Received: 4 April 2017 | Revised: 25 August 2017 | Accepted: 4 October 2017
DOI: 10.1002/dac.3460

SPECIAL ISSUE - COMMUNICATION AND COMPUTATION COOPERATION: PRINCIPLES, ALGORITHMS AND SYSTEMS

WILEY

Design, modeling, and analysis of online combinatorial double auction for mobile cloud computing markets

Yuchao Zhang[1] | Ke Xu[2] | Xuelin Shi[2] | Haiyang Wang[3] | Jiangchuan Liu[4] | Yong Wang[5]

[1]School of Software, Beijing University of Posts and Telecommunications, Beijing, China

[2]Computer Science and Technology, Tsinghua University, Beijing, China

[3]Computer Science and Technology, University of Minnesota at Duluth, Duluth, Minnesota, USA

[4]College of Natural Resources and Environment, South China Agricultural University, Guangzhou, Guangdong, China

[5]School of Social Sciences, Tsinghua University, Beijing, China

Correspondence
Ke Xu, Room 4104, FIT Building, Tsinghua University, Beijing, China.
Email: xuke@mail.tsinghua.edu.cn

Funding information
National Natural Foundation of China, Grant/Award Number: 61472212; National Science and Technology Major Project of China, Grant/Award Number: 2015ZX03003004; National High Technology Research and Development Program of China (863 Program), Grant/Award Number: 2013AA013302 and 2015AA015601; EU Marie Curie Actions CROWN, Grant/Award Number: FP7-PEOPLE-2013-IRSES-610524; National Natural Science Foundation of China, Grant/Award Number: U1301253; Science and Technology Planning Project of Guangdong Province, China, Grant/Award Number: 2015B010110006

Summary

With the burgeoning of cloud service companies, cloud computing is becoming an efficient means of providing computing resources. Amazon EC2, Rackspace, Google App, and Microsoft Azure are attracting more and more users over the Internet these years. However, in mobile cloud computing (MCC), traditional cloud pricing models can no longer support the above popular applications, because user behaviors are dynamic and time sensitive. As an MCC application is the combination of communication services (eg, wireless access services) and computation services (eg, cloud services), it lacks new auctions for capturing the feature of MCC markets based on the communication and computation cooperation (3C).

In this paper, we design an efficient double-sided combinatorial auction model in the context of 3C-based MCC to mitigate this problem. We first propose the framework of online combinatorial double auctions to model mobile cloud computing market. On this base, we give four principles of design requirements, which can make the scheme more efficient and practical, and then we design a new winner determination algorithm that shows how the auction mechanism decides commodity allocation and transaction prices. At last, we conduct a series of experiments to deep analyze the property of our mechanism. The experiment results indicate that the proposed online auction mechanism obtains comparable allocation efficiency to the social optimal solution.

KEYWORDS

combinatorial double auction, mobile cloud computing, pricing model, resource allocation

1 | INTRODUCTION

Being able to leverage on-demand accesses to computing infrastructures, cloud computing is emerging as a promising resource sharing platform. While Internet cloud services are burgeoning, the existing cloud markets develop extremely slowly with respect to the pricing mechanism. For example, Amazon EC2[1] charges \$0.03 to \$0.12

 Copyright © 2017 John Wiley & Sons, Ltd.

per hour for each virtual machine (VM) instance, and the prices are fixed. When the price of commodity has a well-known value, which is common knowledge to both sellers and buyers, such a posted-offer pricing model is commonly used. So a buyer acts simply as a price taker who has just 2 choices: accept the price or not. As a price taker, a buyer cannot affect the price of the commodity. Although such fixed pricing schemes are acceptable to small enterprises and markets, they cannot be adopted in dynamic cloud computing markets because of the dynamism and diversity.

To mitigate this problem, auction-based instances are emerging in cloud markets. Such *Spot Instances* brings more freedom to users, because this scheme allows customers to bid for unused resources and to run instances whenever their bids exceed the current price. Many different auction mechanisms are therefore proposed to make resource allocation and pricing policies in cloud markets.[2-4] But in mobile cloud computing (MCC) market, where services are provided through the cooperation between communication resources (eg, network bandwidth and wireless spectrum) and computation resources (eg, CPU and GPU), these traditional single-sided auction models are no longer efficient for such cloud applications. In particular, Sharrukh et al[5] have proved that auctions enjoy distinct advantages over other schemes when items are complementary. In other words, the auctioned items will have a higher value as a set (eg, sell multiresources as a set) than as separate parts. It is known that the usage of mobile computing is increasing rapidly. Many researchers[6-8] point out that both the consumer and enterprise markets for mobile cloud computing applications have become abundant in many industries, such as health, medicine, business, social networking, travel, entertainment, and news. With both communication and computation resource requirements of these cloud-based mobile applications, it is in urgent need to develop a smarter auction model to meet such increasing demands in MCC markets based on the communication and computation cooperation (3C).

In this paper, we introduce a double-sided combinatorial auction model for the MCC market. In particular, we design a framework of online combinatorial double auctions and a winner determination problem (WDP) model to provide better services to both MCC applications and users. To solve this problem, we develop a decomposition algorithm, which can calculate item prices for winners in each auction. Moreover, a bidding language for mobile users to express valuations is also investigated. Extensive experiment results show that the proposed online auction mechanism successfully obtains high allocation efficiency, and are even comparable to the optimal solution.

The rest of this paper is structured as follows. Section 2 reviews the related works, such as combinatorial and double auctions. Section 3 proposes the framework of the combinatorial double auction in MCC. Section 4 describes our novel bidding language $\mathcal{L}_{MU}$ for mobile users. The model and algorithm of WDP for our auction mechanism are introduced in Section 5. Then the evaluation results are given in Section 6. Finally, Section 7 concludes the paper.

2 | RELATED WORK

As cloud computing is designed to be a market-oriented computing paradigm, resource allocation and pricing are always hot topics. Bidding and auctions* are deemed to be effective solutions to grid and computing resource markets.[9,10] Meanwhile, MCC is a representative example of cloud services, thus has natural advantages to adopt auctions. In this section, we first present some related works on combinatorial and double auctions[11,12] and then review the auctions about resource allocation and pricing.

2.1 | Combinatorial and double auctions

Based on the supply and demand in cloud markets, auction is one of the most effective and economic ways to set commodity price. The auction model supports different negotiation models between sellers and buyers, like one-to-many (for instance, single-sided auction) or many-to-many (eg, double auction). Auctions have multiple flexible form; players are allowed to bid for one item or sets of items once a time. Therefore, the design of auction mechanism is a hot spot in microeconomics.

Combinatorial auctions allow bids for bundles of items, providing a great way to allocate multiple distinguishable items among bidders.[13] Combinatorial auctions can make bidders flexibly reveal their preferences on the replaceable or complementary relationships of items, which can decline the bidding risk, increase the revenue, and thus improve the economic efficiency of the auctions remarkably. The main topics concerned in combinatorial auctions are bidding languages, winner determination, and mechanism design.

In recent years, combinatorial auctions have attracted significant interest as automated mechanisms to buy and sell bundles of scarce resources, for example, the virtual machine instances in Zaman and Grosu.[14] The development of the Internet and e-commerce has provided a

*An "auction" is a process of buying and selling goods or services. "Bidding" is an offer to set a price by an individual for a product or service. An auction consists of several stages, like offering goods (or services) up for bid, taking bids, and buying the item to the highest bidder.

wonderful platform for combinatorial auctions. Combinatorial auctions have been applied in wide economic domains successfully, including truckload transportation, industrial procurement, radio spectrum auction, and airport time slots.

Today's e-commerce combinatorial auction platforms usually only support one-to-many negotiations. In such single-sided auctions, an auction is initialized by one auctioneer, and then buyers begin to bid in the auction. For a market with limited number of buyers or sellers, single-sided auctions are well suited. But when a market consists of numerous of buyers and sellers, these mechanisms are usually noneffective. That is because there are different items and auctions in these complex markets, buyers or sellers should bid repeatedly to maximize their profits, and they also have to consider the multiple possible outcomes of each participate auctions. This computational burden in combinatorial auctions hinders the trades for players. To relieve this computational burden, many recent researches are developing double auctions to promote transactions. [15]

Double auctions are many-to-many negotiations. In double auctions, multiple buyers and sellers can bid at the same time. Gjerstad and Dickhaut[16] give strategies for sellers and buyers in a double auction (DA) market. After that, Bredin and Parkes[17] present a general method to design truthful DAs, and the dynamic pricing rules can ensure that no agent can benefit from misreporting its arrival time, duration, or value. Along with this, DAs were applied by researchers in many areas, such as spectrum auctions,[10,18] mobile crowdsourcing,[19] mobile data offloading,[20] and many other research fields. Indeed, today's exchanges apply variants of double auctions[21] (for example, New York Stock Exchange [NYSE], NASDAQ, and the major foreign exchange).

2.2 | Auctions in MCC markets

Auction-based mechanisms have been adopted in various fields such as network bandwidth, wireless spectrum, energy industries, and advertisements, which investigate how participants behave in a competition for resources. Auctions in computing can date back to 1968 when Sutherland[22] designed a processing time allocation method using auctions. After that, many market-based resource allocation strategies have been proposed, some of which are applied to grid computing and scheduling. [9,23]

Cloud computing appeared as a more effective market-oriented computing paradigm than grid computing, so currently, researchers start to investigate the economic aspects of cloud computing policies. For example, to solve the resource allocation problem across multiple clouds, Buyya et al[2] proposed an infrastructure of federated clouds. To allocate computing resource and virtual machines, Prasad et al[24] and Zaman et al[5] adopted combinatorial auctions. Furthermore, for cloud application allocation, Lee et al[25] implemented a real-time group auction system. Besides these, Zhang et al[4] put forward an online auction framework for cloud computing, Fujiwara et al[26] proposed a market-based resource allocation mechanism that enables participants to trade future and current services in the forward market and the spot market, respectively. In our former work,[3] we have proposed a continuous double auction mechanism and a bidding strategy for cloud markets; in this scheme, both cloud users and CSPs can maximize their profits.

There are few works introducing auction-based resource allocation mechanisms in MCC markets. "An auction mechanism for resource allocation in mobile cloud computing systems" by Niyato et al[27] is one of the works, which developed an auction mechanism with discount factors for resource allocation. But our mechanism considers a combinatorial double auction mechanism, which enables mobile users and resource providers to submit bids and asks simultaneously. The advantage of this mechanism is that we can supports users to bid sets of commodities at one time, and this makes our mechanism more efficient.

3 | THE FRAMEWORK OF THE MCC COMBINATORIAL DOUBLE AUCTION

We consider a platform for the MCC markets where multiple mobile users and resource providers can buy and sell commodities in a combinatorial double auction manner. Our solution is efficient for resource allocation in MCC and appeals to mobile users and the MCC providers. On one hand, mobile users can bid bundles of applications and services on the platform with little mobile data usage. On the other hand, providers can supply sets of commodities in each auction.

3.1 | Design principles

A feasible auction model for MCC should meet 4 principles as detailed below.

Support resource combination. In wireless MCC services, services are provided by remote clouds, and mobile users access these services via wireless networks (like 2G, 3G, or WiFi), as shown in Figure 1. In this framework, base stations or WiFi access points (AP) provide radio (bandwidth) resources, and remote cloud provides computing and storage resources. Obviously, if a mobile user wants to use cloud services, he needs to buy both wireless access services and cloud resources. Therefore, in a feasible

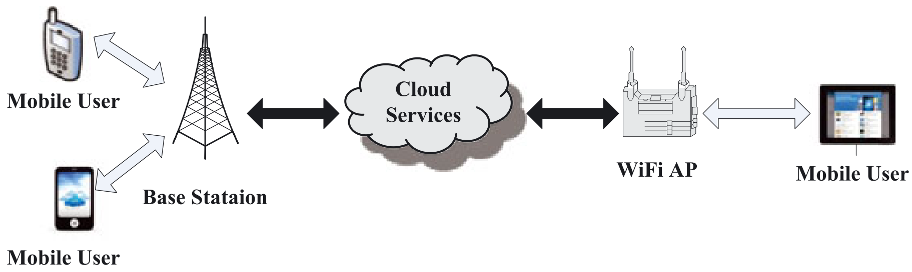


FIGURE 1 Mobile cloud computing enabling mobile users to access cloud services through wireless networks

MCC market, an effective auction mechanism should allow users to bid for sets of items (sometimes called bundles), rather than to bid different items in many sequential auctions.

Support various applications. Different from traditional content service providers, MCC providers usually offer various services besides computing and storage resources, like natural language translating, image processing, and multimedia search.[7] As mentioned previously, mobile users are attracted to buy a package of commodities together. For example, an increasing number of users prefer to take photos by using mobile devices. However, due to the limited storage space, they prefer to upload some photos to online storage servers when there are inadequate space in their mobile devices. An MCC provider P1 supplies such storage services. To get additional profits, P1 also provides an animator application and other image-processing applications. Thus, a mobile user Peter may buy 1 GB storage space and 10 animators for 1 year. Thus, when Peter takes a new photo, he can upload it to the storage servers of P1. If he wants to make an animation, he can select photos on the servers and submit them to the animator application. The application will run on the remote servers and return the result to Peter. Because there are more and more such applications in the MCC markets, the scale of one auction may be large. Consequently, an effective combinatorial auction mechanism is vital for the MCC markets, which should quickly determine the winners and prices of an auction consisting of many users and providers.

Be energy-efficient. In wired clouds, energy efficiency is not a big concern. But energy is of particular importance for MCC market, because both transmission and computation consume the limited energy of mobile devices. To attract mobile users to take part in auctions, a mechanism allows users to submit bids when transmitting data as few as possible with few computations to construct their bids. Furthermore, mobile users access online auction platforms via different wireless networks, so the less data transmission gets the lower cost. Therefore, a concise bidding language is vital for mobile users.

Be simple. In the current cloud markets, users often rent cloud resources to support their applications. [28,29] Due to the complexity of online auctions, a feasible auction model would be more acceptable to mobile users if it is very simple to use.

In summary, in such competitive MCC markets populated by mobile users and MCC providers, combinatorial auctions are feasible to solve resource allocation and pricing problems. Every mobile user, who demands sets of various commodities, can bid bundles in one combinatorial auction. Moreover, double auctions can also be adopted to improve market efficiency, in which buyers and sellers can submit bids simultaneously.

There are some significant differences between our solution and the existing mechanisms. First, we design a combinatorial double auction mechanism for the MCC resource allocation. This mechanism enables mobile users and resource providers to submit bids and asks simultaneously; thus, users can bid sets of commodities in one auction. Second, we also design a novel bidding language for mobile users.

3.2 | Framework overview

On an online auction platform, "bid" is the price a buyer's willingness to pay and "ask" is a seller's (or provider's) asking price. The platform first collects bids from mobile users and collects asks from MCC providers, then it computes the winners and the prices according to the WDP solution. Figure 2 shows the overview of the framework.

It is an electronic bidding platform, which can be easily accessed via the Internet and make use of e-commerce technologies. The platform plays the role of an auctioneer, on which mobile users can submit bids for bundles of items, while the MCC providers can submit asks. In addition, auctions are all performed in an online manner, ie, users and providers can take part in auctions whenever they need, and the platform can determine winners and price instantaneously as soon as auctions close.

The auction on the platform has 3 states: the registration stage, the bidding stage, and the winner determination stage. In the registration stage, the bulletin board presents all the information about resources, the related parameters

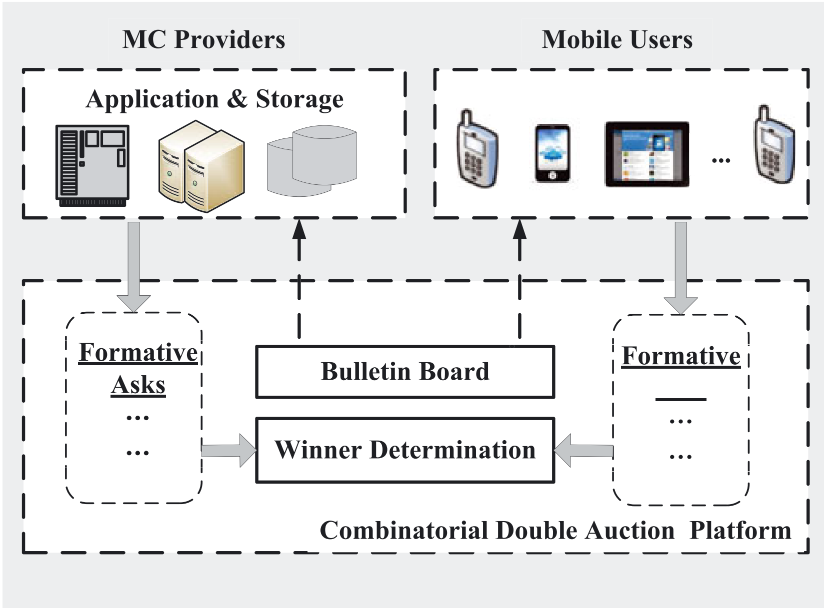


FIGURE 2 A framework of the mobile cloud computing combinatorial double auction platform

of mobile users, and the MCC providers, then every player is certified. In the bidding stage, buyers (providers) begin to submit bids (asks). At last, in the winner determination stage, the determination module calculates the winners and prices according to the combinatorial double auction mechanism.

Specifically, there are 2 main modules in this platform, one is the formative bidding language, and the other is the winner determination algorithm. In the bidding language module, the electronic bidding platform translates a user's specific demands into requests described in the formatting bidding language, the user's heterogeneous demands thus turn into regulated and consistent forms. The proposed scheme can further ensure that the details of user requirements can be revealed. So far, each request from users and each ask from MCC providers are submitted to the platform. In the winner determination module, the platform should compute winners and prices based on the auction mechanism while achieving high social welfare. At last, the bidding results are announced to users and providers, and once the charging and payment procedures are complete, the connections are established.

The waiting time of users in this process is acceptable, because the bidding platform is able to implement many transactions simultaneously, and the calculation takes not much time in each bidding period. The bidding language can transmit bidding functions in a succinct way to the platform, reducing computational complexity significantly. Also, we will evaluate the computational efficiency and the execution iterations of this scheme in the evaluation section.

Before, we describe the 2 main modules, we first give the bidding rules of the platform in the next subsection, and then through a simple example, we will show how this platform works. Later in Sections 4 and 5, we will detail the bidding language $\mathcal{L}_{MU}$ and the winners determination algorithm, respectively.

3.3 | The market rules

In the proposed MCC mechanism, the auction platform acts as a central auctioneer who receives the bids and asks, and then it performs all the computation to find the optimal allocation of items to bidders. To improve trading efficiency, the following market rules are defined.

Bidding period rule. The bidding period $\boldsymbol{t_{bp}}$ can be 1 day, 1 hour, etc. During one bidding period, players including buyers and providers submit their bids and asks, and at the end of this bidding period, the auction closes and the market clears. At an auction, only one bid or ask can be submitted by each mobile user or MCC provider. At the end of the bidding period, all bids and asks are opened. Furthermore, auction results, ie, winners and prices, are also published.

Combination rule for users. Each mobile user usually has heterogenous demands and valuations for commodities, so a bid of user i can work for bundles of items (a subset of all the available commodities) and a valuation (willingness to pay).

Combination rule for providers. Each provider has different kinds of resources to sell, so an ask of provider j can also work for multiple items. Also, the ask should contain the offered price per unit for commodity and the quantity of this commodity.

Minimum bid rule. To prevent unreasonably low bids and guarantee the trading process, we define the minimum bid allowed in the market, $\boldsymbol{B_{min}}$. It can be set according to history transaction prices or 0.

Maximum ask rule. Similarly, to prevent unreasonably high asks and ensure the trading process, we also define the maximum ask allowed in the market, $\boldsymbol{A_{max}}$. It can also be set according to history transaction records or $+\infty$.

These above rules are published in the MCC auction platform. Each user and provider who participates in auctions must obey the rules. The rules not only ensure auction efficiency but also enable users and providers to understand the auction mechanism. In addition, the records of transaction history published on the platform help users to decide their valuations on various services and applications.

3.4 | Scenario of a combinatorial double auction on the platform

With the online auction platform and bidding rules, mobile users and the MCC providers can trade by auctions. Users often have a variety of demands, and providers also supply various services and applications. For a mobile user who needs 2 applications, bidding 2 items in one combinatorial auction is more efficient than bidding twice in 2 sequential auctions. Moreover, double auctions allow users and providers to bid simultaneously in the same auction, which also improves market efficiency. Advantages of the combinatorial double auction for the MCC markets can be revealed in the scenario shown in Figure 3.

During the bidding period, both users and providers may submit bids and asks. Each user can bid one or bundles of items, while each provider can supply multiple units of commodities. In Figure 3, Tom seeks to buy 2 units of Item 1 before bidding \$4. Peter bids \$5 for one unit of Items 1 and 2. Linda bids \$8 for one unit of Item 1 and 2 units of Item 2. All the 3 users submit their bids to the online auction platform. Two MCC providers, P1 and P2, take part in the auction. P1 sells 2 units of Item 1 at price \$2. P2 sells 3 units of Item 2 at price \$3. Both providers also submit their asks to the platform.

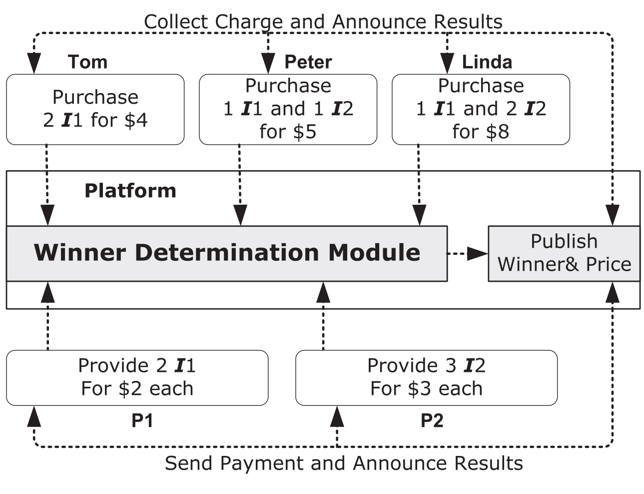


FIGURE 3 A scenario of an auction: consisting of 2 providers and 3 users bidding for 2 commodities

At the end of the bidding period, the winner determination module computes winners and prices of the auction; the algorithm of which will be described in Section 5. Then the results are announced to users and buyers. In the auction, Peter wins one unit of Item 1 and one unit of Item 2, and Linda wins one unit of \$1 and 2 units of Item 2. However, Tom loses to Peter and Linda.

While in the traditional sequential auction, there are 2 disadvantages described as follows: Firstly, users and providers cannot bid simultaneously in one auction, so if Tom wins in the Item 1 subauction, both Peter and Linda will lose the auction because they cannot get Item 1. Secondly, one can just bid for one item at one time, so the auction will be repeated for 5 times in this scenario, which reduces the market efficiency.

As shown in the above scenario, the combinatorial double auction mechanism is effective and flexible for mobile users and MCC providers. It not only ensures competitive bidders and offers simultaneously, but also allows users to bid for bundles of items at one time. However, for applying the auction mechanism to real markets, there are 2 key problems to solve: a concise bidding language for users and a feasible WDP algorithm. The solutions will be given in the following sections.

4 | THE BIDDING LANGUAGE

In our MCC auction platform, the bidding language transform module is implemented in the client side. Players use bidding language to concisely express their natural valuation functions, and bidding languages can translate users' specific demands into requests. In this section, we first analyze the heterogenous mobile user valuations and then introduce a novel bidding language $\mathcal{L}_{MU}$, which can represent various user demands. After that, we discuss the advantages and contributions of the proposed $\mathcal{L}_{MU}$ in combinatorial double auctions.

4.1 | Heterogenous mobile user valuations

In a specific combinatorial auction, let $\boldsymbol{R}$ represent the set of all goods for sale, and each buyer could have a different valuation for each subset $\boldsymbol{S}$ of $\boldsymbol{R}$. This differentiation results in $2^{|R|}-1$ possible bids because $\boldsymbol{R}$ has $2^{|R|}-1$ different subsets.

Furthermore, for a particular user, items' value may depend on each other, ie, item value to a buyer depends on whether he/she possesses another related item. There are 2 situations. (1) Items are substitutable, ie, resources (like storage) from different places are the same to users, so these resources have similar value to the user. (2) Items are complementary, ie, users prefer to get all of them or none of them. Take online photo posting as an example, wireless connection and storage resource are complementary.

The above relationship can be defined as follows.

Definition 1. Given a mobile user i, a valuation $v_i^{\{r\}}$ for a commodity r. Items a and b are substitutable if $v_i^{\{a,b\}} < v_i^{\{a\}} + v_i^{\{b\}}$, and these 2 items are complementary if $v_i^{\{a,b\}} \geq v_i^{\{a\}} + v_i^{\{b\}}$. Specifically, when $v_i^{\{a,b\}} = v_i^{\{a\}} + v_i^{\{b\}}$, these 2 independent items can also be viewed as complementary.

Different items' valuations lie on users' utilities, which denotes a user's satisfactory with the allocated resources. The total utilities of a user does not always equal to the sum of each commodity's utility due to the complementarity and substitutability. Therefore, auction mechanisms (including our MCC mechanism) always aim at maximizing buyers' utilities and sellers' payoffs. We use the following equation to formulate the user utility:

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

where S is the commodity set, $U_i(S)$ is the utility of user i, and v_i^S is the valuation of S. P_i^r is the final price of item r paid by user i. The utility function also reflects the complementarity and substitutability among items from users' valuations:

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

In (2), $h_i \geq 0$ can be viewed as the premium, and $l_i \geq 0$ means discount in one auction. From buyer's point of view, he/she would like to pay more if he/she can buy 2 complementary items in one auction. On the contrary, he/she would not buy 2 substitutable items unless there is a discount.

Such heterogenous user demands in combinatorial auctions is difficult to express; many bidding languages try to address this problem. Some encode the bid information in a succinct and simple manner. But there is always a trade-off between expressiveness and simplicity, like any other languages. In the next subsection, we will first review the existing bidding languages before introducing our novel language.

4.2 | Semantics of bidding languages

Bidding languages are used to model bidding patterns. Single-minded bidding language (or atomic bidding language) is one of the most common methods. It describes user demands only in one form: for available items $\boldsymbol{R}$, a user i chooses a subset $\boldsymbol{S}$ in valuation v_i^S. [30]

Obviously, the above single-minded bidding language cannot express complementarity and substitutability. Therefore, some other bidding languages are proposed. For example, in OR language, items are ORed together and turn to be a bundle-value pair; any pair or numbers of these pairs can be accepted in auctions. For example, $(\{x\},2)OR(\{y,z\},3)$ implies a value of 2 for $\{x\}$ and a value of 5 for $\{x,y,z\}$. Although OR is good at expressing complementarity, it cannot express substitutability. XOR is another bidding language, which can express any valuation function, but in XOR, only one bundle-value pair can be accepted. Here is another example, $(\{x\},2)XOR(\{y,z\},3)$ implies a value of 2 for $\{x\}$ and a value of 3 for $\{x,y,z\}$. While XOR is more expressive than OR, OR has its own advantages (eg, specify valuations more succinctly). Therefore, many researches try to combine OR and XOR so as to take the fully advantage of both. The representative new languages are XOR-of-ORs, OR-of-XORs, and some logical languages. [31]

As we can see, any bidding language is good at expressing just partial patterns, and it is not always possible to compare 2 bidding languages accurately due to different advantages. Generally speaking, complicated bidding languages are usually more efficient than the simple languages (atomics, OR, and XOR), because they can express various combinatorial bids. But at the same time, the complicated languages cost much than simple languages when computing WDP results.

In our online MCC auction platform, we adopt a complicated bidding language to allows mobile users to submit different kinds of combinatorial bids; meanwhile, the performance of our platform is still good. The main reasons are as follows: (1) In the MCC markets, most of the mobile users are nonprofessional traders, who cannot design various bids for multiple combinations. (2) The auction platform is efficient enough to conduct numerous transactions simultaneously; each bidding period just cost an acceptable overhead and will end up in acceptable time. Auctions can thus be held frequently. Furthermore, our proposed bidding language $\mathcal{L}_{MU}$ restricts the kinds of combinations, and this constraint largely reduces computational complexity.

In a $\mathcal{L}_{MU}$ bid, semantics can be expressed in the following Backus-Naur Form (BNF):

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

To be specific, a $\mathcal{L}_{MU}$ bid can be one of the following 4 forms according to the 3 equations:

1. **Atomic bid.** $(\langle S, v^S \rangle)$ is an atomic bid, which means to bid a set of commodities S ($S \subseteq R$) with the valuation v^S ($v^S \in \mathcal{N}$, and $v^S/|S| \geq b_{min}$). Single-minded language can also express the atomic bid. v^S is the valuation for commodity set S, which consists of $|S|$ commodities. While b_{min} is the minimum bid for a single commodity, so the minimum bid for $|S|$ commodities should be at least $|S| \times b_{min}$, ie, the value of v^S. In other words, $v^S \geq |S| \times B_{min}$. Thus, $v^S/|S| \geq b_m in$.
2. **Combinatorial bid**. $(\langle S_1, v^{S_1} \rangle \rightarrow \langle S_2, v^{S_2} \rangle)$ is a combination of 2 atomic bids by an operator $\rightarrow$, where $S_1, S_2 \subset R$ and $S_1 \cap S_2 = \phi$. This form is convenient for users to bid substitutable goods. The equivalent representation in XOR language is

$$(\langle S_1, v^{S_1} \rangle \rightarrow \langle S_2, v^{S_2} \rangle) \Longleftrightarrow (\langle S_1, v^{S_1} \rangle XOR \langle S_2 \cup S_1, v^{S_2} \rangle). \tag{3}$$

The user may obtain S_1 or $S_2 \cup S_1$, but not both of them.
3. **Atomic bid with quantity range.** $(\langle S, v^S \rangle)^{\leq n}$ denotes an atomic bid with quantity range, which means a user wants to buy at most n units of the atomic bid ($n \in N$, and $n > 1$).
4. **Combinatorial bid with quantity range.** $(\langle S_1, v^{S_1} \rangle \rightarrow \langle S_2, v^{S_2} \rangle)^{\leq n}$ is a combinatorial bid with quantity range, which means a user wants to buy at most n units of the combinatorial bid ($n \in N$, and $n > 1$).

For mobile users who want to buy one unit of each type commodity, they can use the first 2 bidding forms, and for users who want to buy n copies of the same commodity, they can use the later 2 forms. Let Bid be one bid in $\mathcal{L}_{MU}$; the equivalent representation in OR language for multiunits of Bid can be represented as follow:

$$Bid^{\leq n} \Longleftrightarrow \underbrace{(Bid \textbf{ OR } Bid \textbf{ OR OR } Bid)}_{n}. \tag{4}$$

Thus, user can get groups of commodities (up to n groups).

In Figure 3, the users Tom, Peter, and Linda submit their bids as follows: $B_T = (\langle I_1, \$2 \rangle)^{\leq 2}$, $B_P = (\langle \{I_1, I_2\}, \$5 \rangle)$ and $B_L = (\langle \{I_1, I_2, I_2\}, \$8 \rangle)$.

4.3 | Advantages over previous languages

The design for bidding languages involves a trade-off between expressiveness and simplicity, so our $\mathcal{L}_{MU}$ also needs to express heterogenous demands of mobile users in a concise way. Compared with OR, XOR, and other complicated logical bidding languages, $\mathcal{L}_{MU}$ has the following advantages.

Ease of use. The semantics of $\mathcal{L}_{MU}$ is easy to understand, and bidders can express their demands in the correct format of $\mathcal{L}_{MU}$. The general mobile users cannot handle too many logical operators, such as OR, XOR, and AND. Therefore, they prefer to submit simple bids rather than apply various logical operators to combine bids.

Considering the following scenario, there are 3 types of services to auction, ie, storage services, GIF animator services, and Flash maker services, denoted as a, b, and c, respectively. A user Peter submits an atomic bid: $B = (\langle \{a, b\}, \$5 \rangle)$, which means he wants to buy one unit of storage service and GIF animator service, which is valued complementary. If Peter expects he has many pictures to be saved and processed to GIF, he can submit $(\langle \{a, b\}, \$5 \rangle)^{\leq 3}$, which means he can get 3 copies of them at most. Furthermore, Peter deems GIF animator service and Flash-maker service are substitutable, and he buys both services only when there will be a discount. He can submit a combinatorial bid $(\langle \{a, b\}, \$5 \rangle \rightarrow \langle \{c\}, \$6 \rangle)$.

Representing quantity ranges. The previous bidding languages cannot express buyers' demands for multiunits of goods directly. If a buyer needs 3 units of the good a at most, he can submit a bid denoted as $\langle \{a\}, \$2 \rangle OR \langle \{a\}, \$2 \rangle OR \langle \{a\}, \$2 \rangle$ in OR. His demand even cannot be expressed in XOR. However, our $\mathcal{L}_{MU}$ provides a simple way to represent quantity ranges.

Conciseness.$\mathcal{L}_{MU}$ is concise in 2 respects. Firstly, the quantity ranges of users' demands can be represented simply. Secondly, applying a binary operator $\rightarrow$ to express a bid consisting of substitutable goods needs less characters than that used in XOR. For example, Peter deems GIF animator service and Flash-maker service are substitutable, so his bid expressed in $\mathcal{L}_{MU}$ is $(\langle \{a, b\}, \$5 \rangle \rightarrow \langle \{c\}, \$6 \rangle)$, while in XOR is $(\langle \{a, b\}, \$5 \rangle XOR \langle \{a, b, c\}, \$6 \rangle)$.

Low cost of wireless network transmission. The bids of mobile users are submitted to the auction platform via various wireless networks. The features of $\mathcal{L}_{MU}$ simplify the bids, especially expressing quantity ranges and demands for substitutable goods. Therefore, the costs of wireless network transmission are reduced.

5 | THE WINNER DETERMINATION PROBLEM

WDP (or the combinatorial allocation problem, CAP) represents for the problem that which set of bids to accept;

TABLE 1 Main notations used in WDP

Notation	Explanation
$\boldsymbol{R}$	The set of commodities
$\boldsymbol{I}$	The set of bidders
$\boldsymbol{J}$	The set of providers
$\boldsymbol{A}$	The set of asks offered by providers
$\boldsymbol{B}$	The set of bids submitted by bidders
v^S	The valuation for a set of commodities $S(S \subseteq R)$
$x(S, i)$	Whether S is allocated to bidder i

this is a computational problem concerning how to allocate items to buyers efficiently after the auction platform receives the bids and asks. So the WDP model and its algorithm directly effects the efficiency of an auction mechanism.

Given the notations in Table 1, a general WDP model of a single-sided combinatorial auction can be stated as follows: in an auction, given the set of commodities $\boldsymbol{R}$, the set of bidders $\boldsymbol{I}$ and the set of bids $\boldsymbol{B}$ submitted by all the bidders, we can find an item allocation to bidders, which maximizes the auctioneer's revenue. More formally, the model can be denoted as:

$$\begin{aligned} & max \sum_{i \in I} \sum_{S \subseteq R} B_i(S) x(S, i) \\ & s.t. \sum_{r \in S} \sum_{i \in I} x(S, i) \leq 1 \quad \forall r \in R \\ & \qquad \sum_{S \subseteq R} x(S, i) \leq 1 \quad \forall i \in I \\ & \qquad x(S, i) \in 0, 1 \quad \forall S \subseteq R, i \in I \end{aligned} \tag{5}$$

$\boldsymbol{S}$ is a subset of $\boldsymbol{R}$, ie, $\boldsymbol{S} \subseteq \boldsymbol{R}$. $B_i(S)$ is a bid for $\boldsymbol{S}$ submitted by bidder i. Without loss of generality, let $B_i(S) \geq 0$. If $\boldsymbol{S}$ is allocated to bidder i, $x(S, i) = 1$, otherwise $x(S, i) = 0$.

It is an integer programming problem, which has been proved to be NP-hard.[32] This problem is difficult for a large set of commodities $\boldsymbol{R}$, specifically if bids exist for all subsets of commodities.

Our solution is a many-to-many auction mechanism, ie, combinatorial double auction, which allows buyers and sellers bid simultaneously in one auction. The general combinatorial auctions are single sided; therefore, the WDP model described in problem (1) is unsuitable for our MCC combinatorial double auction. Obviously, the objective of such double auctions is to maximize total surpluses (social welfare) of all participators, including both buyers and sellers. To this end, in our auction mechanism, the WDP is formulated as an optimization problem, ie, to maximize the total social welfare of all the users and providers.

In our combinatorial double auction, $\boldsymbol{R}$ represents for set of commodities, $\boldsymbol{I}$ is the set of mobile users, and $\boldsymbol{J}$ stands for the set of MCC providers. Given $\boldsymbol{B} = \{B_1, \ldots, B_i, \ldots, B_{|I|}\}$ is the set of submitted bids, and $\boldsymbol{A} = \{A_1, \ldots, A_j, \ldots, A_{|J|}\}$ is the set of charged asks, our model can find an allocation that maximizes total social welfare. To formulate a feasible WDP model, the bids and asks need to be preprocessed before being used in our MCC combinatorial double auction.

5.1 | Preprocessing of bids and asks

$\mathcal{L}_{MU}$ enables 4 bidding forms (atomic bid, combinatorial bids, atomic bid with quantity range, and combinatorial bid with quantity range). To simplify WDP, we can transform various bids into the one-unit atomic bid by introducing the concepts of dummy goods and subusers. See the following transformations:

Transformation 1. Any atomic bid with quantity range, $B_i = (\langle S, v^S \rangle)^{\leq n}$. Suppose this bid is submitted by n subusers, this bidding form can then be transformed to n atomic bids $(\langle S, v^S \rangle)$.

Transformation 2. Any combinatorial bid, $B_i = (\langle S_1, v^{S_1} \rangle \rightarrow \langle S_2, v^{S_2} \rangle)$. Suppose su_i^1 submits $(\langle S_1 \cup dummy_i, v^{S_1} \rangle)$ and su_i^2 submits $(\langle S_1 \cup S_2 \cup dummy_i, v^{S_2} \rangle)$. We introduce a dummy good $(dummy_i)$ and 2 subusers (su_i^1, su_i^2); this bidding form can then be transformed to 2 atomic bids.

Transformation 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.

Thus, the original WDP model can be converted to a new one, which is expressed only by the atomic bidding format. The solution to the new WDP model can be converted back to the solution to the original WDP model. In other words, once we get the solution to the new WDP model, we can easily calculate the solution to the original WDP model.

In the same way, we also introduce subproviders to simplify the asks submitted by providers. The following transformation guides the detailed process:

Transformation 4. Any ask offering multiple goods, $(\langle r_1, c_j^{r_1}, q_j^{r_1} \rangle, \langle r_2, c_j^{r_2}, q_j^{r_2} \rangle, \ldots, \langle r_m, c_j^{r_m}, q_j^{r_m} \rangle)$. Suppose they are submitted by m subproviders, this bidding form can then be transformed to m simple asks $(\langle r_m, c_j^{r_m}, q_j^{r_m} \rangle)$. The solution to the new WDP is the solution to the original WDP.

The preprocessing is shown in Algorithm 1.

Algorithm 1 Preprocessing of bids and asks

INPUT:
1) $\boldsymbol{R}$: the set of the commodities; $\boldsymbol{I}$: the set of the mobile users; $\boldsymbol{J}$: the of the MCC providers set;
2) $\boldsymbol{B} = \{B_1, \dots, B_i, \dots, B_{|I|}\}$: the set of bids; $\boldsymbol{A} = \{A_1, \dots, A_j, \dots, A_{|J|}\}$: the set of asks.
OUTPUT:
1) $\hat{\boldsymbol{B}}$: the set of atomic bids;
2) $\hat{\boldsymbol{A}}$: the set of simple asks;
3) $\hat{\boldsymbol{R}}$: the set of goods and dummy goods;
4) $\hat{\boldsymbol{I}}$: the set of users and subusers;
5) $\hat{\boldsymbol{J}}$: the set of providers, dummy providers, and subproviders.
Initialization ($\hat{\boldsymbol{I}}, \hat{\boldsymbol{J}}, \hat{\boldsymbol{B}}, \hat{\boldsymbol{A}} = \emptyset, \hat{\boldsymbol{R}} = \boldsymbol{R}$)
for all $B_i \in \boldsymbol{B}$ **do**
    if B_i is not an atomic bid **then**
        Transform B_i to a group of atomic bids S_b
$\hat{\boldsymbol{B}} = \hat{\boldsymbol{B}} \cup S_b, \hat{\boldsymbol{I}} = \hat{\boldsymbol{I}} \cup \{subusers\}$
        for all $dummy_i$ **do**
            $\hat{\boldsymbol{J}} = \hat{\boldsymbol{J}} \cup \{dp_i\}, \hat{\boldsymbol{A}} = \hat{\boldsymbol{A}} \cup \{(\langle dummy_i, 0, 1\rangle)\}$
$\hat{\boldsymbol{R}} = \hat{\boldsymbol{R}} \cup \{dummy_i\}$
        end for
    else
        $\hat{\boldsymbol{B}} = \hat{\boldsymbol{B}} \cup B_i, \hat{\boldsymbol{I}} = \hat{\boldsymbol{I}} \cup \{user_i\}$
    end if
end for
for all $A_j \in \boldsymbol{A}$ **do**
    if $|A_j| \geq 1$ **then**
        Transform A_j to a group of simple asks S_a
$\hat{\boldsymbol{A}} = \hat{\boldsymbol{A}} \cup S_a, \hat{\boldsymbol{J}} = \hat{\boldsymbol{J}} \cup \{subproviders\}$
    else
        $\hat{\boldsymbol{A}} = \hat{\boldsymbol{A}} \cup A_j, \hat{\boldsymbol{J}} = \hat{\boldsymbol{J}} \cup \{provider_j\}$
    end if
end for

After the original bids and asks are transformed according to the above transformations, each user/subuser only submits one atomic bid, and each provider/subprovider only submits one atomic ask. The new commodity set is denoted as $\hat{\boldsymbol{R}}$, which also includes the dummy goods. Similarly, there are new sets $\hat{\boldsymbol{I}}$ of buyers, $\hat{\boldsymbol{J}}$ of sellers, $\hat{\boldsymbol{B}}$ of transformed bids, and $\hat{\boldsymbol{A}}$ of simplified asks. Each item in $\hat{\boldsymbol{B}}$ is denoted as $\hat{B}_i = \langle S_i, v_i\rangle$, and S_i presents the bundle the buyer i bids. Similarly, each item in $\hat{\boldsymbol{A}}$ is $\hat{A}_j = \langle r_j, c_j, q_j\rangle$, and r_j presents the good that the seller j sells.

5.2 | The WDP model

In the preprocessing period, the original bids and asks are transformed into atomic ones, so the origin WDP model can be formulated as follow:

$$
\begin{aligned}
& max\left(\sum_{i\in\hat{I}} x_i U_i(S_i) + \sum_{j\in\hat{J}} y_j W_j(r_j)\right) \\
& s.t. \sum_{i\in\hat{I}, r\in\hat{B}_i(1)} x_i = \sum_{j\in\hat{J}, r=\hat{A}_j(1)} y_j \quad \forall r \in \hat{R} \\
& \quad y_j \in \{0, 1, \dots, q_j\} \quad \forall j \in \hat{J} \\
& \quad x_i \in \{0, 1\} \quad \forall i \in \hat{I}
\end{aligned}
\tag{6}
$$

where $\boldsymbol{x_i}$ denotes whether the buyer i participant in the auction and $\boldsymbol{y_j}$ denotes seller j's transaction quantity, $(x_i, y_j), i \in \hat{I}, j \in \hat{J}$ specify the auction result. To obtain origin allocation results, we should map subusers (subproviders) to original mobile users (MCC providers); thus, the origin MCC resource allocation is acquired.

Equation 6 is to maximize the total social welfare of all the participators (both users and providers), denoted as

$Z(\boldsymbol{x}, \boldsymbol{y})$. Social welfare reflects the efficiency of an auction mechanism. $U_i(S)$ is buyer i's utility function, which has been defined in Equation 1, $W_j(r)$ is seller j's surplus function, which can be formulated as follow:

$$W_j(r) = P_j^r - c_j^r, \tag{7}$$

where P_j^r denotes that seller j sells item r at price P_j^r and c_j^r is seller j's origin offered price. Therefore, once selling out one unit of commodity r, seller j can obtain $W_j(r)$ surplus.

Therefore, the object of (6) can be rewritten as

$$Z(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i \in \hat{I}} x_i \left(v_i - \sum_{r \in S_i} P_i^r \right) + \sum_{j \in \hat{J}} y_j (P_j - c_j). \tag{8}$$

Because

$$\sum_{i \in \hat{I}} x_i \sum_{r \in S_i} P_i^r = \sum_{j \in \hat{J}} y_j P_j, \tag{9}$$

we have

$$Z(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i \in \hat{I}} x_i v_i - \sum_{j \in \hat{J}} y_j c_j. \tag{10}$$

Up to now, the origin WDP problem can finally transformed into the following integer program:

$$\begin{aligned} (IP)\ z_{IP} = max & \left(\sum_{i \in \hat{I}} v_i x_i - \sum_{j \in \hat{J}} c_j y_j \right) \\ s.t. & \sum_{i \in \hat{I}} b_{ri} x_i - \sum_{j \in \hat{J}} a_{rj} y_j = 0 \quad \forall r \in \hat{R} \\ & y_j \in \{0, 1, \dots, q_j\} \quad \forall j \in \hat{J} \\ & x_i \in \{0, 1\} \quad \forall i \in \hat{I}. \end{aligned} \tag{11}$$

To present the first constraint clearly, 2 matrixes $\boldsymbol{b}$ and $\boldsymbol{a}$ are used. The $\boldsymbol{b}$ is a $|\hat{R}| \times |\hat{I}|$ matrix, and each element is 0 or 1, ie, $b_{ri} \in \{0, 1\}$. Because all the original bids are transformed to the atomic bids, one commodity appears at most once in each atomic bid. If buyer i bids the commodity r, $b_{ri} = 1$. Otherwise, $b_{ri} = 0$. Similarly, $\boldsymbol{a}$ is a $|\hat{R}| \times |\hat{J}|$ matrix, and each element is also 0 or 1. If seller j offers the commodity r, $a_{rj} = 1$. Otherwise, $a_{rj} = 0$. Furthermore, there is only one element being 1 in each column of $\boldsymbol{a}$, because all the original asks are transformed to the form $\hat{A}_j = \langle r_j, c_j, q_j \rangle$, which only consists of one type of commodities. After the original bids and asks are preprocessed according to the transformation methods, the matrixes $\boldsymbol{b}$ and $\boldsymbol{a}$ are initialized.

To decide the specific transaction prices, we design a decomposition algorithm to relax the above integer problem $\mathcal{P}$ into a linear formulation, and the detailed pricing mechanism will be introduced in the next subsection.

5.3 | The decomposition algorithm and pricing mechanism

The optimized problem IP is a special case of the origin NP-hard WDP problem,[32] so it is also NP-hard. Therefore, how to find an optimal allocation solution and transaction prices of each commodity is important.

To find the optimal solution to the general WDP problem defined in (5), there are 2 approaches. The first one is the exact method, which relaxes the problem into a larger feasible region; this helps to solve the origin problem more easily. The optimal solution value of this relaxed problem is the upper bound of the origin problem.[32] The second approach is to use standard artificial intelligence (AI) searches.[33] Given the submitted bids, this method searches all the possible allocations. Besides these, for different problem sizes and structures, there are different algorithms with satisfactory performance. However, there are no general-purpose algorithms that can solve all instances due to the applicability of combinatorial auctions. Furthermore, there is few research works on the double combinatorial auctions.

In our solution, we first decompose IP to find a computationally efficient way to solve the double auction problem. Using a linear programming problem, we reformulate the origin problem, and then we can solve the new linear programming problem in polynomial time. Once we work out the optimal value to the new linear dual problem, the optimal solution to the original problem can be obtained.

To relax the original problem IP, we move the first constraint into the objective function with a penalty term (the Lagrangean relaxation). Then we get the following new problem LR:

$$\begin{aligned} (LR)\ \ z_{LR}(\lambda) = & \, maxL(\boldsymbol{x}, \boldsymbol{y}; \lambda) \\ & s.t.\ 0 \le y_j \le q_j \quad \forall j \in \hat{J} \\ & 0 \le x_i \le 1 \quad \forall i \in \hat{I} \end{aligned} \tag{12}$$

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

$$\begin{aligned} L(\boldsymbol{x}, \boldsymbol{y}; \lambda) = & \sum_{i \in \hat{I}} v_i x_i - \sum_{j \in \hat{J}} c_j y_j \\ & + \sum_{r \in \hat{R}} \lambda_r \Big(\sum_{j \in \hat{J}} a_{rj} y_j - \sum_{i \in \hat{I}} b_{ri} x_i \Big) \end{aligned} \tag{13}$$

and λ is the Lagrangean multipliers vector, $\lambda = (\lambda_1, \dots, \lambda_r, \dots, \lambda_{|\hat{R}|})$.

Thus, we get the Lagrange duality problem LD of the origin problem:

$$\begin{aligned} (LD)\ \ z_{LD} = & \, min z_{LR}(\lambda) \\ & s.t.\ \lambda_r \ge 0 \quad \forall r \in \hat{R} \end{aligned} \tag{14}$$

As there are many subgradient algorithms for the Lagrangean relaxation, z_{LD} can be easily solved. Here, we adopt the subgradient algorithm in Fumero[34] because our problem can be converted into a traveling salesman problem (TSP). Therefore, the subgradient of the Lagrangean function $L(\boldsymbol{x},\boldsymbol{y};\lambda)$ in our problem can be defined as follow:

$$g = \partial L(\boldsymbol{x},\boldsymbol{y};\lambda)/\partial\lambda, \tag{15}$$

where $\lambda^{(k)}$ represents for iteration, which is generated according to the following recursion:

$$\lambda^{(k+1)} = \lambda^{(k)} + t^{(k)}\mathrm{g}^{(k)}. \tag{16}$$

At the point $\lambda^{(k)}$, $t^{(k)}$ represents for the step size and $\mathrm{g}^{(k)}$ is the subgradient of the function $L(\boldsymbol{x},\boldsymbol{y};\lambda)$. Phase I in Algorithm 2 shows the subgradient algorithm.

Algorithm 2 Winner determination algorithm of the MCC combinatorial auction

INPUT:
The output of Algorithm 1.
OUTPUT:
1) The allocation decision $\boldsymbol{WB}$, $\boldsymbol{WA}$;
2) The vector of transaction prices $\boldsymbol{P}$.
Construction of $\boldsymbol{b}$, $\boldsymbol{a}$, and get IP

▷ **Phase I: Optimization**

Relax IP to LD by the Lagrangean Relaxation multipliers λ
Initialization $k = 1$, $\lambda^{(1)} = (1, \dots, 1)$, $\mathrm{g}^{(1)}$, $\varepsilon > 0$
while $\mathrm{g}^{(k)} \geq \varepsilon$ **do**
    Compute $\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(k)}$
    $t^{(k)} = (\bar{L} - L(\boldsymbol{x}^{(k)},\boldsymbol{y}^{(k)};\lambda^{(k)}))/\|\mathrm{g}^{(k)}\|^2$
    $\lambda^{(k+1)} = \lambda^{(k)} + t^{(k)}\mathrm{g}^{(k)}$
    $k = k + 1$
end while

▷ **Phase II: Transformation of Optimal Solution**

Remove dummy good prices from λ, $\boldsymbol{P} = (\lambda_1, \dots, \lambda_{|R|})$
for $r = 1$ to $|R|$ **do**
    With $\boldsymbol{x}, \boldsymbol{y}$, merge allocations of subusers and subproviders into $\boldsymbol{WB}$, $\boldsymbol{WA}$

end for

How to calculate transaction prices is the key point. Consider the constraint $\sum_{i\in\hat{I}} b_{ri}x_i - \sum_{j\in\hat{J}} a_{rj}y_j = 0 (\forall r \in \hat{R})$, this means that the total demands of all users should be equal to the total supplies of all providers. According to the Lagrangean multipliers, λ acts as a price vector. Therefore, once the Lagrangean dual problem LD is solved, we can then obtain the optimal vector $\lambda = (\lambda_1, \dots, \lambda_r, \dots, \lambda_{|\hat{R}|})$. So the transaction price of the commodity r is λ_r. As one type of goods only has one price in the MCC auction market, so the dummy goods' trade prices are all 0.

Following the above steps, we present our detailed MCC combinatorial auction mechanism in Algorithm 2, which introduces how the auction mechanism decides commodity allocation and transaction prices.

The allocation decision ($\boldsymbol{WB}$, $\boldsymbol{WA}$) and transaction prices ($\boldsymbol{P}$) are obtained by executing Algorithm 2, and these results are published on the auction platform. $\boldsymbol{WB}$ is a $|R| \times |I|$ matrix, where WB_{ri} denotes the amount of commodity r allocated to user i. Similarly, $\boldsymbol{WA}$ is a $|R| \times |J|$ matrix, where WA_{rj} denotes the quantity of good r sold by the provider j. $\boldsymbol{P}$ is a vector of $|R|$ elements, where P_r denotes the transaction price of commodity r. The platform matches users and providers according to $\boldsymbol{WB}$ and $\boldsymbol{WA}$ in sequence. Then it reports the user allocation to providers. The match result and amount that each user needs to pay are calculated and announced to users. Once the transaction is completed, the connection between users and providers will be established and providers start to provide services to users.

Obviously, Algorithm 2 is individually rational because mobile users will never be charged more than their valuations. Besides, it is budget balanced because the total profits of all providers are equal to the total payments of all users.

6 | EVALUATION

The main objective of our auction mechanism is to allocate MCC resources effectively. Therefore, we focus on examining the allocation performance of our mechanism under mobile users' illustrative demands and providers' offer distributions. Furthermore, the computational efficiency is also an important criterion, for mechanisms should be designed to require computations as few as possible. Because the popular simulation softwares (eg, CloudSim) support neither auction protocols nor price generation, we simulate auctions with different kinds of scales to evaluate our solution.

There are some research works applying various auction mechanisms to allocate cloud or mobile cloud resources. However, we do not compare our auction mechanism with these solutions for the following reasons.

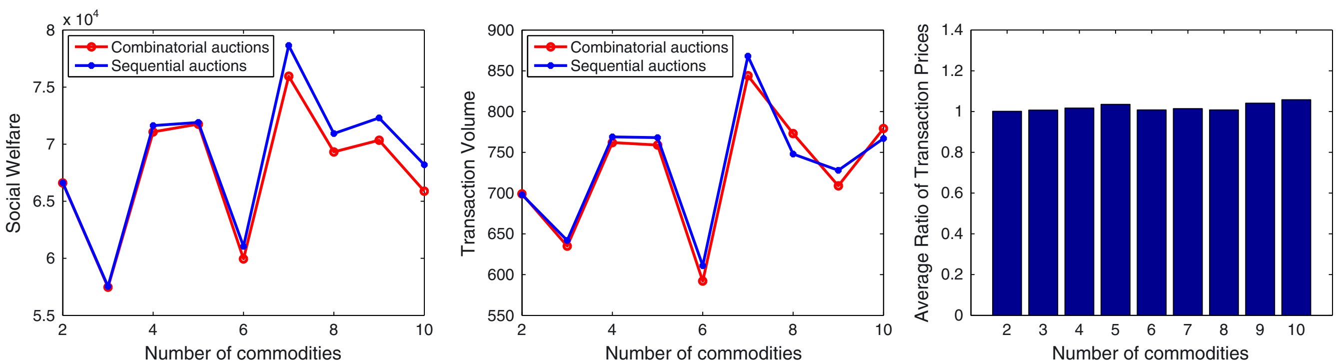
Firstly, none of the prior solutions has implemented online combinatorial double auctions. The existing solutions are single-sided auctions, which only support one-to-many negotiations. In addition, most of them are noncombinatorial auctions, ie, the bidders only bid one type of goods in one auction. Our solution enables buyers and sellers to submit bids simultaneously and allows users to bid a bundle of goods at one auction.

Secondly, the existing solutions often compare themselves with some theoretic auction mechanisms, which can get the social optimal allocation decisions but have not been applied in real markets.[4,24] For example, Zhang et al[4] compared their solution with the Vickrey-Clarke-Groves (VCG) mechanism. Although such theoretic mechanism can get the allocation decision with the optimal social welfare, it cannot be followed in real-world cloud markets because it is either too costly or too difficult for users to understand. If the social welfare of a feasible auction is close to such a theoretic optimal, it also proves the auction performance is acceptable. In this section, we compare our solution with the sequential single-minded double auctions on allocation performance.

We consider the following simulation scenario: The set of commodities is $\boldsymbol{R}$, the set of mobile users is $\boldsymbol{I}$, and the set of MCC providers is $\boldsymbol{J}$. Because Algorithm 1 can preprocess all the complicated original bids and asks without loss of generality, suppose each user submits only one atomic bid $(\langle S, v^S \rangle)$, and each provider offers only one ask $(\langle r, c, q \rangle)$. User i bids for the bundle S_i, which is a subsets of $\boldsymbol{R}$ (selected from the $2^{|\boldsymbol{R}|} - 1$ cases).

To show the comparison results between our MCC combinatorial double auction and the traditional single-minded auction, $|\boldsymbol{R}|$ sequential auctions $a_1, \dots, a_r, \dots, a_{|\boldsymbol{R}|}$ are constructed, 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 is divided into 3 single-minded auction bids sa_l, sa_m and sa_n, and the value for each bid is set to $v_i^S/3$. We use the Marshallian path[35] to match bids and asks to provide the optimal solutions. Specifically, we evaluate the following six criteria:

1. **Social welfare.** Denoted by E_s, social welfare is the total payoffs/utilities of both sellers and buyers. The social welfare of the proposed combinatorial auction $E_s(CA)$ is the optimal value of objective function $Z(\boldsymbol{x}, \boldsymbol{y})$, and that of the sequential auctions $E_s(SA)$ is the sum of $|\boldsymbol{R}|$ sequential auctions: $E_s(SA) = \sum_{r \in \boldsymbol{R}} E_s(a_r)$.
2. **Transaction volume.** Denoted by E_v, transaction volume is the quantity of the goods transacted successfully, ie, the amount of the transactions, which reflects the market efficiency of the auction mechanisms. The larger the transaction volume is, the better the market efficiency is. The transaction volume of the proposed combinatorial auction is $E_v(CA) = \sum_{j \in \boldsymbol{J}} y_j$, and that of the sequential auctions is $E_v(SA) = \sum_{r \in \boldsymbol{R}} E_v(a_r)$.
3. **Average ratio of transaction prices.** Denoted by α, average ratio of transaction prices is the average ration of transaction price in our combinatorial auction and that in sequential auctions. $\alpha = (\sum_{r \in \boldsymbol{R}} P_r^{CA} / p_r^{SA}) / |\boldsymbol{R}|$.
4. The **Computational efficiency.** Computational efficiency is the number of iterations, which denotes the overall efficiency and performance of the whole platform.



Social welfare of combinatorial auctions compared with sequential auctions

Transaction volume of combinatorial auctions compared with sequential auctions

Ratio of transaction prices

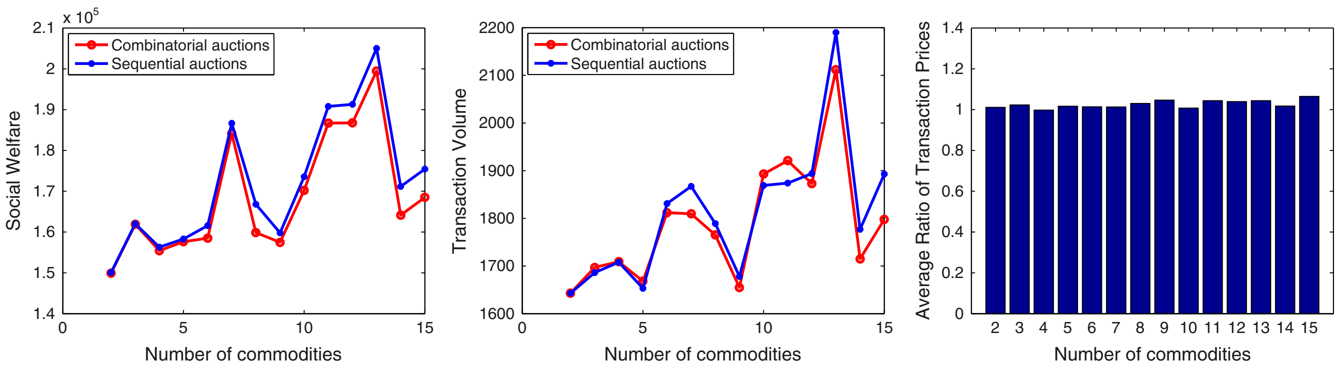
FIGURE 4 Allocation performance of the proposed mobile cloud computing combinatorial double auction mechanism (Scenario 1)

5. **Number of players.** $|\boldsymbol{R}|$, $|\boldsymbol{I}|$, and $|\boldsymbol{J}|$ are the number of commodities, users, and providers, respectively. These parameters can influence the computational efficiency significantly.
6. **Running time.** Denoted by $|\boldsymbol{T}|$, running time represents for the waiting time of players in the auction. The whole process will be considered low-latency if $|\boldsymbol{T}|$ falls in an acceptable range.

The comparison results between combinatorial auction and sequential auction are shown in Figures 4 and 5; social welfare, transaction volume, and transaction price on different scales are measured. User amount and provider amount are fixed, and the commodity amount is increasing. In Figure 4, $|\boldsymbol{J}| = 100$, $|\boldsymbol{I}| = 2000$ and $|\boldsymbol{R}| = 2, 3, \dots 10$. In Figure 5, $|\boldsymbol{J}| = 200$, $|\boldsymbol{I}| = 5000$, and $|\boldsymbol{R}| = 2, 3, \dots 15$. From these simulation results, we can see that with the increasing commodity number, the performance of the proposed combinatorial auctions is always close to the optimal solution. Further, the transaction prices of our mechanism are stable.

To evaluate the computational efficiency, we analyze the number of iterations when computing the optimal solution, and the results on different scales are shown in Figure 6. Figure 6A shows the influence on different user amount while the provider amount $|\boldsymbol{J}|$ is set to 100. Figure 6B shows the influence on different provider amounts while the user amount $|\boldsymbol{I}|$ is set to 2000. To better illustrate the influence, we conduct 2 scenarios on different scales. The solid line denotes the curve when the number of commodities $|\boldsymbol{R}|$ is set to 5, while the dashed line denotes the curve when $|\boldsymbol{R}|$ is set to 10. From the result we can see that the iterations is about 40 even when there is 10 commodities, 2000 users and 100 providers. These results prove that the bidding platform is efficient enough to calculate winners and prices in a particular auction in large scales.

To better illustrate the influence on different amounts of players (commodities $|\boldsymbol{R}|$, users $|\boldsymbol{I}|$, providers $|\boldsymbol{J}|$), we conduct a large number of experiments in the above 2 scenarios. The amount of users and providers are both on the increase, and there are 5000 users and 100 providers at

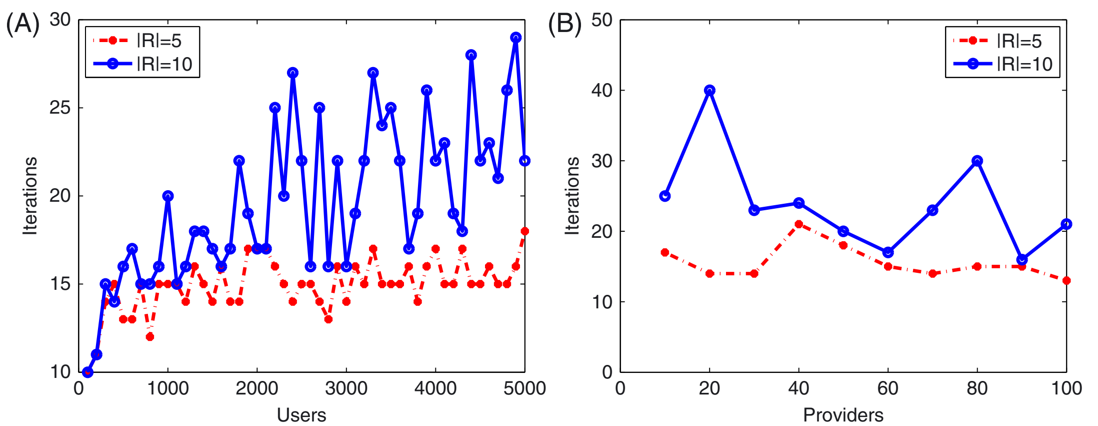


Social welfare of combinatorial auctions compared with sequential auctions

Transaction volume of combinatorial auctions compared with sequential auctions

Ratio of transaction prices

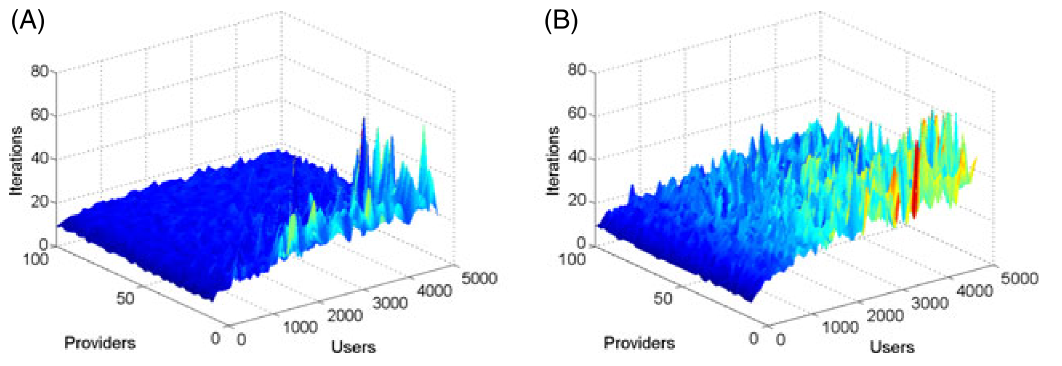
FIGURE 5 Allocation performance of the proposed mobile cloud computing combinatorial double auction mechanism (Scenario 2)



Iterations with different amount of users

Iterations with different amount of providers

FIGURE 6 Computational efficiency of the proposed mobile cloud computing combinatorial double auction mechanism with different amount of users and providers (The number of iterations can represent the computational efficiency)



Iterations with different user and provider amounts while $|\boldsymbol{R}| = 5$

Iterations with different user and provider amounts while $|\boldsymbol{R}| = 10$

FIGURE 7 Iterations of the proposed mobile cloud computing combinatorial double auction mechanism

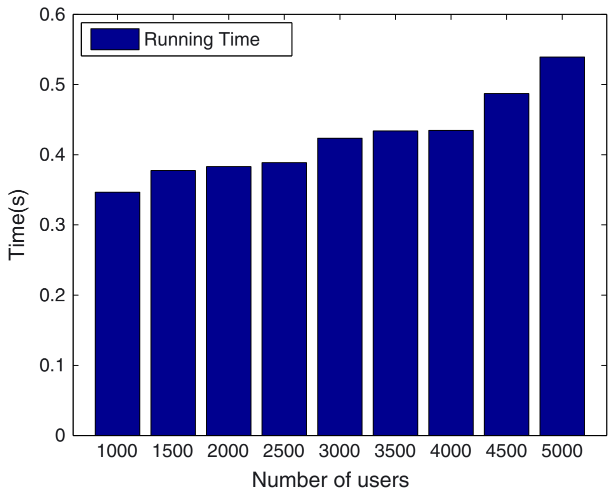


FIGURE 8 Running time of different amount of users while $|\boldsymbol{R}| = 10$

most. In Figure 7A, the amount of commodities $|\boldsymbol{R}|$ is set to 5, and in Figure 7B, the $|\boldsymbol{R}|$ is set to 10. The peak value of iterations is 59, and the mean value is about 23.68. From these results, we can conclude that the number of interactions will increase along with the number of commodities and players, which is still acceptable.

As to the running time, we make a timer for the platform and analyze the time cycle of the whole process. When there are 10 commodities and 100 providers, we test the running time with different amounts of users (from 1000 to 5000). Figure 8 shows the running time, and each bar stands for the average value of 100 experiments. From these results, we can see that the latency of this platform is about 500 milliseconds even there are thousands of players.

For comparison, we investigate the response time of a web service (PayPal) for a customer application and find that despite the occasional service blip, the PayPal service has about 450 milliseconds,[36] which indicates that the user experience and application performance of PayPal is truly nice. Further, according to Akamai,[37] 47% of consumers expect a website to load in 2 seconds or less, and 40% of consumers will abandon a website that takes more than 3 seconds to load. Besides, in a 2012 analysis of Google Analytics data, Google found that the web's median page load speed is 2.7 seconds (4.8 seconds for mobile). So iWeb Technologies[38] suggests that 2 seconds is the benchmark under which you should aim to keep your response speeds. Therefore, we can conclude that our platform can meet users' latency requirements.

Overall, from the above simulation results, we can get the following conclusions: First, the social welfare and transaction volume of the proposed MCC combinatorial double auction are close to the optimal solution, which proves that the allocation efficiency of our approach is high. Second, the transaction prices of our approach are stable. Third, the WDP algorithm can obtain the optimal results and can converge with acceptable latency.

7 | CONCLUSION

In this paper, we first design a combinatorial double auction mechanism in the 3C-based MCC market. To implement the mechanism, we then design an online auction framework, which enables mobile users to bid bundles of items in one auction. On base of these, we also design a novel bidding language to facilitate mobile users to express their valuations. To efficiently decide winners and prices of each auction, a WDP model is formulated. At last, we conduct a series of experiments, and the results show that the proposed mechanism obtains nice allocation performance that are close to the optimal solution and the system performance is stable.

ACKNOWLEDGMENTS

This work was supported by the National Natural Foundation of China (61472212), National Science and Technology Major Project of China (2015ZX03003004), the National High Technology Research and Development Program of China (863 Program) (2013AA013302, 2015AA015601), and EU Marie Curie Actions CROWN (FP7-PEOPLE-2013-IRSES-610524). Jiangchuan Liu's work was supported by the National Natural Science Foundation of China (U1301253) and Science and Technology Planning Project of Guangdong Province, China (2015B010110006).

ORCID

Yuchao Zhang http://orcid.org/0000-0002-0135-8915

REFERENCES

1. Amazon EC2 pricing. http://aws.amazon.com/ec2/pricing/; June 21, 2013.
2. Buyya R, Ranjan R, Calheiros RN. Intercloud: Utility-oriented federation of cloud computing environments for scaling of application services. *Algorithms and Architectures for Parallel Processing*. Busan, Korea:Springer; 2010:13-31.
3. Shi X, Xu K, Liu J, Wang Y. Continuous double auction mechanism and bidding strategies in cloud computing markets. *Bibliographic Code: 2013arXiv1307.6066S*. July, 2013.
4. Zhang H, Li B, Jiang H, Liu F, Vasilakos AV, Liu J. A framework for truthful online auctions in cloud computing with heterogeneous user demands. In: Proceedings of the 2013 IEEE Infocom. Turin, Italy; 2013:1510-1518.
5. Zaman S, Grosu D. Combinatorial auction-based mechanisms for VM provisioning and allocation in clouds. In: 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID). Ottawa, Canada; 2012:729-734.
6. Dinh HT, Lee C, Niyato D, Wang P. A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless Commun Mobile Comput*. 2013;13(18):1587-1611.
7. Fernando N, Loke SW, Rahayu W. Mobile cloud computing: a survey. *Future Gener Comput Syst*. 2013;29(1):84-106.
8. Sanaei Z, Abolfazli S, Gani A, Buyya R. Heterogeneity in mobile cloud computing: taxonomy and open challenges. *Commun Surv Tutorials, IEEE*. 2014;16(1):369-392.
9. Buyya R, Abramson D, Giddy J, Stockinger H. Economic models for resource management and scheduling in grid computing. *Concurr Comput Pract Experience*. 2002;14(13-15):1507-1542.
10. Zhang Y, Xu K, Wang H, Liu J, Zhong Y, Chen W. Performance and incentive of teamwork-based channel allocation in spectrum access networks. *IEEE 23rd International Symposium on Quality of Service (IWQoS)*. 2014:255-260.
11. Cramton P, Shoham Y, Steinberg R. Combinatorial auctions. 1-19; 2006.
12. McAfee RP. A dominant strategy double auction. *J Econ Theory*. 1992;56(2):434-450.
13. De Vries S, Vohra RV. Combinatorial auctions: a survey. *INFORMS J Comput*. 2003;15(3):284-309.
14. Zaman S, Grosu D. Combinatorial auction-based allocation of virtual machine instances in clouds. *J Parallel Distrib Comput*. 2013;73(4):495-508.
15. Chu LY. Truthful bundle/multiunit double auctions. *Manage Sci*. 2009;55(7):1184-1198.
16. Gjerstad S, Dickhaut J. Price formation in double auctions. *Games Econ Behav*. 1998;22(1):1-29.
17. Bredin J, Parkes DC. Models for truthful online double auctions. *Bibliographic Code: arXiv:1207.1360*. July, 2012:50-59.
18. Wang Wei, Liang Ben, Li Baochun. Designing truthful spectrum double auctions with local markets. *Mobile Comput IEEE Trans on*. 2014;13(1):75-88.
19. Wei Y, Zhu Y, Zhu H, Zhang Q, Xue G. Truthful online double auctions for dynamic mobile crowdsourcing. In: 2015 IEEE Conference on Computer Communications (infocom). Hong Kong, China; 2015:2074-2082.
20. Iosifidis G, Gao L, Huang J, Tassiulas L. An iterative double auction for mobile data offloading. In: International Symposium on Modeling & Optimization in Mobile, Ad Hoc & Wireless Networks. Tsukuba Science City, Japan; 2013:154-161.
21. Vytelingum P, Cliff D, Jennings NR. Strategic bidding in continuous double auctions. *Artif Intell*. 2008;172(14):1700-1729.
22. Sutherland IE. A futures market in computer time. *Commun Acm*. 1968;11(6):449-451.
23. Wolski R, Plank JS, Brevik J, Bryan T. Analyzing market-based resource allocation strategies for the computational grid. *Int J High Perform Comput Appl*. 2001;15(3):258-281.
24. Prasad G, Vinu, Prasad. A combinatorial auction mechanism for multiple resource procurement in cloud computing. In: International Conference on Intelligent Systems Design and Applications. Kochi, India; 2012:337-344.
25. Lee C, Wang P, Niyato D. A real-time group auction system for efficient allocation of cloud internet applications; 2013.
26. Fujiwara I, Aida K, Ono I. Applying double-sided combinational auctions to resource allocation in cloud computing. In: IEEE/IPSJ International Symposium on Applications and the Internet. Seoul, Korea (South); 2010:7-14.
27. Niyato D, Zhang Y, Wang P. An auction mechanism for resource allocation in mobile cloud computing systems. In: International Conference on Wireless Algorithms, Systems, and Applications. Zhangjiajie, China; 2013:76-87.
28. Menache I, Ozdaglar A, Shimkin N. Socially optimal pricing of cloud computing resources. In: Proceedings of the 5th International ICST Conference on Performance Evaluation Methodologies and Tools. Paris, France; 2011:322-331.
29. Ostermann S, Iosup A, Yigitbasi N, Prodan R, Fahringer T, Epema D. A performance analysis of EC2 cloud computing services for scientific computing. *Cloud Computing*. Springer; 2010:115-131, Lisbon, Portugal.
30. Lehmann D, Oćallaghan LI, Shoham Y. Truth revelation in approximately efficient combinatorial auctions. *J ACM (JACM)*. 2002;49(5):577-602.
31. Boutilier C, Hoos H. Bidding languages for combinatorial auctions. In: International Joint Conference on Artificial Intelligence. Seattle, Washington, USA; 2001:1211-1217.
32. Sandholm T. Approaches to winner determination in combinatorial auctions. *Decis Support Syst*. 2000;28(1):165-176.
33. Sandholm T, Suri S. Bob: improved winner determination in combinatorial auctions and generalizations. *Artif Intell*. 2003;145(1):33-58.
34. Fumero F. A modified subgradient algorithm for Lagrangean relaxation. *Comput Oper Res*. 2001;28(1):33-52.

35. Brewer PJ, Huang M, Nelson B, Plott CR. On the behavioral foundations of the law of supply and demand: human convergence and robot randomness. *Exp Econ*. 2002;5(3):179-208.
36. Mappic S. How fast are your web services? July 08, 2015. https://blog.appdynamics.com/cloud/how-fast-are-your-web-services/.
37. Akamai. Content delivery network (cdn) and cloud computing services. February 17, 2015. https://www.akamai.com/.
38. iWeb Technologies. Understanding, analyzing and reducing latency. February 20, 2014. http://blog.iweb.com/en/2014/02/understanding-analyzing-reducing-latency/13176.html.

How to cite this article: Zhang Y, Xu K, Shi X, Wang H, Liu J, Wang Y. Design, modeling, and analysis of online combinatorial double auction for mobile cloud computing markets. *Int J Commun Syst*. 2017;e3460. https://doi.org/10.1002/dac.3460