auctioneer (i.e., seller) locally determines the winning bid candidates and the charge required to pay for each buyer.
- The auctioneer notifies the auction decision including the winning bid candidates and the payment required, to the buyers.
- Due to the distributed nature, the seller may cheat on the payment decided by itself, so as to gain higher utility. To address this issue, after getting the auction decision, the buyers need to collaboratively evaluate whether the auctioneer cheats on the decided payment. If detecting cheating activity, the current auctioneer will be removed from the set of the possible trading sellers at all the buyers.
- As the buyer may win its bids at multiple sellers, it chooses the final seller that it is willing to make trade with, according to the defined criteria.

Note that the above distributed auction process can be conducted round by round, until achieving satisfied system efficiency.

### *B. Problem Formulation*

Without loss of generality, we define the following variants:

- $p_i^b$: The price per unit resource charged by buyer $b_i$;
- $p_j^s$: The payment per unit resource that seller $s_j$ can get;
- $x_{i,j}$: Let it be $1$ if buyer $b_i$ wins its bid at seller $s_j$, otherwise, let it be $0$.

The auction mechanism must follow the following rules:

- The buyer $b_i$ is willing to win its bid at $s_j$, only if the price charged by it is no more than its bid $c_{i,j}$, i.e.,

$$p_i^b \le c_{i,j}, \forall b_i \in B. \tag{1}$$

- The seller $s_j$ can accept the bid from $b_i$, only if the payment it obtains is no less than the minimum price $\widetilde{p}$. That is,

$$p_j^s \ge \widetilde{p}, \forall s_j \in S. \tag{2}$$

- The overall amount of resources required by the buyers that win the bids at $s_j$, must be no more than the maximum amount of resources available at $s_j$, i.e.,

$$\sum_{i=1}^{m} x_{i,j} r_i \le R_j, \forall s_j \in S. \tag{3}$$

- Each buyer can only win its bid from at most one seller, as executing task on multiple mobile devices will incur extra migration cost, e.g., traffic incurred among mobile devices. That is,

$$\sum_{j=1}^{n} x_{i,j} \le 1, \forall b_i \in B. \tag{4}$$

Let $V_i = (v_{i,1}, v_{i,2}, \cdots, v_{i,n})$ be the true valuation vector of buyer $b_i$, where $v_{i,j}$ denotes the true value per unit resource that $b_i$ can obtain if it wins bid from $s_j$. It is noted that such true information is only known to the buyer itself, and because of selfishness, it may submit untruthful information to gain higher utility, e.g., $c_{i,j} \ne v_{i,j}$. Then, following the above rules,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2818147, IEEE Transactions on Services Computing



we can define the utilities of the buyer $b_i$ and seller $s_j$, denoted as $U_i^b$ and $U_j^s$ respectively, as follows:

$$U_i^b = \left( \sum_{j=1}^{n} x_{i,j} v_{i,j} - p_i^b \right) r_i, \quad (5)$$

$$U_j^s = \sum_{i=1}^{m} x_{i,j} r_i p_j^s. \quad (6)$$

As seen, if $U_i^b \geq 0$, it means that buyer $b_i$ is allocated to the seller with a valuation no less than the charged price. On the other hand, $U_j^s$ denotes the revenue earned at $s_j$, by allocating the resources to the buyers.

### C. Auction Objectives

As described above, each seller (also acted as an auctioneer) makes a local auction decision to determine the set of the wining bid candidates, and the corresponding payments, followed by the final seller selection at each selected buyer. An efficient auction mechanism should satisfy the following desirable properties:

- *Computational Efficiency*: The designed auction mechanism should be able to get results with a polynomial time complexity.
- *Individual Rationality*: The auction mechanism must guarantee that each winning buyer pays no more than its bid, and each seller gets no less than the minimum price that it asks for.
- *Truthfulness*: The auction mechanism must be able to guarantee that no bidder can improve its own utility by submitting a bid different from its true valuation. In other words, for $\forall b_i \in B$, $U_i^b$ can be maximized only if $C_i = V_i$. Furthermore, the proposed mechanism should also guarantee the truthfulness of the sellers on their payment decisions.
- *Budget Balance*: There must be no economic loss for each auction round. That is, the expense that each auctioneer charge all its winning buyers can afford the total payment that the seller get, i.e.,

$$\sum_{i=1}^{m} x_{i,j} r_i p_i^b \geq \sum_{i=1}^{m} x_{i,j} r_i p_j^s, \forall j. \quad (7)$$

Targeting at the above properties, we design a distributed and efficient auction mechanism which is computational tractable, individual rational, truthful, and budget balanced.

## III. Auction Mechanism Design

In this section, we first introduce the detailed process of each auction round, which includes making auction decisions at each auctioneer, evaluating the payment decision with the collaboration of buyers, and determining the final seller at each buyer. Then, we use an example to illustrate how the proposed mechanism works. Finally, we theoretically analyze the properties of the mechanism.

**Algorithm 1:** Auction decision making process at $s_j$

```
begin
    x'_{i,j} = 0, p^s_j = 0, p^b_{i,j} = 0 for ∀i, j;
    𝔹_j = {b_i | c_{i,j} ≥ p̃, r_i ≤ R_j, t_i ≤ T_j, ∀b_i ∈ B};
    sort the buyers in 𝔹_j into 𝔹_j = {b_{l_1}, ··· , b_{l_{|𝔹_j|}}} such
    that c_{l_1,j} ≥ c_{l_2,j} ≥ ··· ≥ c_{l_{|𝔹_j|},j};
    if Σ_{b_i∈𝔹_j} r_i ≤ R_j then
        set x'_{i,j} = 1 for ∀b_i ∈ 𝔹_j;
        set the price p^b_{i,j} = p̃ for ∀b_i ∈ 𝔹_j, and p^s_j = p̃;
    end
    if Σ_{b_i∈𝔹_j} r_i > R_j then
        k = arg max_k { Σ_{k'=1}^{k} r_{l_{k'}} ≤ R_j, ∀b_{l_{k'}} ∈ 𝔹_j };
        for ∀k' ∈ {1, 2, ··· , k} do
            set x'_{l_{k'},j} = 1, and p^b_{l_{k'},j} = c_{l_{k+1},j};
        end
        set p^s_j = c_{l_{k+1},j};
    end
end
```

### A. Auction Decision Making Process

The auction decision is made locally at each seller, i.e., auctioneer. We now use seller $s_j \in S$ as an example.

To ease presentation, we define $x'_{i,j} = 1$ if buyer $b_i$ is chosen as one of the winning buyer candidates of $s_j$, otherwise, $x'_{i,j} = 0$. According to Eq. (1) and Eq. (2), $b_i$ can win its bid at $s_j$, only if it satisfies the following three conditions: 1) its bid is no less than the minimum price $\widetilde{p}$, 2) the resources required by $b_i$ can be met by $s_j$, and 3) the minimum staying time required by task $b_i$, i.e., $t_i$, is no more than the least staying time of device $s_j$, $T_j$. Without loss of generality, we use $\mathbb{B}_j$ to denote the set of the buyers that satisfy the above three conditions of $s_j$. Hence, initially, $x'_{i,j} = 0$ for $\forall i, j$, and

$$\mathbb{B}_j = \{b_i | c_{i,j} \geq \widetilde{p}, r_i \leq R_j, t_i \leq T_j, \forall b_i \in B\}. \quad (8)$$

We then choose the winning buyer candidates from $\mathbb{B}_j$. We re-sort the buyers in $\mathbb{B}_j$ according to the non-increasing order of their bids, e.g., $\mathbb{B}_j = \{b_{l_1}, b_{l_2}, \cdots, b_{l_{|\mathbb{B}_j|}}\}$, where

$$c_{l_1,j} \geq c_{l_2,j} \geq \cdots \geq c_{l_{|\mathbb{B}_j|},j}. \quad (9)$$

We now consider the buyers in $\mathbb{B}_j$ as follows. To ease understanding, we use $p_{i,j}^b$ to denote the price that $b_i$ requires to pay per unit resource, if $b_i$ is selected as one of the winning buyer candidates of $s_j$.

- If $\sum_{b_i \in \mathbb{B}_j} r_i \leq R_j$, it means that $s_j$ has enough resources to support all the buyers in $\mathbb{B}_j$. In this case, we add all the buyers in $\mathbb{B}_j$ as the candidates of the winning buyers, i.e., $x'_{i,j} = 1$ if $b_i \in \mathbb{B}_j$. Moreover, we set the price $p_j^s = \widetilde{p}$, and $p_{i,j}^b = \widetilde{p}$ for $\forall b_i \in \mathbb{B}_j$.
- If $\sum_{b_i \in \mathbb{B}_j} r_i > R_j$, we calculate the first $k$ buyers whose bids are the highest and can be satisfied by seller $s_j$, i.e.,

$$k = \arg\max_k \left\{ \sum_{k'=1}^{k} r_{l_{k'}} \leq R_j, \forall b_{l_{k'}} \in \mathbb{B}_j \right\}. \quad (10)$$

Then, all the buyers in $\{b_{l_{k'}}|1 \leq k' \leq k\}$ are selected as the candidates of the winning buyers at $s_j$, i.e., $x'_{l_{k'},j} = 1$ when $1 \leq k' \leq k$. Furthermore, we set the price $p^b_{i,j} = c_{l_{k+1},j}$ when $1 \leq k' \leq k$, and $p^s_j = c_{l_{k+1},j}$.

The detail of the above process is presented in Algorithm 1. All the sellers in $S$ follow the same procedure as $s_j$.

### B. Payment Evaluation Process

Due to the distributed nature of auction decision making process, the seller which acts as the auctioneer may cheat on the final payment decided at it, so as to get higher utility. For example, seller $s_j$ may notify untruthful payments, $\overline{p}^b_{i,j} \neq p^b_{i,j}, \overline{p}^s_j \neq p^s_j$, to the buyers. To address this issue, a payment evaluation process is now proposed, which requires the collaborations of the buyers.

After receiving the auction decision, each buyer will know whether it wins in the auction or not. Then, the collaboration of the buyers on evaluating each seller, e.g., $s_j$, can be conducted as follows:

- For each buyer $\forall b_i \in \mathbb{B}_j$, if it fails the auction at $s_j$, it compares the payment decided by $s_j$, $p^s_j$, and its own bid $c_{i,j}$. If $p^s_j = c_{i,j}$, it broadcasts an acknowledgement message to confirm the truthfulness of the seller.
- For each buyer $\forall b_i \in \mathbb{B}_j$, if it wins the auction at $s_j$, it waits the acknowledgements from other buyers. After a certain period, if no acknowledgement is received and the payment $p^s_j > \widetilde{p}$, the buyer judges that seller $s_j$ makes cheat on the payment, and removes it from the set of its winning seller candidates. Alternatively, if no acknowledgement is received but the payment $p^s_j = \widetilde{p}$, the buyer judges that the seller $s_j$ is truthful.

Before analyzing the above payment evaluation process, we first get that

**Corollary 1** *For seller $s_j$, the true payment decided by Algorithm 1 should be either $\widetilde{p}$ or the bid claimed by the buyer whose bid is the largest among all the buyers failing in the auction.*

*Proof:* The above corollary can be easily obtained by considering the two situations in Algorithm 1, which is thus omitted here. ∎

We then analyze the proposed payment evaluation process as follows.

**Lemma 1** *With the above payment evaluation process, seller $s_j$ cannot improve its revenue with cheated payment $\overline{p}^b_{i,j}$, where $\overline{p}^b_{i,j} \neq p^b_{i,j}$.*

*Proof:* We prove the above lemma by considering the following two situations:

Firstly, if $\sum_{b_i \in \mathbb{B}_j} r_i \leq R_j$, the true payment should be $p^b_{i,j} = p^s_j = \widetilde{p}$. We now consider the untruthful case. If $\overline{p}^b_{i,j} < p^b_{i,j}$, the constraint in Eq. (2) will be violated. Alternatively, if $\overline{p}^b_{i,j} > p^b_{i,j}$, according to the payment evaluation process, no acknowledgement will be received at the buyers, and thus it is evaluated as the untruthful seller. Under such a circumstance, $s_j$ will be removed from the set of winning seller candidates, which results in $0$ utility. Hence, in this situation, seller $s_j$ will not cheat on its decision.

Secondly, if $\sum_{b_i \in \mathbb{B}_j} r_i > R_j$, the true payment $p^b_{i,j} = p^s_j = c_{l_{k+1},j}$, where the first $k$ buyers in $\mathbb{B}_j$ win the auction. We now consider the untruthful case. Due to the rationality, we must have $\overline{p}^b_{i,j} > p^b_{i,j} = c_{l_{k+1},j}$ for the winning buyer $b_i$, as otherwise, the revenue of $s_j$ will be reduced. However, to pass the evaluation process of the buyers, the payment $\overline{p}^b_{i,j}$ must be equal to one of the bids claimed by the buyers failing in the auction. In other words,

$$\overline{p}^b_{i,j} \in \{c_{l_{k+2},j}, \cdots, c_{l_{|\mathbb{B}_j|},j}\}. \tag{11}$$

As $c_{l_{k+1},j} \geq c_{l_{k+2},j} \geq \cdots \geq c_{l_{|\mathbb{B}_j|},j}$, the revenue with untruthful payment will be no more than that with truthful payment.

In summary, seller $s_j$ cannot improve its utility with untruthful payment. ∎

**Algorithm 2:** Final seller determination at $b_i$

```
begin
    p_i^b = 0;
    if Σ_{s_j∈S} x'_{i,j} = 0 then
        set x_{i,j} = 0, for ∀s_j ∈ S;
    end
    if Σ_{s_j∈S} x'_{i,j} = 1 then
        set x_{i,j} = x'_{i,j}, for ∀s_j ∈ S;
        set p_i^b = p_{i,j}^b, where x_{i,j} = 1;
    end
    if Σ_{s_j∈S} x'_{i,j} > 1 then
        for ∀s_j ∈ {s_j | x'_{i,j} = 1} do
            calculate U_{i,j} = (v_{i,j} − p_{i,j}^b) r_i;
        end
        j* = arg max_j {U_{i,j} | x'_{i,j} = 1, ∀s_j ∈ S};
        set x_{i,j*} = 1, and x_{i,j} = 0 for any other j ≠ j*;
        set p_i^b = p_{i,j*}^b;
    end
end
```

### C. Final Seller Determination Process

With the above subsection, each seller selects some appropriate buyers as the candidates to win the bids. However, it is noted that a buyer may be assigned to multiple sellers, which cannot satisfy the rule defined in Eq. (4). Hence, for each buyer, it is still necessary to determine the final seller.

The final seller determination process is conducted locally at each buyer. For buyer $b_i \in B$, according to $\{x'_{i,j}\}$, there are three cases: 1) $\sum_{s_j \in S} x'_{i,j} = 0$; 2) $\sum_{s_j \in S} x'_{i,j} = 1$; and 3) $\sum_{s_j \in S} x'_{i,j} > 1$.

Case 1 ($\sum_{s_j \in S} x'_{i,j} = 0$): this means $b_i$ was not selected as the candidate by any sellers in $S$, and hence cannot win its bid. As such, we set $x_{i,j} = 0$, for $\forall j$.

Case 2 ($\sum_{s_j \in S} x'_{i,j} = 1$): it means that buyer $b_i$ wins its

| | Required resources | Bids submitted by buyers | | | | |
|---|---|---|---|---|---|---|
| | | $s_1$ | $s_2$ | $s_3$ | $s_4$ | $s_5$ |
| $b_1$ | 2 | 6 | 0 | 5 | 0 | 6 |
| $b_2$ | 3 | 4 | 6 | 0 | 3 | 6 |
| $b_3$ | 3 | 0 | 4 | 5 | 4 | 0 |
| $b_4$ | 2 | 6 | 0 | 6 | 0 | 5 |

Fig. 2. An illustration to the auction mechanism design

bid at a single seller. In this case, for buyer $b_i$, we have

$$x_{i,j} = \begin{cases} 1, & \text{if } x'_{i,j} = 1; \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Moreover, we set the price that $b_i$ requires to pay per unit resource is

$$p_i^b = p_{i,j}^b, \text{where } x_{i,j} = 1. \tag{13}$$

Case 3 ($\sum_{s_j \in S} x'_{i,j} > 1$): in this case, buyer $b_i$ is selected by multiple sellers in the above subsection. To guarantee each task can be executed at a single device, we choose the final seller of $b_i$ as follows:

- For each seller $s_j$, where $x'_{i,j} = 1$, we calculate the utility of the buyer $b_i$ if $b_i$ wins the bid to $s_j$, as follows:

$$U_{i,j} = \left(v_{i,j} - p_{i,j}^b\right) r_i. \tag{14}$$

- Among all the possible sellers in $\{s_j | x'_{i,j} = 1, \forall s_j \in S\}$, we select the one with the largest utility as the final seller, e.g., $s_{j^*}$, where

$$U_{i,j^*} = \max\left\{U_{i,j} | x'_{i,j} = 1, \forall s_j \in S\right\}. \tag{15}$$

In other words, buyer $b_i$ will win its bid to seller $s_{j^*}$, i.e.,

$$x_{i,j} = \begin{cases} 1, & \text{if } j = j^*; \\ 0, & \text{otherwise.} \end{cases} \tag{16}$$

Furthermore, we set the price $p_i^b = p_{i,j^*}^b$.

The detailed process is described in Algorithm 2.

### D. An Illustration

We now use an example in Fig. 2 to illustrate how the proposed mechanism works. We consider five sellers $\{s_1, s_2, \cdots, s_5\}$ and four buyers $\{b_1, b_2, \cdots, b_4\}$ in the system. The amounts of resources available at seller $s_1, s_2, \cdots, s_5$ are $4, 5, 6, 8, 5$ respectively. Correspondingly, the amount of resources required for each buyer and the bid submitted for each seller are given in Fig. 2. We also assume that the minimum price set by the system is $\widetilde{p} = 3$. To ease presentation, suppose that the least staying time of $s_j$ is higher than the minimum staying time requirements of all the tasks: $T_j \geq t_i$ for $\forall i, j$.

**Auction Decision Making Process**: Initially, $x'_{i,j} = 0$ for $\forall i, j$, $p_j^s = 0$ for $\forall j$, and $p_{i,j}^b = 0$ for $\forall i, j$. The sellers make decisions locally.

- $s_1$: $\mathbb{B}_1 = \{b_1, b_2, b_4\}$, and after sorting, $\mathbb{B}_1 = \{b_1, b_4, b_2\}$. As $\sum_{b_i \in \mathbb{B}_1} r_i > R_1$, we calculate $k = 2$. Hence, $x'_{1,1} = x'_{4,1} = 1$. We get $p_{1,1}^b = p_{4,1}^b = c_{2,1} = 4$, and $p_1^s = 4$.
- $s_2$: after sorting, $\mathbb{B}_2 = \{b_2, b_3\}$. As $\sum_{b_i \in \mathbb{B}_2} r_i > R_2$, we calculate that $k = 1$. Hence, $x'_{2,2} = 1$. We then set $p_{2,2}^b = c_{3,2} = 4$, and $p_2^s = 4$.
- $s_3$: after sorting, $\mathbb{B}_3 = \{b_4, b_1, b_3\}$. As $\sum_{b_i \in \mathbb{B}_3} r_i > R_3$, we get $k = 2$. Then, $x'_{4,3} = x'_{1,3} = 1$. Moreover, we can obtain that $p_{4,3}^b = p_{1,3}^b = c_{3,3} = 5$, and $p_3^s = 5$.
- $s_4$: after sorting, we have $\mathbb{B}_4 = \{b_3, b_2\}$. Since $\sum_{b_i \in \mathbb{B}_4} r_i < R_4$, $x'_{3,4} = x'_{2,4} = 1$. Then, we set $p_{3,4}^b = p_{2,4}^b = \widetilde{p}_4 = 3$, and $p_3^s = 3$.
- $s_5$: after sorting, we have $\mathbb{B}_5 = \{b_1, b_2, b_4\}$. As $\sum_{b_i \in \mathbb{B}_5} r_i > R_5$, we calculate $k = 2$. Then, $x'_{1,5} = x'_{2,5} = 1$, and set $p_{1,5}^b = p_{2,5}^b = c_{4,5} = 5, p_5^s = 5$.

**Final seller determination process**: This procedure is also locally conducted at each buyer.

- $b_1$: as $\sum_{s_j \in S} x'_{1,j} > 1$, we need to compare the utilities obtained at all the possible sellers $\{s_1, s_3, s_5\}$ where $x'_{1,1} = x'_{1,3} = x'_{1,5} = 1$. Obviously, $b_1$ can get the highest utility at $s_1$, $U_{1,1} = (6-4)*2 = 4, U_{1,3} = (5-5)*2 = 0, U_{1,5} = (6-5)*2 = 2$. Hence, $s_1$ is the final seller selected by $b_1$, i.e., $x_{1,1} = 1$, and $x_{1,j} = 0$ for $\forall j \neq 1$.
- $b_2$: it is noted that $\sum_{s_j \in S} x'_{2,j} > 1$, and $b_2$ is selected as the winning buyer candidates at $s_2, s_4, s_5$. Then, we compare their utilities, and obtain that $b_2$ achieves its best utility at $s_2$, where $U_{2,2} = (6-4)*3 = 6$. Hence, $s_2$ is the final seller of $b_2$, i.e., $x_{2,2} = 1$, and $x_{2,j} = 0$ for $\forall j \neq 2$.
- $b_3$: as $\sum_{s_j \in S} x'_{3,j} = 1$, we get that $x_{3,4} = 1$, while $x_{3,j} = 0$ for $j \neq 4$.
- $b_4$: It is noted that $\sum_{s_j \in S} x'_{4,j} > 1$, i.e., both $s_1$ and $s_3$ choose $b_4$ as their winning buyer candidates. Because $b_4$ gets more utility at $s_1$ than at $s_3$, $s_1$ is thus chosen as the final seller of $b_4$, i.e., $x_{4,1} = 1$, and $x_{4,j} = 0$ for $\forall j \neq 1$.

With the above mechanism, the successful trading pairs are $(b_1, s_1)$, $(b_2, s_2)$, $(b_3, s_4)$ and $(b_4, s_1)$, with unit-resource trading price/payment 4, 4, 3 and 4, respectively.

### E. Performance Analysis

We now analyze the performance of the proposed auction mechanism in achieving the computational efficiency, individual rationality, truthfulness, and budget balance.

**Lemma 2** *The proposed auction mechanism is computationally efficient.*

*Proof:* To prove the above lemma, we need to analyze the computation complexity of Algorithm 1 and Algorithm 2.

- Auction decision making process (Algorithm 1): For each seller $s_j$, the complexity of calculating and sorting the buyers in $\mathbb{B}_j$ is $O(m \log_2 m)$. Then, calculating $\sum_{b_i \in \mathbb{B}_j} r_i$ and selecting the appropriate buyer candidates consumes at most $O(m)$ time-complexity. Hence, the complexity of selecting the winning bid candidates at each seller is $O(m \log_2 m)$.

- Final seller determination process (Algorithm 2): For each buyer $b_i$, calculating $\sum_{s_j \in S} x'_{i,j}$ and setting the appropriate value for each $x_{i,j}$ runs with $O(n^2)$ time-complexity. Hence, the complexity of determining the final seller is $O(n^2)$.

To sum up, the complexity of the proposed auction mechanism is polynomial, and thus computationally efficient. ■

We further get that

**Lemma 3** *The proposed auction mechanism is individually rational for both the buyers and the sellers.*

*Proof:* According to Algorithm 1, there are only two cases that buyer $b_i$ can be selected as one of the winning buyer candidates of seller $s_j$.

Case 1 ($\sum_{b_i \in \mathbb{B}_j} r_i \leq R_j$): In this case, $b_i$ is selected as one of the winning buyer candidates of $s_j$, only if $c_{i,j} \geq \widetilde{p}$, as otherwise, $b_i \notin \mathbb{B}_j$. Since the price of buyer $b_i$ and the payment of seller $s_j$ are both set to $p^b_{i,j} = p^s_j = \widetilde{p}$, if $b_i$ is selected as one of the buyers of $s_j$, we have

$$c_{i,j} \geq \widetilde{p} = p^b_{i,j}; \tag{17}$$

$$p^s_j = \widetilde{p}. \tag{18}$$

Hence, in this case, buyer $b_i$ is not charged a price that is greater than its bid, and seller $s_j$ will not be rewarded with a payment that is less than its asked minimum price.

Case 2 ($\sum_{b_i \in \mathbb{B}_j} r_i > R_j$): According to Algorithm 1, seller $s_j$ only selects the first $k$ buyers in $\mathbb{B}_j$ as the candidates, where $k = \arg\max_k \left\{ \sum_{k'=1}^{k} r_{l_{k'}} \leq R_j, \forall b_{l_{k'}} \in \mathbb{B}_j \right\}$. Then, if $b_i$ can be selected as one of the winning buyer candidates, we must have $b_i \in \{b_{l_1}, b_{l_2}, \cdots, b_{l_k}\}$, and $c_{i,j} \geq c_{l_k,j}$. As the price $p^b_{i,j} = c_{l_{k+1},j}$ and $p^s_j = c_{l_{k+1},j}$, if $b_i$ is selected as one of the winning buyer candidates of $s_j$, we have

$$c_{i,j} \geq c_{l_k,j} \geq c_{l_{k+1},j} = p^b_{i,j} = p^b_i; \tag{19}$$

$$\widetilde{p} \leq c_{l_{k+1},j} = p^s_j. \tag{20}$$

Hence, the buyer $b_i$ is not charged more than its bid, and the seller $s_j$ is rewarded no less than its asked minimum price.

Therefore, the proposed auction mechanism is individually rational for both the buyers and the sellers. ■

We further analyze the truthfulness as follows.

**Lemma 4** *The proposed auction mechanism guarantees the truthfulness of the bids submitted by the buyers in the auction.*

*Proof:* To ease presentation, we use $B^c_j$ and $B^w_j$ to denote the set of the winning buyer candidates and the final buyers of seller $s_j$ with Algorithm 1 and Algorithm 2, respectively, i.e., $B^c_j = \{b_i | x'_{i,j} = 1, \forall b_i \in B\}$, $B^w_j = \{b_i | x_{i,j} = 1, \forall b_i \in B\}$. In other words, $B^w_j \subseteq B^c_j$. We now consider the case when the buyer $b_i$ submits untruthful price information, i.e., $c_{i,j} \neq v_{i,j}$. Without loss of generality, we use $\overline{U}^b_i$ and $\overline{p}^b_{i,j}$ to denote the utility of $b_i$, and the price charged by $b_i$ at $s_j$ respectively, when $b_i$ submits untruthful information.

For $\forall b_i \in B$, there are two cases when $b_i$ submits truthful information ($c_{i,j} = v_{i,j}$): 1) $b_i \in \bigcup_{s_j \in S} B^w_j$, and 2) $b_i \notin \bigcup_{s_j \in S} B^w_j$.

1) For buyer $b_i \in \bigcup_{s_j \in S} B^w_j$, it must win its bid at one of the sellers. Assume that $b_i$ wins its bid at seller $s_j$, i.e., $x_{i,j} = 1, b_i \in B^c_j$. We now consider the case when $b_i$ submits untruthful information.

- Buyer $b_i$ loses its bid at all of the sellers: In this case, we have $\overline{U}^b_i = 0$. According to Lemma 3, with truthful submission, i.e., $c_{i,j} = v_{i,j}$, we can obtain that $U^b_i = (v_{i,j} - p^b_i) r_i \geq 0$. Hence, $\overline{U}^b_i \leq U^b_i$.
- Buyer $b_i$ also wins its bid at seller $s_j$: As described in Algorithm 1, the charging price of buyer $b_i$ is either $\widetilde{p}$ or the bid of the $(k+1)$-th buyer in $\mathbb{B}_j$, which is independent of the bid submitted by $b_i$. Thus, we have $\overline{U}^b_i = U^b_i$.
- Buyer $b_i$ wins its bid, but at a different seller $s_{j'}$, where $j' \neq j$. Then, the following two subcases are to be considered.
  - $x'_{i,j'} = 1$ when $b_i$ submits truthful bid: this means that $b_i$ is also selected as one of the winning buyer candidates at $s_{j'}$ in Algorithm 1 with truthful bid. The reason that $b_i$ finally wins its bid at $s_j$, instead of $s_{j'}$, is that the utility obtained at $s_{j'}$, $(v_{i,j'} - p^b_{i,j'}) r_i$, must be no more than the utility obtained at $s_j$, $(v_{i,j} - p^b_{i,j}) r_i$. That is,

$$\left(v_{i,j'} - p^b_{i,j'}\right) r_i \leq \left(v_{i,j} - p^b_{i,j}\right) r_i. \tag{21}$$

    Moreover, as the price charged by $b_i$ at $s_j$ is independent of the bid sent by $b_i$, we still have $p^b_{i,j'} = \overline{p}^b_{i,j'}$, even if $b_i$ sends untruthful bid. Therefore, we can obtain that

$$\begin{aligned} \overline{U}^b_i &= \left(v_{i,j'} - \overline{p}^b_{i,j'}\right) r_i \\ &= \left(v_{i,j'} - p^b_{i,j'}\right) r_i \\ &\leq \left(v_{i,j} - p^b_{i,j}\right) r_i = U^b_i. \end{aligned} \tag{22}$$

  - $x'_{i,j'} = 0$ when $b_i$ submits truthful bid: it means $b_i$ is not selected as one of the winning buyer candidates of $s_{j'}$ in Algorithm 1 with truthful bid. There are two cases for this result. In the first case, $b_i \notin \mathbb{B}_{j'}$, i.e., $c_{i,j'} = v_{i,j'} < \widetilde{p}$. When $b_i$ wins its bid at $s_{j'}$ untruthfully, it pays $\overline{p}^b_{i,j'}$. Nevertheless, according to Lemma 3, winning its bid must satisfy that $\overline{p}^b_{i,j'} \geq \widetilde{p}$. Hence,

$$\begin{aligned} \overline{U}^b_i &= (v_{i,j'} - \overline{p}^b_{i,j'}) r_i \\ &< (\widetilde{p} - \overline{p}^b_{i,j'}) r_i \\ &\leq (\overline{p}^b_{i,j'} - \overline{p}^b_{i,j'}) r_i \\ &= 0 \leq U^b_i. \end{aligned} \tag{23}$$

    In the second case, $b_i \in \mathbb{B}_{j'}$, but $b_i$ is not included in $B^c_{j'}$. This case only happens when the truthful bid $v_{i,j'} \leq c_{l_{k+1},j'}$, where $k = |B^c_{j'}|$. Nevertheless, $b_i$ wins its bid at $s_{j'}$ untruthfully. Hence, the untruthful bid $c_{i,j'} \geq c_{l_{k+1},j'}$. As the final payment of seller $s_{j'}$ is either $c_{l_k,j'}$ or $c_{l_{k+1},j'}$, i.e., $\overline{p}^b_{i,j'} \geq c_{l_{k+1},j'}$, we

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2818147, IEEE Transactions on Services Computing



obtain that

$$\begin{aligned}\overline{U}_i^b &= (v_{i,j'} - \overline{p}_{i,j'}^b) r_i \\ &\leq (v_{i,j'} - c_{l_{k+1},j'}) r_i \\ &\leq (c_{l_{k+1},j'} - c_{l_{k+1},j'}) r_i \\ &= 0 \leq U_i^b. \end{aligned} \quad (24)$$

2) For buyer $b_i \notin \bigcup_{s_j \in S} B_j^w$, it must not win its bid at any seller with truthful information. In other words, $U_i^b = 0$. There are two cases when $b_i$ submits untruthful bid.

- $b_i$ still does not win its bid at any seller: in this case, we have $\overline{U}_i^b = U_i^b = 0$.
- $b_i$ wins its bid at seller $s_j$ untruthfully: we now consider the reason that $b_i$ does not win its bid at $s_j$ in truthful case.
  - $v_{i,j} < \widetilde{p}$: in this case, $b_i$ cannot win its bid at $s_j$ obviously. Nevertheless, if $b_i$ wins its bid at $s_j$, its untruthful bid $c_{i,j}$ must be no less than $\widetilde{p}$. Moreover, when $b_i$ wins its bid, we must have $\overline{p}_{i,j}^b \geq \widetilde{p}$. Hence, we have

$$\overline{U}_i^b = (v_{i,j} - \overline{p}_{i,j}^b) r_i < (\widetilde{p} - \overline{p}_{i,j}^b) r_i \leq 0. \quad (25)$$

  - $v_{i,j} \geq \widetilde{p}$, but $b_i$ is not included in $B_j^c$: It means that in truthful case, we have $b_i \in \mathbb{B}_j$, but $b_i \notin B_j^c$, i.e.,

$$\sum_{b_i \in \mathbb{B}_j} r_i \geq R_j, \quad (26)$$

$$v_{i,j} \leq c_{l_{k+1},j}, \text{ where } k = |B_j^c|. \quad (27)$$

As $b_i$ wins its bid at $s_j$ untruthfully, its submitted bid $c_{i,j}$ must be no less than $c_{l_k,j}$, as otherwise, $b_i$ cannot be included in $B_j^c$. Moreover, as $\sum_{b_i \in \mathbb{B}_j} r_i \geq R_j$, according to Algorithm 1, the price $\overline{p}_{i,j}^b$ is either $c_{l_k,j}$ or $c_{l_{k+1},j}$. As $c_{l_k,j} \geq c_{l_{k+1},j}$, we have $\overline{p}_{i,j}^b \geq c_{l_{k+1},j}$. Then, it is easy to get that

$$\overline{U}_i^b = (v_{i,j} - \overline{p}_{i,j}^b) r_i \leq (c_{l_{k+1},j} - \overline{p}_{i,j}^b) r_i \leq 0. \quad (28)$$

To sum up, submitting untruthful bid cannot increase the utility of the buyer. In other words, only with the truthful bid, the maximum utility of the buyer can be achieved. Thus, we prove the above lemma. ∎

**Lemma 5** *The proposed auction mechanism is budget-balanced.*

*Proof:* According to Algorithm 1 and Algorithm 2, if $x_{i,j} = 1$, it means buyer $b_i$ will win its bid at seller $s_j$, and the price paid by $b_i$ is equal to the payment obtained at $s_j$, i.e., $p_i^b = p_j^s$. Hence, we have

$$\sum_{i=1}^{m} x_{i,j} r_i p_i^b = \sum_{i=1}^{m} x_{i,j} r_i p_j^s, \forall j. \quad (29)$$

In other words, the budget balance can be achieved, which thus proves the lemma. ∎

According to the above analysis, our proposed auction mechanism holds the following properties: computational efficiency, individual rationality, guaranteeing the truthfulness of the bids, and budget balance.

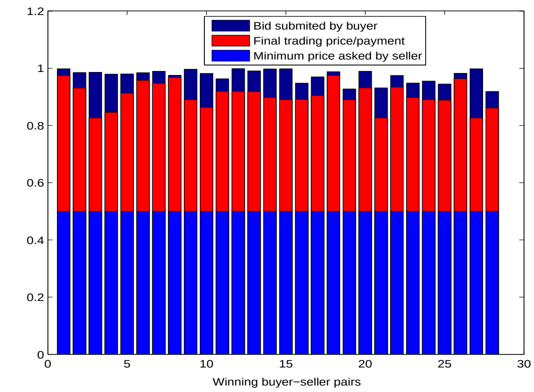


Fig. 3. Performance on individual rationality and budget balance

## IV. Simulation Results

In this section, we present simulation results to evaluate the performance of the proposed auction mechanism. Similar to [28], we generate the bids submitted by the buyers and the truthful valuation of the buyers according to a uniform distribution within $[0, 1]$, and the minimum price of the sellers $\widetilde{p} = 0.5$. In the simulations, we investigate the performance of the proposed scheme on the individual rationality, the impact of untruthful activities on the utilities that can be achieved at the buyers, and the system efficiency obtained by the auction mechanism.

### A. Performance on Individual Rationality and Budget Balance

We now evaluate the performance of our mechanism on achieving the individual rationality and budget balance.

In this simulation, we set $m = n = 50$, randomly generate the amount of resources required by each buyer $b_i$ within $[3, 10]$, and the amount of resources available at each seller $s_j$ within $[10, 20]$, i.e., $r_i \in [3, 10], R_j \in [10, 20]$. Among the winning buyers and their corresponding sellers, we choose 28 pairs of them, and for each pair, we show the bid submitted by the buyer, final trading price/payment of the buyer/seller, and the minimum prices asked by the trading seller.

As shown in Fig. 3, the final price paid by each buyer is no more than the bid submitted by the buyer, and the final payment earned at each seller is no less than the minimum price asked by each seller. In other words, the individual rationality of both buyers and sellers is well achieved by the proposed scheme. Moreover, as described in Algorithm 1 and Algorithm 2, the price paid by each buyer is the same as the payment earned at its corresponding seller. Thus, the budget balance can also be achieved in the simulation.

### B. Performance on the Truthfulness of the Bidders

To evaluate the truthfulness of the bidders, we study two settings in the simulations.

In the first setting, we only simulate the untruthful activity of buyer $b_i$ on a specific seller $b_j$, and investigate the utility of $b_i$ obtained at $s_j$. i.e., $(v_{i,j} - p_{i,j}^b) r_i x_{i,j}$. As shown in Fig. 4 (a), the truthful valuation $v_{i,j}$ is set to 0.7975, and $b_i$ can

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2818147, IEEE Transactions on Services Computing



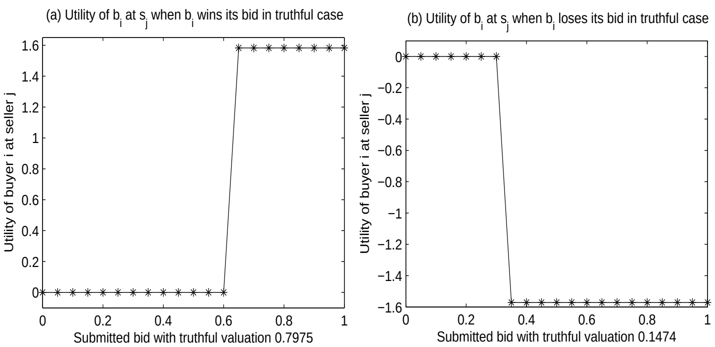


Fig. 4. Utility comparison of buyer $b_i$ at $s_j$ in truthful and untruthful cases

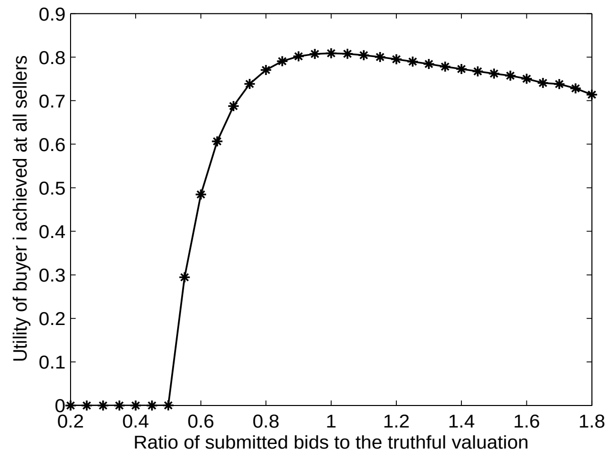


Fig. 5. Utility comparison of buyer $b_i$ at all sellers under different bids

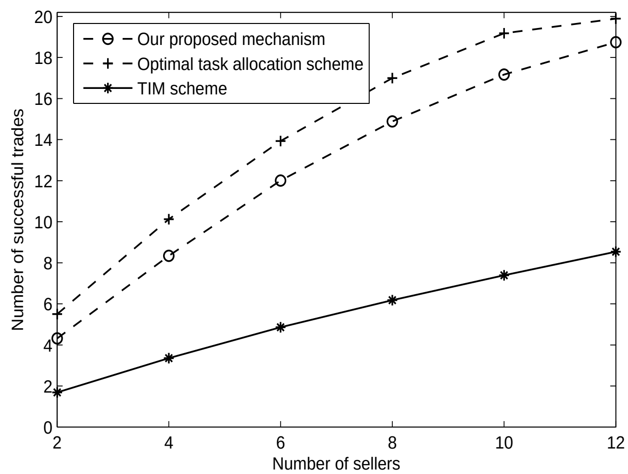


Fig. 6. Comparison of the number of successful trades

win its bid at $s_j$ if it submits the truthful bid. However, $b_i$ may sends untruthful bid to $s_j$, which is shown in x-axis. As analyzed in Lemma 4, submitting untruthful bids cannot increase the utility of $b_i$ at $s_j$. Similarly, in Fig. 4 (b), we investigate the other case when $b_i$ loses its bid at seller $s_j$ with truthful valuation $0.1474$. As seen, although $b_i$ can win its bid at $s_j$ by submitting a bid much higher than its truthful valuation, e.g., more than $0.4$, its utility obtained at $s_j$ is much less than $0$. In other words, cheating on bids not only cannot increase its profit, but also may hurt its own utility.

In the second setting, we investigate the utility of $b_i$ that can be obtained at all the sellers, i.e., $\sum_{s_j \in S}(v_{i,j} - p^b_{i,j})r_i x_{i,j}$. As shown in Fig. 5, the value in x-axis is defined as the ratio of the submitted bid to the truthful valuation, i.e.,

$$\frac{c_{i,j}}{v_{i,j}}, \forall c_{i,j} \in C_i. \tag{30}$$

For example, if the ratio is $0.4$, the bid $c_{i,j} = 0.4 * v_{i,j}$. It is observed that $b_i$ achieves the maximum utility when the ratio is 1, as in this case, the submitted bid $c_{i,j}$ is exactly the same as the truthful valuation $v_{i,j}$. The smaller the gap between the submitted bids and the truthful valuation, the larger the utility achieved by $b_i$. We would like to note that when the ratio is small, e.g., between $0$ and $0.5$, the utility of the buyer is $0$. This is reasonable, as when the buyer deliberately submits a very low untruthful bid, the buyer is unlikely to win the auction, according to Algorithm 1. In addition, when the bid is less than the truthful valuation, increasing the submitted bid can improve the utility, while when the bid is higher than the truthful valuation, increasing it will decrease the utility. In other words, only with truthful bid, the maximum utility of the buyer can be obtained, which guarantees the truthfulness.

### *C. Performance on System Efficiency*

As described in the previous sections, this paper mainly aims to achieve the individual rationality, budget balance, computational efficiency, and truthfulness of the bidders. Nevertheless, system efficiency is also extremely important for a practical system. Here, system efficiency is defined as the number of successfully trades made by the proposed auction mechanism. Generally, the more the number of the successfully trades, the better the system efficiency, as more tasks can be successfully allocated to the appropriate mobile devices for executions. To improve the system efficiency, we run the proposed auction mechanism round by round, until no further winning bids can be added.

To evaluate the performance on system efficiency, we introduce two baseline algorithms, named *optimal task allocation scheme* and *TIM scheme* respectively. Specifically, optimal task allocation scheme only aims to find the optimal matching between the tasks and the mobile devices, with the objective of maximizing the total number of tasks that can be allocated to the appropriate mobile device, which however, does not consider the possible untruthful activities of the bidders. TIM (Truthful Incentive Mechanism) is proposed in [28], which also achieves the individual rationality, truthfulness of the bidders and the budget balance. However, TIM is only suitable in homogeneous system, where the buyers and the sellers require and hold the same amount of resources, respectively. Hence, it only makes one-to-one trade between the buyers and the sellers.

As shown in Fig. 6, we fix the number of buyers (i.e., tasks) into 20, while varying the number of sellers (i.e., mobile devices) within $[2, 4, 6, 8, 10, 12]$. We can see that the gap between the proposed mechanism and the optimal

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2018.2818147, IEEE Transactions on Services Computing



task allocation scheme is not large. Moreover, optimal task allocation scheme cannot guarantee the truthfulness of bids submitted by the bidders, as it only aims to optimize the number of tasks allocated to the mobile devices. It is also observed that although TIM scheme achieves the truthfulness of the buyers, its system efficiency is much worse than ours. This is possible, as TIM restricts that each seller can only be matched with at most one buyer, while in our mechanism, as long as the seller has enough resources, it can accept multiple buyers. Moreover, Fig. 6 shows that when the number of sellers increases, the system efficiency for all the three schemes gets improved. The reason is that more sellers means more resources in the system, and can be used to satisfy the resource requirements of more tasks.

## V. Conclusion

In this paper, we consider mobile task allocation problem in mobile cloud computing. To incentivize the mobile devices to participate in task execution, we construct an auction model to facilitate the resource trading between the owner of the tasks and the mobile devices. Specifically, the owners of the tasks, acted as the buyers, submit bids to compete for the resources available at the mobile devices, acted as the sellers. A distributed auction mechanism is then designed to fairly allocate the tasks, and determine the trading prices of the resources. Furthermore, an efficient payment evaluation process is proposed to prevent against the possible dishonest activity of the seller on the payment decision, through the collaboration of the buyers. We prove that the proposed auction mechanism achieves certain desirable properties, including computational efficiency, individual rationality, truthfulness guarantee of submitted bids and the sellers, and budget balance. We also validate the performance of the proposed mechanism through simulations.

## References

[1] J. Paczkowski, “iPhone owners would like to replace battery,” Available: http://digitaldaily.allthingsd.com/20090821/iphone-owners-wouldlike- to-replace-battery-att, Aug. 2009.

[2] A. P. Miettinen, and J. K. Nurminen, “Energy efficiency of mobile clients in cloud computing,” in *Proceedings of the 2nd USENIX conference on hot topics in cloud computing*, pp. 1–7, Berkeley, CA, USA, 2010.

[3] N. Balasubramanian, “Energy consumption in mobile phones: a measurement study and implications for network applications,” in *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference (IMC)* pp. 280–293, Chicago, Illinois, USA, 2009.

[4] M. Satyanarayanan, “Fundamental challenges in mobile computing” in *Proceedings of ACM Symposium on Principles of Distributed Computing* (PODC), pp. 1–7, Philttdelphia PA, USA, 1996.

[5] A. Khan, M. Othman, S. Madani and S. U. Khan, “A survey of mobile cloud computing application models,” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 393–413, 2014.

[6] A. P. Miettinen and J. K. Nurminen, “Energy efficiency of mobile clients in cloud computing,” in *Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing*, pp. 1–7, Boston, MA, 2010.

[7] E. Cuervo, A. Balasubramanian, D. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, “Maui: making smartphones last longer with code offload,” in *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services* (ACM MobiSys), pp. 49–62, San Francisco, CA, USA, 2010.

[8] B. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, “Clonecloud: elastic execution between mobile device and cloud,” in *Proceedings of the sixth Conference on Computer Systems* (EuroSys), pp. 301–314, Salzburg, Austria, 2011.

[9] P. Shu, F. Liu, H. Jin, M. Chen, F. Wen, Y. Qu, B. Li, “eTime: Energy-efficient transmission between cloud and mobile devices,” in *Proceedings of IEEE International Conference on Computer Communications (INFOCOM)*, pp. 195–199, Turin, 2013.

[10] Y. Mao, C. You, J. Zhang, K. Huang and K. B. Letaief, “A survey on mobile edge computing: the communication perspective,” IEEE Communications Surveys & Tutorials, vol. 19, no. 4, pp. 2322–2358, 2017.

[11] X. Chen, L. Pu, L. Gao, W. Wu and D. Wu, “Exploiting massive D2D collaboration for energy-efficient mobile edge computing,” IEEE Wireless Communications, vol. 24, no. 4, pp. 64–71, 2017.

[12] M. Ra, A. Sheth, L. Mummert, P. Pillai, D. Wetherall, and R. Govindan, “Odessa: enabling interactive perception applications on mobile devices,” in *Proceedings of the 9th international conference on Mobile systems, applications, and services* (MobiSys), pp. 43–56, NY, USA, 2011.

[13] Y. Kao, B. Krishnamachari, M. Ra,and F. Bai, “Hermes: latency optimal task assignment for resource-constrained mobile computing,” in *Proceedings of IEEE International Conference on Computer Communications* (INFOCOM), pp. 1894–1902, Hong Kong, 2015.

[14] E. Miluzzo, R. C. Ceres and Y. Chen, “Vision: mClouds - computing on clouds of mobile devices,” in *Proceedings of the third ACM workshop on Mobile Cloud Computing and Services* (MCS'12), pp. 9–14, New York, NY, USA, 2012.

[15] F. Liu, P. Shu, H. Jin, L. Ding, J. Yu, D. Niu, and B. Li, “Gearing resource poor mobile devices with powerful clouds: architectures, challenges, and applications,” *IEEE Wireless Communications*, vol. 20, no. 3, pp. 14–22, 2013.

[16] A. Mtibaa, K. A. Harras, and A. Fahim, “Towards computational offloading in mobile device clouds,” in *Proceedings of IEEE International Conference on Cloud Computing Technology and Science*, pp. 331–338, Bristol, 2013.

[17] S. Noor, R. Hasan and M. Haque, “Cellcould: a novel cost effective formation of mobile cloud based on bidding incentives,” in *Proceedings of IEEE International Conference on Cloud Computing*, pp. 200–207, Anchorage, AK, 2014.

[18] J. Song, Y. Cui, M. Li, J. Qiu and R. Buyya, “Energy-traffic tradeoff cooperative offloading for mobile cloud computing,” in *Proceedings of IEEE 22nd International Symposium of Quality of Service (IWQoS)*, pp. 284–289, Hong Kong, May. 2014.

[19] X. Wang, X. Chen, W. Wu, N. An, L. Wang, “Cooperative application execution in mobile cloud computing: a stackelberg game approach,” *IEEE Communications Letters*, vol. 20, no. 5, pp. 946–949, 2016.

[20] J. Fan, X. Wei, R. Li, and Q. Sun, “A cooperative incentive structure for mobile cloud computing,” *Parallel and Cloud Computing Research*, vol. 03, pp. 5–10, 2015.

[21] X. Wang, X. Chen, W. Wu, “Towards truthful auction mechanisms for task assignment in mobile device clouds,” in *Proeedings of IEEE International Conference on Computer Communications (INFOCOM)*, Atlanta, GA, USA, May 2017.

[22] N. Nisan, T. Roughgarden, E. Tardos, and V. V Vazirani, Algorithmic game theory, Cambridge University Press, 2007.

[23] V. Krishna, Auction Theory, 2nd ed. Academic Press, Aug. 2009.

[24] E. Clarke, “Multipart pricing of public goods,” Public Choice, vol. 11, no. 1, pp. 17–33, 1971.

[25] W. Vickrey, “Counterspeculation, auctions, and competitive sealed tenders,” *The Journal of Finance*, vol. 16, no. 1, pp. 8–37, 2012.

[26] L. Zhang, S. Ren, C. Wu and Z. Li, “A truthful incentive mechanism for emergency demand response in colocation data centers,” in *Proceedings of IEEE Conference on Computer Communications* (INFOCOM), pp. 1–9, Kowloon, Hong Kong, 2015.

[27] X. Zhang, G. Xue, R. Yu, D. Yang, J. Tang, “Truthful incentive mechanisms for crowdsourcing,” in *Proceedings of IEEE Conference on Computer Communications* (INFOCOM), pp. 2830–2838, Kowloon, Hong Kong, 2015.

[28] A. L. Jin, W. Song, P. Wang, D. Niyato, P. Ju, “Auction mechanisms toward efficient resource sharing for cloudlets in mobile cloud computing,” *IEEE Transactions on Services Computing*, vol. PP, no. 99, pp. 1–14, doi: 10.1109/TSC.2015.2430315, 2015.

[29] A. L. Jin, W. Song, W. Zhuang, “Auction-based resource allocation for sharing cloudlets in mobile cloud computing,” *IEEE Transactions on Emerging Topics in Computing*, vol. PP, no. 99, pp. 1–12, doi: 10.1109/TETC.2015.2487865, 2015.

**Dr. Xiumin Wang** is an Associate Professor in the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. She received her B.S. from the Department of Computer Science, Anhui Normal University, China, in 2006, and a joint Ph.D. degree from University of Science and Technology of China (USTC) and City University of Hong Kong (Cityu), in 2011. She did her postdoc at Singapore University of Technology and Design from 2011 to 2012. Her research interests include mobile computing, wireless networks, and algorithm design and optimization.

**Yang Sui** received his B.S. and M.S. degrees from the School of Computer and Information, Hefei University of Technology, Hefei, China, in 2014 and 2017, respectively. He is now working as a researcher at China Electronic Technology Corporation (CETC). His interests include wireless networks, network coding, mobile computing.

**Dr. Jianping Wang** is an associate professor in the Department of Computer Science at City University of Hong Kong. She received the B.S. and the M.S. degrees in computer science from Nankai University, Tianjin, China in 1996 and 1999, respectively, and the Ph.D. degree in computer science from the University of Texas at Dallas in 2003. Jianping's research interests include dependable networking, optical networks, cloud computing, service oriented networking and data center networks.

**Dr. Chau Yuen** Dr. Chau Yuen received the B. Eng and PhD degree from Nanyang Technological University, Singapore in 2000 and 2004 respectively. Dr Yuen was a Post Doc Fellow in Lucent Technologies Bell Labs, Murray Hill during 2005. He was a Visiting Assistant Professor of Hong Kong Polytechnic University in 2008. During the period of 2006C2010, he worked at the Institute for Infocomm Research (Singapore) as a Senior Research Engineer. He is now an Associate Professor at Singapore University of Technology and Design. He serves as an Associate Editor for IEEE Transactions on Vehicular Technology. In 2012, he received IEEE Asia-Pacic Outstanding Young Researcher Award.

**Dr. Weiwei Wu** received the B.S. degree in South China University of Technology, China, and the PhD degree from City University of Hong Kong and University of Science and Technology of China (P.R.China) in 2011. Since 2012, he has been an associate professor in Southeast University, P.R. China. His research interest mainly lies in optimization in wireless networks, algorithm design and analysis, scheduling, cloud computing and game theory.