Online Combinatorial Double Auction for Mobile Cloud Computing Markets

Ke Xu*, Yuchao Zhang*, Xuelin Shi*, Haiyang Wang†, Yong Wang‡ and Meng Shen§
*Department of Computer Science and Technology, Tsinghua University, China
Email: xuke@mail.tsinghua.edu.cn, zhangyc14@mails.tsinghua.edu.cn, shixuelin@sina.com
†Department of Computer Science, University of Minnesota at Duluth, USA
Email: haiyang@d.umn.edu
‡School of Social Sciences, Tsinghua University, China
Email: wang.yong@tsinghua.edu.cn
§School of Computer Science, Beijing Institute of Technology, China.
Email: shenmeng@bit.edu.cn
Tsinghua National Laboratory for Information Science and Technology

***Abstract*—The emergence of cloud computing as an efficient means of providing computing as a form of utility can already be felt with the burgeoning of cloud service companies. Notable examples including Amazon EC2, Rackspace, Google App and Microsoft Azure have already attracted an increasing number of users over the Internet. However, due to the dynamic behaviors of some users, the traditional cloud pricing models cannot well support such popular applications as Mobile Cloud Computing (MCC). To mitigate this problem, we take our first steps towards the design of an efficient double-sided combinatorial auction model in the context of mobile cloud computing. In particular, we carefully develop the framework of online combinatorial double auctions and apply a Winner Determination Problem (WDP) model for the proposed auction mechanism. The experiment results indicate that the allocation efficiency of our proposed online auction mechanism is comparable to the social optimal solution.**

I. Introduction

Cloud computing is emerging as a promising paradigm that enables on-demand and elastic access to computing infrastructures. Despite the burgeoning of Internet cloud services, the existing cloud markets are still in the premature stages with respect to their pricing structures. Amazon EC2, for example, advertises $\$0.03 - 0.12$ per hour for each of its Virtual Machine (VM) instances, depending on their types. Such a posted-offer pricing model is commonly used when the commodity to be priced has a well-known value that is common knowledge to both sellers and buyers, and a buyer is simply a price-taker that chooses whether or not to pay the price, complete the transaction, and acquire the commodity. Such a fixed pricing scheme, while perhaps acceptable to a small group of enterprise and individual users, essentially shut the door upon the vast majority of potential cloud users.

To mitigate such a problem, the auction-based instances are widely suggested in the cloud market. Such *Spot Instances* allow the customers to bid on unused resources (e.g., EC2 VMs) and run those instances as long as their bids exceed the current spot price, bringing more freedom to users. Researchers therefore proposed different auction mechanisms to implement resource allocation and pricing in cloud markets [1] [2] [3] [4]. However, these single-sided single-minded auction models cannot well support such popular cloud applications as Mobile Cloud Computing (MCC). In particular, Sharrukh Zaman added detailed reasons in [5] that auctions have clear advantages over others when the auctioned items are complementary. A survey from Juniper Research [6] states that the consumer and enterprise market for cloud-based mobile applications is expected to mount to *$9.5* billion by 2014. It is thus important to develop a smarter auction model to support such an elevating demand.

To better support the MCC applications and users, we carefully design the framework of online combinatorial double auctions and apply a WDP model for the proposed auction mechanism. We further develop a decomposition algorithm to solve WDP, which can effectively determine winners as well as prices of each auction in affordable time. Moreover, we also investigate a bidding language to facilitate mobile users to express valuations concisely, and nearly reach the social optimal solution. Our experiment results show that the allocation efficiency of our proposed online auction mechanism is comparable to the social optimal solution and computationally feasible.

The rest of this paper is organized as follows: Section 2 reviews some related work, and Section 3 proposes a framework of the MCC combinatorial double auction. Section 4 describes the bidding language, while the model and algorithm of WDP are presented in Section 5. Then the simulation results are given in Section 6, and Section 7 concludes the paper.

II. Related Work

As MCC is the combination of wireless access services and cloud services, it is feasible to apply auctions in MCC markets. In this section, we will present related work on auctions (especially combinatorial and double auctions), and then review the use of auctions to implement resource allocation and pricing in cloud and MCC markets.

This work has been supported in part by NSFC Project (61170292, 61472212, 61161140454), National Science and Technology Major Project(2012ZX03005001), 973 Project of China (2012CB315803), 863 Project of China (2013AA013302) and EU MARIE CURIE ACTIONS EVANS (PIRSES-GA-2013-610524).

A. *Combinatorial and Double Auctions*

Auctions are effective economic ways for setting the price of commodities based on supply and demand in real-world markets. The auction model supports one-to-many (for instance, single-sided auction) or many-to-many (e.g., double auction) negotiations between sellers and buyers, and reduces negotiations to a single value (i.e., price). Moreover, in an auction, players may be allowed to bid for one commodity or sets of items at one time.

Now many e-commerce platforms adopting Combinatorial Auctions (CAs) only support one-to-many negotiations, i.e., single-sided auctions. One auctioneer initials an auction before many buyers bid, and vice versa. Weijie Shi gave an online framework for resource provisioning in [7], and by a series of one-round optimizations, the authors applied a randomized auction then approximated the one-round social welfare optimization. Although single-sided auctions are well-suited for markets with a limited number of buyers or sellers, these mechanisms are non-effective when the markets consist of numerous of buyers and sellers. To maximize the profits, a potential buyer or seller may bid repeatedly in various auctions, and to relieve this computational burden and promote transactions, many recent researches have been devoted to the double auctions [8].

Double auctions are many-to-many negotiations, which enable multiple buyers and sellers to bid simultaneously in one auction. Indeed, the major exchanges today, like NASDAQ, New York Stock Exchange (NYSE) and the major foreign exchange (FX), apply variants of double auctions [9].

B. *Auctions in cloud and MCC Markets*

The use of auctions in computing dates back to 1968 when Sutherland [10] proposed allocating processing time in a single computer via auctions. Then with the development of grid computing, many market-based resource allocation strategies were brought out, some of which applied auction mechanisms to grid scheduling [11] [12].

Currently researchers are investigating the economic aspects of cloud computing from different points of view. Buyya *et al.* [1] proposed an infrastructure of federated clouds for auction-based resource allocation across multiple clouds. Prasad *et al.* [13] and Zaman *et al.* [5] used combinatorial auctions to implement computing resources and virtual machine allocation. Furthermore, Lee *et al.* [2] brought out a real-time group auction system for cloud application allocation. Zhang *et al.* [3] put forward a framework for online auctions in cloud computing. In our former work [4], a continuous double auction mechanism was proposed for cloud markets, and a bidding strategy was designed for cloud users and CSPs to maximize their profits.

For MCC resource and application allocation, there is little work introducing auction mechanisms to MCC markets. Niyato *et al.* [14] developed an auction mechanism with premium and discount factors for resource allocation in MCC systems. The major difference between our work and the current work is that we are considering a combinatorial double auction mechanism for MCC markets, which enables mobile users and MCC providers to submit bids and asks simultaneously and supports users to bid sets of commodities at one time.

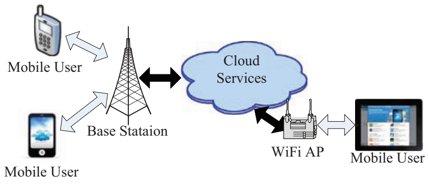


Fig. 1. MCC enabling mobile users accessing cloud services through wireless networking

III. The Framework of MCC Combinatorial Double Auction

We consider a platform for MCC markets where multiple mobile users and MCC providers respectively buy and sell commodities in a combinatorial double auction manner.

A. *Design Requirements*

As shown in Fig.1, the mobile users access services provided by remote clouds via wireless networks, like 2G, 3G or WiFi. Mobile communication base stations or WiFi access points provide radio resources (i.e., bandwidth), while remote cloud provide applications, computing and storage resources.

A feasible auction model for MCC should meet the following requirements: Firstly, the MCC services are the combinations of wireless services and cloud computing resources. If mobile users want to use cloud services, they also need to buy wireless access services. In a feasible MCC market, commodities cover wireless access services as well as applications, computing and storage resources. Secondly, energy efficiency is of particular importance for mobile devices. Moreover, both transmission and computation consume energy. Thirdly, different from traditional CSPs, MCC providers usually offer various applications besides computing utilities and storage resources, such as image processing, natural language translating, and multimedia search [15]. Finally, in current cloud markets cloud users often rent resources to support websites, or run scientific computing [16] [17]. On the other hand, the simple auction rules are more acceptable to mobile users.

B. *Framework Overview*

The online auction platform collects the bids and asks from mobile users and MCC providers respectively, and it computes who and how to win the auctions. The overview of the framework is shown in Fig.2, and the detailed information is in [18].

It is an electronic bidding platform, and the auction on it can be divided into 3 states: the registration stage, the bidding stage and the winner determination stage. In the registration stage, the information of all resources, the related parameters of mobile users, and MCC providers are presented on the bulletin board, and every player is certified. Then in the bidding stage, users can submit bids and providers can submit asks. Eventually, the winners and prices are computed by the determination module according to the combinatorial double auction mechanism.

Furthermore, mobile users can download a bidding application, which executes on mobile devices to translate user's specific demands into requests described in the bidding

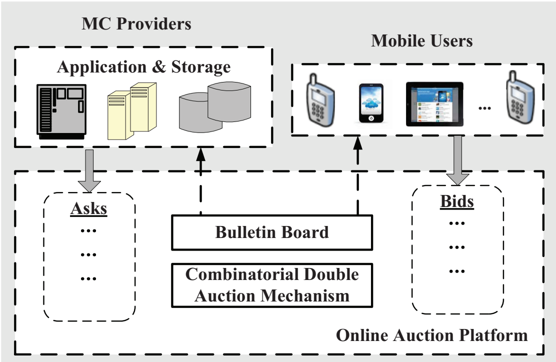


Fig. 2. A Framework of the MCC Combinatorial Double Auction Platform

language, by which user's heterogeneous demands can be restricted to regulated and consistent forms while the details of the requirements can still be revealed. Each request is then submitted to the platform. MCC providers also can submit asks of commodities they want to sell. After an auction closes, the platform computes winners and prices based on the auction mechanism, and then announces the results to users and providers who establish the connection and start to run/host applications once the charging and payment are complete.

The bidding rules of the platform are given in the next subsection. Section 4 describes the bidding language for mobile users, and Section 5 presents how to determine winners in the MCC combinatorial double auction.

C. The Market Rules

In our MCC auction framework, the platform acts as a central auctioneer. To facilitate bidders and improve trading efficiency, some market-rules are defined as follows.

Rule 1: The platform prescribes the **Bidding Period**, $\boldsymbol{t_{bp}}$, can be one day, several hours, etc. During the Bidding Period, mobile users and MCC providers are allowed to submit bids and asks, by the end of which the auction closes and the market clears.

Rule 2: A bid of user i can be for bundles of items, denoted as $\boldsymbol{B_i = \mathcal{L}_{MU}(\langle S, v_i^S \rangle)}$. $\boldsymbol{S}$ is a subset of the available commodities in the auction. v_i^S is a valuation (willingness to pay) of user i for $\boldsymbol{S}$.

Rule 3: An ask of provider j can be for multiple units of items, denoted as $\boldsymbol{A_j = (\langle r_1, c_j^{r_1}, q_j^{r_1} \rangle, \ldots, \langle r_k, c_j^{r_k}, c_j^{r_k} \rangle, \ldots, \langle r_m, c_j^{r_m}, q_j^{r_m} \rangle)}$. $c_j^{r_k}$ is the offered price per unit for commodity r_k of provider j, and $q_j^{r_k}$ is the quantity of commodity r_k.

Rule 4: To prevent unreasonably low bids and speed up the trading process, the **Minimum Bid** allowed in the market, $\boldsymbol{B_{min}}$, is defined.

Rule 5: In the same way, to prevent unreasonably high asks and speed up the trading process, the **Maximum Ask** allowed in the market, $\boldsymbol{A_{max}}$, is defined.

The above rules are published on the MCC auction platform. As long as users and providers take part in auctions, they must submit bids and asks according to the rules. A scenario of the combinatorial double auction on the platform is given in [18].

IV. The Bidding Language for Mobile Users

Bidding language is a language for expressing valuation functions, and a good one which allows bidders to concisely express natural valuation functions. In our MCC auction framework, the bidding language is implemented in the mobile client side to translate user's specific demands into requests. In this section, we first analyze different types of mobile user valuations, and then we put forward a novel bidding language $\mathcal{L}_{MU}$ to represent heterogenous user demands concisely and consistently. At last, we discuss the novel contributions of $\mathcal{L}_{MU}$ in MCC combinatorial double auctions.

A. Heterogenous Mobile User Valuations

In general, let $\boldsymbol{R}$ be the set of all the types of goods for sale in a CA, a buyer could have a different valuation for every subset $\boldsymbol{S}$ of $\boldsymbol{R}$. Because $\boldsymbol{R}$ has $2^{|R|} - 1$ different subsets, there are $2^{|R|} - 1$ possible bids to specify in the CA.

Furthermore, how valuable one item is to a buyer may depend on whether he/she possesses another item. On one hand, some of these items are substitutable (e.g., users can use storage from different places) that they have similar functions to the users. On the other hand, some items are complementary that users will need them as a bundle (e.g., users need both wireless connection and storage for online photo posting).

The complementary and substitutable items in an MCC market can be defined as follows.

Definition 1: A mobile user i has a valuation for a commodity r, denoted as $v_i^{\{r\}}$. For user i, items a and b are **substitutable** if $v_i^{\{a,b\}} < v_i^{\{a\}} + v_i^{\{b\}}$, and items a and b are **complementary** if $v_i^{\{a,b\}} \geq v_i^{\{a\}} + v_i^{\{b\}}$. Especially, when items a and b are independent, are also be viewed as complementary because $v_i^{\{a,b\}} = v_i^{\{a\}} + v_i^{\{b\}}$.

The different valuations for various items lie on user's utilities. A user is satisfactory with the allocated resources, which is referred to as the utility. Because of the complementarity and substitutability, a mobile user's total utilities do not always equal to the sum of his/her utility of each commodity. An efficient auction mechanism should maximize buyer's utilities and seller's payoffs, so does our MCC mechanism. Thus we formulate the user utility functions as follow:

$$U_i(S) = v_i^S - \sum_{r \in S} P_i^r \tag{1}$$

where $U_i(S)$ is the utility of user i on the commodity set S, v_i^S is the valuation of S, and P_i^r is the transaction price on which user i gets the item r. The utility function can reveal the complementarity and substitutability because user's valuations can express them, i.e.,

$$U_i(\{a,b\}) = \begin{cases} U_i(\{a\}) + U_i(\{b\}) + h_i \\ \quad \textit{for a, b is complementary} \\ U_i(\{a\}) + U_i(\{b\}) - l_i \\ \quad \textit{for a, b is substitutable} \end{cases} \tag{2}$$

In (2), $h_i \geq 0$ and $l_i \geq 0$ can be viewed as the premium and discount in one auction respectively. From the standpoint of the buyer, if he/she can buy two complementary items a and b in one auction successfully, he/she is willing to pay more. However, two substitutable items are bought at one time only when he/she can get a discount.

To express such heterogenous user demands in CAs, many bidding languages were brought up, which are meant to provide the syntax for encoding bid's information in a succinct and simple manner. Similar to any language, there is a trade-off between expressiveness and simplicity. In the next subsection, we review these bidding languages and put forward our novel language to express heterogeneous demands of mobile users in MCC markets.

B. Semantics of Bidding Languages

Bidding languages basically try to efficiently model different bidding patterns. The most common method is the single-minded bidding language, or called atomic bidding language. It can only describe user demands as follow: a user i chooses $\boldsymbol{S}$, a subset of available items $\boldsymbol{R}$, for valuation v_i^S [19].

Obviously, the single-minded bids are not expressive enough to distinguish complementarity and substitutability, so the OR, XOR, and other bidding language are put forward. In OR, bundle-value pairs are ORed together, and any number of these pairs may be accepted in an auction. For example, $(\{a\}, 3)OR(\{b, c\}, 4)$ implies a value of 3 for $\{a\}$ and a value of 7 for $\{a, b, c\}$. OR is good for expressing complementarity, but bad for expressing substitutability. XOR can express any valuation function, which simply XOR together all bundle-value pairs. It means that only one of the bundle-value pairs can be accepted in an auction. For example, $(\{a\}, 3)XOR(\{b, c\}, 4)$ implies a value of 3 for $\{a\}$ and a value of 4 for $\{a, b, c\}$. While XOR is more expressive than OR, there are valuations that can be specified more succinctly by OR. However, they sometimes are not very concise, therefore some solutions try to combine OR and XOR to get benefits of the both. The result introduces new languages: OR-of-XORs, XOR-of-ORs, and the logical language [20].

Often each language is good in expressing some patterns and weak or unable in expressing some other patterns. While certain languages can be compared based on their expressivity, it is not always possible to accurately compare two bidding languages. However, the complicated bidding languages are obviously more efficient to express various combinatorial bids than the simple languages (atomics, OR and XOR), while the former are more expensive on computing costs of WDP than the latter.

If our online MCC auction platform adopts a complicated bidding language, although it allows mobile users to submit many kinds of combinatorial bids, the performance of the platform will still be good. Because in MCC markets, mobile users are non-professional traders, they often cannot design multiple combinations of various bids. Furthermore, if the auction platforms are efficient enough to implement many transactions immediately at the end of bidding period with acceptable costs, the auctions can be held frequently and the users do not need to bid many goods in one auction. Therefore, our novel bidding language $\mathcal{L}_{MU}$ restricts the kinds of combinations that bidders may bid on, which not only transmits this bidding function in a succinct way to the platform but also reduces computational complexity.

The semantics of an $\mathcal{L}_{MU}$ bid can be expressed in Backus-Naur Form (BNF) as follows:

$$BID ::= (Comb\_Bid)|(Comb\_Bid)^{\leq n}$$
$$Comb\_Bid ::= Atom\_Bid|Atom\_Bid{\rightarrow}Atom\_Bid$$
$$Atom\_Bid ::= \langle S, v^S \rangle$$

Therefore, an $\mathcal{L}_{MU}$ bid can be one of four forms:

1) **An atomic bid**, $(\langle S, v^S \rangle)$, means a user bids a set of commodities S ($S \subseteq R$) with the valuation v^S ($v^S \in \mathcal{N}$, and $v^S/|S| \geq b_{min}$). It is same as a single-minded language, which can express complementarity.
2) **A combinatorial bid**, i.e., two atomic bids joined by a binary operator $\rightarrow$, is denoted as $(\langle S_1, v^{S_1} \rangle \rightarrow \langle S_2, v^{S_2} \rangle)$, where $S_1 \subset R$, $S_2 \subset R$, and $S_1 \bigcap S_2 = \phi$. When a mobile user wants to bid substitutable goods, he/she can express in the form. The equivalent representation of this form in our $\mathcal{L}_{MU}$ and XOR language are:
$$(\langle S_1, v^{S_1} \rangle \rightarrow \langle S_2, v^{S_2} \rangle) \Longleftrightarrow (\langle S_1, v^{S_1} \rangle XOR \langle S_2 \cup S_1, v^{S_2} \rangle) \quad (3)$$
The user may be allocated S_1 or $S_2 \cup S_1$, but the two cannot appear simultaneously.
3) **An atomic bid with quantity range**, $(\langle S, v^S \rangle)^{\leq n}$, means a user wants to buy the atomic bid up to n units ($n \in N$, and $n > 1$).
4) **A combinatorial bid with quantity range**, $(\langle S_1, v^{S_1} \rangle \rightarrow \langle S_2, v^{S_2} \rangle)^{\leq n}$, means a user wants to buy the combinatorial bid up to n units ($n \in N$, and $n > 1$).

The first two forms are suitable for mobile users who just buy one unit of each type commodity, and the latter two allow users to buy n copies of the same bid. Let *Bid* is one atomic or combinatorial bid in $\mathcal{L}_{MU}$, multi-units of *Bid* can be represented by **OR** as follows:

$$Bid^{\leq n} \Longleftrightarrow \underbrace{(Bid\, \boldsymbol{OR}\, Bid\, \boldsymbol{OR} \ldots \boldsymbol{OR}\, Bid)}_{n} \quad (4)$$

The user can get one group of goods described in *Bid*, or *2* groups, at most n groups.

V. The Winner Determination Problem

The problem of identifying which set of bids to accept has usually been dubbed the WDP, or the combinatorial allocation problem (CAP), which is a computational problem of how to efficiently determine the item allocation once the bids and asks have been submitted to the auction platform. The efficiency of an auction mechanism depends on the WDP model and its algorithm.

While the general WDP model of a single-sided combinatorial auction has been proved to be **NP-hard** [21], our solution is a many-to-many auction mechanism. Obviously, the objective of such double auctions should be maximizing the total surpluses of all traders, including buyers and sellers. Therefore, the WDP of our auction mechanism is formulated as an optimization problem to maximize the **social welfare**, i.e., the total payoffs/utilities of the users and providers.

In our combinatorial double auction, there are set $\boldsymbol{R}$ of commodities, set $\boldsymbol{I}$ of mobile users, and set $\boldsymbol{J}$ of MCC providers. Given set $\boldsymbol{B} = \{B_1, \ldots, B_i, \ldots, B_{|I|}\}$ of bids submitted by all users, and set $\boldsymbol{A} = \{A_1, \ldots, A_j, \ldots, A_{|J|}\}$ of asks offered by all providers, find an allocation of goods to users, which maximizes the social welfare. To formulate a feasible WDP model for our MCC combinatorial double auction, the bids and asks need to be preprocessed.

A. Preprocessing of Bids and Asks

$\mathcal{L}_{MU}$ enables every user to submit one of four types bids (an atomic bid, an atomic bid with quantity range, a combinatorial bids and a combinatorial bid with quantity range). To simplify WDP, the **dummy goods** and **sub-users** are introduced to transform various bids into one format: the one-unit atomic bid. The bid transformation can be conducted according to the following theorems:

Theorem 1: Any atomic bid with quantity range submitted by a user, $B_i = (\langle S, v^S \rangle)^{\leq n}$, can be transformed to n atomic bids $(\langle S, v^S \rangle)$. Suppose they are submitted by n sub-users. The solution to the origin WDP can be obtained by the solution to the new WDP.

Theorem 2: Any combinatorial bid, $B_i = (\langle S_1, v^{S_1} \rangle \rightarrow \langle S_2, v^{S_2} \rangle)$, can be transformed to 2 atomic bids by introducing a dummy good ($dummy_i$) and 2 sub-users (su_i^1, su_i^2). Suppose su_i^1 submits $(\langle S_1 \cup dummy_i, v^{S_1} \rangle)$ and su_i^2 submits $(\langle S_1 \cup S_2 \cup dummy_i, v^{S_2} \rangle)$. The solution to the origin WDP can be obtained by the solution to the new WDP.

Theorem 3: Any combinatorial bid with quantity range, $B_i = (\langle S_1, v^{S_1} \rangle \rightarrow \langle S_2, v^{S_2} \rangle)^{\leq n}$, can be transformed to $2 \times n$ atomic bids. The solution to the origin WDP can be obtained by the solution to the new WDP.

In the same way, **sub-providers** are also introduced to simplify the asks offered by the MCC providers. The transformation of asks can be conducted according to the following theorem:

Theorem 4: Any ask offering more than one type of goods, $(\langle r_1, c_j^{r_1}, q_j^{r_1} \rangle, \langle r_2, c_j^{r_2}, q_j^{r_2} \rangle, \ldots, \langle r_m, c_j^{r_m}, q_j^{r_m} \rangle)$, can be transformed to m simple asks $(\langle r_m, c_j^{r_m}, q_j^{r_m} \rangle)$. Suppose they are submitted by m sub-providers. The solution to the origin WDP can be obtained by the solution to the new WDP.

The preprocessing is shown in Algorithm 1.

Algorithm 1 Preprocessing of bids and asks

Input:
1) The set $\boldsymbol{R}$; the set $\boldsymbol{I}$; the set $\boldsymbol{J}$;
2) The set $\boldsymbol{B} = \{B_1, \ldots, B_i, \ldots, B_{|I|}\}$ of bids, the set $\boldsymbol{A} = \{A_1, \ldots, A_j, \ldots, A_{|J|}\}$ of asks

Output:
The set $\hat{\boldsymbol{B}}$ of atomic bids, the set $\hat{\boldsymbol{A}}$ of simple asks, the set $\hat{\boldsymbol{R}}$ of goods and dummy goods, the set $\hat{\boldsymbol{I}}$ of users and sub-users, the set $\hat{\boldsymbol{J}}$ of providers, dummy provider and sub-providers,.

```
Initialization of Î,Ĵ,B̂,Â = ∅, R̂ = R
for all B_i ∈ B do
  if B_i is not an atomic bid then
    Transform B_i to a groups of atomic bids S_b
    B̂ = B̂ ∪ S_b, Î = Î ∪ {subusers}
    for all dummy_i do
      Ĵ = Ĵ ∪ {dp_i}, Â = Â ∪ {(⟨dummy_i, 0, 1⟩)}
      R̂ = R̂ ∪ {dummy_i}
    end for
  else
    B̂ = B̂ ∪ B_i, Î = Î ∪ {user_i}
  end if
end for
for all A_j ∈ A do
  if |A_j| ≥ 1 then
    Transform A_j to a groups of simple asks S_a
    Â = Â ∪ S_a, Ĵ = Ĵ ∪ {subproviders}
  else
    Â = Â ∪ A_j, Ĵ = Ĵ ∪ {provider_j}
  end if
end for
```

B. The WDP Model

As the original bids and asks are processed in our MCC combinatorial double auction, the WDP model can be formulated as follows:

$$\begin{aligned} &\max(\sum_{i \in \hat{I}} x_i U_i(S_i) + \sum_{j \in \hat{J}} y_j W_j(r_j)) \\ &s.t. \sum_{i \in \hat{I}, r \in \hat{B}_i(1)} x_i = \sum_{j \in \hat{J}, r = \hat{A}_j(1)} y_j \quad \forall r \in \hat{R} \\ &\quad y_j \in \{0, 1, \ldots, q_j\} \quad \forall j \in \hat{J} \\ &\quad x_i \in \{0, 1\} \quad \forall i \in \hat{I} \end{aligned} \tag{5}$$

where $\boldsymbol{x_i}$ denotes whether the buyer i trades in the allocation, and $\boldsymbol{y_j}$ denotes the transaction quantity of the seller j. The variables $(x_i, y_j), i \in \hat{I}, j \in \hat{J}$ specify the auction result. By mapping sub-users and sub-providers to origin mobile users and MCC providers respectively, the MCC resource allocation is acquired.

The object of (6) is to maximize the total utilities of the mobile users and MCC providers, i.e., social welfare, denoted as $Z(\boldsymbol{x}, \boldsymbol{y})$. An auction mechanism is efficient if the allocation maximizes social welfare. $U_i(S)$ is the utility function of the buyer i, which has been defined in (1). $W_j(r)$ is the surplus function of the seller j, and is formulated as follow:

$$W_j(r) = P_j^r - c_j^r \tag{6}$$

where P_j^r is the transaction price on which the seller j sells the item r, and c_j^r is the offered price submitted by the seller j. The seller j can obtain the surplus $W_j(r)$ by selling one unit of commodity r.

Therefore, the object of (6) can be presented as:

$$Z(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i \in \hat{I}} x_i (v_i - \sum_{r \in S_i} P_i^r) + \sum_{j \in \hat{J}} y_j (P_j - c_j) \tag{7}$$

Because

$$\sum_{i \in \hat{I}} x_i \sum_{r \in S_i} P_i^r = \sum_{j \in \hat{J}} y_j P_j \tag{8}$$

we have

$$Z(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i \in \hat{I}} x_i v_i - \sum_{j \in \hat{J}} y_j c_j \tag{9}$$

Therefore, our WDP problem can be solved by the following integer program:

$$(IP)\ z_{IP} = \max(\sum_{i\in\hat{I}} v_i x_i - \sum_{j\in\hat{J}} c_j y_j)$$
$$s.t. \sum_{i\in\hat{I}} b_{ri}x_i - \sum_{j\in\hat{J}} a_{rj}y_j = 0 \quad \forall r \in \hat{R}$$
$$y_j \in \{0,1,\ldots,q_j\} \quad \forall j \in \hat{J}$$
$$x_i \in \{0,1\} \quad \forall i \in \hat{I} \tag{10}$$

In the next subsection, we design the decomposition algorithm to relax the problem $\mathcal{P}$ to a linear formulation, and bring up a pricing mechanism to decide transaction prices.

C. The Decomposition Algorithm And Pricing Mechanism

The optimization problem IP is also NP-hard because it is a special case of the general WDP problem defined in (5), which has been proved to be NP-hard [21].

Two approaches are used to find the optimal solution to the general WDP of the single-sided combinatorial auctions, shown in (5). The first one is the exact method, which replaces the given problem by one with a larger feasible region that is more easily solved. The upper bound on the optimal solution value is obtained by solving a relaxation of the optimization problem [21]. The second approach is to conduct one of the standard Artificial Intelligence (AI) searches over all the possible allocations, given the bids submitted [22]. Several algorithms with satisfactory performance for problem sizes and structures occurred in practice have been developed. However, because of the wide applicability of combinatorial auctions, one cannot hope for a general-purpose algorithm that can efficiently solve every instance of this problem.

To design a computationally efficient algorithm for our combinatorial double auction problem, we first decompose IP. Our decomposition algorithm reformulates the problem to a linear programming problem, which can then be solved in polynomial time with a subgradient algorithm. Then we rely on the solution to the linear dual problem and use its optimal value.

We adopt the Lagrangean relaxation to relax the first constraint of our original problem IP by moving it into the objective function with a penalty term. Then we get the Lagrangean relaxation problem LR:

$$(LR)\ \ z_{LR}(\boldsymbol{\lambda}) = \max L(\boldsymbol{x},\boldsymbol{y};\boldsymbol{\lambda})$$
$$s.t.\ 0 \le y_j \le q_j \quad \forall j \in \hat{J}$$
$$0 \le x_i \le 1 \quad \forall i \in \hat{I} \tag{11}$$

$L(\boldsymbol{x},\boldsymbol{y};\boldsymbol{\lambda})$ is the Lagrangean function and be defined as:

$$L(\boldsymbol{x},\boldsymbol{y};\boldsymbol{\lambda}) = \sum_{i\in\hat{I}} v_i x_i - \sum_{j\in\hat{J}} c_j y_j + \sum_{r\in\hat{R}} \lambda_r (\sum_{j\in\hat{J}} a_{rj}y_j - \sum_{i\in\hat{I}} b_{ri}x_i) \tag{12}$$

and $\boldsymbol{\lambda}$ is a vector of Lagrangean multipliers, $\boldsymbol{\lambda} = (\lambda_1,\ldots,\lambda_r,\ldots,\lambda_{|\hat{R}|})$.

Therefore, we get the Lagrangen dual problem LD of the primal problem:

$$(LD)\ \ z_{LD} = \min z_{LR}(\boldsymbol{\lambda})$$
$$s.t.\ \lambda_r \ge 0 \quad \forall r \in \hat{R} \tag{13}$$

Computing z_{LD} is easy, since there are many subgradient algorithms for the Lagrangean relaxation. Our problem can be deemed as a case of the Traveling Salesman Problem (TSP), and then the subgradient algorithm is applied. A subgradient of the Lagrangean function $L(\boldsymbol{x},\boldsymbol{y};\boldsymbol{\lambda})$ is defined as:

$$g = \partial L(\boldsymbol{x},\boldsymbol{y};\boldsymbol{\lambda})/\partial\boldsymbol{\lambda} \tag{14}$$

Iterate $\boldsymbol{\lambda}^{(k)}$ is generated according to the update recursion:

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + t^{(k)}g^{(k)} \tag{15}$$

where $t^{(k)}$ being a scalar representing the step size and $g^{(k)}$ a subgradient of the function $L(\boldsymbol{x},\boldsymbol{y};\boldsymbol{\lambda})$ at the point $\boldsymbol{\lambda}^{(k)}$.

The key point is to decide transaction prices. The constraint $\sum_{i\in\hat{I}} b_{ri}x_i - \sum_{j\in\hat{J}} a_{rj}y_j = 0(\forall r \in \hat{R})$ restricts that the total demand of all mobile users is equal to the total supply of all providers. Because the vector of the Lagrangean multipliers relaxes it, the $\boldsymbol{\lambda}$ can be interpreted as a price vector. When the Lagrangen dual problem LD is solved, the optimal vector $\boldsymbol{\lambda} = (\lambda_1,\ldots,\lambda_r,\ldots,\lambda_{|\hat{R}|})$ is also obtained. Therefore, the transaction prices of the commodities are decided, and λ_r is the transaction price of the commodity r. In our MCC combinatorial double auction, one type of goods only has one price. The trade prices of the dummy goods all equal to 0.

VI. Simulation and Evaluation

We consider a simulation scenario with a set $\boldsymbol{R}$ of commodities, a set $\boldsymbol{I}$ of mobile users and a set $\boldsymbol{J}$ of MCC providers. Suppose each user only submits an atomic bid $(\langle S, v^S\rangle)$, and each provider only offers an ask for one type of goods $(\langle r,c,q\rangle)$. The bundle S_i that user i bids is selected from the $2^{|\boldsymbol{R}|}-1$ subsets of $\boldsymbol{R}$ randomly. User values, provider's offered prices and quantities are random numbers.

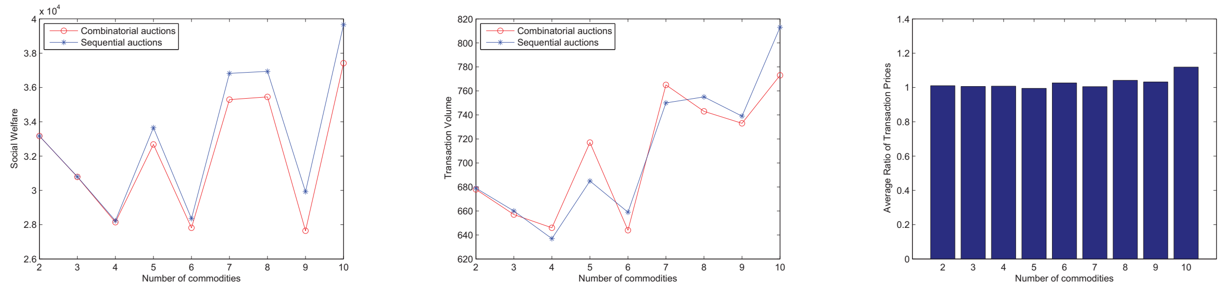
To compare the allocation performance of the proposed combinatorial double auction with the single-minded auctions, we can construct $|\boldsymbol{R}|$ sequential auctions $a_1,\ldots,a_r,\ldots,a_{|\boldsymbol{R}|}$, where the $r-th$ auction sells the commodity r. If user i submits $B_i = (\langle S_i, v_i^S\rangle), S_i = \{r_l, r_m, r_n\}$, the bid can be divided into *3* bids for the single-minded auctions sa_l, sa_m and sa_n, and the value for each bid is set as $v_i^S/3$. We evaluate the following three criterions: The **Social Welfare**, E_s; The **Transaction Volume**, E_v; The **Average Ratio of Transaction Prices**, α.

In order to provide optimal solutions to each single-minded auction a_r, we use the Marshallian path to match bids and asks [23].

Figures 3 and 4 give the comparisons of the combinatorial auctions and sequential auction on social welfare, transaction volume and transaction price in two scenarios on different scales. The amount of users and that of providers are fixed, while the number of commodities is on the increase. In Fig. 3, $|\boldsymbol{I}| = 2000, |\boldsymbol{J}| = 100$ and $|\boldsymbol{R}| = 2,3,\ldots 10$. While in Fig. 4, $|\boldsymbol{I}| = 5000, |\boldsymbol{J}| = 200$ and $|\boldsymbol{R}| = 2,3,\ldots 15$.

Comparing the simulation results, we can observe that as the number of commodities increases, the allocation performance of combinatorial auctions is always close to the optimal sequential auctions. Furthermore, the transaction prices are stable.

To evaluate the computational efficiency of our mechanism, we analyze how many iterations are executed in simulation markets on different scales, as shown in Fig. 5. The number of commodities is fixed, $|\boldsymbol{R}| = 10$. In Fig. 5a, the amount of

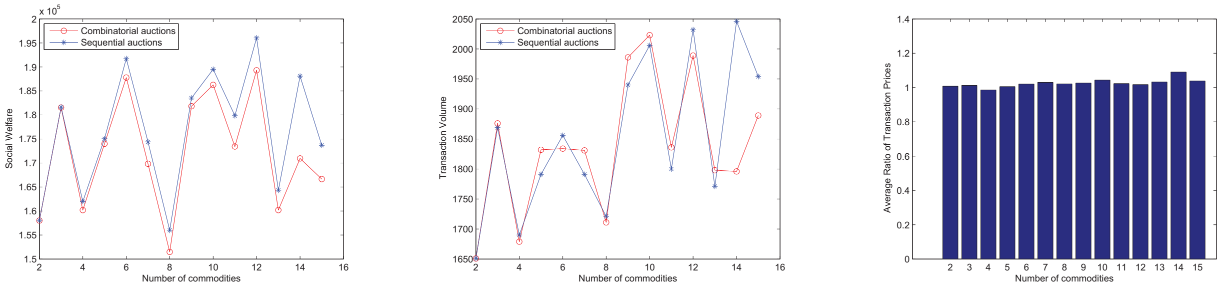


(a) Social welfare of combinatorial auctions compared with sequential auctions

(b) Transaction volume of combinatorial auctions compared with sequential auctions

(c) Ratio of transaction prices

Fig. 3. Allocation performance of the proposed MCC combinatorial double auction mechanism (Scenario 1)

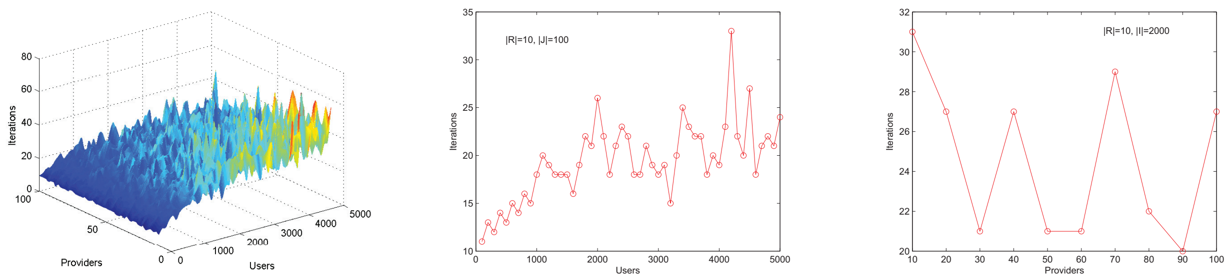


(a) Social welfare of combinatorial auctions compared with sequential auctions

(b) Transaction volume of combinatorial auctions compared with sequential auctions

(c) Ratio of transaction prices

Fig. 4. Allocation performance of the proposed MCC combinatorial double auction mechanism (Scenario 2)



(a) Iterations with different amount of users and providers

(b) Iterations with different amount of users

(c) Iterations with different amount of providers

Fig. 5. Simulation results: computational efficiency (iterations executed to get the optimal solution) of the proposed MCC combinatorial double auction mechanism

users and that of providers are both on the increase. Figure 5b gives the iteration curve with different amount of users, when the number of providers is also fixed, $|\boldsymbol{J}| = 100$. Figure 5c shows iterations varying with the amount of providers with fixed numbers of commodities and users. The results prove that the algorithm is feasible.

Overall, from the simulation results we can conclude that: first, allocation efficiency of our approach is high because the social welfare and transaction volume of our approach are close to the social optimal solution and transaction prices are stable. Second, our WDP algorithm is convergent and can obtain the optimal results with acceptable iterations.

VII. Conclusion

In this paper, we apply the combinatorial double auction mechanism in MCC resource allocation and design an online auction framework to implement the mechanism. It enables mobile users to bid bundles of cloud services at one auction. For the consideration of facilitating mobile users, a novel bidding language is designed to express user's valuations concisely. Then a model of WDP for our auction mechanism is formulated, which can determine winners and prices of each auction in affordable time. The experiment results show that the allocation performance of our solution is very close to the social optimal allocation and the computational cost is also feasible.

REFERENCES

[1] R. Buyya, R. Ranjan, and R. N. Calheiros, "Intercloud: Utility-oriented federation of cloud computing environments for scaling of application services," in *Algorithms and architectures for parallel processing*. Springer, 2010, pp. 13–31.

[2] C. Lee, P. Wang, and D. Niyato, "A real-time group auction system for efficient allocation of cloud internet applications," 2013.

[3] H. Zhang, B. Li, H. Jiang, F. Liu, A. V. Vasilakos, and J. Liu, "A framework for truthful online auctions in cloud computing with heterogeneous user demands," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 1510–1518.

[4] X. Shi, K. Xu, J. Liu, and Y. Wang, "Continuous double auction mechanism and bidding strategies in cloud computing markets," *arXiv preprint arXiv:1307.6066*, 2013.

[5] S. Zaman and D. Grosu, "Combinatorial auction-based mechanisms for vm provisioning and allocation in clouds," in *Cluster, Cloud and Grid Computing (CCGrid), 2012 12th IEEE/ACM International Symposium on*. IEEE, 2012, pp. 729–734.

[6] S. Perez, "Mobile cloud computing: $9.5 billion by 2014," http://www.juniperresearch.com/, 2010.

[7] W. Shi, L. Zhang, C. Wu, Z. Li, and F. C. Lau, "An online auction framework for dynamic resource provisioning in cloud computing," in *Proceedings of the ACM SIGMETRICS/international conference on Measurement and modeling of computer systems*. ACM, 2014, pp. 71–83.

[8] L. Y. Chu, "Truthful bundle/multiunit double auctions," *Management Science*, vol. 55, no. 7, pp. 1184–1198, 2009.

[9] P. Vytelingum, D. Cliff, and N. R. Jennings, "Strategic bidding in continuous double auctions," *Artificial Intelligence*, vol. 172, no. 14, pp. 1700–1729, 2008.

[10] I. SUTHERLAiND, "A futures market in computer time," 1968.

[11] R. Buyya, D. Abramson, J. Giddy, and H. Stockinger, "Economic models for resource management and scheduling in grid computing," *Concurrency and computation: practice and experience*, vol. 14, no. 13-15, pp. 1507–1542, 2002.

[12] R. Wolski, J. S. Plank, J. Brevik, and T. Bryan, "Analyzing market-based resource allocation strategies for the computational grid," *International Journal of High Performance Computing Applications*, vol. 15, no. 3, pp. 258–281, 2001.

[13] G. Vinu Prasad, S. Rao, and A. S. Prasad, "A combinatorial auction mechanism for multiple resource procurement in cloud computing."

[14] D. Niyato, Y. Zhang, and P. Wang, "An auction mechanism for resource allocation in mobile cloud computing systems." 2013.

[15] N. Fernando, S. W. Loke, and W. Rahayu, "Mobile cloud computing: A survey," *Future Generation Computer Systems*, vol. 29, no. 1, pp. 84–106, 2013.

[16] S. Ostermann, A. Iosup, N. Yigitbasi, R. Prodan, T. Fahringer, and D. Epema, "A performance analysis of ec2 cloud computing services for scientific computing," in *Cloud Computing*. Springer, 2010, pp. 115–131.

[17] I. Menache, A. Ozdaglar, and N. Shimkin, "Socially optimal pricing of cloud computing resources," in *Proceedings of the 5th International ICST Conference on Performance Evaluation Methodologies and Tools*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2011, pp. 322–331.

[18] X. Ke, Z. Yuchao, S. Xuelin, W. Haiyang, W. Yong, and S. Meng, "Online combinatorial double auction for mobile cloud computing markets," http://www.thucsnet.org/uploads/2/5/2/8/25289795/online.pdf, 2014.

[19] D. Lehmann, L. I. Oćallaghan, and Y. Shoham, "Truth revelation in approximately efficient combinatorial auctions," *Journal of the ACM (JACM)*, vol. 49, no. 5, pp. 577–602, 2002.

[20] C. Boutilier and H. H. Hoos, "Bidding languages for combinatorial auctions," in *International Joint Conference on Artificial Intelligence*, vol. 17, no. 1. LAWRENCE ERLBAUM ASSOCIATES LTD, 2001, pp. 1211–1217.

[21] T. Sandholm, "Approaches to winner determination in combinatorial auctions," *Decision Support Systems*, vol. 28, no. 1, pp. 165–176, 2000.

[22] T. Sandholm and S. Suri, "Bob: Improved winner determination in combinatorial auctions and generalizations," *Artificial Intelligence*, vol. 145, no. 1, pp. 33–58, 2003.

[23] P. J. Brewer, M. Huang, B. Nelson, and C. R. Plott, "On the behavioral foundations of the law of supply and demand: Human convergence and robot randomness," *Experimental economics*, vol. 5, no. 3, pp. 179–208,