A Combinatorial Double Auction Model for Cloud Computing Based on the Average Price of Cloud Resource Packages

Xiaojie ZONG*, Tiezhu ZHANG, Yusa WANG and Linxiao LI

School of Computer Science & Information Engineering, Zhejiang Gongshang University, Hangzhou, Zhejiang 310018, China

Abstract — At present, the cloud computing market is a competitive market where the information is asymmetric. There are multiple providers and users participating in cloud service transactions. But market monopolization and providers excessively pursuing maximal interests pose urgent problems. In order to solve these problems, this paper proposes a model named CDATM. First it analyzes the advantages and disadvantages of two pricing models: the double auction and combinatorial auction. Then it establishes a K-Pricing Scheme to distribute the market surplus to buyers and providers in a ratio of 1:2. Combining the cloud resources' characteristics of complementary, this paper describes the combinatorial double auction of cloud service problem as a 0-1 integer-programming problem, then matches resources and price the cloud products after sorting the resource packages’ average price. Secondly, this paper improves the utility of buyers and providers in the algorithm implementation, so as to get the CDATM model based on the average price of the cloud service combination. Finally, through a comparative analysis using simulation, the validity of the model is verified from the perspective of utility, transaction rate, price stability and supply-demand relationship.

Keywords - Cloud computing service, pricing, combinatorial double auction, CDATM.

I. INTRODUCTION

The management of cloud service resources is a dynamic process. It needs to optimize the allocation of resources and provide resources sharing. It should maintain the balance of supply-demand relationship; at the same time maximize their interests. At present, the cloud computing market is a competitive market where the information is asymmetric. There are multi-providers and multi-users participating in the cloud service transactions. Therefore, how to efficiently complete the cloud service trade and give a proper pricing strategy has become a major problem in the cloud computing market.

Now several cloud service providers have respectively proposed their pricing strategies [1]: Microsoft Windows Azure uses a pricing strategy with fixed ratio; In Amazon EC2, providers can dynamically set prices of resources, and users bid for the resources; Providers of Heroku Add-ones sell their resources through setting prices in advance. Cloud computing resources are alternative, and the same unit of cloud resources has different prices among different providers. The cloud computing market has gradually formed, and now the more important task is to analyze the management issues of cloud computing resources from the perspective of market.

Weinhardt [2] et al. pointed out that, the pricing model of pay-as-you-go is the most widely used model by cloud service providers. They adopt a pricing strategy of fixed proportion in resources, storage and data transmission, but this method cannot price discriminatively according to the users’ different demand. Shang Shifeng [3] et al. established a double auction-pricing model of cloud service for multi-type resources, but it only considers the situation of single provider and single user. Qian Wang [4] et al. focused on the situation that multi-users apply to a single cloud service provider for multi-type resources. They establish the model of combinatorial double auction, but it only considers the corner market with single cloud service provider. Kant Umesh [5] et al. took the situation of multi-providers and multi-users into consideration, and came up with a many-to-many double auction model and a resource allocation algorithm. But due to the grid is open source and free, business model and market mechanism such two management methods are limited when applied to grid computing.

On the basis of the above literatures, we consider that cloud resources are dynamic and complementary in the cloud computing market where the information is asymmetry. Both of the buyers and sellers should apply pay-as-bid auction to multi-type services on the basis of the combination of different kinds and numbers. This paper studies the advantages and disadvantages of double auction and combinatorial auction model, then presents a combinatorial double auction model named CDATM. What’s more, it adopts the K-Pricing Scheme [6] to distribute the market surplus to buyers and providers with a ratio of K. It not only solves the problem of monopoly so as to efficiently complete the trade matching, but also works out an appropriate pricing strategy to reduce the number of transactions and improve the overall utility of transaction.

II. PROBLEM DESCRIPTION

There are three main types of roles in the auction model, including resource provider, resource consumer and auctioneer. In order to facilitate the analysis of cloud service transaction and pricing, this paper sets up three basic assumptions: (1) The prices of buyers and providers are sealed; (2) The information of buyers and providers in the transaction is asymmetry; (3) Cloud service products are leased to the users through the network. However, there is...
transmission loss due to the providers’ regional difference, this paper ignores such considerations so as to simplify the problem.

In the double auction model, the transaction of cloud service is a game issue where the information is asymmetry. This paper draws an equilibrium bidding strategy of cloud service transaction on the basis of Bayesian double auction pricing model. Sharrukh Zaman [7] et al. proposed a combinatorial double auction model where a single provider supplies multi-type cloud services. It uses Greedy Algorithm to get the mechanism of cloud resources allocation and the pricing strategy. Then sort descending the buyers according to the average price of per unit cloud resource. At last, according to the sorting result, the auctioneer picks out the transaction, which satisfies the constraint condition, and at the same time the buyers’ benefit is relatively large. But this model can only ensure that the cloud service providers’ interests are maximum.

This paper introduces $n$ cloud service providers and $m$ buyers to transact $l$ kinds of different cloud services (including CPU, memory, hard disk storage and network bandwidth). We define the utility $U_j$ to be the absolute difference of true price valuation and the final trading prices. This paper improves the utility through the algorithm so as to produce the correct incentives. After the cloud service package $A_j$ and combinatorial bidding $P_j$ are submitted to the cloud resource auctioneer, it adopts the K-Pricing Scheme to calculate the matrix of transaction price, and then comes to the auction results. The second line is the constraint to ensure the providers can provide the corresponding request resources; the third line means that the participants win or lose in the auction when $X_j$ is 0 or 1.

Next, the design and implementation of CDATM’s auction algorithm and pricing algorithm are introduced in detail. The steps are as follows:

1. According to model CDATM and the submitted set $B_i$, the auctioneer finds out the winning provider and buyer, and then comes to the auction results $X_j$.

2. Calculate the average price of each participant’s auction price, and the formula is as follows:

$$aP_j = \frac{P_j}{\sum a_i}$$ (2)

3. Sort the participants according to the average price $aP_j$, calculated by Eq.(2). The providers are sorted in ascending according to the $aP_j$, and we put the serial number into the list $s^{l}_i$. Resources are classified according to the type of product, and we sequentially put them into the list $s^{l}_i$ (It’s a matrix with $nW$ rows and $l$ columns where $nW$ shows a sum of the winning provider). The buyers are sorted in descending according to the average price, and we also put the serial number into the list $b^{l}_i$.

Use K-Pricing Scheme to calculate the transaction price matrix of cloud service, denoted as $T(b,s)$. It shows the price that the bth buyer in list $b^{l}_i$ transacts with the sth provider in list $s^{l}_i$. The formula is as follows:

$$T(b,s) = KaP_{s(b)} + (1 - K)aP_{s(e)}, K \in (0,1)$$ (3)
Distribute and price the cloud services according to the order of list \( sl \). In the matching transaction, the buyer firstly transacts with the first resource provider. If the resource meets the buyer’s needs, then the two parties make a deal. Otherwise, the buyer will transact with the next provider who has such resource until the rest of transactions are completed. The algorithm of CDATM is described as Fig. (1):

**Algorithm CDATM**

1. **Input**: \( A, B, bl, sl, sli \)
2. **Output**: \( P, P_w \), where \( n \) means the number of provider, \( n \) means the number of buyer.
3. **Initialization**: Make \( c = 1, i = 1, v = 1 \), and introduce two matrixes with \( mw \) rows and \( nw \) columns, respectively, denoted as \( Di \) and \( Dl \). \( Di \) is a transaction distribution matrix of \( P_i \), \( Dl \) is a transaction price matrix of \( P_i \).
4. **According to the lists of \( bl \) and \( sl \), determine the transaction price.** In order to evenly distribute the market surplus to both trading parties, set the \( K \) in Eq.(3) equal to 1/2. Compare the absolute values of \( B(i, bl(c)) \) and \( sl(v,i,v) \), where \( B(i, bl(c)) \) means the number of the \( i \)th resource requested by the \( c \)th buyer. In the matrix \( sl \), \( sl(v,i,v) \) means the element in row \( i \) and column \( v \), where \( v \) means the column number of the \( v \)th resource in the list \( sl \).
5. If \( B(i, bl(c)) \leq sl(v,i,v) \)
   \[ A(c, v) = A(c, v) + B(i, bl(c)); \]
   \[ D(c, v) = D(c, v) + B(i, bl(c)) \cdot (aP_i + aP_{svi}); \]
   \( sl(v,i,v) = sl(v,i,v) + B(i, bl(c)) \)
   \( Bl(i, bl(c)) = 0; \)
   Jump to the step ②;
else
   \[ A(c, v) = A(c, v) - sl(v,i,v); \]
   \[ D(c, v) = D(c, v) - sl(v,i,v) \cdot (aP_i + aP_{svi}) / 2; \]
   \( Bl(i, bl(c)) = Bl(i, bl(c)) + sl(v,i,v); \)
   \( sl(v,i,v) = 0; \)
   \( v = v + 1; \)
   Keep operating in step ②;
   The element values of matrix \( sl \) are negative. \( A(c, v) \) means the quantity of product \( i \) provided by the \( v \)th cloud service provider \( s(v) \) to the \( c \)th buyer \( bl(c) \). \( D(c, v) \) is the bargain price of the both side of transaction.
7. **Judge whether the service resources requested by the \( c \)th buyer are met.**
   If All service resources requested by this user are met; Jump to the step ②;
   else
   \( i = i + 1; \)
   Jump to the step ②;
9. **Judge whether the service resources requested by all buyers in the list \( bl \) are met.**
10. **End**

After completing all the transactions, we can get the total revenue and expenses of both sides of transaction, expressed as \( P_s \) and \( P_w \), respectively.

\[
P_n = (\sum_{v=1}^{nw} \sum_{i=1}^{nw} D_i ) P_i, P_m = (\sum_{c=1}^{mw} \sum_{i=1}^{nw} D_i ) P_i
\]

**IV. SIMULATION AND ANALYSIS OF THE ALGORITHM CDATM**

This paper uses Matlab to simulate the algorithm. We set \( A, B, C \) such three different kinds of cloud service products and 15 people participating in the auction. The buyers’ valuations are respectively in the range of \([5, 10], [10, 15], [15, 20] \), and the providers’ valuations are respectively in the range of \([4, 8], [8, 12], [12, 16] \). And we set \( m_i = n_i = 20 \). First of all, assume that both sides of the transaction price honestly. Table 1 shows all situations of transaction, where No.1-15 represent the auction of providers, and No.16-30 represent the bid of buyers.

**TABLE I. PARAMETERS IN THE COMBINATORIAL DOUBLE AUCTION**

<table>
<thead>
<tr>
<th>No.</th>
<th>( A )</th>
<th>( B )</th>
<th>( C )</th>
<th>( P_s )</th>
<th>No.</th>
<th>( A )</th>
<th>( B )</th>
<th>( C )</th>
<th>( P_s )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( -20 )</td>
<td>( -6 )</td>
<td>( -17 )</td>
<td>( -436 )</td>
<td>16</td>
<td>3</td>
<td>13</td>
<td>13</td>
<td>461</td>
</tr>
<tr>
<td>2</td>
<td>( -17 )</td>
<td>( -17 )</td>
<td>( -16 )</td>
<td>( -297 )</td>
<td>17</td>
<td>4</td>
<td>3</td>
<td>105</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>( -16 )</td>
<td>( -14 )</td>
<td>( -4 )</td>
<td>( -259 )</td>
<td>18</td>
<td>3</td>
<td>4</td>
<td>20</td>
<td>462</td>
</tr>
<tr>
<td>4</td>
<td>( -10 )</td>
<td>( -1 )</td>
<td>( -19 )</td>
<td>( -342 )</td>
<td>19</td>
<td>0</td>
<td>20</td>
<td>266</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>( -11 )</td>
<td>( -8 )</td>
<td>( -9 )</td>
<td>( -248 )</td>
<td>20</td>
<td>20</td>
<td>6</td>
<td>11</td>
<td>329</td>
</tr>
<tr>
<td>6</td>
<td>( -3 )</td>
<td>( -18 )</td>
<td>( -14 )</td>
<td>( -331 )</td>
<td>21</td>
<td>18</td>
<td>18</td>
<td>3</td>
<td>469</td>
</tr>
<tr>
<td>7</td>
<td>( -20 )</td>
<td>( -1 )</td>
<td>( -4 )</td>
<td>( -219 )</td>
<td>22</td>
<td>13</td>
<td>19</td>
<td>20</td>
<td>795</td>
</tr>
<tr>
<td>8</td>
<td>( -5 )</td>
<td>( -18 )</td>
<td>( -13 )</td>
<td>( -429 )</td>
<td>23</td>
<td>2</td>
<td>3</td>
<td>15</td>
<td>314</td>
</tr>
<tr>
<td>9</td>
<td>( -17 )</td>
<td>( -10 )</td>
<td>( -4 )</td>
<td>( -240 )</td>
<td>24</td>
<td>1</td>
<td>10</td>
<td>215</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>( -7 )</td>
<td>( -6 )</td>
<td>( -7 )</td>
<td>( -175 )</td>
<td>25</td>
<td>17</td>
<td>20</td>
<td>0</td>
<td>309</td>
</tr>
<tr>
<td>11</td>
<td>( -5 )</td>
<td>( -2 )</td>
<td>( -7 )</td>
<td>( -128 )</td>
<td>26</td>
<td>11</td>
<td>18</td>
<td>19</td>
<td>703</td>
</tr>
<tr>
<td>12</td>
<td>( -17 )</td>
<td>( -8 )</td>
<td>( -17 )</td>
<td>( -437 )</td>
<td>27</td>
<td>17</td>
<td>18</td>
<td>15</td>
<td>722</td>
</tr>
<tr>
<td>13</td>
<td>( -1 )</td>
<td>( -19 )</td>
<td>( -13 )</td>
<td>( -429 )</td>
<td>28</td>
<td>2</td>
<td>10</td>
<td>3</td>
<td>218</td>
</tr>
<tr>
<td>14</td>
<td>( -8 )</td>
<td>( -11 )</td>
<td>( -4 )</td>
<td>( -253 )</td>
<td>29</td>
<td>17</td>
<td>19</td>
<td>3</td>
<td>370</td>
</tr>
<tr>
<td>15</td>
<td>( -9 )</td>
<td>( -16 )</td>
<td>( -6 )</td>
<td>( -315 )</td>
<td>30</td>
<td>18</td>
<td>8</td>
<td>8</td>
<td>326</td>
</tr>
</tbody>
</table>

According to the above data and 0-1 integer programming model in Formula (1), the optimal result is \( x_i = (0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 1) \). Thus we can conclude that the providers whose number are \( 2, 3, 4, 5, 6, 8, 9, 10, 11, 12, 13, 14, 15 \) win the transaction, and the buyers whose number are \( 16, 17, 18, 19, 20, 21, 22, 24, 25, 26, 27, 30 \) win the transaction. According to the transaction model and the pricing algorithm proposed in this paper, we can get the corresponding transaction allocation and price matrix. Finally, buyers and providers are sorted according to the average price \( AP_i \) calculated in the Formula (2). In the
sorted order, the list of cloud service provider is $sl = [2 \ 3 \ 9 \ 10 \ 5 \ 11 \ 6 \ 15 \ 12 \ 14 \ 8 \ 13]$, the list of buyer is $bl = [24 \ 18 \ 16 \ 22 \ 26 \ 27 \ 19 \ 21 \ 17 \ 30 \ 20 \ 25]$. Their total prices of the transaction respectively are as follows:

$$P_A = (475.72, 390.58, 349.70, 233.41, 324.19, 158.96, 417.54, 352.91, 470.92, 256.95, 356.36, 387.68, 334.53).$$

It is the total revenue of cloud service providers.

$$P_B = (152.42, 342.95, 350.36, 618.31, 572.58, 596.98, 234.15, 428.01, 106.25, 348.10, 377.58, 381.75).$$

It is the total expenses of buyers.

Next, illustrate the algorithm by an example. From the Table 1, we can see that the No.24 buyer requests the cloud service combination package $(1,1,10)$ at the price of 215. It means that this buyer would like to buy one unit A, one unit B and 10 units C. However, the top one provider——No.2 provider, provides the combination package $(17,17,6)$. Therefore, it would seem that the No.24 buyer buys the resources of A and B from the No.2 provider, but the resource C of the No.2 provider provided is not sufficient to meet the needs of the buyer. So the No.24 buyer has to buy 6 units and 4 units C separately from the No.2 provider ranking first in the list and the No.3 provider ranking second in the list. The final transaction price can be calculated according to the Formula (4). So the final transaction price of No.24 buyer is 152.42.

All above results are based on the case that the buyers submit the true price valuations. However, if the No.24 buyer attempts to obtain greater benefits by making a false price, that is to say, he reports a price lower than his true price valuation to push down the price of transaction. Then using the algorithm proposed in this paper, we can find that the buyer’s total expenses will increase rather than decrease. Because when the buyer deliberately submits a lower price, on the meanwhile, his position in the list of $bl$ is changed. It means that he will trade with the cloud service provider who has a higher average bidding, thereby increase his total transaction costs. To sum up, this method in a certain extent meets the principle of incentive compatibility. Both of the buyers and providers can’t get higher utility by reporting a false bidding, thereby this model can encourage them to submit their real price valuations.

A. The Utility Analysis Of Both Sides Of Transaction

Calculate the average utility of both sides of transaction according to the above simulation results. The calculating formulas are $U_i = V_i - P_i$, and the results are $AU_i = \frac{U_i}{\sum a_i}$, shown in Fig. (2) and Fig. (3).

From the Fig. (2) and Fig. (3), we can find that the average utility decreases by the bidding price of buyers. When the utility of buyer is disutility, we can motivate buyers to improve their bid price. Similarly, the average utility decreases by the bidding price of providers. That is, when the products are sold at a low price, the cloud computing market will compensate these providers. Thus it can be seen that, this algorithm can effectively complete the transaction and pricing of cloud services products.

B. The Exchange Rate

In this paper, the exchange rate is defined as $RT$ to express the efficiency of the algorithm. In the above examples of simulation results, there are 25 participants winning to make the transaction, that is, there are 5/6 of participants successfully completing the transaction allocation and pricing. The formula of $TR$ is shown below:

$$TR = \frac{\text{length}(sl) + \text{length}(bl)}{N} \times 100\% \quad (5)$$

In this paper, 3 price ranges are set to analyze the transaction rate, as shown in Table 2.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Buyer</td>
<td>[5,15]</td>
<td>[15,20]</td>
</tr>
<tr>
<td></td>
<td>Provider</td>
<td>[5,15]</td>
<td>[15,20]</td>
</tr>
<tr>
<td>2</td>
<td>Buyer</td>
<td>[10,25]</td>
<td>[15,30]</td>
</tr>
<tr>
<td></td>
<td>Provider</td>
<td>[5,20]</td>
<td>[10,25]</td>
</tr>
<tr>
<td>3</td>
<td>Buyer</td>
<td>[15,30]</td>
<td>[20,35]</td>
</tr>
<tr>
<td></td>
<td>Provider</td>
<td>[5,20]</td>
<td>[10,25]</td>
</tr>
</tbody>
</table>

Each number of the buyer and provider is 20, and we simulate 100 times in each price range. The simulation results are shown in Figure. (4):
As can be seen from Fig. (4), in the third price range, about 80% of the 40 participants win from the auction and participant in the transaction. Even though in the first price range, the exchange rate is also about 50%. In addition, with the price range of the buyer is gradually greater than the price range of the provider, the exchange rate of both sides will increase significantly, which is also consistent with the real economic phenomena.

C. The Price Stability

The pricing algorithm should satisfy the stability principle. In this paper, the price stability of the algorithm is measured by means of the average transaction price. The formula is as follows:

\[
\bar{P}_b = \frac{\sum_{j=1}^{b_l}(a_{blj})P_{blj}}{\phi(b)l}, \bar{P}_s = \frac{\sum_{j=1}^{s_l}(a_{slj})P_{slj}}{\phi(s)l} \quad (6)
\]

\(\bar{P}_b\) and \(\bar{P}_s\) respectively mean the average transaction price of buyers and providers. \(\phi(b)\) and \(\phi(s)\) are the numbers of buyers and providers. \(P_{blj}\) and \(P_{slj}\) respectively mean the transaction price of buyers and providers who win from the auction.

This paper takes the first price range in Table 2 for example. We simulate 100 times, and get the average transaction price of each auction. The results shown in Figure (5):

The results show that the average transaction prices of buyers and providers basically fluctuate within 3. It means that this algorithm is a relatively stable pricing strategy.

D. The Analysis Of Supply-Demand Relationship

In economics, an effective pricing method can accurately reflect the supply-demand relationship of market.

In this paper, we define three kinds of cloud computing market, respectively are oversupply, balance between supply and demand, undersupply, as shown in Table 3.

Assume that there are 30 participants and 3 kinds of cloud service in the transaction, and the valuations ranges of these services are [5,20], [10,25], [15,30], respectively.

![Figure. (5). The average transaction price of buyers and providers](image)

![Figure. (6). The average transaction price in three kinds of cloud computing market](image)

![Figure. (6). The average transaction price in three kinds of cloud computing market](image)
analyzes the limitations of double auction pricing model and combined auction pricing model, then establishes a model of combinatorial double auction for the cloud computing named CDATM. With the simulation, we respectively analyze the utility, exchange rate, price stability and the supply-demand relationship of buyers and providers, thereby the proposed algorithm is verified. In addition, although the CDATM algorithm effectively completes the trade matching and gives the pricing strategy, it is limited in some cases that a type of cloud computing service is far more important than other cloud services, and there are large differences among the prices of different cloud services, so it needs to be improved to adapt to different application scenarios. What’s more, this paper doesn’t consider the requirements of QoS (Quality of Service), more efforts should be done in a future study.

ACKNOWLEDGEMENTS
This work is supported by Zhejiang Provincial Natural Science Foundation of China (No.Y13G020041).

REFERENCES