A Double Auction-Based Approach for Multi-User Resource Allocation in Mobile Edge Computing

Yanlin Yue, Wen Sun and Jiajia Liu
School of Cyber Engineering, Xidian University, Xi'an, Shaanxi, 710071, China
yueyanlin@outlook.com, sunwen@xidian.edu.cn, liujiajia@xidian.edu.cn

***Abstract*—Mobile edge computing (MEC), as an emerging technique, brings computing resource near to the user end, and thus offers prompt and high-bandwidth service. In MEC, edge servers typically provide their limited computing resource to an appropriate set of users and expect to be rewarded for that, while users pay for the service. In this paper, We utilize the network economics to solve the problem of resource allocation in MEC to maximize system efficiency. We consider a multi-user and multi-server scenario with locality-awareness, i.e., an edge server can only serve multiple MDs in the vicinity, and propose a single-round double auction scheme based on breakeven in MEC (SDAB), which is proved to be individual rationality and truthful. The simulation results indicate that SDAB can significantly improve the system efficiency of MEC as compared with the existing works, while maintaining the economic property of budget balance.**

***Keywords*—mobile edge computing, resource allocation, network economics, double auction.**

I. Introduction

With the unprecedented growth of mobile devices (MDs), such as smartphones or Internet of things (IoT) equipment, more and more highly complex and computationally intensive applications like interactive gaming, image recognition are running on MDs [1], [2]. However, running these computation-intensive applications at the MDs is constrained by limited resource capacity and high battery consumption of the MDs. Mobile cloud computing (MCC) creates a suitable scheme by offloading the computation tasks to a distant centralized cloud [3], [4]. However, the transmission delay caused by long-distance transmission and network congestion makes the scheme unfeasible for real-time applications.

As a new emerging prominent computing paradigm, mobile edge computing (MEC) has been introduced [5]–[7]. The MEC offers MDs proximity cloud computing service through edge servers that bring storage and computation resources to the edge of mobile network. Hence, compared with the MCC, the MEC can reduce significantly latencies and improve reliability [8], [9]. Although MEC is a promising technology that enhances the quality of service (QoS), it faces many challenges [10], [11], such as mobility management, locality-aware resource allocation for various edge severs and MDs, and network economics for profit-driven edge servers and MDs. What needs to be stressed is that resource allocation and network economics exhibit locality in MEC where the MEC provides only limited computational and storage resources and the number of applications being served may be constrained [12]. Moreover, MDs and edge servers typically belong to different camps, and they only care their own interests. Thus how to design an incentive framework to appropriately encourage the participation of MDs and edge servers, whereas allocating computing resource with locality-awareness to maximize the system efficiency is a key issue.

There have been existing studies on how to utilize the network economics to solve the problem of resource allocation [13]–[15]. Bidirectional interaction between MDs and edge servers can be well constructed using auction scheme. Iosifidis *et al*. [16] assume that all mobile network operators can share multiple access points. They proposed a double-auction mechanism to maximize the difference between the benefits and costs of computing offloading. In the works of [17], the authors modelled the offloading decision as a double auction scheme, and proposed an algorithm to improves the auction efficiency. Zhao *et al*. [18] formulated the resource allocation problem as a multi-round combinational double auction considering the different types of virtual machines can be integrated into a bundle to be bid and the QoS level is taken into consideration in their scheme. Jin *et al*. [19] designed an efficient incentive mechanism to fully reap the interests of offloading by charging MDs and rewarding cloudlets.

The existing works provide valuable insights for resource allocation. However, they did not joint the limited computing resources and the locality characteristics of edge servers. Due to the limited resources of edge severs, there exists locality where MDs' computing tasks can only be offloaded to their nearby edge servers, and the number of MDs served by the edge servers may be constrained. To compensate for this deficiency, in this paper, we joint network economics and resource allocation to maximize the number of successful trades on the basis of the literature [19]. To be more specific, we design a single-round double auction scheme based on breakeven (SDAB) for the resource allocation. Different from the literature [19], we consider the situation where each seller can be assigned to multiple buyers, and the simulation results indicate that the performance of SDAB can significantly improve the system efficiency of MEC while maintaining the economic properties of budget balance.

The rest of this paper is organized as follows. Section II describes the system model of this work. Section III proposes our auction scheme, i.e., SDAB. In Section IV, we give numerical results. Finally, we conclude in Section V.

978-1-5386-2070-0/18/$31.00 ©2018 IEEE

II. System Model

A. System Model

As illustrated in Fig. 1, MDs try to connect to edge servers to offload their computing tasks. Taking into account the spatial distributions of edge servers and their locality-aware characteristics, MDs have different preferences for different edge servers. For example, MD 1 tends to connect to the 1^{st} edge server and MD 3 tends to connect to the 1^{st} or the 2^{nd} edge server. On the other hand, edge servers should be motivated to contribute their resources to mobile devices. Therefore, there is a trade between the MDs requesting the resources and the MEC offering the resources, and MDs as the buyers and edge severs as the sellers. As a trusted third party, the auctioneer is responsible for the trading between MDs and edge severs while determining winning traders and clearing prices. It should be emphasized that the information submitted by buyers and sellers to the auctioneer is private and every trader has no knowledge of others [20].

Considering m edge severs (sellers) and n MDs (buyers). The set of edge servers can be denoted by $S = \{1, 2, ..., j, ..., m\}$, and j means the j^{th} edge sever, i.e., seller j. At the start of auction, the edge server announces their resources to the auctioneer, $R = \{r_1, r_2, ..., r_j, ..., r_m\}$, where r_j is the number of remaining resource units for seller j. Each seller asks a reward for the resources it provides, i.e., ask. The asks of all sellers can be denoted by an ask vector $A = \{a_1, a_2, ..., a_j, ..., a_m\}$, where a_j is the ask of seller j.

Let $B = \{1, 2, ..., i, ..., n\}$ be the set of buyers, and i denotes the i^{th} buyer. In order to obtain the computing resources, buyer i requests resources from every edge server it can connect to and has a bid price vector $D_i = \left\{d_i^1, d_i^2, ..., d_i^j, ..., d_i^m\right\}$, where d_i^j is the maximum price how much buyer i is willing to pay for seller j. For buyer i, the values of the elements in D_i may be different, due to its location relationship with the edge severs. For all buyers, the bid matrix can be denoted by $D = \{D_1; D_2; ...; D_n\}$. Note that we assume the computing task is atomic and cannot be divided in this paper.

B. Double Auction Framework in MEC

Given these necessary information, such as S, B, A, R and D, how to decide the mapping between winning sellers and winning buyers under the limited computing resources of edge servers and determine the price of MDs and the payment of edge servers, is of great significance. Let W_b be the winning buyer set and W_s be the winning seller set. For each winning buyer $i \in W_b$, we have a winning seller $j \in W_s$ that corresponds to it, i.e., $\forall i \in W_b, \delta(i) = j \in W_s$, where $\delta(x)$ is the mapping function.

In order to determine the price and the payment, we need to introduce a concept that is not mentioned above, i.e., utility. For each buyer, utility refers to the difference between the true valuation of the computing resources it achieved and the price that it has to pay. For sellers, utility is equal to the payment minus the cost for processing computing tasks. To further illustrate this term, we assume that buyer i achieved the resources of seller j. v_i^j is the true valuation of buyer i on the resources achieved from seller j, and let p_i^b be the price that buyer i has to pay. p_j^s is the payment of seller j recevied from all buyers that have achieved its resources, and c_j is the cost of seller j. The utility of buyer i (denoted as u_i^b) and seller j (denoted as u_j^s) are defined as follows:

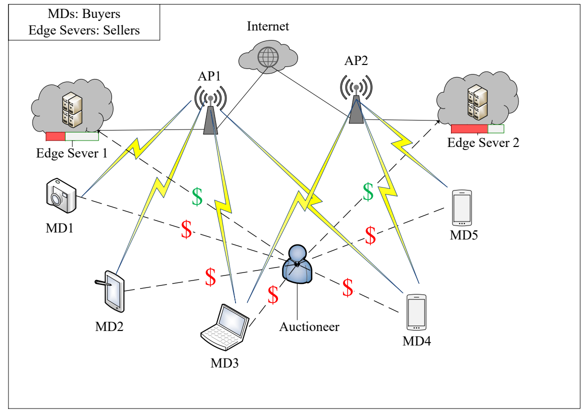


Fig. 1. An illustration of resource allocation in MEC. Each MD/buyer bids for computation resource with a bid price. Each edge server/seller is paid for providing computational and storage resources and the auctioneer determines winning traders and clearing prices.

$$\begin{cases} u_i^b = v_i^j - p_i^b & b_i \in W_b \\ u_j^s = t_j^s - c_j & s_j \in W_s \end{cases} \tag{1}$$

Note that if $i \notin W_b$, the utility of buyer i will be zero, in other words, it fails to bid. $u_i^b > 0$ means that buyer i succeeds in earning the resources of an edge sever and benefits from the trade, the larger the value of u_i^b , the greater the profit. The true valuation v_i^j of buyer i indicates the true price that buyer i is willing to pay for the service. In order to avoid manipulation, the bid price shall equals to the true valuation of service, i.e. $v_i^j = d_i^j$.

The utility of the auctioneer is equal to the total charge from the MDs minus the payment to the edge servers. It can be expressed by the following formula:

$$u_a = \sum_{b_i \in W_b} p_i^b - \sum_{s_j \in W_s} p_j^s \tag{2}$$

For seller j, a set of MDs can be served if the following two conditions are met:

1) the demanding computation resources by the selected MDs are less than the capacity of edge server;
2) all the matched bids are not less than its ask, i.e., $d_i^j \geq a_j$.

For the convenience of readers, the important notations are given in Table I.

III. SDAB Scheme

In this paper, we go further on the basis of [19]. For the sake of comparison of the later simulation and the convenience

TABLE I
NOTATION SETTING

Symbol	Meaning
B	Set of buyers
S	Set of sellers
b_w^{qi}	Set of buyers that win the services from seller qi
D	Bid matrix of all buyers
D_c	Bid vector of candidates
D_i	Bid vector of buyer i
A	Ask vector of all sellers
A_c	Ask vector of candidates
R	Resource vector of all sellers
W_s	The set of winning sellers
W_b	The set of winning buyers
v_i^j	The valuation of buyer i on service from seller j
p_i^b	Price charged to buyer i
p_j^s	Payment rewarded to seller j
P_b	Set of price charged winning buyers
P_s	Set of payment rewarded to winning sellers
u_i^b	Utility of b_i
u_j^s	Utility of s_j
M_c	Mapping set of candidates
M_w	Mapping set of winners

of the readers, we used almost the same notations as in [19] . Note that this work is different from [19] in the aspects that our work allows an edge server to serve multiple users while literature [19] is just a one-to-one resource allocation model. Moreover, we eliminate the duplicate buyers (the same buyer who may be selected by other sellers) during the winner determination. For the sellers in the backward order, during buyer elimination, the candidate buyers are eliminated (as these buyers have been selected by the sellers sorted before them), and they lose the chance to incorporate other potential buyers.

A. Detailed Description of SDAB

In this section, we propose a single-round double auction scheme based on breakeven (SDAB) to solve the resource allocation problem in MEC, and then we can prove that the scheme satisfies truthfulness and individual rationality. SDAB contains two stages: filtering & candidate set determination and winner set determination & pricing.

In the first stage, we sort the bids in descending order and sort the asks in ascending order, and then we determine the candidate set based on sorted bids and asks. For asks, we sort in ascending order $A^{'} = \{a_{q1}, a_{q2}, ..., a_{qm}\}, 0 \leq a_{q1} \leq a_{q2} \leq ... \leq a_{qm}$. After which we sort all the non-zero elements in D in descending order $D^{'} = \{d_{p1}^{q1}, d_{p2}^{q2}, ..., d_{pg}^{qg}\}, d_{p1}^{q1} \geq d_{p2}^{q2} \geq , ..., \geq d_{pg}^{qg} > 0$, where d_{pg}^{qg} is the bid value of buyer pg to seller qg, and it is the g^{th} element in D', $g = |D'|$. It needs to be emphasized that each buyer may bid on different sellers. Therefore, one buyer may appear multiple times, and a seller may also appear multiple times.

According to breakeven, we eliminate the buyers and sellers that do not satisfy the criteria. In this paper, we choose the median as the breakeven just like in literature [19], i.e., the median ask of sellers is selected as breakeven $a_{q\mu}$, where $\mu = \lceil \frac{m+1}{2} \rceil$. The bid threshold is $d_{p\varphi}^{q\varphi}$ in $D^{'}$, $\varphi = \underset{\varphi}{argmax}\{d_{p\varphi+1}^{q\varphi+1} < a_{q\mu}\}$, i.e., $d_{p\varphi}^{q\varphi}$ is the minimum value greater than or equal to $a_{q\mu}$. Then we remove the asks that greater than or equal to $a_{q\mu}$, $A_c \leftarrow A^{'} \setminus \{a_{q\mu}, a_{q(\mu+1)}, ..., a_{qm}\}$, and all the bids that less than $d_{p\varphi}^{q\varphi}$ are removed, $D_c \leftarrow D^{'} \setminus \{d_{p\varphi+1}^{q\varphi+1}, d_{p\varphi+2}^{q\varphi+2}, ..., d_{pg}^{qg}\}$. In the end we have the mapping relationship of the candidates. For example, if $d_{pi}^{qi} \in D_c$ & $a_{qi} \in A_c$ then (pi, qi) will be the successful mapping. Let M_c be the mapping set of candidates, then $(pi, qi) \in M_c$. It is important to emphasize that M_c is an ordered set that its elements are ordered in decreasing order of the bid of candidate buyers (D_c).

In the second stage, starting with the first element in M_c, we consider the following scenarios based on the M_c. For the convenience of narration, we take $(pi, qi) \in M_c$ as an example.

- If all restrictions are met, parties to the transaction are added to the winner set, i.e., $W_b \leftarrow W_b \cup \{pi\}, W_s \leftarrow W_s \cup \{qi\}$, and delete other elements related to pi in M_c.
- If restrictions do not hold, the price of the selected feasible buyers (denoted by b_w^{qi} , so b_w^{qi} is a set of buyers that win the resources from seller qi) for seller qi is determined the same as the highest bid who loses the trade $\forall k \in b_w^{qi}$, $p_k^b \leftarrow d_{pi}^{qi}$, d_{pi}^{qi} is the highest bid who loses the trade with seller qi. Meanwhile, in M_c, delete all the elements that bid on but failed to obtatin the resource from seller qi.
- In the end, the remaining elements of M_c are the winning matched pairs, i.e., $M_w \leftarrow M_c$, M_w is the mapping set of winners.

For buyers who have not priced yet in W_b,we uniformly charge with a price $d_{p\varphi}^{q\varphi}$. For each seller in W_s, for example, $qi \in W_s$, whenever seller qi serves a buyer, it will be receive a reward $a_{q\mu}$. The total payment of seller qi, i.e., $p_{qi}^s = a_{q\mu} \times num$, where num is the number of elements in b_w^{qi}.

B. Economic Properties

Our proposed double auction scheme satisfies the properties of truthfulness and individual rationality. In this subsection, we wiil prove that.

Theorem 1. *SDAB satisfies individual rationality.*

Proof: Individual rationality means that no one will lose by participating in the auction. The price of a particular winning buyer is less than than its bid price. The payment of a winning seller is greater than its ask. For SDAB, it satisfies individual rationality when the utility of all traders participating in the auction is greater than or equal to zero. If a buyer or seller fails in the auction, the utility of the buyer or seller is zero. For a winning seller j, its received payment $p_j^s \geq a_{q\mu}$ and its asks is a_j. In our algorithm, for

Algorithm 1: Multi-user to multi-sever scheme with double auction : SDAB(B, S, A, D, R).

Input: B, S, A, D, R;
Output: W_b, W_s, M_w, P_b, P_s;

Initialize:$W_b = \emptyset, W_s = \emptyset, M_w = \emptyset, P_s = \emptyset$;
Order the non-zero elements of D in decreasing order $D^{'} = \{d_{p1}^{q1}, d_{p2}^{q2}, ..., d_{pg}^{qg}\}, d_{p1}^{q1} \geq d_{p2}^{q2} \geq, ..., \geq d_{pg}^{qg} > 0$;
network economics Order the elements of A in increasing order $A^{'} = \{a_{q1}, a_{q2}, ..., a_{qm}\}, 0 \leq a_{q1} \leq a_{q2} \leq ... \leq a_{qm}$;
Find the median of $A^{'}$ as $a_{q\mu}, \mu = \lceil \frac{m+1}{2} \rceil$;
Find the minimum value equal to or greater than $a_{q\mu}$ from $D^{'}$ as $d_{p\varphi}^{q\varphi}, \varphi = \underset{\varphi}{argmax}\{d_{p\varphi+1}^{q\varphi+1} < a_{q\mu}\}$;
$A_c = A^{'} \backslash \{a_{q\mu}, a_{q(\mu+1)}, ..., a_{qm}\}$;
$D_c = D^{'} \backslash \{d_{p\varphi+1}^{q\varphi+1}, d_{p\varphi+2}^{q\varphi+2}, ..., d_{pg}^{qg}\}$;
$M_c = \{(pi, qi) | d_{pi}^{qi} \in D_c \,\&\, a_{qi} \in A_c\}$, $n = |M_c|$;
$P_b = zeros(1, n)$, $P_{b,i}$ is the enement of P_b and it means the price of buyer i ;
for $i = 1$ *to* n **do**
  if $r_{qi} > 0$ **then**
    $W_b \leftarrow W_b \cup \{pi\}$;
    $b_w^{qi} \leftarrow b_w^{qi} \cup \{pi\}$;
    if $qi \notin W_s$ **then**
      $W_s \leftarrow W_s \cup \{qi\}$;
    end
    $Del \leftarrow \{(pi, qk) | (pi, qk) \in W_c \& k \neq i\}$;
    $M_c \leftarrow M_c \backslash Del$, $n \leftarrow n - |Del|$;
    $r_{qi} \leftarrow r_{qi} - 1$;
  else
    $\forall k \in b_w^{qi}$, $p_k^b \leftarrow d_{pi}^{qi}$, $P_{b,k} \leftarrow \{p_k^b\}$;
    $Del \leftarrow \{(pk, qi) | (pk, qi) \in W_c \& k \geq i\}$;
    $M_c \leftarrow M_c \backslash Del$, $n \leftarrow n - |Del|$;
  end
end
while $(pi, qi) \in M_c \,\&\, p_{pi}^b = 0$ **do**
  $P_{b,pi} \leftarrow \{d_{p\varphi}^{q\varphi}\}$;
end
$P_b \leftarrow P_b \backslash \{P_{b,i} | p_i^b = 0\}$;
$p_j^s \leftarrow a_{q\mu}, \forall a_j \in W_s$, $P_s \leftarrow \{p_j^s\}$;
$M_w \leftarrow M_c$;
return W_b, W_s, M_w, P_b, P_s;

any $j \in W_s$, we have $a_j \leq a_{q\mu} \leq p_j^s$ and for any $i \in W_b$, we have $d_i^j \geq p_i^b \geq d_{p\varphi}^{q\varphi}$. Thus SDAB is individually rational for buyers and sellers. On the other hand, for any mapping relationship (i, j) in M_w, $p_i^b \geq a_{q\mu}$ is always correct, i.e., $u_a \geq 0$.Therefore, we conclude that SDAB satisfies individual rationality. ■

Theorem 2. *SDAB is truthful.*

Proof: If an auction is monotonous and it gives all winning sellers a reward of the critical ask and charges critical bid to all winning buyers, then it is truthful [21]. Keeping other conditions constant, for biuyer i, if $d_i^j \geq p_i^b$, it will win the service and and if $d_i^j < p_i^b$, it will fail. At this point, we say that p_i^b is critical. Similarly, for seller j, p_j^s is critical when seller j wins the service by submitting $a_i < p_j^s$ and fails to bid by submitting $a_i > p_j^s$, given others' donditions remain unchanged. That is to say, the critical bid is the minimum bid for a buyer to win the auction and the critical ask is the maximum ask for a seller to win the auction, Therefore, critical bid and critical ask are thresholds that determines their auction results. Their value depends on the bids of other buyers ($resp.$ the asks of other buyers) and how to select the winners from the participants. A winning seller always be in the winner set as long as $a_j < a_{q\mu}$. On the contrary, if seller j asks $a_j > a_{q\mu}$, it will not be added into the winning set W_s according to SDAB. Therefore, for all sellers, $a_{q\mu}$ is the critical. On the other hand, there are two kinds of situations for the critical bid.

When the resources of seller j are all exhausted, i.e., $r_j \leq 0$, the highest failed bid (for example, d_{pi}^{qi}) is charged from buyer b_w^{qi}. In this situation, the highest failed bid d_{pi}^{qi} is the critical bid for winning buyers that belong to b_w^{qi}. When buyer pi bids $d_{pi}^{qi} < d_{pj}^{qi}$, it will be sorted after d_{pj}^{qi}. Then d_{pj}^{qi} will be selected instead of d_{pi}^{qi}.

In another situation, buyers in b_w^{qi} have to pay for the service achieved at a price of $d_{p\varphi}^{q\varphi}$, $d_{p\varphi}^{q\varphi}$ is critical for all buyers that belong to b_w^{qi}. For a winning buyerpi, it will remain in the set of W_b if its bid $d_{pi}^{qi} \geq d_{p\varphi}^{q\varphi}$. On the other hand, if buyer pi bids $d_{pi}^{qi} < d_{p\varphi}^{q\varphi}$, it won't be added into the winner set.

Therefore, sellers achieve their critical asks and buyers are charged critical prices in SDAB. Under the constraint of monotonicity, we proved that SDAB is truthful. ■

IV. Performance Evaluation

This section presents the simulation results for evaluating SDAB performance and gives a brief introduction to simulation settings.

A. Simulation Settings

We consider the scenario where the MDs are randomly deployed. 100 MDs and 10 edge severs are assumed to be present in the area. The asks of edge severs and the bid prices for MDs are randomly generated following a uniform distribution within the range of [2,10] and [0,14] without otherwise specified. The computing task is atomic and cannot be divided, and the computing capacity of edge severs are randomly chosen from [2, 10]. The available channels for a base station is about 100, which means at most 100 MDs connected with the base station at the same time constrained by communication capabilities. We may change the number of MDs or edge severs, the bids of MDs, or the asks of edge severs for evaluating the performance of SDAB, which are covered in more detail in other subsections.

B. System Efficiency

Fig. 2 shows the number of successful trades for different numbers of buyers, given the number of edge severs remain

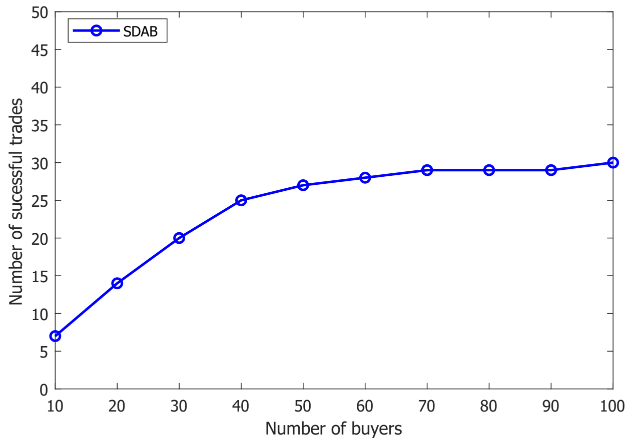


Fig. 2. The number of successful matching over varying number of buyers.

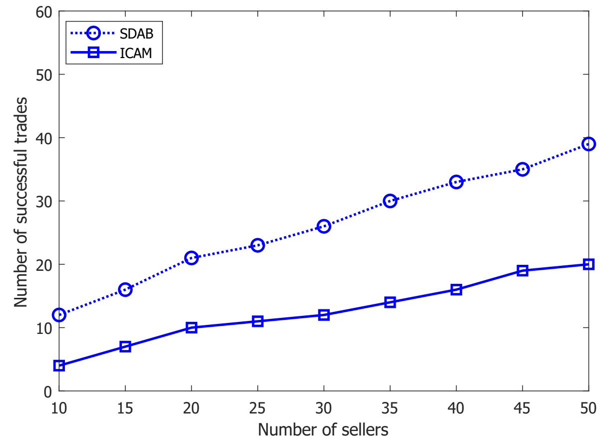


Fig. 3. The number of successful matching over varying number of sellers.

unchanged. As shown in Fig. 2, before about 40 buyers, the number of the successful trades of SDAB grows almost linearly, but then it increases slowly. It is the edge server capacity that constrains the number of successful matching. When buyers grow from 10 to 40, edge servers have sufficient resources to serve the MDs. However as the number of buyers grows, edge servers don't have enough resources. Thus, the rate of growth begins to decline. From this figure, we can understand that when the resources are sufficient, our method can make more than half of the MDs be served.

Fig. 3 compares SDAB and ICAM [19] in terms of the number of the successful trades under different numbers of sellers, given the number of edge MDs remain unchanged (50 MDs) . Because ICAM only considers one edge server serving one MD and one MD simultaneously bidding only one edge server, SDAB performs better than ICAM. Here, we use the number of the successful trades to measure system efficiency. As the figure shows, when the number of MDs reaches 50, the number of successful trades approaches 40 in SDAB while in ICAM, the number of successful trades does not exceed 20. Due to the constraint of breakeven, the number of the successful trades will not exceed half amount of sellers in ICAM. Nevertheless, if one edge server can serve multiple MDs, ICAM will be less affected by breakeven.This is the another reason that SDAB outperforms ICAM.

C. Individual Rationality and Budget Balance

The bids, prices, payments and asks are shown in Fig. 4. The price paid by buyers in the winner set will not exceed their respective bids, while each winning seller receives a payment not less than its ask from the auctioneer. Therefore, SDAB is individually rational. The results show that the edge servers get enough compensation to encourage them to share resources. On the other hand, MDs are allocated the required resources and pay no more than their bid prices for these resources. Therefore, the MDs are also motivated to request resources from the MEC. For the entire mechanism, neither the buyer nor the seller, their utility will not be negative during the

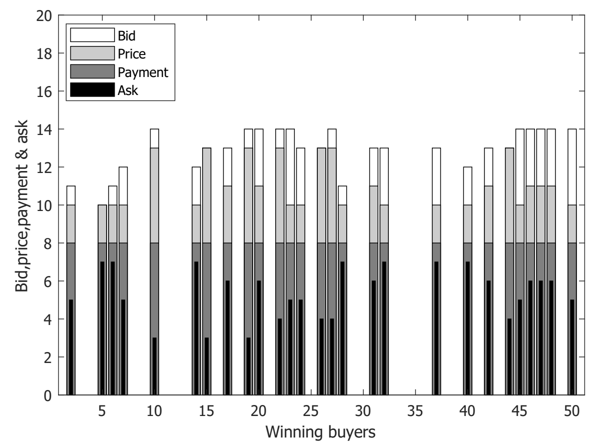
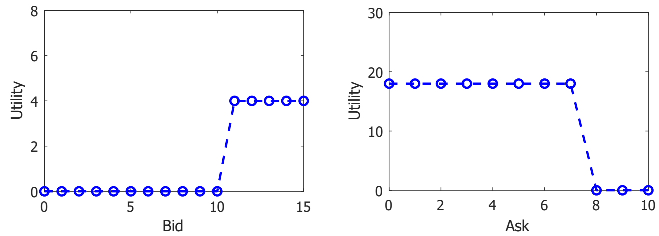


Fig. 4. The comparison of bids, pricing, payment and ask. Each winning buyer is charged a price not higher than its bid, while each winning seller receives a payment not less than its ask from the auctioneer.

auction process. Thus SDAB satisfies budget balance and the utility of the acutioneer $u_a > 0$. This result will encourage the auctioneer to host the transaction.

D. Truthfulness

Fig. 5 depicts the economic property of truthfulness. When the buyer's bid falls below the critical value, it will fail in the auction and its utility will be zero. And once it bids above the critical value, it wins the trade and utility is a fixed constant. Similarly, when the seller's ask lower than the critical value, its utility is a constant value and the utility becomes zero when the seller's ask is greater than the critical value. Since one buyer has multiple bids, in order to verify the truthfulness of the buyer, we take the maximum bid for simulation and set its other bids to 0. According to our algorithm, this setting is reasonable.



(a) Buyer (b) Seller

Fig. 5. The truthfulness of buyer and seller. The utility is zero when the buyer/seller loses the auction. Otherwise, the utility is a constant.

V. Conclusions

In this paper, we joint network economics and resource allocation in MEC to maximize the number of successful trades. Particularly, we designed a double auction scheme in MEC, namely a single-round double auction scheme based on breakeven (SDAB), which proved to satify individual rationality and truthfulness. Although some other economic properties, such as, budget balance, have not been proved, we can see that they are satisfied in our simulation analysis. The simulation results show that the algorithm SDAB can significantly optimize the system efficiency by maximizing the number of successful trades.

References

[1] C. Zhu, V. C. M. Leung, X. Hu, L. Shu, and L. T. Yang, "A review of key issues that concern the feasibility of mobile cloud computing," in *Proc. Physical and Social Computing 2013 IEEE Int. Conf. Green Computing and Communications and IEEE Internet of Things and IEEE Cyber*, Aug. 2013, pp. 769–776.

[2] G. Han, X. Yang, L. Liu, M. Guizani, and W. Zhang, "A disaster management-oriented path planning for mobile anchor node-based localization in wireless sensor networks," *IEEE Transactions on Emerging Topics in Computing*, vol. PP, no. 99, p. 1, 2017.

[3] Z. Liu, Y. Feng, and B. Li, "Socialize spontaneously with mobile applications," in *Proc. IEEE INFOCOM 2012*, Mar. 2012, pp. 1942–1950.

[4] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless communications and mobile computing*, vol. 13, no. 18, pp. 1587–1611, 2013.

[5] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.

[6] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing-a key technology towards 5g," *ETSI white paper*, vol. 11, no. 11, pp. 1–16, 2015.

[7] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1628–1656, 2017.

[8] C. Zhu, H. Wang, X. Liu, L. Shu, L. T. Yang, and V. C. Leung, "A novel sensory data processing framework to integrate sensor networks with mobile cloud," *IEEE Systems Journal*, vol. 10, no. 3, pp. 1125–1136, 2016.

[9] D. Huang, P. Wang, and D. Niyato, "A dynamic offloading algorithm for mobile computing," *IEEE Transactions on Wireless Communications*, vol. 11, no. 6, pp. 1991–1995, 2012.

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

[11] H. Li, G. Shou, Y. Hu, and Z. Guo, "Mobile edge computing: progress and challenges," in *Mobile Cloud Computing, Services, and Engineering (MobileCloud), 2016 4th IEEE International Conference on*. IEEE, 2016, pp. 83–84.

[12] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, "Collaborative mobile edge computing in 5g networks: New paradigms, scenarios, and challenges," *IEEE Communications Magazine*, vol. 55, no. 4, pp. 54–61, Apr. 2017.

[13] S. Paris, F. Martignon, I. Filippini, and L. Chen, "An efficient auction-based mechanism for mobile data offloading," *IEEE Transactions on Mobile Computing*, vol. 14, no. 8, pp. 1573–1586, Aug. 2015.

[14] L. Fawcett, M. Broadbent, and N. Race, "Combinatorial auction-based resource allocation in the fog," in *Proc. Fifth European Workshop Software-Defined Networks (EWSDN)*, Oct. 2016, pp. 62–67.

[15] B. Neethu and K. R. R. Babu, "Dynamic resource allocation in market oriented cloud using auction method," in *Proc. Int. Conf. Micro-Electronics and Telecommunication Engineering (ICMETE)*, Sep. 2016, pp. 145–150.

[16] G. Iosifidis, L. Gao, J. Huang, and L. Tassiulas, "A double-auction mechanism for mobile data-offloading markets," *IEEE/ACM Transactions on Networking (TON)*, vol. 23, no. 5, pp. 1634–1647, 2015.

[17] L. Xu, J. Wang, A. Nallanathan, and Y. Li, "Resource allocation based on double auction for cloud computing system," in *Proc. IEEE 18th Int. Conf. High Performance Computing and Communications; IEEE 14th Int. Conf. Smart City; IEEE 2nd Int. Conf. Data Science and Systems (HPCC/SmartCity/DSS)*, Dec. 2016, pp. 1538–1543.

[18] Y. Zhao, Z. Huang, W. Liu, J. Peng, and Q. Zhang, "A combinatorial double auction based resource allocation mechanism with multiple rounds for geo-distributed data centers," in *Proc. IEEE Int. Conf. Communications (ICC)*, May 2016, pp. 1–6.

[19] A.-L. Jin, W. Song, and W. Zhuang, "Auction-based resource allocation for sharing cloudlets in mobile cloud computing," *IEEE Transactions on Emerging Topics in Computing*, 2015.

[20] J. Trevathan and W. Read, "Privacy and security concerns for online share trading," 2007.

[21] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic game theory*. Cambridge University Press Cambridge, 2007, vol. 1.