

Combinatorial Double Auction Winner Determination in Cloud Computing using Hybrid Genetic and Simulated Annealing Algorithm

Ali Sadigh Yengi Kand, Ali Asghar Pourhaji Kazem

Department of Computer Engineering,
Tabriz Branch, Islamic Azad University,
Tabriz, Iran

Abstract—With the advancement of information technology need to perform computing tasks everywhere and all the time there, in cloud computing environments and heterogeneous users have access to different sources with different characteristics that these resources are geographically in different areas. Due to this, the allocation of resources in cloud computing comes to the main issue is considered a major challenge to achieve high performance. Due to the nature of cloud computing is a distributed system to account, comes to business, economic methods such as auctions are used to allocate resources for decentralization. As an important economic bilateral hybrid auction model is the perfect solution for the allocation of resources in cloud computing, on the other hand, providers of cloud resources similarly, their sources of supply combined addressing. One of the problems auction two-way combination with maximum benefit for the parties to the transaction is the efficient allocation of resources to the problem of determining an auction winner is known. Given that the winning auction is NP-Hard. It results in a problem, several methods have been proposed to solve it. In this dissertation, taking into account the strength simulated annealing algorithm, a modified version of it is proposed for solving the winner determination in combinatorial double auction problem in cloud computing. The proposed approach is simulated along with genetic and simulated annealing algorithms and the results show that the proposed approach finds better solutions than the two mentioned algorithms.

Keywords—Cloud computing; double auction; winner determination; genetic algorithm; simulated annealing

I. INTRODUCTION

Resource management is one of the key challenges in cloud computing and cloud data center management [1]. Most cloud providers use fixed price mechanisms to allocate resources to users. But these mechanisms do not provide an efficient and acceptable allocation of resources, and in fact cannot maximize the profitability of cloud resource providers [2], [3]. In such a situation, cloud-based economic models are suited to tune, deliver, and demand resources. An appropriate option for allocating resources in cloud computing is the use of bidding mechanisms. Among the auctioning mechanisms, the most appropriate mechanism used in cloud computing for

allocation and pricing of resources is a combinatorial auction mechanism.

In this way, prices depend on the conditions of demand and supply rather than the fair exchange between cloud providers and users. Considering the above, the use of combinatorial double auction to allocate resources in cloud computing can be a very appropriate model [4]–[7]. The double auction mechanism consists of two steps. The first step is to determine the winning bidder by solving an optimization model that aims to maximize social welfare by taking into account the payment of users and the profit of the providers. The second step is the allocation and pricing of resources among the winners. However, it has been proved that the problem of winner determination of the auctions is a NP-hard problem and hence the researchers are using heuristic, meta-heuristic and greedy methods to solve it [2], [8]. Considering the above issues regarding allocation of resources in cloud computing, in this paper, a modified simulated annealing algorithm has been used to determine the winner of the auction in the allocation of cloud resources. The modified simulated annealing algorithm in each step, instead of using a neighboring solution, uses several neighboring solutions. This change in the base simulated annealing algorithm leads to early convergence and improves the final solution found. In the rest of this paper, in Section 2, related works about economic models for resource allocation in cloud computing are introduced. In Section 3, formal definition of the problem is provided. The proposed mechanism is outlined in Section 4 and the simulation and experimental results are evaluated in Section 5. Finally, Section 6 concludes the paper.

II. RELATED WORKS

Economic models provide different policies and tools for allocating resources in cloud systems. In cloud computing, users compete with other users as well as resource owners with other resource owners [9], [10]. Economic models can be based on the transaction or payment of the resource price. In cloud computing, providers and owners of resources with financial incentives provide their users with cloud resources. Taking into account the points mentioned, the use of decentralized methods is a good way to manage resources in cloud computing. Economic solutions are appropriate because

they have a decentralized structure and also motivate the owners of the resources to participate in their resources in the cloud [11], [12]. Another economic model is that both the user's goals and the objectives of the owners of the resource are taken into consideration in the process of resource allocation.

To date, several market-based resource allocation models and algorithms have been proposed for cloud computing environments. In the rest of this section some of them have been discussed. Wang et al. [13] conducted a study for resource allocation using an English combinatorial auction in cloud computing environments. The resource marketing price was resolved by an English combinatorial auction model, which is concentrated principally on maximizing the seller's profit and reducing the execution time for the winner determination. In 2013 [14], a virtualized resource allocation mechanism to assign CPU resources in virtualized machines was proposed. This work tried to overcome the unfairness issue of resource allocation in cloud computing. This work is essentially concentrated on enhancing the system resource utilization, and it is not considered to be of benefit to the user and service provider.

Another deficiency of this model is that it was restricted to virtual machine and different types of resources were not considered.

Xu presented CDA-CCRA, a new cloud computing resource allocation model based on combinatorial double auction mechanism for more effective resource utilization in cloud computing [15]. The CDA-CCRA model can simultaneously satisfy the users and providers requirements and significantly reduce transactions. Sabzevari et. Al have been proposed one double combinatorial auction based resource allocation approach for cloud computing environments [2]. The main goal of their study is to allocate economic resources in a way that lead to increase social welfare. Their proposed approach uses imperialist competitive algorithm for winner determination and genetic algorithm for resource allocation and payment schemes.

III. PROBLEM DEFINITION

Auctions that bidders can offer a combination of resources has recently been taken into consideration. Compared to a non-trading auction, the combinatorial auction has a high performance. In the cloud computing environment distributed resources, including computing resources, storage resources, network bandwidth, and so on, compete with each other to execute user's work, and as a result, a combinatorial auction is appropriate for allocating resources. In double auction, both buyer and seller can submit their offers. Compared to a one-way auction in which several buyers compete for goods sold by a vendor, a double auction prevents of monopolies. The combinatorial double auction offers not only the benefits of a combinatorial auction, but also the needs of both buyers and sellers and therefore is suitable for cloud resource allocation. The purpose of the combinatorial double auction is to maximize the overall profit by taking into account this limitation that the number of units selected from the resources in the buyer's combined packages does not exceed the number of units provided by the vendors.

Suppose there is a R resource set containing k resources. After both parties offer their offers to the broker, the broker must perform the auction which is known as the winner determination of the auction problem. This problem is described in (1), (2) and (3).

$$\text{Max} \sum_{j=1}^n P_j x_j \quad (1)$$

$$\sum a_{ij} x_j \leq 0, \forall i \in K \quad (2)$$

$$x_j \in \{0,1\}, \forall j \in \{1,2,\dots,n\} \quad (3)$$

The set of proposed packages is $\bar{B} = \{B_1, B_2, \dots, B_j, \dots, B_n\}$, where n is the number of combined packets of resources. Each bid B_j is (a_j, p_j) , where $a_j = (a_{1j}, \dots, a_{ij}, \dots, a_{kj})$ and a_{ij} represents the number of units requested from resource i . Also, p_j is the offered price for package j . If $p_j > 0$, it is a buyer's offer, and if $p_j < 0$, then it is considered as the seller's offer. Also, if $x_j = 1$, that is, the packet is assigned and if $x_j = 0$, that is, not assigned. Finding the best x_j values to maximize formula (1) with the constraints (2) and (3) is the same as determining the winner of the auction, which is considered as programming 0-1 and is an NP-hard problem.

IV. PROPOSED APPROACH

In this section, the various stages of the proposed approach, which uses the hybrid genetic and simulated annealing algorithm to determine the winner of the auction in the allocation of cloud computing resources, is presented.

A. Hybrid Genetic and Simulated Annealing

In order to increase the efficiency of the simulated annealing algorithm, various researchers have tried to combine this algorithm with genetic algorithm using different methods, and the results of the combination of these two algorithms show an increase in efficiency. To combine these two algorithms, the total number of iteration of the hybrid algorithm is shared between the simulated annealing and genetic algorithms. The genetic algorithm performs the first half of the iteration of the hybrid algorithm and the simulated annealing algorithm uses the best chromosome of the final population of genetic algorithm as the initial solution. In addition, the simulated annealing algorithm performs the other half of the iteration of the hybrid algorithm and finally converges to the optimal solution. Considering that the simulated annealing algorithm does not start with a completely randomized solution, and in fact the best solution obtained from the genetic algorithm is considered as an initial solution for it, it will be seen a significant increase in the quality of the final solution.

B. Encoding

An important step in evolutionary algorithms is how to encode and display a solution. Each problem solution in the proposed approach is an array of binary numbers of length n , in which n is the number of offers. Array members as mentioned, there are binary numbers in which 1 means

acceptance of the offer and 0 means the denial of the relevant offer. Fig. 1 illustrates an example of a solution for eight offers.



Fig. 1. An example of an offer (n=8) and the corresponding solution.

C. Fitness Function

Fitness function is one of the important concepts in every heuristic and meta-heuristic approach. In determining the winner of the auction, the suitability of a solution is obtained from the sum of the winning bidder's bid. The fitness function is shown in (4).

$$\sum_{j=1}^n \text{Price}_j x_j \tag{4}$$

where $x_j \in \{0,1\}$

D. Selection Operator

One of the methods for selecting chromosomes in the genetic algorithm is roulette wheel selection. In the proposed approach, this method is used for the selection operator. This approach is because all individuals are mapped to neighborhoods based on their fitness level. The size of the area is determined by each individual according to its fitness size and then a random number is generated and, depending on the size of the number, the individual is selected. This process is repeated so that the desired number of parents (reproductive population) is provided.

E. Crossover Operator

Genetic algorithm of the proposed approach uses uniform crossover. In uniform crossover, for each gene of a chromosome a random number is produced between 0 and 1. If the generated random number is less than 0.5, the gene is inherited from the first chromosome and otherwise it inherited from the second chromosome. The second child's chromosome is also obtained by using reverse mappings. Fig. 2 shows how to perform a uniform crossover operator on two sample chromosomes.

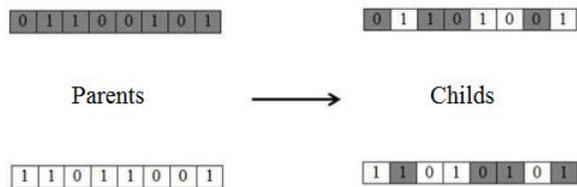


Fig. 2. How to make a uniform crossover on two sample chromosomes.

F. Mutation Operator

In the genetic algorithm of the proposed approach to apply the mutation operator, a random gene is selected from the chromosome, and its value, which is a binary number, is complemented. In fact, if the value of the selected gene is zero, it is converted to one and vice versa.

G. Temperature Initialization and Reduction

Initialization of temperature as well as its reduction in each iteration of simulated annealing algorithm is crucial step. The initial value of the temperature is considered T_{in} , and (5) is used to reduce it.

$$T_{p+1} = \alpha T_p \tag{5}$$

In (5), α which is the coefficient of temperature reduction is a real value in the interval (0,1). If the value is close to 1, it causes a slow decrease in temperature, and thus allows the algorithm to search a large space of solutions and accepts many of the displacements to the optimal solution. After several experiments and taking into account the items mentioned in the proposed approach, the value of α is considered equal to 0.992.

H. Generation of Neighboring Solutions

The generation of neighboring solutions in the simulated annealing algorithm is generally carried out using one or more approaches from known mutation approaches. In the proposed hybrid algorithm, neighboring solutions are generated by random selecting the one component of current solution and completing its value.

V. SIMULATION AND EXPERIMENTAL RESULTS

To evaluate the proposed approach, Matlab software is used for simulation. In addition, to compare the performance of the proposed approach, genetic and simulated annealing algorithms are also simulated. To carry out all tests, a Dell computer with a Core i5, 2 GHz processor and 4 GB of main memory was used.

Taking into account the fact that the proposed method uses a hybrid meta-heuristic algorithm, the results of the convergence as well as stability experiments are presented in the rest of this section. Also, the results of the proposed approach, which is named GASA in the rest of this section, have been compared with the results of genetic algorithm (GA) and simulated annealing (SA). In the convergence experiment, GASA, SA and GA algorithms are executed for two different test scenarios, and Fig. 3 and 4 show the results. In the first scenario, the number of participants is 200, and in the second scenario, the number of participants is 500. In the graphs, the horizontal axis indicates the iteration number of the algorithm and the vertical axis shows the value of the fitness function.

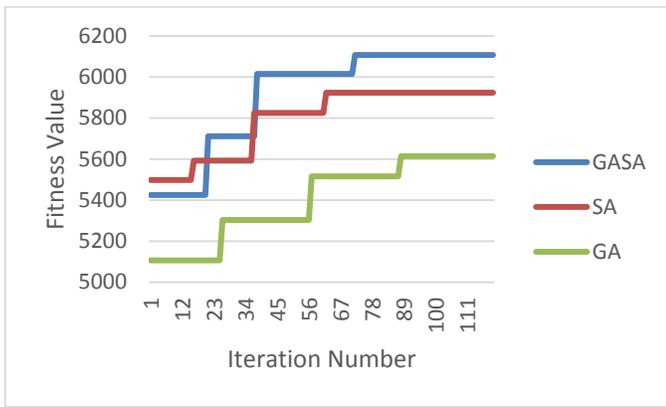


Fig. 3. Convergence test (Scenario 1).

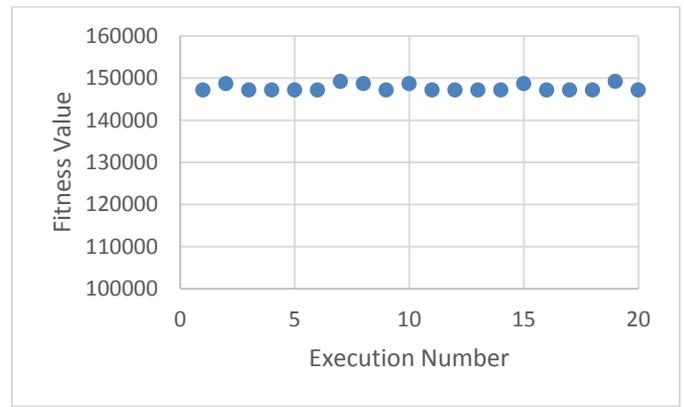


Fig. 6. Stability test (Scenario 2).

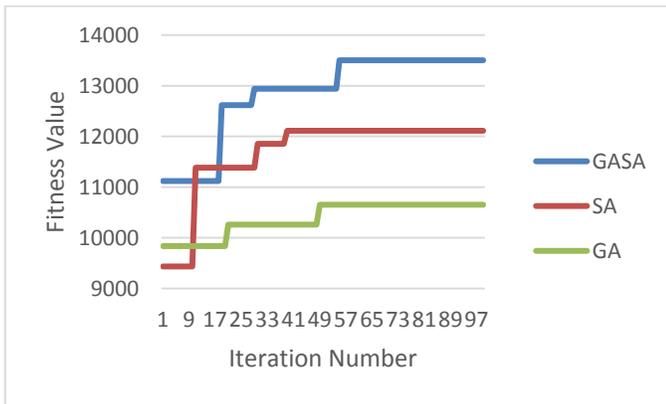


Fig. 4. Convergence test (Scenario 2).

The results of convergence test show that the proposed approach has a good convergence rate and finds better solutions compared to the two SA and GA algorithms.

Meta-heuristic algorithms have a nondeterministic and random nature, so it is necessary to examine the stability of these algorithms. The stability of an algorithm is whether the algorithm generates the same responses for various and different executions. To verify the stability of the GASA algorithm for the two scenarios mentioned, the algorithm is executed 20 times, and the value of the fitness function in each run is shown in Fig. 5 and 6. The horizontal axis in the diagrams shows the execution number of the algorithm and the vertical axis of the fitness value.

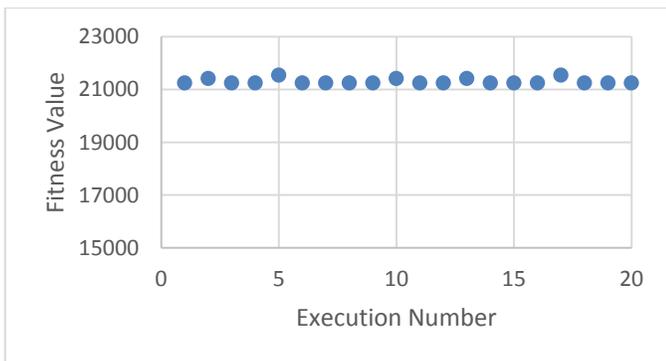


Fig. 5. Stability test (Scenario 1).

Examining the results of the stability test shows proper stability of the proposed approach which converges to optimal solution every time the algorithm is executed.

VI. CONCLUSION AND FUTURE WORKS

Resource management is one of the key challenges in cloud computing and cloud data center management. Most cloud providers use fixed price mechanisms to allocate resources to users. But these mechanisms do not provide an efficient and acceptable allocation of resources, and in reality they cannot maximize the profitability of cloud resource providers. In such a situation, cloud-based economic models are appropriate for the regulation, presentation and demand of resources. In this paper, combinatorial double auctioning has been used to allocate resources in cloud computing. Given the fact that the issue of determining the winner of the auction is in the category of NP-hard problems, in this paper, a combination of genetic and simulated annealing algorithms are used to solve it. The integration of two genetic and simulated annealing algorithms allows for better solutions to the problem. The results of the experiments performed in the MATLAB environment showed that the proposed approach has a good convergence rate and is also very good in terms of stability. Also, the results of experiments performed on two different scenarios showed that the proposed approach produces more suitable solutions compared to the two genetic simulated annealing algorithms.

In the future, researchers can use new meta-heuristic algorithms such as forest optimization algorithm, krill herd optimization algorithm and etc. to optimize the winner determination problem in combinatorial double auction in cloud computing.

REFERENCES

- [1] A. M. Sampaio and J. G. Barbosa, "Chapter Three - Energy-Efficient and SLA-Based Resource Management in Cloud Data Centers," in *Energy Efficiency in Data Centers and Clouds*, vol. 100, A. R. Hurson and H. Sarbazi-Azad, Eds. Elsevier, 2016, pp. 103–159.
- [2] R. A. Sabzevari and E. B. Nejad, "Double Combinatorial Auction based Resource Allocation in Cloud Computing by Combinatorial using of ICA and Genetic Algorithms," *International Journal of Computer Applications*, vol. 110, no. 12, 2015.
- [3] H. Wang, H. Tianfield, and Q. Mair, "Auction Based Resource Allocation in Cloud Computing," *Multiagent Grid Syst.*, vol. 10, no. 1, pp. 51–66, 2014.

- [4] X. Deng, P. Goldberg, B. Tang, and J. Zhang, "Revenue maximization in a Bayesian double auction market," *Theoretical Computer Science*, vol. 539, pp. 1–12, 2014.
- [5] I. Fujiwara, K. Aida, and I. Ono, "Applying double-sided combinatorial auctions to resource allocation in cloud computing," in *Proceedings - 2010 10th Annual International Symposium on Applications and the Internet, SAINT 2010*, 2010, pp. 7–14.
- [6] F. Nassiri-Mofakham, M. A. Nematbakhsh, A. Baraani-Dastjerdi, N. Ghasem-Aghaee, and R. Kowalczyk, "Bidding strategy for agents in multi-attribute combinatorial double auction," *Expert Systems with Applications*, vol. 42, no. 6, pp. 3268–3295, 2015.
- [7] Y. LIU, Y. LIU, X. MA, and K. LIU, "Solving {WDP} in combinatorial double auction based on trading strategy," *The Journal of China Universities of Posts and Telecommunications*, vol. 19, Supple, pp. 148–152, 2012.
- [8] F. Gorbazadeh and A. A. Pourhaji Kazem, "Hybrid Genetic Algorithms for Solving Winner Determination Problem in Combinatorial Double Auction in Grid," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 1, no. 2, 2012.
- [9] M. Mihailescu and Y. M. Teo, "Dynamic resource pricing on federated clouds," in *Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, 2010, pp. 513–517.
- [10] J. Stößer, D. Neumann, and C. Weinhardt, "Market-based pricing in grids: On strategic manipulation and computational cost," *European Journal of Operational Research*, vol. 203, no. 2, pp. 464–475, 2010.
- [11] A. Haque, S. M. Alhashmi, and R. Parthiban, "A survey of economic models in grid computing," *Future Generation Computer Systems*, vol. 27, no. 8, pp. 1056–1069, 2011.
- [12] L. Rodero-Merino, E. Caron, A. Muresan, and F. Desprez, "Using clouds to scale grid resources: An economic model," *Future Generation Computer Systems*, vol. 28, no. 4, pp. 633–646, 2012.
- [13] X. w. Wang, X. y. Wang, and M. Huang, "A resource allocation method based on the limited English combinatorial auction under cloud computing environment," in *2012 9th International Conference on Fuzzy Systems and Knowledge Discovery*, 2012, pp. 905–909.
- [14] C. Jiang, L. Duan, C. Liu, J. Wan, and L. Zhou, "VRAA: virtualized resource auction and allocation based on incentive and penalty," *Cluster Computing*, vol. 16, no. 4, pp. 639–650, 2013.
- [15] J. Xu, "A Cloud Computing Resource Allocation Model Based on Combinatorial Double Auction," in *2016 3rd International Conference on Information Science and Control Engineering (ICISCE)*, 2016, pp. 5–8.