International Journal of Grid Distribution Computing
Vol. 8, No.5, (2015), pp.117-126
http://dx.doi.org/10.14257/ijgdc.2015.8.5.11

Research of Resource Allocation in Cloud Computing Based on Improved Dual Bee Colony Algorithm

Wu Ju-Hua

(Computer and Information Engineering College of Xinxiang University Xinxiang HeNan 453000 China)
hnxxwjh@yeah.net

Abstract

The allocation of resources in cloud computing has always been the focus of research, and as for how to improve the implementation speed, response time and utilization rate of system, a resource allocation scheduling optimization method in cloud computing based on improved double bee colony algorithm is proposed in this paper. To introduce difference algorithm in artificial ant colony and improve its double bee colony can avoid the bee colony to fall into local optimum, while by introducing chaos algorithm, the convergence speed and precision of the optimal solution in the improved algorithm can be accelerated and the best resources in clouding computing can be found quickly to a certain extent. The simulation experiments show that the algorithm in this paper has improved the algorithm performance efficiently, which not only optimizes the resource allocation ability in cloud computing system, but also improves the resource utilization rate in cloud computing.

Keywords*: Resource Allocation in Cloud Computing, Double Bee Colony, Chaos Algorithm, Difference Algorithm*

1. Introduction

Currently, cloud computing is a model to regard IT resources as services, because more and more enterprises do not need to get resources from the Internet by investing in network infrastructure [1-2]. It has always been an important criterion in measuring the performance of clouding computing as how to reasonably schedule and allocate resources in cloud computing, and particularly, it is the focus of cloud computing as how to deal with the allocation and scheduling of multi-users' multi-tasks in cloud computing. In Literature [3], an elastic load balancing mechanism of cloud computing based on green computing resources strategies, and the experimental results show that under this load balancing mechanism, the overall quality of service has improved significantly. In Literature [4], a resource allocation algorithm based on fuzzy quotient space theory is proposed, and this algorithm can effectively satisfy users' task QoS, and improve resource utilization rate. In Literature [5], virtualization technologies are used to categorize resources to form multi-level resources, and complete resource scheduling with different strategies. The simulation experiment has shown that resource utilization rate has been improved. In Literature [6-7], a random integer programming is proposed to optimize resources, and make dynamic allocation of resources by examining resource overhead of various stages. In Literature [8], genetic algorithm, basic ant colony algorithm and resource scheduling algorithm are proposed to allocate cloud computing resources for users reasonably with the target of the completion time of the best overall task.

Artificial bee colony (ABC) [9] is a random optimization algorithm proposed in 2005 and based on bees' group intelligent search. It has the characteristics of simple parameter setting, and easy implementation, etc. but it is also easy to fall into local optimum like other intelligent algorithms. In this paper, the ABC is proposed to be combined with the differential evolution algorithm on the basis of ABC algorithm to obtain a dual-colony

ISSN: 2005-4262 IJGDC

structure, which can ensure the independent optimization of two colonies. Meanwhile, information exchange is conducted, which not only can maintain the diversity of this colony algorithm, but also maintain search capability of the overall situation. The test functions show that the improved algorithm can improve the convergence speed and time. By the correspondence of the optimum location of bee colony in ABC algorithm to the resource-processing capacity of nodes in cloud computing, time can be saved and the efficiency of resource allocation in cloud computing can be improved efficiently.

2. Cloud Computing Model

It is an overall problem as how to allocate resources in cloud computing more reasonably, because it needs to satisfy multiple task requirements put forward by cloud customers as possible as time can allow, and make resource allocation more reasonable while maintaining high utilization rate of the system. It is the key of the entire cloud computing environment efficiency to allocate resources as possible and effective as possible. From the perspective of task allocation of virtual machine in cloud computing, the following formula (1) is satisfied:

$$s.t. \sum_{\substack{1\le i\le n \\ 1\le j\le m}} x_{ij}t_{ij} + x_{ij}c_{ij} + x_{ij}s_{ij} \le F \quad (1)$$

Herein, $M = \{1, 2 \ldots m\}$ represents the number of virtual nodes, $N = \{1, 2 \ldots n\}$ represents the number of tasks, $\min_t F_{\min}$ represents the maximum objective function with the minimum completion time in cloud computing environment, x_{ij} represents that the task i occupies the virtual node j, 1 means occupied, and others mean vacancy. t_{ij} means the time of task i on virtual node j, c_{ij} means the costs of task i on virtual node j, s_{ij} means resource utilization of task i on virtual node j.

3. Basic Algorithm

3.1. Bee Colony

In artificial bee colony, there are three main kinds of bees. The first kind is call way-directing fees, who find sources of food. The role of way-directing bees is to adopt some certain probabilities to exchange with other bees through the location and quality of food sources. The second kind, also called accompanying bees, determine the quality of food source through the first way-directing bees, and their main role is to determine whether the later bees will collect this food. The third kind of bee, also called scout bees, further collect the new food.

In bee colony, the process of finding food is one to selecting the optimal solution, and then the last food source is the space value of optimizing solutions to the problem. In artificial bee colony, initialize N honey-collecting phases of food source, and in the process of finding food by way-directing bees, they will always compare the quality of food that has already been found with those found last time, rank yields of these food sources, and send the ranking to accompanying bees, who will use the rotation algorithm to select these food sources according to the yields of these food sources. Then the result will be that food with higher yields will have more probability to be selected and after determining the food sources, accompanying bees will generate new food sources around it so as to make comparison and get the food source with better quality.

In bee colony, the probability of honey choice is:

$$P_t = \frac{F(\omega_i)}{\sum_{t=1}^{S} F(\varpi_t)} \quad (2)$$

In the formula, ω_i represents the i honey source, $F(\omega_i)$ is the fitness function value at ω_i, herein, and i belongs to $\{1,2.........S\}$. The formula for food source locations of way-directing bees and accompanying bees is:

$$y_{ij} = x_{ij} + rand \times (x_{ij} - x_{kj}) \quad (3)$$

In the formula, $k, j \in \{1,2.......N\}$, herein, x_{ij} is the location of current food source, x_{kj} is the location of a certain food source around at random, and y_{ij} is the location of bees' new food source. After the individual i has searched the food source for certain times, new food source is generated and its location is:

$$x_i^j = x_{\min}^j + t \times (x_{\max}^j - x_{\min}^j) \quad (4)$$

In the formula, t is a random number in (0,1), $x_{\max}$ is the maximum value of variables, and $x_{\min}$ is the minimum value of variables.

3.2 Differential Genetic Algorithm

The difference algorithm adopts the feature of real number coding used in genetic algorithm, and it is an optimization algorithm that can make random search in continuous space and aims at the NP original colonies in swarm intelligent algorithm. Each solution $X_i = (x_{i1}, x_{i2} \cdots x_{in})$ is a solution with n dimensional vector, and it is mainly made up of variation, crossover and selection.

(1) Variation operation: in difference algorithm, variation method in formula (5) is chosen in this paper.

$$y_{ij} = x_{best,j} + F \times (x_{r1} - x_{r2}) \quad (5)$$

In formula (5), $x_{best,j}(t)$ is the j dimensional vector of the best individual in the current colony, F is random factor that is used mainly to control the convergence degree of difference vectors, the set value is between [0,1].

(2) Crossover operation: make crossover of the intermediate individual y_i with the parent individual x_i through a certain probability selection and obtain new individuals:

$$z_{i,j} = \begin{cases} y_{i,j}, & t \in [0,1] \\ x_{i,j}, & otherwise \end{cases} \quad (6)$$

In formula (6), it can be ensured that a random integer between 0 and 1 can appear in the crossover process, and there is at least one component coming from $y_{i,j}$.

(3) Selection operation. "Greedy" selection strategy is used in difference algorithm to choose individuals so as to ensure that the ones with the best fitness are selected to the next generation. New individual $z_{i,j}$ and individual $y_{i,j}$ of the last generation are generated after variation and crossover operation. Otherwise, $y_{i,j}$ will remain unchanged and go directly into the next generation.

3.3 Chaos Algorithm

The chaos algorithm is to make comparison in the traversal process, keep the relative excellent solution in the process of comparison, and find the optimal solution meeting the condition in the process of iteration and searching. The thoughts are as follows:

$$for \;\; j = 1 \;\; to \;\; k \;\; do$$

$$T_{i,j} = Cos(\pi T_{i,j})$$

$$X_i^{'} = X_{iworst}(t) + T_{i,j}(X_{ibest}(t) - X_{iworst}(t))$$

$$endfor$$

4. Improved Double Bee Colony Algorithm

4.1. Theory of Algorithm

Because the artificial bee colony algorithm and difference algorithm have different ways to produce individuals, they have different effects in the optimization process. A double colony artificial difference bee colony algorithm is proposed based on the analysis of the previous two algorithms. Its thought is to divide the original colony into two groups and adopt artificial bee colony in the first group while difference algorithm in the second group. In the first group, formula (3) is still used in the individual variation, which is made up of the variation individual itself and a random individual so as to maintain the diversity of colony and make search from the perspective of the whole situation. In the second group, through the renewal of formula (5), the variation individual comes from the current optimal individual so as to ensure that the local searching ability is strengthened and the convergence speed is fast. Mix the strategies of the first group and the second group so as to ensure that the algorithm's convergence is accelerated while keeping the diversity of colonies and that the algorithm can have convergence efficiency of the whole situation in reasonable angle. In addition, in formula (2), because there are some differences between the bees' food source and individual of the field and meanwhile, the bee individual of field is chosen at random, the difference of food source quality of the current individual and the neighboring one has not been considered to a certain extent. And in bees' constantly searching process, the factor of food source renewal has not be considered, and it reflects that bee colony does not have sufficient ability to control the optimal food source in the optimization process, which causes the slow-down of convergence speed and thus affects the time to finding the optimal solution for bee colony algorithm. In order to better solve bees' choice of food source in double bee colony algorithm, chaos algorithm is introduced to make it possible that the original solution is capable to choose food source according to their quality and thus ensure that in the searching process, when choosing the neighboring individuals, the accompanying bees can compare the quality of the current food source and its neighboring ones, and balance the information exchange with the neighboring bees timely. The renewed formula is as follows:

$$y_{ij} = x_{ij} + \cos(rand \times w \times (x_{ij} - x_{kj})) \times (x_{ij} - x_{kj}) \quad (7)$$

When the food source quality of neighboring individual is superior to the current one, and then $w > 1$; when the food source quality of the current individual is superior to the neighboring one, and then $w < 1$. Herein, the value of w is as follows:

$$w = \begin{cases} ((iter_{\max} - iter_i) / iter_{aver})^{\alpha} \\ ((iter_i - iter_{\min}) / iter_{aver})^{\beta} \end{cases}$$
$$iter_{aver} = \frac{sqrt(iter_{\max} + iter_{\min})}{iter_{\max} - iter_{\min}} \quad (8)$$

In formula (5), $iter_{\max}$ refers to the maximum iteration number, $iter_{\min}$ refers to the minimum iteration number, $iter_i$ represents the frequency of searching neighboring food source at the i time, α is a constant number, which is relatively reasonable between [0,0.5], and β is also a constant number, which is relatively reasonable between [1, 1.5].

4.2 Algorithm Steps

Step1：Set the original parameter, colony amount, maximum iteration number, minimum and minimum iteration number, herein, set that the amount of way-directing bees and accompanying bees account for 50% respectively and there is one scout bee. Set the adaptive value of good food source at random.

Step2: Arrange the initial solution by chaos algorithm and get the adaptive value of the optimal food source.

Step3: According to the food source found by the way-directing bees in step3, accompanying bees generate the optimization strategy according to formula (3) and (5).

Step4: Rank the yields by using formula (2) in these two strategies respectively.

Step5: Compare the quality of the current food source and the neighboring food source by the income probability, and choose to store food source that is superior to the current conditions through formula (7) and (8).

Step6: If the adaptive value of honey source cannot be found after the cycling has gone through a limited times, and then forgive it.

Step7: Termination conditions are satisfied and the previous set cycling times are achieved.

5. Simulation Experiment Analysis

In order to verify the performance of this algorithm, the test is carried out in two parts in this paper: one is to test the algorithm performance and another is to schedule resources in cloud computing.

5.1. Algorithm Performance Test

In order to verify the effectiveness of the algorithm, a benchmark functions in Literature [9] are used to make comparison test so as to verify the effectiveness of the algorithm and analyze its performance. Make test on the basis of Matlab of Windows.

(1) Sphere Function

$$f(x)=\sum_{i=1}^{n}x_i^2 \quad -100\le x_i\le 100$$

(2) Quaric Function

$$f(x)=\sum_{i=1}^{n}x_i^4+random[0,1] \; -5.12\le x_i \le 5.12$$

(3)Arkely Function

$$f(x)=-20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2})-\exp(\frac{1}{n}\sum_{i=1}^{n}\cos 2\pi x_i)+20+e \quad -32\le x_i\le 32$$

In this paper, aiming at setting the value of α and β in formula (8) and starting from searching the quality effects of values, the value of the three functions is set as [0.1,0.5], the value of β is set between [1.1,1.5], and the value of α is set between [0.1, 0.5]. The test results of the three functions are shown as in Table 1-3, herein, aver refers to the average value, st refers to the variance, min represents the minimum value and max represents the maximum value, and the value of D is 2.

Table 1. Test Results of Sphere Function

Sphere Function		Colony Amount is 20		Colony Amount is 50		Colony Amount is 100	
		Before Improvement	After Improvement	Before Improvement	After Improvement	Before Improvement	After Improvement
D=2	Aver	0.0005129	1.17E-05	5.17E-10	6.95E-17	2.21E-15	1.19E-15
	St	0.0006312	1.43E-06	4.29E-09	5.05E-17	3.74E-15	1.84E-17
	Min	0.0016145	2.71E-05	2.13E-13	1.21E-17	1.38E-20	3.16E-20
	Max	0.0059761	0.08E-07	1.81E-09	7.13E-17	2.27E-20	7.61E-20

Table 2. Test Results of Quartic Function

Quartic Function		Colony Amount is 20		Colony Amount is 50		Colony Amount is 100	
		Before Improvement	After Improvement	Before Improvement	After Improvement	Before Improvement	After Improvement
D =2	Aver	2.4731459	4.752141	0.724698	0.685412	0.008025	0.006212
	St	2.8915210	3.045263	1.963254	1.871412	0.045621	0.023135
	Min	0.2418141	0.087452	0.004241	0.003254	0.341261	0.281276
	Max	9.3426341	9.212332	8.465712	7.864127	0.274578	0.192312

Table 3. Test Results of Ackely Function

Ackely Function		Colony Amount is 20		Colony Amount is 50		Colony Amount is 100	
		Before Improvement	After Improvement	Before Improvement	After Improvement	Before Improvement	After Improvement
D =2	Aver	3.728912	2.728742	1.524164	1.014589	1.15E-12	1.32E-14
	St	3.045236	2.801547	1.425412	1.025475	2.22E-13	0.84E-14
	Min	0.176984	0.138523	1.718745	1.612689	1.16E-15	1.57E-16
	Max	0.721457	0.694578	1.694521	1.527812	1.49E-15	1.51E-16

It can be shown from Table1 to Table 3 that with the increase of colony amount, the convergence accuracy of bee colony improves constantly, and it has also been shown through the test functions that the algorithm in this paper has some certain improvements both in the computing accuracy and stability.

4.2. Task Allocation of Improved Double Bee Colony Algorithm in Cloud Computing

Test is carried out by using the CloudSim [12] platform, and the hardware mainly includes Core i3CPU, 4GDDR3 and Windows Xp, and the software uses matlab2012 to make simulation. Set 300 virtual tasks, 10 virtual nodes, and 500 times of iteration. Make comparison of the algorithm in this paper with algorithms in Literature [2] and [3]. And the comparison, carried out in cloud computing models, is shown in Figure 1-5.

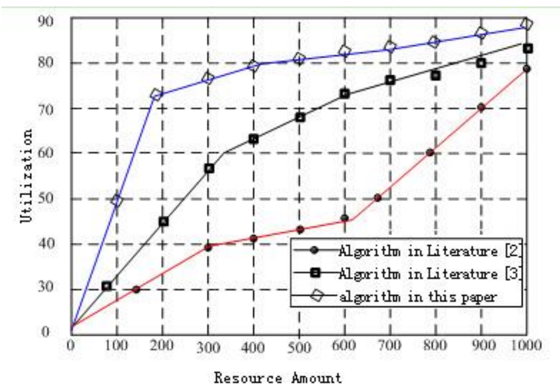

Figure 1. Resource Utilization Comparison of 3 Resource-load Algorithms

资源数目：Resource Amount
利用率：Utilization

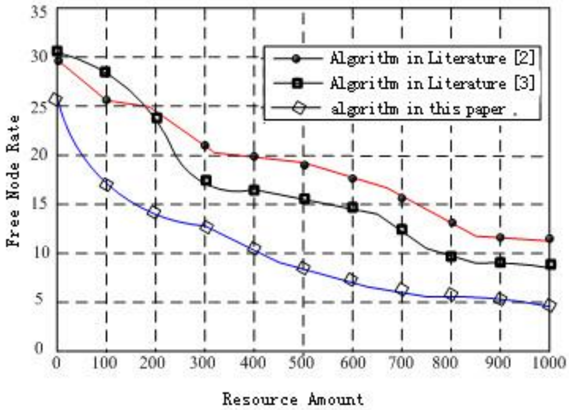

Figure 2. Energy-Consumption Time Comparison of 3 Resource-load Algorithms

空闲节点率：Free Node Rate
资源数目：Resource Amount

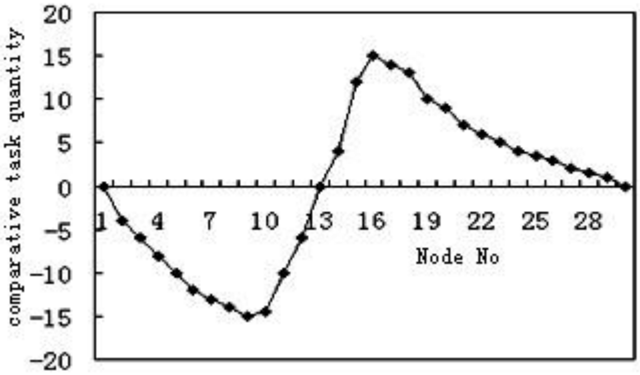

Figure 3. Assignments of Nodes with Different Performance When There are 100 Tasks

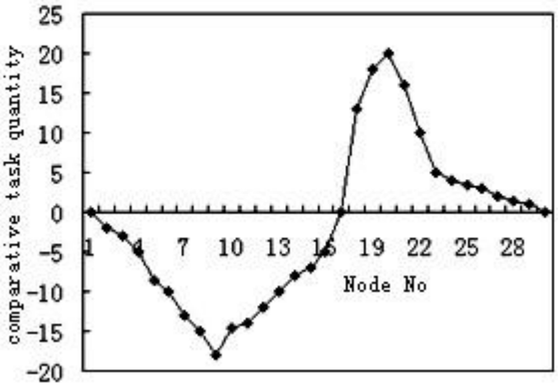

Figure 4. Assignments of Nodes with Different Performance When There are 200 Tasks

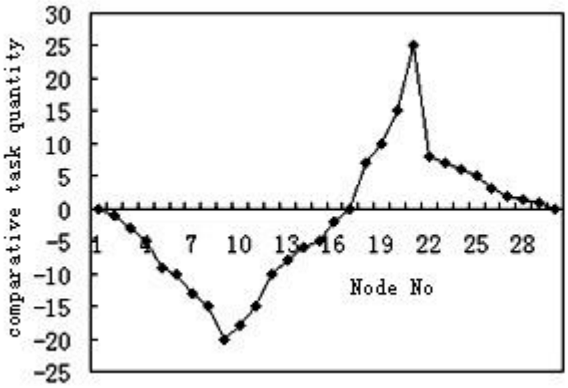

Figure 5. Assignments of Nodes with Different Performance When There are 300 Tasks

As it can be seen from Figure 1-2, the algorithm in this paper is significantly superior to the algorithms in the other two literatures in terms of the optimization effects of scheduling resources. And it can also be found from Figure 3-5 about the node performance with different tasks that the algorithm in this paper has better effects both in timing and performance. The algorithm in this paper has fewer advantages in average transmission time and implementation time when there are relatively fewer tasks, but has demonstrated its advantages when there are a lot of tasks. The main reason is that the improved bee colony algorithm can better allocate resources in cloud computing models and with the increasing amount of tasks, the time of prediction model is approaching the actual time.

6. Conclusion

It is an important problem as how to reasonably use resources in cloud computing. In this paper, genetic differential algorithm is introduced into the bee colony algorithm, making the improvement of the global and local searching ability of the improved algorithm. Besides, chaos algorithm is introduced into the bee colony algorithm so as to improve the bee colony's efficiency to find good food source quickly and solve the resource-allocation problem as well as improve the efficiency of allocation resources on the cloud computing platform.

References

[1] L. Vaquero, L. Rodero-Marino, J. Caceres, *et al*., "A break in the clouds, towards a cloud definition", SIGCOMM Computer Communication Review, vol. 39, no. 1, **(2009)**, pp. 50-55.
[2] Xuyu, "Research on Resource Schedule Model under Cloud Environment", Computer simulation, vol. 30, no. 5, **(2013)**, pp. 362-365.
[3] D. Yao, G. Tao and C. J. Jie, "Fleet elastic load balancing mechanism in cloud environment", Journal of Computer Applications, vol. 33, no. 3, **(2013)**, pp. 830-833.
[4] Q. Ping and L. L. Peng, "Task Scheduling Algorithm Based on Fuzzy Quotient Space Theory in Cloud Environment", Journal of Chinese Computer Systems, vol. 34, no. 8, **(2013)**, pp. 1793-1797.
[5] R. Buyya, D. Abramson, J. Giddy, *et al.*, "Economic models for resource management and scheduling in grid computing", Concurrency and Computing, vol. 14, **(2002)**, pp. 1507-1542.
[6] J. Nimis, A. Anandasivam, N. Borissov, *et al*., "SORMA-business cases for an open grid market concept and implementation", Proceedings of the 5th International Workshop on Grid Economics and Business Models(GECON 08), **(2008)**, pp.173-184.
[7] A. J. Younger, G. V. Laszewski, W. Li-Zhe, *et al*., "Efficient resource management for cloud computing environments", Green Computing International Conference, **(2010)**, pp. 357-364.
[8] C. H. Zhao, S. S. Zhang, Q. F. Liu, *et al.*, "Independent tasks scheduling base on genetic algorithm in cloud computing", 5^{th} International Conference on Wireless Communications, Networking and Mobile Computing, vol. 8, **(2009)**, pp. 5548-5551.
[9] D. Karaboga, "An idea based on honey bee swarm for numerical optimization", Technical Report-TR06

Erci- yes University, Engineering Faculty. Computer Engineering Department, **(2009)**.
[10] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization", Artificial Bee Colony (ABC) algorithm", Journal of Global Optimization, vol. 39, no. 3, **(2007)**, pp. 459-471.
[11] R. N. Calheiros, R Ranjan, R. Cafd and Buyya R, "CloudSim, a novel framework for modeling and simulation of cloud computing infrastructures and services, **(2009)**, pp. 9-3.
[12] R. N. Calheiros, R Ranjan, R. Cafd and Buyya R, "CloudSim, a novel framework for modeling and simulation of cloud computing infrastructures and services **(2009)**, pp. 9-3.
[13] J. Y. Wu, W. J. Li, J. P. Huang, J. Lin, D. R. Zhang, "Key techniques for Mobile Internet a survey SCIENTIA SINICA Informationis, vol. 45, no. 1, **(2015)**, pp. 45-69.

Author

Wu Junhua, (1979) Male, bachelor, Research Direction: Computer Science & Technology, Algorithm Analysis, etc.