Cloud Resource Combinatorial Double Auction Algorithm Based on Genetic Algorithm and Simulated Annealing

Bing Hu[1], Lin Yao[1], Yong Chen[2], and Zinxin Sun[1](✉)

[1] Key Laboratory of Broadband Wireless Communication and Sensor Network Technology, Nanjing University of Posts and Telecommunications, Nanjing, China
sunzx@njupt.edu.cn

[2] Nanjing LongYuan Microelectronic Company Limited, Nanjing, China
andychenyin@163.com

Abstract. In this paper, on the basis of the analysis of common market model and some economic theories in the cloud computing resource management process, we propose a cloud resource management model based on combinatorial double auction. In order to solve the winner determination problem (WDP) in the combinatorial double auction, a cloud resource combinatorial double auction algorithm based on genetic algorithm and simulated annealing algorithm is proposed. Simulation results reveal that the algorithm combines genetic algorithm with simulated annealing algorithm (SAGA) outperforms genetic algorithm on fitness value and stability, and as the number of bidders increase, the solution have higher fitness value can be obtained.

Keywords: Cloud computing resource · Combinatorial double auction · Genetic algorithm (GA) · Simulated annealing (SA)

1 Introduction

Cloud computing is the mainstream way to provide web services. Based on the internet platform, it is primarily using virtualization technology to provide customers with a flexible, dynamic web services. Many researchers predicted that "the core competition of cloud computing in the future is in cloud datacenter" [1]. Cloud datacenter is a concentration place used for containing the computing equipment resources, and provides energy and maintenance for the computing equipment. It can be constructed separately, located within other buildings or distributed in multiple systems that in different geographical locations. Cloud resources together to serve multiple customers through a multi-tenant pattern. Although the resources exist in distributed sharing way in the physical, it is presented to users in the form of a single overall in the logical eventually [2]. The resource scheduling technology for cloud datacenter is the core of the cloud computing applications, and the key technology for cloud computing to realize

J.-H. Lee and S. Pack (Eds.): QShine 2016, LNICST 199, pp. 435–445, 2017.
DOI: 10.1007/978-3-319-60717-7_43

large-scale applications, improve system performance and reduce energy consumption.

Cloud computing can conveniently on-demand access to a common set of configurable computing resource (such as, network, servers, storage device, application program and service). These resources can be quickly provided and released, while minimizing management cost or the interference of service provider. For the users, the management of computing resources is transparent. Users obtain the service they needed by paying to the resource providers without purchasing, maintaining and managing the infrastructure. Consequently, it is crucial to select appropriate providers based on the users requirement to complete the allocation of resource.

The researches of grid computing and distribution computing are worth cloud resource scheduling algorithm using for reference. According to the scheduling characteristics of cloud computing, many different scheduling strategies have been proposed. Resource scheduling problem is a NP-hard problem. A number of heuristic algorithms have been proposed to achieve a linear optimal. With an aim towards the biodiversity of resources and tasks in cloud computing, Zuo *et al.* propose a resource cost model and an improved ant colony optimization algorithm to solve the multi-objective optimization-scheduling problem [3]. Based on the diversity of tasks and the dynamic factors of resources, a multiqueue interlacing peak scheduling method is proposed to balance loads and improve the effects of resource allocation and utilization effectively in [4]. In order to accurately reflect the resource states, a self-adaptive threshold based dynamically weighted load evaluation method, which is evaluates the load state of the resource through a dynamically weighted evaluation method, is presented in [5]. However, since cloud computing services have many QoS constraints, practical scheduling scheme of cloud computing system is very few.

According to the economic characteristics of cloud computing, a cloud resource management model based on the economics is designed in [6]. And auction model is widely used in the cloud computing environment, because it is easy to realize and the required price information is smaller. Auctions can be categorized into single-side auctions, double-side auctions and combinatorial auctions according to the number of providers and users and the amount of resources. Combinatorial double auction is a combination of combinatorial auction and double auction. In the combinatorial double auction, both sides submit bids for multiple items. Compared with other auction mechanism, combinatorial double auctions solve the problem that bidder and auctioneer have unequal status, at the same time, reduce the auction times and improve the trade efficiency [7].

The key problem in combinatorial double auction is the winner determination problem (WDP). The process for solving combinatorial double problem not only involve the process of two-player game, but also result in large number of feasible solutions since the uncertainty of combination. If we simply adopt the method of exhaustion, the efficiency is very low. Moreover, it has been proved that the combinatorial double problem is a NP-hard problem. Xia et al. [8] reduce a general combinatorial double auction to a combinatorial single-sided auction, which is a multi-dimensional knapsack problem. Hsieh and Liao [9] take advantages of the surplus of combinatorial double auctions to reward winners based on the surplus of auctions, and proposed a computationally efficient approximate algorithm to tackle the complexity issue in combinatorial double auctions. In this paper, we propose a combinatorial double auction model based on genetic algorithm and simulated annealing algorithm, combined with the advantages of the two algorithms to solve the cloud resources allocation problem based on combinatorial double auction.

The organization of this paper is as follows. A cloud resource management model based on combinatorial double auction is proposed in Sect. 2. In Sect. 3, the algorithm of combinatorial double auction for cloud resources based on SAGA is elaborated. Section 4 presents the simulation results. Finally, conclusions are presented in Sect. 5.

2 Cloud Resource Management Model Based on Combinatorial Double Auction

The cloud resource management model based on combinatorial double auction is described in Fig. 1. This model is consisted of cloud resource consumer, cloud resource service agent, auctioneer, and cloud resource provider. Cloud Resource Provider (CRP) and Cloud Resource Consumer (CRC) are two main members in the market-oriented cloud resource management. Generally, CRC regards the Cloud Resource Agent (CRA) as its representative to complete all the work. Auctioneer formulates auction rules, CRC and CRP use different strategies to achieve their respective purposes under the precondition of abiding by the rules.

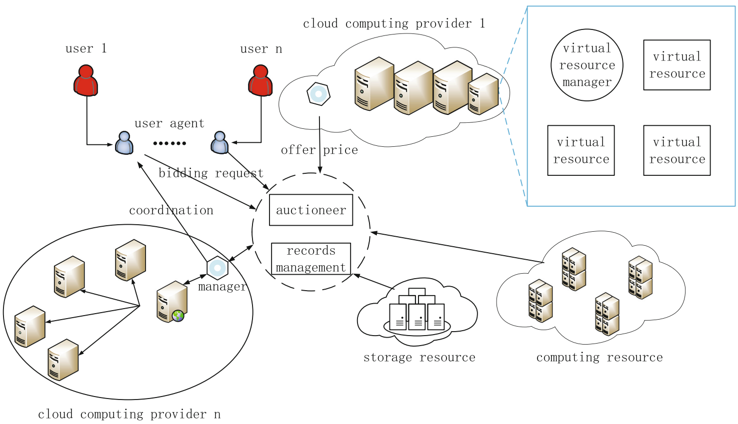

Fig. 1. Cloud resource management model based on combinatorial double auction.

The main auction steps in the cloud resource management are as follows.

(1) CRP provides price for the auctioneer.
(2) CRA submits a bid to the auctioneer.
(3) Auctioneer decides allocation policy according to the corresponding strategies.
(4) The CRP and the CRA who complete the assignment can coordinate with the requests related to the performance via Service Level Agreements (SLA).
(5) If coordination is successful, the auction is over, else, the auctioneer will choose other CRP, until there is no CRP can complete or a CRP is selected to complete a secondary assignment of CRA.

3 Cloud Resource Combinatorial Double Auction Algorithm Based on SAGA

3.1 Problem Description

In the process of the auction, buyers and sellers provide their own combinatorial bid document for the auctioneer, the bid document contains the number of the goods that they need and the price or the bid price of the goods. The auctioneer integrates the bid document, under the premise of ensuring that the number of each goods that buyer needs is not exceeding the number that seller provides, selecting an appropriate allocation scheme to maximize the social surplus.

Assuming that in the process of the auction there are m bidders and k types of different cloud resources (including CPU, memory, hard disk, bandwidth and so on) are auctioned by n auctioneers. Then the total number of participants in the auction is N, $N = m + n$.

Definition 1. $\mathbf{a}$ *is the resource combinatorial bundle provided by the resource providers and users, resource bundle of the j-th participant denote as* $\mathbf{a}_j = (a_{1j}, \cdots, a_{ij}, \cdots, a_{kj})$. a_{ij} *is the resource quantity of i submitted by the participant j, if $a_{ij} > 0$, it denotes that participant j is a buyer and the number of the i-th resource that j requested is a_{ij}; if $a_{ij} < 0$, it denotes that participant j is the resource provider and the number of the i-th resource provided by j is a_{ij}.*

Definition 2. p_j *is the amount the bidder is willing to pay for bundle j, if $p_j > 0$, it is regarded as a buy bid; if $p_j < 0$, it is regarded as a sell bid.*

Definition 3. *The set of participant auction items is* $B = (B_1, B_2, \cdots, B_i, \cdots, B_n)$, *bid document of the j-th participant B_j can be represented as* $B_j = (\mathbf{a}_j, p_j)$.

Therefore, the auctioneer can describe the problem in the following form:

$$\max \sum_{j=1}^{N} p_j x_j \tag{1}$$

$$s.t. \sum_{i=1}^{N} a_{ij} x_j \leq 0, \ \ \forall i \in I \tag{2}$$

$$X = [x_1, \cdots, x_n], \ \ x_i \in \{0, 1\}, \ \forall j \in \{1, \cdots, N\} \tag{3}$$

where x_j stands for the results of allocation, if $x_j = 1$, it denotes that the j-th bids win the bid; if $x_j = 0$, it denotes that the j-th bids lose the bid. Formula (1) is the objective function, formula (2) is the constraint function, and formula (3) simplifies the problem into 0-1 programming problem. The objective of the model is to maximize social surplus while satisfying the constraint that the resource number of the winning buyer does not exceed the number supplied by the winning seller, i.e. supply is greater than demand. So the problem is the 0-1 programming problem, but also a NP-hard problem.

3.2 Algorithm Description

Currently, genetic algorithm (GA) is still one of the best methods to solve the winner determination problem (WDP). GA has good performance in global search, it can find out all the solutions in the solution space in a short time, while without falling into a rapid decline snare of local optimal solution, and distributed computing can be carried out conveniently to speed up the solution by using its intrinsic parallelism. Simulated annealing algorithm (SA) is a stochastic searching optimization algorithm based on Mente Carlo iterative solution strategy. Its starting point is the similarity between the annealing process of solid material and the general combinatorial optimization problem. Starting from an initial temperature, with the decreasing of temperature, the optimal solution is searched in solution space combined with the probabilistic jumping property, i.e. the local optimal solution can jump out probabilistically and tend to global optimal eventually.

GA converge to the optimal solution of the problem with probability 1. However, in practical application, the poor ability of searching the local optimal and prematurity phenomenon will be came out in GA. SA is a stochastic algorithm, which can avoid the problem falling into a local optimal solution. But its control ability to the whole search domain is rather poor so that the global optimal solution is not always found out. Hence, a combination of genetic algorithm and simulated annealing algorithm is proposed in this paper. Take advantages of the strong global search capability of genetic algorithm and strong local search capability of simulated annealing algorithm, during each genetic process, simulated annealing method is used for each individual.

Firstly, the algorithm obtains an initial population randomly, and then searches the global optimal solution via an improved genetic algorithm, if a more optimal solution is not found through genetic algorithm, local optimal search is performed by simulated annealing algorithm. When the initial temperature of the simulated annealing algorithm reduced to zero, genetic algorithm is used for global search again. Iterate this process until satisfy the end condition.

Algorithm 1. Cloud resources combinatorial double auction algorithm based on SAGA

```
Initialize the variables of GA and SA
create an initial population randomly
   for i from 1 to generation number
       for j from 1 to population size
         select parents
         create new_solutions with applying crossover and mutation on parents
          Δt = fitness( parents) - fitness( new_solutions)
       if Δt < 0
         new_solutions accept to new generation
       else
         if exp(Δt/T) > rand(0 ~ 1)
           new_solutions accept to new generation
         else
           parents go to new generation
         end if
       end if
   end for
   decrease T
   if the stop conditions are satisfied stop the algorithm
end for
```

First of all, the specific content of the genetic algorithm and simulated annealing algorithm is defined.

(1) Chromosome coding

Assuming that the number of people participated the auction is N and the chromosome X to be solved is represented by a binary string with length N, where $X = [x_1, \cdots, x_n],\ x_i \in \{0, 1\},\ \forall j \in \{1, \cdots, N\}$, x_j represents the allocation results, if $x_j = 1$, it denotes that the j-th bids win the bid; if $x_j = 0$, it denotes that the j-th bids lose the bid. For example, assume $X = 10110110...01$, it means participants 1, 3, 4 and so on win the bid, others lose the bid.

(2) Fitness function

Fitness function is the only standard for judging the quality of the solution. In this paper, the value of the objective function serves as the fitness function, i.e. $f(X) = \sum_{j=1}^{N} p_j x_j,\ \ x_j \in \{0, 1\}$ represents the size of the obtained social surplus when the current chromosome is $X = [x_1, \cdots, x_n],\ x_i \in \{0, 1\}$. The greater the value of $f(X)$ is, the closer the X get to the optimal solution.

(3) Parents selection

In this paper, parents selection is selected by calculating the proportion of fitness value of each individual in the total fitness value of all the individuals.

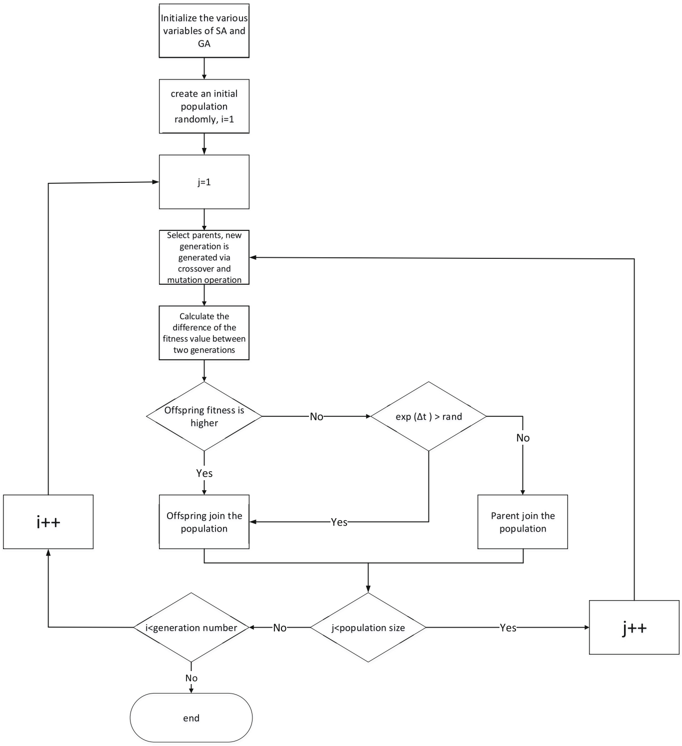

Fig. 2. Flow chart for cloud resources combinatorial double auction algorithm based on SAGA.

The larger the proportion is, the greater possibility of the individual is selected. So, the probability that the current group $P = [X_1, \cdots, X_N]$ is selected can be expressed as

$$p(X_j) = \frac{X_j}{\sum_{i=1}^{N} X_i} \tag{4}$$

(4) Crossover

Nonuniform arithmetic crossover is adopted in crossover operation. Suppose that the individuals who are going to cross are X_1^s and X_2^s respectively, then two new individuals are generated via a crossover.

$$\begin{cases} X_1^{s+1} = \alpha X_2^s + (1-\alpha)X_1^s \\ X_2^{s+1} = \alpha X_1^s + (1-\alpha)X_2^s \end{cases} \tag{5}$$

$$\alpha = e^{(-\alpha_0 S/s)} \tag{6}$$

where α_0 is the crossover coefficient, S represents the maximum iteration number of genetic algorithm, s represents the number of the current iteration.

(5) Mutation operation

This paper adopts the method of the mutation position invert, that is, change the original gene 1 to 0 in a probabilistic manner, and the original gene 0 is mutated into 1.

(6) Simulated annealing algorithm

After the crossover and mutation operation, simulated annealing algorithm is performed.

(7) End condition

When the fitness value of the optimal individual reached the optimal solution or the number of iterations reached the maximum iteration number, output the optimal solution, and the iteration is over.

The specific algorithm is described in Algorithm 1.

Flow chart for cloud resources combinatorial double auction algorithm based on SAGA is shown in Fig. 2.

4 Simulations and Results Analysis

Since CloudSim was the most advanced simulator among the simulation environments, and it had great properties such as scaling well and having a low simulation overhead [10], in this paper, the CloudSim toolkits was selected as the experimental simulation tool. Before the test, the parameters involved in SAGA need to be set. Table 1 lists the parameters value needed in the algorithm.

Table 1. Main parameters of algorithm

Parameter name	Parameter value
Population size	200
Chromosome number	16
Crossover probability	0.5
Mutation probability	0.05
Population genetic number	20
Changes on temperature T	T = T * 0.9

Suppose there are three kinds of resources A, B and C in the auction. Each buyer and seller offer their bid documents, which including the resource number and price, to the auctioneer. Combinatorial resource bundles for buyer and seller are listed in Table 2. The total number of participants is 16, the number of users is 6, and the remaining is providers.

Table 2. The sample of combinatorial resource bundle

Number	1	2	3	Price	Number	1	2	3	Price
1	0	3	3	104	2	4	3	3	136
3	3	3	4	144	4	2	0	1	33
5	4	3	2	124	6	1	5	1	93
9	−2	−3	−1	−80	10	−1	−3	0	−36
11	−2	−2	0	−38	12	0	−1	−3	−70
13	−3	−3	−1	−93	14	−2	0	−3	−71
15	0	−3	−3	−76	16	−3	−1	−1	−54

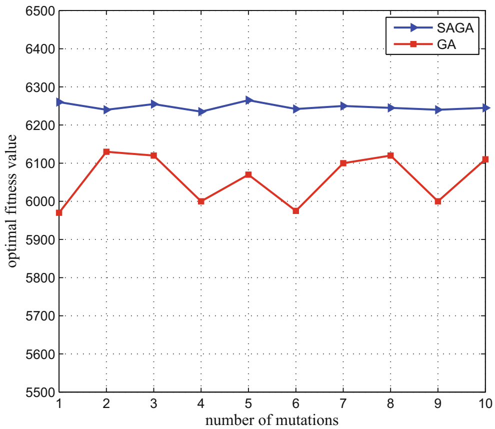

Fig. 3. The convergence of fitness value of the optimal solution for the SAGA and GA.

In order to judge the performance of SAGA algorithm, SAGA algorithm and general GA was compared to observe their convergence. Suppose there are 400 bidders in the simulation experiments. After 10 mutations, Fig. 3 shows the convergence of fitness value of the optimal solution for the genetic algorithm (GA) and genetic algorithm mixed with simulated annealing algorithm (SAGA). We can observe that SAGA has higher fitness value than GA, and the difference among every fitness values of SAGA is smaller than that of GA. Hence SAGA has better stability.

Then optimization results and computing time are compared between SAGA and GA under different number of participants. 10 groups of test samples with different number of bid documents are selected in the test process. Their fitness value and computing time is presented in Figs. 4 and 5 respectively. Although the computing time of SAGA is longer than that of GA with the number of bidders increase, SAGA can obtain the solution with a higher fitness value.

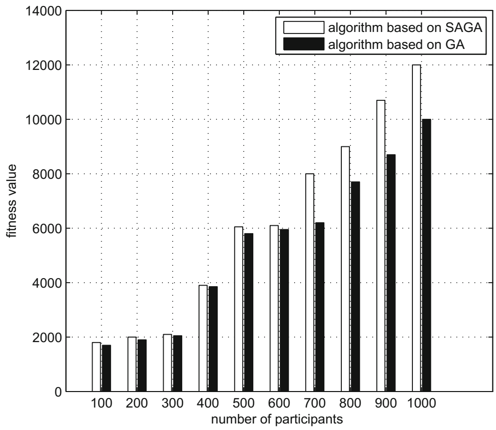

Fig. 4. Fitness value under different number of participants.

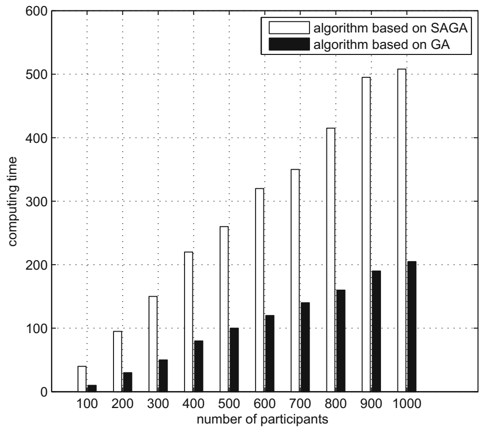

Fig. 5. Computing time under different number of participants.

5 Conclusion

The purpose of cloud computing is to realize collaborative work and resource sharing, while the heterogeneity and dynamic of various resources in the cloud computing and diversity of user requirements make the resource management abnormal complexity under the environment of cloud computing. However, due to the economic characteristics of cloud computing itself, the allocation algorithm based on economic model become a research hotspot in the cloud computing resource allocation. In this paper, we propose a cloud market model based on combinatorial double auction, and in order to solve the winner determination problem in the combinatorial double auction, a cloud resource combinatorial double auction algorithm based on SAGA is presented. The simulation results clearly illustrated that the proposed method has higher fitness value and stability than that of GA, and the proposed method can obtain the solution with a higher fitness value as the bidders increase.

Acknowledgement. This paper was supported by the National Natural Science Foundation of China (Nos. 61170276, 61373135); Project for Production Study and Research of Jiangsu Province (Grant No. BY2013011); Science and Technology Enterprises Innovation Fund Project of Jiangsu Province (Grant No. BC2013027); Key University Science Research Project of Jiangsu Province (Grant No.12KJA520003); Natural Science Foundation of Jiangsu Province of China (Grant No. BK20140883).

References

1. Barroso, L.A., Clidaras, J., Holzle, U.: The datacenter as a computer: an introduction to the design of warehouse-scale machines. Synth. Lect. Comput. Archit. **8**(3), 1–154 (2013)
2. Armbrust, M., Fox, A., Griffith, R., Joseph, A.D.: A view of cloud computing. Commun. ACM **53**(4), 50–58 (2010)
3. Zuo, L., Shu, L., Dong, S., et al.: A multi-objective optimization scheduling method based on the ant colony algorithm in cloud computing. IEEE Access **3**, 2687–2699 (2015)
4. Zuo, L., Dong, S., Shu, L., et al.: A multiqueue interlacing peak scheduling method based on tasks classification in cloud computing. IEEE Syst. J. **PP**(99), 1–13 (2016)
5. Zuo, L., Shu, L., Dong, S., et al.: Dynamically weighted load evaluation method based on self-adaptive threshold in cloud computing. Mob. Netw. Appl. 1–15 (2016)
6. Buyya, R., Yeo, C.S., Venugopal, S.: Market-oriented cloud computing: vision, hype, and reality for delivering it services as computing utilities. In: 10th IEEE International Conference on High Performance Computing and Communications (HPCC), pp. 5–13 (2008)
7. Samimi, P., Teimouri, Y., Mukhtar, M.: A combinatorial double auction resource allocation model in cloud computing. Inf. Sci. (2014)
8. Xia, M., Stallaert, J., Whinston, A.B.: Solving the combinatorial double auction problem. Eur. J. Oper. Res. **164**(1), 239–251 (2005)
9. Hsieh, F.S., Liao, C.S.: Schemes to reward winners in combinatorial double auctions based on optimization of surplus. Electron. Commer. Res. Appl. **14**(6), 405–417 (2015)
10. Son, J., Dastjerdi, A.V., Calheiros, R.N.: CloudSimSDN: modeling and Simulation of software-defined cloud data centers. In: 2015 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), pp. 475–484 (2015)