

Negotiation Based Combinatorial Double Auction Mechanism in Cloud Computing

Zakir Ullah¹, Asif Umer¹, Mahdi Zaree², Jamil Ahmad¹, Faisal Alanazi^{3,*}, Noor Ul Amin¹,
Arif Iqbal Umar¹, Ali Imran Jehangiri¹ and Muhammad Adnan¹

¹Department of Information Technology, Hazara University Mansehra, Mansehra, 21300, Pakistan

²Tecnologico de Monterrey, School of Engineering and Sciences, Zapopan, 45201, Mexico

³Prince Sattam bin Abdulaziz University, College of Engineering, Department of Electrical Engineering, Alkharj, 16273, Saudi Arabia

*Corresponding Author: Faisal Alanazi. Email: faisal.alanazi@psau.edu.sa

Received: 22 November 2020; Accepted: 11 April 2021

Abstract: Cloud computing is a demanding business platform for services related to the field of IT. The goal of cloud customers is to access resources at a sustainable price, while the goal of cloud suppliers is to maximize their services utilization. Previously, the customers would bid for every single resource type, which was a limitation of cloud resources allocation. To solve these issues, researchers have focused on a combinatorial auction in which the resources are offered by the providers in bundles so that the user bids for their required bundle. Still, in this allocation mechanism, some drawbacks need to be tackled, such as due to the lower average bid price the users are dropped from the auction process. To solve this problem, we proposed a “Negotiation based Combinatorial Double Auction Mechanism for Resource Allocation (N-CDARA) in cloud computing”. The proposed method negotiates with dropped users. Lower average bid price users are asked by our proposed mechanism to increase their bids, as by the quoted bids they will be dropped by the auctioneer. Most of the users that are close to winning accept the proposal and increase their bid prices. The proposed mechanism is implemented in a CloudSim simulation toolkit. Results are compared with the latest model and performance study shows that in our proposed scheme more users win and get their requested services and the utilization of offered services is increased up to 18.4% than the existing schemes.

Keywords: Cloud computing; combinatorial double auction negotiation; resource utilization; users serve

1 Introduction

With the improvement in human society, the fundamental and necessary services are provided commonly in such a way that everyone can obtain access to them easily. At the present, utilities such as water, electricity, gas, and telephone are important to accomplish daily life resources. These utilities need to be present for consumers at any time. Consumers then pay the providers based



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

on the consumption of the utilities. In 1969, one of the chief scientist of the Advanced Research Project Agency Network (ARPANET) project which seeded the internet, Leonard Kleinrok said; “At present computer networks are still during their infancy, but as they grow up and become sophisticated, we will probably see the spread of computer utilities such that present electric and telephone utilities will serve individual homes and offices across the country”. Computing services are available on-demand in the contemporary society like other daily life services [1].

1.1 Cloud Computing

The cloud computing concept was presented by John McCarthy in 1960. Cloud computing is an emerging trend in the field of IT that enables the movement of data and computational processes from the desktop and portable PCs into a large data center. With the wide use of the internet, now the application can be delivered as a service anywhere in the world. The main advantages of CC are to efficiently use the distributed resources, add them together to gain higher throughput and solve large-scale computation problems [2]. Commonly used definition of cloud computing (CC) is, “A model for enabling convenient, on-demand network access to a shared pool of configurable computing resources e.g., network, servers, storage, applications and services that can be rapidly provisioned and released with minimal management effort or service provider interactions” [3]. Computing resources are being changed to a model composed of services that are commoditized and transferred in such a way as traditional utilities such as electricity, gas, water, and telecommunication. The consumers access their required services, but they do not know the physical location and delivery process of those services [4]. CC refers to both the application delivered as a service over the internet, hardware and system application in the datacenter that furnishes those services. The hardware, data center and software are what we call a cloud. A cloud is known as a public cloud, when it is made available to “pay as you go” manner to the public, while the internal data centers of businesses or other companies that are not available to the public are called the Private Cloud [5].

1.2 Cloud Computing Service Models

Generally, cloud computing provides services such as “Software as a Service” which are delivered via the internet by the cloud application, thus it eliminates the need of installing and running the applications on the user system, “Platform as a Services” that provides the computing platform. All demanding applications by the client are available there. Thus, the developer does not need to buy or install any software or hardware and “infrastructure as a Service” in which the customers do not need to purchase the required server datacenter or network resources. They need to pay only for the time duration they use the services. [2]

1.3 Resources Allocation

Resource allocation is the process of allocating services to the end-users in a cloud market. It is responsible for the performance of all cloud systems. An optimal resource allocation process is necessary to satisfy the bidders in the cloud market. Resources can be defined as “anything provided by a cloud provider as a service, such as CPU power, Storage, Memory, Bandwidth, application, software etc.” To gain maximum advantage of the cloud system, it is essential to provide an efficient resource allocation mechanism to fulfill the user’s requests as well as to maximize the cloud service provider profit [6].

1.4 Auction

Auction is the management by which the potential buyers and sellers compete to buy and sell services. The implementation and decentralization of the auction are easy and suitable for distributed systems. The auction uses the procedure of either one, many, or many to many. In one-to-many procedures, an auction is initiated by one manager and multiple agents can make a bid. In many to many, the auction is initiated by multiple agents and several agents can make a bid. Double auction is the most popular example of many to many auction protocols [7]. Auctions may be single sided in which the users bid for their required resources from multiple available resources in the cloud. A sealed-bid auction is an example of a one-sided auction in which the bidders submit their bids in a specific period. Another type of auction is double-sided, in which the auctioneer receives bids from both users and providers at the same time. This mechanism is used to support the neutral trade between the user and provider [8]. Combinatorial auction is a new technique used in the auction model where the participants bid for the bundle of resources instead of a single resource [9].

1.5 Pricing Models in Cloud Computing

Pricing is the method to decide what the cloud provider will receive for his offered services from the end-users or what the cloud user will pay for his accessed services to the service provider. An efficient and effective pricing model is necessary for commercial cloud services. The model should be valuable for cloud providers as well as for cloud users [9]. From an economic perspective, two main types of pricing models are used in Cloud Computing, the static pricing model, and the dynamic pricing model. In static price, all prices are predetermined and fixed for the whole-time horizon. The time for the offered services is always pre-determined because the services are mostly delivered on a time basis in cloud computing. In a dynamic pricing model, the values are changing frequently depending on some factors like the supply and demand of the resources. It is more useful than static because of its various advantages such as by changes in selling price, it maximizes the profit of providers [10].

1.6 Challenges in Cloud Computing

Cloud computing is mostly used by different industries. There are many demerits in CC that have not been fully addressed. The challenging research issues in cloud computing are automated service provisioning, virtual machine migration, server consolidation, energy management, traffic management and analysis, data security, software framework, storage technology and data management and novel cloud architecture [11]. Bidder drop problem and Resource wastage is also a big challenge because we know that in an auction it is always the demand of cloud providers to utilize maximum resources. For maximum utilization of resources, we need to minimize the bidder drop problem in the auction. To address this problem, we propose a new “Negotiation based, Combinatorial Double Auction Mechanism for Cloud Computing”.

2 Literature Review

We divide our review study into the following sections.

2.1 Single-Sided Auction Based Mechanisms

Vickray auction-based allocation was proposed for the CC in [12]. A Truth-telling scheme was presented in this model. In this auction, the customer’s demand varies between peak and off-peak periods. At the beginning of each period, the users submit their bids to the cloud provider. The

provider determines the prices. The first highest bidder wins the resources under the price of the second-highest bidder. A Resource Batch Matching Auction (RBMA) is proposed in [13] for the optimal allocation of resources. Three valuation metrics were created: market effectiveness, user satisfaction and QoS. The immune evolutionary algorithm was used in this scheme. Based on the historical data of QoS the malevolent bidders are punished in this mechanism. For effective and secure resource allocation a first price sealed bid auction under the game-theoretic framework was proposed in [14]. In this model, each user generates with some level of priorities and thus a single resource may be different for the users according to their priorities. The model has three phases. Each customer seeks a malicious provider in the first phase. The information of the bidders was announced by the provider in the second phase and the third phase, the bidder adjusts their bidding strategy to bid for another resource if his payoff increases by this resource.

2.2 Double Auction Based Mechanisms

An adoptive double auction mechanism (ADAM) was proposed in [15]. To ensure the utility value for the customers and providers, this model trades at their expected price and quantities. Strategy proof, budget-balance and individual rationality are proved in ADAM. User satisfaction and utilization of resources were improved in this model. Sun et al. [16] proposed a Nash Equilibrium-based continuous double auction (NECDA) for resource allocation in cloud computing. In this mechanism, the CDA was presented with the support of Nash Equilibrium. The bidders bid independently of each other and calculate the strategy periodically that results in maximum utility. The auctioneer aimed to accomplish the Nash Equilibrium and maximum social utility. Reference [17] proposed a truthful multi-unit double auction resource allocation mechanism. The mechanism can create high welfare to both, cloud providers as well as cloud users. But in this approach, they only focus on double-sided auction. They have not studied the other categories of the auction such as a combinatorial auction.

2.3 Combinatorial Base Auction

Combinatorial double auction resource allocation was proposed in [7]. They proposed greedy algorithms for calculating the final trade prices. The author claims that their model is an efficient market-based model, because of competition and bidding on unlimited resources on both sides. However, this model was not incentive compatible, some of the most interested bidders were dropped in this model.

Authors in [18] proposed a combinatorial-based auction named as “Combinatorial Economical Double Auction Resource Allocation” for fairness and to maximize overall utilization. The authors determine the genuine providers according to the configuration of CPU and their average bid price and prioritize them. The non-genuine providers were not rejected in this scheme, but they were arranged according to their distance from the genuine provider. But the loss of customers was not taken into consideration in this scheme.

In [19] the author proposed a negotiation-based heuristic for dynamic pricing that considers the behavior of both the service provider and the customer and tries to optimally satisfy both for pricing. In this approach, both agents negotiate directly with each other. In this model an agent creates a limited offer at a specific time, the responding agent can accept or reject the proposal. If it rejects the offer, the agents exchange their roles. But there is no intermediary involved in this model.

For the efficient resource allocation and pricing in CC, a new mechanism was proposed by [9]. The author used an integer linear programming (ILP) algorithm to solve the VMs allocation

problem in a combinatorial double auction. The profit of users and providers was maximized in this model. But the bidder drop problem was not considered in this model. A fair mechanism for combinatorial reverse auction-based cloud market has been proposed by [20]. Priority was assigned to each cloud provider on their bidding information and auction result. If a provider loses more time, he has a higher priority. The higher priority provider has a higher chance of winning. But the priority method was used only for providers. Users are not taken into consideration in this model.

3 Proposed Work

In this section, we formulate our problem and then we discuss the system model and the different entities of the model. We also present here the solution for the problem as negotiation-based combinatorial double auction resource allocation (N-CDARA).

3.1 Problem Formulation

We have solved our problems in the following stages. First, we have run the existing algorithm of CDARA [7], thus we have got the drop bidders. In the second stage, we have checked the average prices of the dropped users and compared them with the provider average prices. The third step calculates the difference between their average prices and multiplies it with the required quantity of the user to get the bid difference of the user for which the user was dropped. In the fourth step, we have asked the users by sending them the bid difference value to increase their bid prices. The auctioneer calculates new bid prices for those users who agree, while the non-agreed users were rejected by the auctioneer. The process was continued till the last user. In the last step, we have run the algorithm again according to the new bid prices of the agreed users. Hence the enhanced bidders got their required resources.

3.2 System Model

There are different entities involved in the model which we have discussed one by one. [Fig. 1](#) presents the entities of the proposed model. Mainly four types of entities are used in the proposed work.

3.2.1 Cloud Customers

Cloud customers are the individuals that access the services and use them according to their needs. The customers have their brokers which work on their behalf.

3.2.2 Brokers

Brokers work on the behalf of the users. Each user possesses a broker. The broker is responsible to send requests to the cloud marketplace to fulfill the user's tasks. The broker generates a bundle of the user-required resources and sends a request to the auctioneer for the required bundle of resources. Brokers are also responsible to generate bid value for the requested bundle.

3.2.3 Cloud Providers

Cloud providers are the personnel or firms that offer their services and charge the user of their services. The cloud provider posts their services on the cloud marketplace and gets the auction result, accepts jobs from the winner users and then provides the services and charges them.

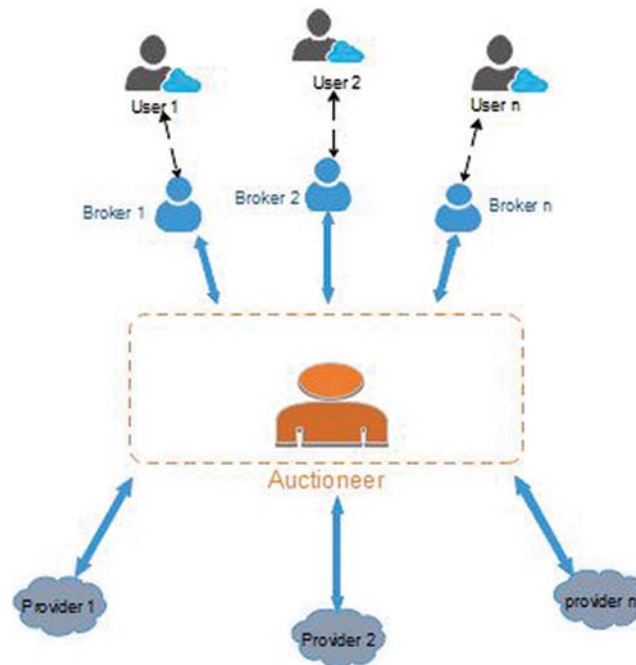


Figure 1: system architecture

3.2.4 Auctioneer

The auctioneer is an intermediary that controls the overall system. The auctioneer gets required and offered bundles of resources from the brokers and providers respectively with their bid values. Then the auctioneer runs the algorithm to determine the winners and calculates the payment amount.

3.3 Proposed N-CDARA

The auctioneer is responsible to apply the N-CDARA method when the bidder drop problem occurs. It will negotiate with dropped users to settle the differences in their prices. We use the existing scheme of CDARA in [7] by adding the N-CDARA procedure. Fig. 2 shows the proposed model, in which users and providers bids for services. CDARA model is used for resources allocation, but before deciding winners our N-CDARA method re-negotiate with users if they increase bid price then may get chance of winning service. The auctioneer starts the auction. The brokers and providers start bidding for their requirements and offer services respectively. Then the auctioneer calculates their bid densities. The broker bid density is calculated concerning their bid value and the time for which the broker needs to use the services. The provider bid density is calculated according to the bid value. Then the auctioneer calculates the average prices and determines the winners of the auction. Finally, the auctioneer will get the dropped bidder list. The auctioneer is responsible to call the proposed N-CDARA method for them. The proposed method negotiates with the dropped bidders. It provides a new pricing list and asks them to enhance their bid values for which they were dropped. The interested users enhance their bid values according to the new price list of the proposed scheme.

The proposed scheme has the following communication phases:

Phase I: The auctioneer announces the start of the auction and informs the bidders.

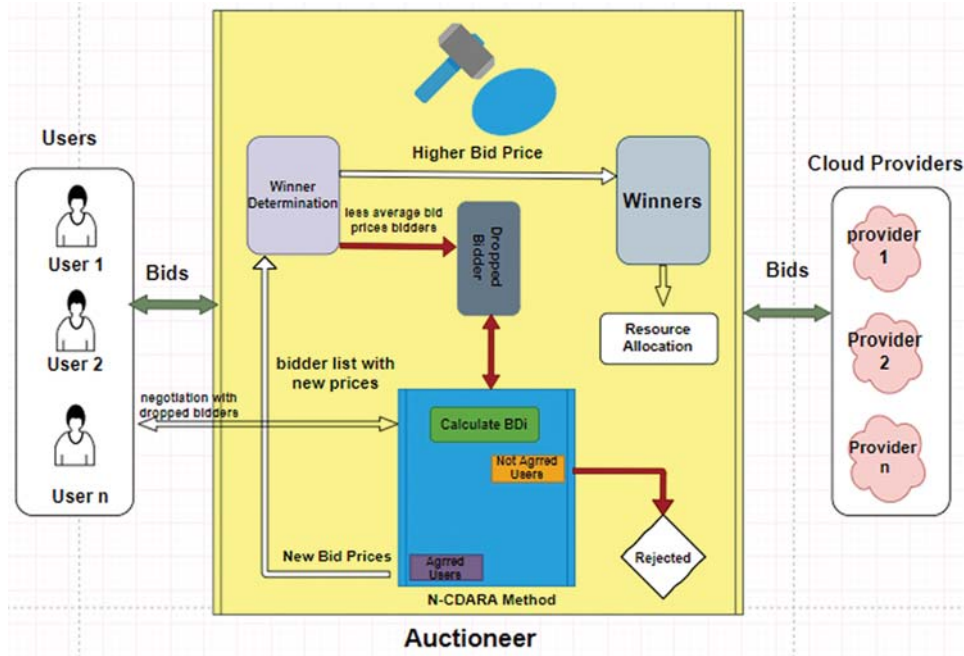


Figure 2: Proposed scheme

Phase II: The brokers and providers send bids.

Broker B_i ($i = 1, 2, 3 \dots n$) generates a bundle of the user-required resources, calculates the price according to the user budget for each bundle and then sends the bid to the auctioneer.

Cloud provider P_j ($j = 1, 2, 3 \dots m$) generate a bundle of offered resources and price value and then sends the bid to the auctioneer.

Phase III: Phase III is the end of the auction. The auctioneer informs both players to stop bidding.

Phase IV: In Phase IV, the winners are determined. The auctioneer determines the winners of the auction by using the data it received from the brokers and providers. The auctioneer sorts the users and providers according to their bid densities to give priority to those users whose bid price is higher and who want to use the services for the long term. We know that there are total M numbers of items for user and provider. By Samimi et al. [7] M is calculated by the Eq. (1) for the user, and the cloud provider by Eq. (2). We have used the same equations. We have calculated user's bid densities by using the Eq. (3) which is used by [7], where b_i is the bid price of user I , M_i is the total number of requested items. t_i is the time duration for which the user needs to use these resources. Bid densities of the providers are calculated by Eq. (4), where b_j is the bid price of provider j and M_j is the total number of offered items by the provider j . Here, we have created two lists of the users, the list of winners (Lw) and a list of drop users (d).

$$M_i = \sum_{k=1}^l (a_k^i \times q_k^i) \tag{1}$$

$$M_j = \sum_{k=1}^l (d_k^j \times q_k^j) \quad (2)$$

$$bdi = \frac{bi}{\sqrt{Mi}} \times t_i \quad (3)$$

$$bdj = \frac{bj}{\sqrt{Mj}} \quad (4)$$

Phase V: In this phase, new bid prices are calculated for the drop bidders who have higher bid densities, but due to less average bid price they were dropped from the auction by the auctioneer represented by DP in this scheme. We have provided a chance to the dropped users by sending them the bid difference due to which they were dropped. For this purpose, first, we have calculated the average prices of providers and users by Eqs. (5) and (6) respectively.

$$api = \frac{b_i}{tq_i} \quad (5)$$

$$apj = \frac{b_j}{tq_j} \quad (6)$$

In Eqs. (5) & (6) tq_i & tq_j is the total number of requested and offered items of user and provider respectively and is calculated by using Eqs. (7) and (8).

$$tq_i = \sum_{k=1}^l q_k^i \quad (7)$$

$$tq_j = \sum_{k=1}^l q_k^j \quad (8)$$

Now, we calculate the bid difference of the users due to which they were dropped by the CDARA [7]. The bid difference (BD_i) of the user is calculated using Eq. (9). After calculating the bid difference of each user, our proposed method will negotiate with them by sending the bid difference (BD_i) values and ask them to enhance their bid values. If the user DP agrees with BD_i value, then the BD_i is added to their bid price (bi) by Eq. (10) to get a new bid price (NB_i) for the user and place it in the agreed user list (L_a). If user dp does not agree with BD_i , the auctioneer rejects it.

$$BD_i = (apd_{iJ} \times q_i) \quad (9)$$

$$NB_i = BD_i + b_i \quad (10)$$

Phase VI: Phase VI determines the winners from the dropped bidders list. Again, we calculate the bid densities of the agreed users in the list (L_a) by using the Eq. (3). Combine the list of winners (L_w) and the list of agreed users (L_a) by Eq. (11). Then we sort the list in decreasing order. Finally, we run the algorithm.

$$CbdL_i = bdL_a + bdL_w \quad (11)$$

Phase VII: Allocation of resources.

The auctioneer allocates resources to the users. Initially, the resources are allocated to the first user in the user's list from the first provider in the cloud provider's list. It was checked that the first provider can fulfill the requested items of the first user in the user's list or not. The quality of the first item requested by the user is compared with the provider's first resource type. The requested amount if available is checked and provided to the user. If the first cloud provider fails to address all the requirements of the first user, the 2nd provider is checked and so on. When all the requests of 1st user are satisfied, the same process is applied for the next user. The method will be continued till the last user.

Phase VIII: Phase VII includes the pricing scheme. We use the same pricing model which is used by [7]. The average trade price of user i and cloud provider j was calculated by the Eq. (12). The average trade price depends on the average price of the user and cloud provider. The price that should be paid by the user i to the cloud provider j was calculated by using the Eq. (13), where aq_i^j is the allocated quantity of resources. The sum of the price paid is the final trade price of the user to the provider which is computed using Eq. (14).

$$ATP_i^j = \left(\frac{ap_i + ap_j}{2} \right) \quad (12)$$

$$PY_i^j = ATP_i^j \times aq_i^j \quad (13)$$

$$FTP_i^j = \sum_{k=1}^l PY_i^j \quad (14)$$

Phase IX: Task allocation and payment.

The broker/user task is allocated to the related provider. The task is performed by the cloud provider and then sends the information about the execution of the task to the broker/user. The broker is responsible to send the payment to the provider.

Algorithm 1: Negotiation based Combinatorial Auction Mechanism

Input: Brokers bids and providers bids

Output: Winners of the auction

Initialization: $I = 1, j = 1$

Step 1: $bidi, bidj$

Step 2: bdi, bdj

Step 3: For ($i = 1; i \leq m; i++$);

```

{
  BDi = (adi, j * qi)
  If (BDi > 0)
  Then
    Ld = Ui
  Else
    Lw = Ui
}

```

End of loop

For dl to dp

(Continued)

```

    Send  $dp$  bid difference from the list  $Ld$ 
    Ask: if  $dp$  agree to increase the price
    Put  $dp$  in list  $La$ 
    Else
    In list  $Lna$ 
    End of loop
Step 4:  $bdLa$ 
Step 5:  $CbdLi = bdLa + bdLw$ 
Step 6: Sort  $CbdLi$ 
Step 7: Exit

```

The input of algorithm 1 are the user's bid and provider's bid. The output of the algorithm is the winner of the auction. i represents the number of brokers and the number of cloud providers is represented by j . The variable k is used as a counter for a resource type. In the first step, the auctioneer announces the start of the auction and informs the users and cloud provider about the start of the auction. In the second step, the bids of the broker and provider are sent to the auctioneer. In the third step, the auctioneer sorts the provider offers and user requests in ascending order for the attribute of the resource type and then calculates the bid densities of each broker and cloud provider. In the next step, the brokers are sorted in descending order and cloud providers are sorted in ascending order according to their bid densities. In the next step, the auctioneer creates a list of winning brokers and a list of drop users which are represented by L_w and L_d respectively based on average bid prices. The bidders that have less than average bid prices due to which they were dropped are placed in the list L_d .

Next, the auctioneer computes the bid difference BD_i of each user in list L_d . Negotiating with less average bid price users by sending them the bid difference value ask them to enhance their bid price. If the user d_p agree, the auctioneer put it in the list of agreed users represented by La otherwise in the list of non-agreed users (Lna) and the list of agreed users and sorts the combined list according to their bid densities.

Algorithm 2: Resource allocation and Pricing for N-CDARA

Input: Combine bid density list of users ($CbdL_i$) and bid density list of providers (bd_j).

Output: Allocation, amount matrix and price matrix.

Step 1: Sort the requests and offers in increasing order based on their attribute values.

Step 2: Sort the user list in descending order and provider list in ascending order according to their bid densities

Step 3: $k = 1$

If ($q[i, k] = 0$) or ($q[j, k] = 0$) or ($a[i, k] > a[j, k]$)

$k = k + 1$;

If ($q[i, k] \leq q[j, k]$)

begin

insert $q[i, k]$ into amount matrix $[i, j]$ for resource k ;

Calculate the price paid of the allocated resource by Eqs. (12) and (13);

insert PY_i^j in the price matrix $[i, j]$ for resource k ;

update $q[j, k] = 0$;

(Continued)

```

    update  $q[i, k] = q[i, k] - q[j, k]$ ;
  End
Else
  Begin
    Insert  $q[j, k]$  in the amount matrix  $[i, j]$  for resource  $k$ ;
    Calculate  $PY_i^j$  of the allocated resource using Eqs. (12) and (13);
    Insert  $PY_i^j$  into the amount matrix  $[i, j]$  for resource  $k$ ;
    update  $q[j, k] = q[j, k] - q[i, k]$ ;
    update  $q[i, k] = 0$ ;
  End
Step 4: If all the requests of  $B_i$  are not satisfied, then:
  Begin
     $j = j + 1$ ;
    go to step 1;
  End
Step 5: If all the requests of  $B_i$  are satisfied, then:
  Exit;

```

The input of algorithm 2 is the combined bid density list of users (*CbdLi*) which have been obtained from algorithm 1 and the bid density list of a cloud provider. The output of the algorithm is the amount matrix and the price matrix which shows the allocated quantities and price paid respectively. In the initial step, the auctioneer sorts the user requests and the provider offers in increasing order according to the attributes, and then sort the broker list in descending order and provider list in ascending order according to their bid densities.

In step second, the first broker requests have been checked in the first cloud provider in the list. The attributes of requested items have been matched with the provider offered item attributes, such as CPU, storage, memory and bandwidth. If the provider can fulfill the requested attribute of the resource type, it would be matched. Otherwise, the next resource type would be checked. If the first cloud provider fails to address all the requests of the first broker, the second provider would be checked for the same purpose. In the next step, when all the requests of the broker were fulfilled, then the next broker request would be accomplished. After addressing the requests of all the brokers in the sorted list, the algorithm is terminated in the last step.

4 Evaluation, Simulation and Scenarios

The detailed analysis of our proposed work was presented by giving the evaluation parameter, their simulation setup and present the scenarios according to the simulation results in this section.

4.1 Evaluation Parameters

The performance of our proposed algorithm was examined based on the following parameters.

4.1.1 Users Served

The number of users that win in the auction. The main focus of this work is to maximize the number of users served in the auction. We can calculate the percentage of users served (PUS) by using the Eq. (15) that win resources in the auction. In Eq. (15), WUn is the number of winner

users and TU is the total number of users.

$$PUS = \frac{WU_n}{TU} \times 100 \quad (15)$$

4.1.2 Services Utilization

The provider offered different types of resources in a cloud market. It is always the demand of the provider to utilize maximum resources in a market. Maximum utilization of resources is the key to the success of a cloud market.

4.1.3 Users Utility

The concession got by the user from the auction is the utility of the user. The user gets utility when the bid price is higher than the price decided by the auctioneer. The utility of the user can be measured by subtracting the final price paid from the bid price with Eq. (16). The overall utility of the users is measured by Eq. (17).

The utility of user i = bid price of user i – final trade price of user i .

$$u_i = b_i - FTP_i \quad (16)$$

$$Tu_i = \sum_{i=1}^m (u_1 + u_2 + u_3 \dots u_m) \quad (17)$$

4.1.4 Provider Utility

The concession got by the provider from the auctioneer is the utility of the provider. We will measure the utility of each provider by subtracting bid price from the final price paid with Eq. (18). Check for all users to whom provider services were allocated. The total utility of the providers can be obtained by Eq. (19).

$$u_j = FTP_j - b_j \quad (18)$$

$$Tu_j = \sum_{j=1}^n (u_1 + u_2 + u_3 \dots + u_n) \quad (19)$$

4.1.5 Fairness

For all the players i.e., users and providers, the double auction mechanism should be fair. Fairness in an auction market reduces the bidder drop problem. Fairness for the users is not included in CDARA. The proposed model is fair for users as well as for providers. In the existing model, fairness for the users is not considered, because some users are very close to the winning users, but they do not know their degree of closeness due to which they should leave the auction. In the proposed model, we have negotiated with them and informed those users about their degree of closeness to win. Our mechanism will give a chance to the lower bid price users.

4.2 Simulation Environment

The creation of a real environment under different configurations and requirements of a system in a repeatable and controllable manner is bothering, expensive and time-consuming for the evaluation of our proposed scheme. Thus, it is not possible to use a real-world cloud environment for the presentation of the proposed work. Hence, the best alternative is to use the simulation tools. CloudSim is a new generalized and extensible framework that allows unified modeling,

simulation and investigation of emerging cloud computing infrastructures and application services in [21]. We have used Cloud Auction (the extension of CloudSim) to test the performance of our proposed scheme. The following experimental setup is built for the evaluation of the proposed system:

There is a total of 27 participants, 20 Brokers and seven providers, but the number of brokers and providers can be increased without any limit. The attributed range is according to the CloudSim range. The user and provider bids are taken from the CDARA [7]. Bid range is 0.012 to 0.1046 cost/MI, processor speed range is 220, 1000 MIPS, the range of storage was 1500 to 40000 MBs, memory range is 256, 512, 1024, 2048 MBs and the bandwidth range is \$ 20 to 100 B/S. The time range is taken as 10 to 60 min. Four types of VM [VM1–VM4] were considered, but it can be increased to unlimited types.

4.3 Scenarios

The following scenarios were created to examine the performance of our proposed N-CDARA model.

4.3.1 Scenario 1 (Allocation and Winner of Auction)

In the existing CDARA model [7], include 27 participants. Seven are the providers and twenty brokers. The author describes that 11 users won in the auction and all 7 providers are the winner of the auction. But, in actual simulation of CDARA only 9 users won in the auction and get resources and 5 providers are the winner who allocates resources to the users. The remaining 11 users and 2 providers were dropped in the CDARA model. Some of the users i.e., B_{12} , B_{13} and B_{16} have higher bid densities, but due to less average bid prices, they were rejected by the CDARA [7]. Some users dropped because of the unavailability of their required resources.

In our proposed work we can calculate the bid difference of those users that has higher bid densities but less average prices, by using the Eq. (20).

$$BD_i = \alpha pd_{ij} \times q_i \quad (20)$$

where BD_i denoted the value of bid difference of the user, αd_{ij} is the average price difference between user and provider and it is calculated by using Eq. (21), q_i is the required quantity of user.

$$\alpha pd_{ij} = ap_j - ap_i \quad (21)$$

$$NB_i = BD_i + b_i \quad (22)$$

After calculating the bid difference of the user (BD_i) we negotiate with the user by sending them the BD value and ask them to increase their bid price. Those users who drop due to less average bid price in CDARA i.e., B_{12} , B_{13} and B_{16} are agreed with the new bid value. Then we calculate the new bid prices (NB_i) for them using Eq. (22) in which b_i is the original bid price of the user. We run the algorithm again hence, they also win resources in our proposed model. Hence in our proposed scheme, the number of winner users is increased. Some users are not agreed with the new bid values and they are rejected. Some of the users i.e., B_9 , B_{15} , etc are dropped due to the unavailability of their required resources because the providers utilize their resources to the users that have higher bid densities. In our proposed scheme the number of winner users was maximized, as 11 users have got their required resources.

4.3.2 Users Served

The focus of our research work is to maximize the number of users served in the auction. In the existing model, the number of users participate in the auction, but few of them win in the auction. In our N-CDARA model, the maximum users win in the auction. We can calculate the percentage of users served (PUS) by using the Eq. (15) that win resources in the auction. WUn is the number of winner users, and TU is the total number of users. Fig. 3 and Tab. 1 show that in our proposed model maximum users served as compare to the existing scheme.

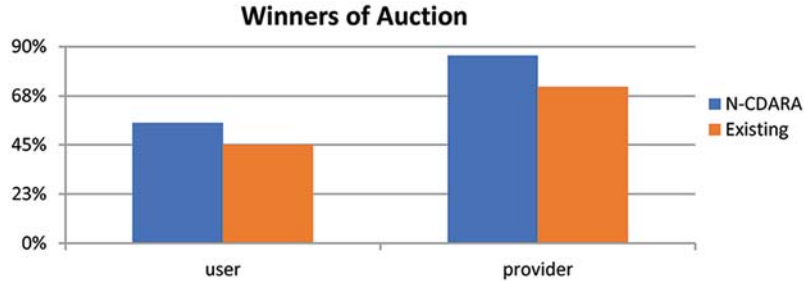


Figure 3: Percentage of the winner users

Table 1: Winner percentage of the users

Model	User (%)	Provider (%)
Existing [7]	$PUS = \frac{9}{20} \times 100 = 45$	$PWP = \frac{5}{7} \times 100 = 71.82$
N-CDARA	$PUS = \frac{11}{20} \times 100 = 55$	$PWP = \frac{6}{7} \times 100 = 85.71$

4.3.3 Scenario 2 Utilization of Resources and Profit of Users and Provider

The utilization of providers' resources is increased in the proposed model therefore the number of winners also increased. More users get an allocation in our proposed scheme so that the utilization of resources is increased. In our proposed model 63% of provider resources are utilized. Tab. 2 and Fig. 4 present the utilization of the provider resources in the existing scheme and the proposed scheme. Our proposed model outperforms the utilization of resources.

Table 2: Utilization of resources

Providers	Total available resources	Utilization in existing model	Utilization in proposed model
P1	14	9	11
P2	10	0	1
P3	12	0	0
P4	10	10	10
P5	12	2	9
P6	9	9	9
P7	9	4	8
Total	76	34	48
Percentage	100	44.7	63.1

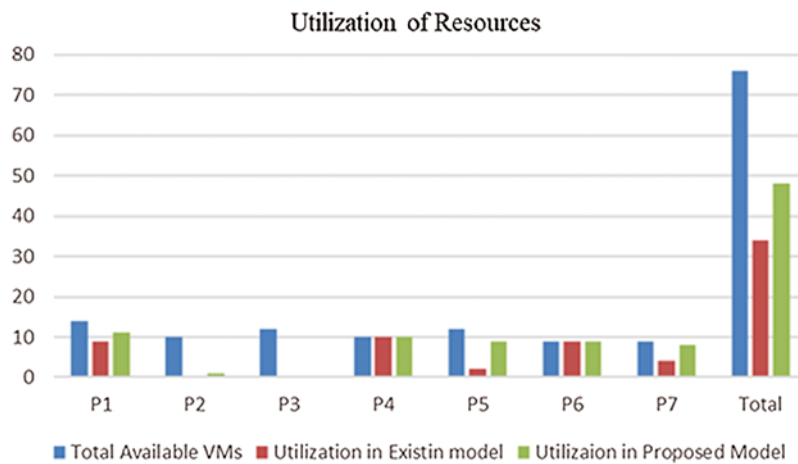


Figure 4: Utilization of provider resources

4.3.4 Users Utility

Users' utility is the concession which they get from the auction. The user gets utility when the bid price is higher than the price decided by the auctioneer. In the existing scheme of CDARA [7] the lower average bid price users were rejected, and some users have negative utility due to low bid price which led to a decrease in the utility of the overall users. In the suggested model all the winner users have positive utilities because there is no such user that bid at a low price. Our proposed mechanism will maximize the overall utility of the user as compared to the existing scheme because maximum users get the resources in the proposed scheme. In our scheme, the lowest bid price provider allocates services to the highest bid price users. The users having higher bid densities get the maximum utility. The utility of the user can be measured by subtracting the final price paid from the bid price, using Eq. (16). The overall utility of the users is measured by Eq. (17). Tab. 3 and Fig. 5 show the utility of the users in our proposed scheme. Fig. 5 clearly shows that our proposed model outperforms as compared to the existing model.

The utility of user i = bid price of user i – final trade price of user i .

Table 3: User utility in the proposed model

User	Bid	Trade price	Utility
U1	0.0308	0.0246	0.0062
U3	0.0318	0.0288	0.0030
U7	0.0422	0.0267	0.0155
U10	0.0471	0.0385	0.0086
U12	0.0399	0.0363	0.0036
U13	0.0456	0.0357	0.0099
U14	0.0461	0.0396	0.0056
U16	0.0399	0.0367	0.0030
U17	0.0201	0.0167	0.0034
U18	0.0239	0.0223	0.0016
U20	0.0533	0.0433	0.0100
Total utility			0.0704

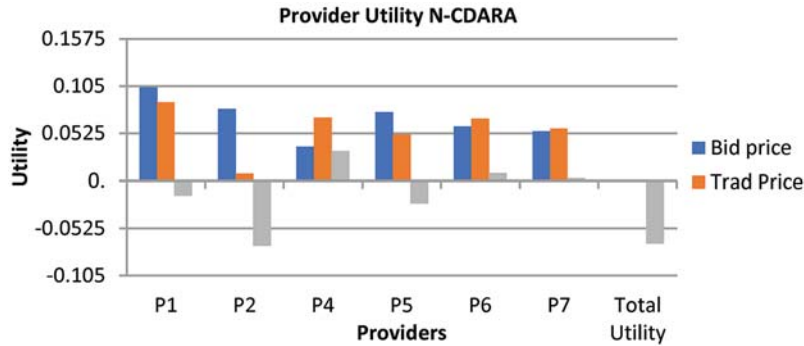


Figure 5: Total utility of providers

4.3.5 Provider Utility

In our scheme, some of the users have agreed to increase their bid price to get their required resources. By increasing the bid price of the users, the provider utility is maximized in this scheme. In our proposed N-CDARA scheme the maximum brokers were to win in the auction due to an increase in the bid price, it directly increases the overall utility of the providers, because maximum offer resources are consumed in the proposed model. Fig. 7 shows the utility of the provider in our proposed N-CDARA model. We can measure the utility of each provider by subtracting the bid price from their final price paid, using Eq. (18). Check for all users to whom provider services were allocated. The total utility of the providers can be obtained by Eq. (19). Fig. 6 shows the utility of the providers in our proposed model while Fig. 7 clearly shows that our proposed model outperforms.

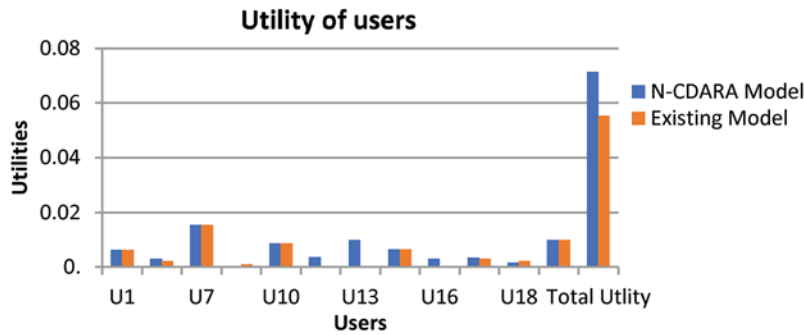


Figure 6: User utility in the proposed model

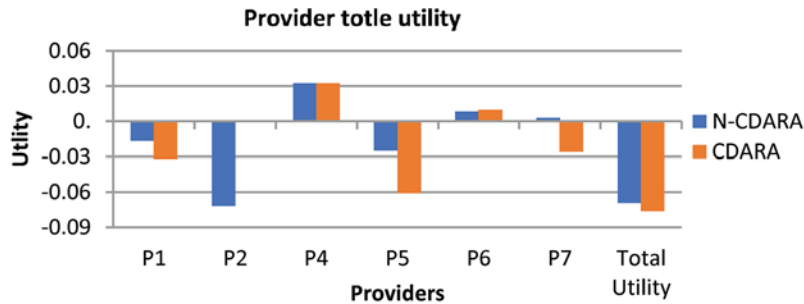


Figure 7: Comparison of provider utility in CDARA and N-CDARA

5 Conclusion

Due to emerging market growth and application of cloud computing, an optimal resource allocation and pricing mechanism is a natural demand of cloud computing that attracts more customers and providers in a cloud market. Researchers in this area have worked to ensure optimal and attractive resource allocation and pricing but still needed optimal algorithms to attract participants. In this work, we have developed a Negotiation based Combinatorial Auction Mechanism for Resource Allocation that ensures to maximize the number of winners in auctions and maximize the utilization of resources. Our proposed scheme uses the advantage of combinatorial auction by adding the negotiation method. We used the combinatorial to provide resources in the form of a bundle and negotiation is used to solve the bidder drop problem in the auction mechanism. In our proposed algorithm more users won the auction and get improved utilities than the existing scheme. For future work we can add the following additions:

It is a challenge for researchers to enhance the system with more users and providers. To design an efficient algorithm that is more attractive for the users and provider. Security architecture is also needed to authorize the users/brokers and providers for auction.

Funding statement: This publication was supported by the Deanship of Scientific Research at Prince Sattam bin Abdulaziz University, Alkharj, Saudi Arabia.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, 2009.
- [2] Y. Jadeja and K. Modi, "Cloud computing-concepts, architecture and challenges," in *2012 Int. Conf. on Computing, Electronics and Electrical Technologies*, Nagercoil, India, IEEE, pp. 877–880, 2012.
- [3] A. Hameed, A. Khoshkbarforousha, R. Ranjan, P. P. Jayaraman, J. Kolodziej *et al.*, "A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems," *Computing*, vol. 98, no. 7, pp. 751–774, 2016.
- [4] R. Buyya, C. S. Yeo and S. Venugopal, "Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities," in *2008 10th IEEE Int. Conf. on High Perf. Computing and Communications*, Dalian, China, IEEE, pp. 5–13, 2008.
- [5] A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski *et al.*, "Above the clouds: A berkeley view of cloud computing," *Dept. Electrical Eng. and Comput. Sciences, University of California, Berkeley, Rep. UCB/EECS*, vol. 28, no. 13, pp. 2009, 2009.
- [6] R. Sudeepa and H. Guruprasad, "Resource allocation in cloud computing," *Int. Journal of Modern Communication Technologies and Research*, vol. 2, no. 4, pp. 19–21, 2014.
- [7] P. Samimi, Y. Teimouri and M. Mukhtar, "A combinatorial double auction resource allocation model in cloud computing," *Information Sciences*, vol. 357, pp. 201–216, 2016.
- [8] F. Sheikholeslami and N. JafariNavimipour, "Auction-based resource allocation mechanisms in the cloud environments: A review of the literature and reflection on future challenges," *Concurrency and Computation: Practice and Experience*, vol. 30, no. 16, pp. e4456, 2018.
- [9] S. A. Tafsiri and S. Yousefi, "Combinatorial double auction-based resource allocation mechanism in cloud computing market," *Journal of Systems and Software*, vol. 137, no. 5, pp. 322–334, 2018.
- [10] B. Pourebrahimi, K. Bertels, G. Kandru and S. Vassiliadis, "Market-based resource allocation in grids," in *2006 Second IEEE Int. Conf. on e-Science and Grid Computing*, Amsterdam, Netherlands, IEEE, pp. 80, 2006.

- [11] Q. Zhang, L. Cheng and R. Boutaba, "Cloud computing: State-of-the-art and research challenges," *Journal of Internet Services and Applications*, vol. 1, no. 1, pp. 7–18, 2010.
- [12] W.-Y. Lin, G.-Y. Lin and H.-Y. Wei, "Dynamic auction mechanism for cloud resource allocation," in *Proc. of the 2010 10th IEEE/ACM Int. Conf. on Cluster, Cloud and Grid, Computing*, Melbourne, VIC, Australia, IEEE Computer Society, pp. 591–592, 2010.
- [13] X. Wang, J. Sun, M. Huang, C. Wu and X. Wang, "A resource auction based allocation mechanism in the cloud computing environment," in *2012 IEEE 26th Int. Parallel and Distributed Processing Symp. Workshops & Ph.D. Forum*, Shanghai, China, IEEE, pp. 2111–2115, 2012.
- [14] H. Hu, Z. Li and H. Hu, "An anti-cheating bidding approach for resource allocation in cloud computing environments," *Journal of Computational Information Systems*, vol. 8, no. 4, pp. 1641–1654, 2012.
- [15] D. Ding, S. Luo and L. Ai, "An adaptive double auction mechanism for cloud resource allocation," *Journal of Communication*, vol. 33, no. Z1, pp. 132–140, 2012.
- [16] D. Sun, G. Chang, C. Wang, Y. Xiong and X. Wang, "Efficient Nash equilibrium based cloud resource allocation by using a continuous double auction," *2010 Int. Conf. on Computer Design and Applications*, Qinhuangdao, China, vol. 1, pp. 94–99, 2010.
- [17] D. Kumar, G. Baranwal, Z. Raza and D. P. Vidyarthi, "A systematic study of double auction mechanisms in cloud computing," *Journal of Systems and Software*, vol. 125, pp. 234–255, 2017.
- [18] R. Singhal and A. Singhal, "A combinatorial economical double auction resource allocation model (CEDARA)," in *2019 Int. Conf. on Machine Learning, Big Data, Cloud and Parallel Computing*, Faridabad, India, IEEE, pp. 497–502, 2019.
- [19] G. Baranwal, D. Kumar, Z. Raza and D. P. Vidyarthi, "A negotiation based dynamic pricing heuristic in cloud computing," *International Journal of Grid and Utility Computing*, vol. 9, no. 1, pp. 83–96, 2018.
- [20] D. Kumar, G. Baranwal, Z. Raza and D. P. Vidyarthi, "Fair mechanisms for combinatorial reverse auction-based cloud market," in *Information and Communication Technology for Intelligent Systems*, Singapore: Springer, pp. 267–277, 2019.
- [21] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose and R. Buyya, "CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and experience*, vol. 41, no. 1, pp. 23–50, 2011.