Task Scheduling Algorithm in Cloud Computing Based on Fairness Load Balance and Minimum Completion Time

Wang Yizhen[1,a], Sun Yongqiang[2] , *Sun Yi[b]

[1]Beijing University of Posts and Telecommunications, Beijing 100876, China

[2]China Waterborne Transport Research Institute

[a]email:2013213437@bupt.edu.cn, [b]email:sunyisse@bupt.edu.cn

Keywords: Cloud computing; Fairness; The minimum completion time; Load balancing; Task scheduling.

Abstract. Task scheduling is one of the key technology of cloud computing, its main goal is to satisfy the fairness of the system resources and tasks more effectively, and at the same time as much as possible to realize the load balance of system resources and reduce the completion time. Combined with the advantages of genetic Algorithm, we put forward a Fair and Balanced and effective genetic Scheduling Algorithm FBEGSA (Fair Balanced Efficient Genetic Scheduling Algorithm).The experimental results show that the algorithm is useful to task scheduling.

I Introduction

Cloud computing is a new computing model,in it's environment, each server uses virtualization technology to become a huge pool of resources, dynamically invoking the virtual machine to provide users to use. How to allocate the user tasks reasonably to different resources and make the whole system's performance can achieve the best results is the key problem to be solved by the task scheduling algorithm. Ships produce a large amount of AIS data every day in the shipping. Through the data analysis of the high risk of water, it puts forward high requirements of the system performance and reaction time.At present, there is no research on fairness,load balance and minimum completion time.

In this paper, combining with the merit of the GA(Genetic Algorithm) which focuses on the fairness of resource and task, the load balance of system resources, and the task scheduling problem in cloud computing environment. The simulation experiments show the feasibility and effectiveness of the FBEGSA algorithm.

II Load Balance and Minimum Completion Time

【1】If the cloud computing environment m resources is expressed as R=$\{r_1,r_2\ldots, r_m\}$,n tasks are represented as U=$\{u_1, u_2\ldots u_n\}$,so the cloud computing system can be described as Cloud= (R, U).

The collection of m resouces'load can be represented as L=$\{l_1, l_2\ldots, l_m\}$.

Predict the Minimum Completion Time of n tasks are done on the m resources can be described as a m*n matrix:

$$\text{PMCT}=\begin{pmatrix} C_{11} \ldots C_{1m} \\ C_{21} \ldots C_{2m} \\ \vdots \quad C_{ij} \quad \vdots \\ C_{n1} \ldots C_{nm} \end{pmatrix}$$

In the matrix PMCT, element C_{ij} represents the Predict the Minimum Completion Time of task $u_i$is done on the resource r_j in the task queque of the cloud computing.

Every user hopes that their own task can be completed in the Predict the Minimum Completion Time. Thinking about the task scheduling,the best result is that each task can be scheduled on the resource with the Predict the Minimum Completion Time and having the smallest load.

This algorithm is to consider the fairness of task scheduling from the perspective of the overall load balancing of the system and reducing the total completion time of the task.

Firstly, take into account load balance and the minimum completion time. Can find a balance between the two factors in task scheduling by the formulas as follows.

The first step is initializing the matrix PMCT and the collection L. The second step is ergodicing all elements in the matrix in order to find the Predict the Minimum Completion Time.

C_{min}=Min(C_{ij}), $1 \le i \le n, 1 \le j \le m$

Find the smallest load of the resource in the collection L. l_{min}=Min(l_j)

Our algorithm use the following formulas to choose tasks to match to the resources on the process of task scheduling.

$$\frac{C_{min}}{C_{ij}^{l_{min}}} \ge \frac{l_{min}}{l_j^{C_{min}}} \quad \text{(A)} \qquad \frac{C_{min}}{C_{ij}^{l_{min}}} < \frac{l_{min}}{l_j^{C_{min}}} \quad \text{(B)}$$

$l_j^{C_{min}}$ is the load value of the resource with the minimum completion time in the all elements in the matrix PMCT. $C_{ij}^{l_{min}}$ is the Predict the Completion Time of the task doing on the resource with the smallest load value.

Choose Normalization formula to represent the paper’s formula. The Normalization formula is generally used to compare with data coming from different sources in the same system. Here we compare time and the load value in the same system in order to choose what we want.

If the results of the comparison satisfy the formula (A) ,that illustrates the D-value of the minimum completion time of the two tasks is bigger than the D-value of the load of the allocating resources. Then choose the task from the matrix PMCT with the Predict the Minimum Completion Time to match with the resource.

If the results of the comparison satisfy the formula (B) ,that illustrates the D-value of the load of the allocating resources of the two tasks is bigger than the D-value of the minimum completion time. Then choose the resource with the smallest load value from the collection L, and find the corresponding task from the matrix PMCT with the Predict the Minimum Completion Time to match with it.

Analyze the property of every algorithm by collecting simulation experimental data.

Our algorithm show huge advantageous of load balance and total completion time of the task. Among, load balance degree is calculated by the following formula:

$$S_{l(i)}^2 = \frac{1}{m}[(l_1 - \bar{l})^2 + (l_2 - \bar{l})^2 + \ldots + (l_m - \bar{l})^2] \quad \text{(C)}$$

So the load balance of the system can be showed as: $S = \sum_{i=1}^{m} S_{l(i)}$ (D)

III Fairness and Load Balance

The core idea of the fairness is to ensure the tasks of the system can be allocated the maximize resources of the system, which contains two aspects of the meaning:the fairness of the arranged tasks and that of the scheduled resources.

Load balance is to balance the arranged methods of the multi-tasks efficiently, and divide the tasks according to the resources nodes ’calculation ability into different resources to schedule. In order to state the fairness and load balance better, then we will show some definitions in the form:

【2】If the expect executing time of the task ui is ET(i),and the actual executing time is AT(i),then the fairness of the task u_i is: $TF(i)=\ln\frac{AT(i)}{ET(i)}$ (1)

When AT(i)≤ET(i),that is the actual executing time is better than expect executing time, TF(i)≤0 represents that u_i is in the state of fairness.When AT(i)>ET(i),that is task consume excess executing time, TF(i)>0 represents that $u_i$is in the state of unfairness.Through the description of resources and tasks for cloud computing environment, and through the comparison of the expected execution time of task ET(i) and the actual execution time of AT(i) ,which can reflex the difference between the actual implementation situation and ideal situation efficiently in the cloud computing environment, and make it as a measure of fairness algorithm.

【3】If the length of the task u_i is represented as Length(i),the size of the task u_i is expressed as Size(i), then the weight of task u_i is W(i). $W(i)=\log_2(Length(i)+Size(i))$ (2)

By evaluating the length and size of the task, it is more accurate to measure the weight of the task.

【4】If the processing capacity of the resource r_j is p(j), and the task has been assigned to is atotal(j), then the fairness of resource r_j is: $RF(j)=\ln\frac{\sum_{i=1}^{atotal(j)} W(i)}{p(j)}$ (3)

When $\sum_{i=1}^{atotal(j)} W(i) = p(j)$, the amount of task system allocate to the resource r_j is equivalent to its handling ability, RF(j)=0 shows that the resource r_j satisfy the fairness;When $\sum_{i=1}^{atotal(j)} W(i) < p(j)$, the amount of task system allocate to the resource r_j is less, at this time, RF (J) <0 which is called Too few unfair;When $\sum_{i=1}^{atotal(j)} W(i) > p(j)$, the amount of task system allocate to the resource r_j is on overload condition, RF(j)>0 which is called Too much unfair.

Through the definition of the formula (3), we can see that: the fairness of resource allocation in cloud computing environment is more considering the amount of resources allocated to the task is proportional to its computing power, too much or too few will affect the fairness of the resources.

The fairness of the system SF will be considered in two aspects: the fairness of the task and the fairness of the resources. The specific definition of the system is as follows:

【5】For cloud computing systems with m resources and n tasks, if the fairness of the task u_i represents as TF(i), the fairness of resource r_j represents as RF(j),so the fairness of the system is SF: $SF=\frac{1}{m} \times \sum_{j=1}^{m} RF(j) + \frac{1}{n} \times \sum_{i=1}^{n} TF(i)$ (4)

Load balance in cloud computing environment will take full consideration of the load balance factor of each resource in the system. And the load balance rate of the resource is mainly thinking about the ratio of the amount of the task has been completed and the allocation of the tasks. The specific definition is as follows:

【6】If the resource $r_j$has been assigned to atotal(j), the amount of the tasks that have been processed is ftotal(j), then the load balance rate Load(j) of the resource r_j is:

$$Load(j)=\ln\frac{ftotal(j)}{atotal(j)} \quad (5)$$

【7】The load balance of resourcer_j is Load(j), then the load balance of the cloud computing system Load is: $Load=\frac{1}{m} \times \sum_{j=1}^{m} Load(j)$ (6)

Through the quantitative qualitative definition of fairness and load balance in the cloud computing environment from the formula (1)-(6), this paper will be demonstrated and tested by a large number of experimental data in the experimental section.

IV Fair Balance and Efficient Genetic Algorithm

Idea of the Algorithm design

FBEGSA algorithm, in the mechanism of parallel search of genetic algorithms, is a kind of effective cloud computing task scheduling algorithm based on fairness, load balance and minimum completion time. FBEGSA algorithm fulfills the requirements of fairness and load balance as far as possible, so as to effectively improve the efficiency of resource utilization and the efficiency of the algorithm execution.

The structure and implementation steps of FBEGSA algorithm are similar to the traditional genetic algorithm, which mainly includes the steps of encoding, fitness calculation, selection, crossing and variation. The traditional genetic algorithm is improved, and the formal definition of FBEGSA algorithm is as follows:

【8】If the encoding method of the FBEGSA algorithm is represented as E, the fitness function is F, the selection operation is S, the crossing operation is C, the variation operation is M, then the FBEGSA algorithm can be described as FBEGSA=(E,F,S,C,M).

Basic operation of the algorithm

In this paper, encoding according to the mapping between resource and task. The value of each gene in the individual is assigned to the resource number of the sub task corresponding to the position. The fitness function of FBEGSA algorithm will be combined with the system's fairness, load balance and minimum time to complete the design, the specific description as follows:

$$F(x_i)=e^{TF(i)+RF(i)+\sqrt{Load(i)\times\sum_{i=1}^{m} S_{l(i)}}}$$

In this paper, we consider the algorithm combining with fairness, load balance and minimum completion time, because the three factors should be considered for the task scheduling and are very important. The three factors are in a parallel state, and they play an important role in the task scheduling , so in the function F use the sum as the weight. Because of the parallel operation, it is a trend of exponential function, so we choose to use the exponential function to express.That is what we guess the result according to the theory,we will do large of experiments to verify our suppose.

The load balance degree of the cloud computing system based on the premise of fairness is formula(5),and the load balance degree of the cloud computing system based on the minimum completion time is formula (D)

In this paper, the load balance degree J(i) is introduced, which is based on the load balance degree of the fairness and the load balance degree of the minimum completion time:

$$J(i)=\sqrt{Load(i)\times\sum_{i=1}^{m} S_{l(i)}} \tag{E}$$

Because the fairness and the minimum completion time are in parallel, their load balance degree play an important role in the task scheduling, which are the same property, so multiply them. And then use extracting a root to get the load balance degree of the system.

So we introduced the fitness function of genetic algorithm. $F(x_i)=e^{TF(i)+RF(i)+J(i)}$

[1]The selection operation is to retain the population with the highest fitness to the next generation. Therefore, the FBEGSA algorithm for the selection of operation S is described as follows:

$$S[P(t)]=\{x|y\in P(t),F(x)\geq F(y),x\in P(t)\} \tag{8}$$

[2]The implementation steps of the crossing operation of the FBEGSA algorithm are as follows:

Step1 According to the crossing probability P_C, select two individual x and y from the population P(t) randomly.

Step2 Random generate variable c_1= random (1, m), m represents length of the chromosome, $1\leq c_1\leq m$;

Step3 Make $i=\lceil c_1 \rceil$,if i=1,then make i=i+1.If i=m, then make i=i-1;

Step4 Exchange the former i bits of the individual x and y, the specific operation is shown in figure 1.

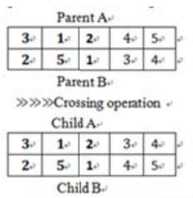

Figure 1-crossing operation of the FBEGSA algorithm

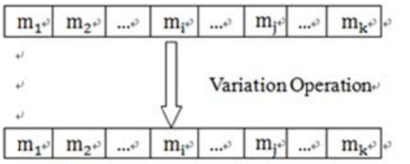

Figure 2- Variation Operation of the algorithm

The implementation steps of the variation of the FBEGSA algorithm are as follows:

Step1 According to the variation probability p_m,randomly select individuals from population P(t);

Step2 Randomly generate two variable m_1=random(1,m), m2=random(1,m), m are the length of the chromosome, $1\leq m, m2\leq m$;

Step3 Make $i=\lceil m_1 \rceil$,$j=\lceil m_2 \rceil$,if i=j, then j=j+1;

Step4 Swap the gene i and gene j of the individual x, and schematic as shown in figure 2.

Implementation steps of the algorithm

The operation steps of the FBEGSA algorithm are as follows:

Step1 Randomly generate initial population P(0)={$x_1, x_2 \dots, x_i, \dots x_{Size}$} in solution space and initial the size, crossing probability P_C, variation probability, maximum evolutionary algebra G_m, and evolutionary algebra t of population.

Step2 Calculates the fitness value of each individual in the population F(x_i),i=1,2,…,Size;

Step3 Carrying out the select operation to population P(t);

Step4 Carrying out the crossing operation to population P(t);

Step5 Carrying out the variation operation to population P(t);

Step6 Make t=t+1. If t≤ G_m,then jump to the Step2; otherwise, the algorithm search process end.

V Experiment Test

Load balance and the minimum completion time. (The algorithm here is called ET algorithm)

Load balance

Figure 1.1 shows the number of resources to 2, the average value of the load balance obtained from the 20 experiments into the different number of tasks

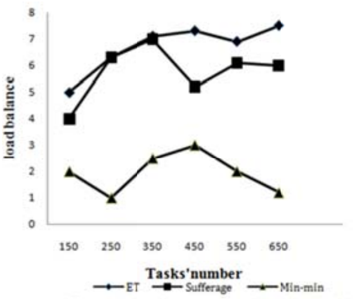

Figure 1.1 ---the average value of the load balance

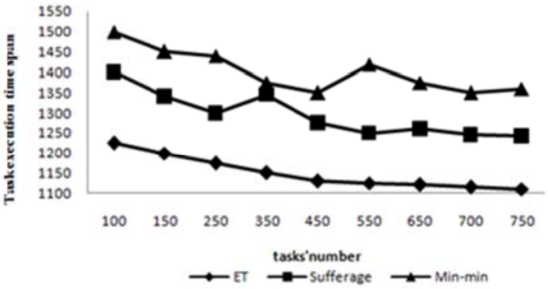

Figure 1.2-The total completion time of the tasks

Figure 1.1 shows, the load balance of the three algorithms decrease in number when the number of the tasks increases. That is the performance of the load balance improves.

The load balance of ET algorithm is a little higher than the Sufferage algorithm, and much higher than the Min-min algorithm.

Above all, the ET algorithm has advantages in load balancing.

Total completion time

Figure 1.2 shows the number of resources into the 30. The total completion time of the task is obtained by the 20 experiments under the different tasks.By it the total completion time of the 3 algorithms increases when the tasks increase. But the total time of ET algorithm is large.The experiments verify that the ET algorithm has great advantages of shortening the total completion time of tasks.

We study a variety of grid task scheduling algorithms.

Design the ET algorithm, which is suitable for the cloud computing environment, based on the advantages of Min-Min and Suffrage algorithm.

Load balance and fairness.(The algorithm here is called FBGSA algorithm)

We use the CloudSim simulation platform to test the effectiveness of the FBGSA algorithm,

Comparing with the TCDE algorithm in reference[2] and the FSA algorithm reference[1], the document has mentioned.

The specific parameters of various algorithms in the implementation process are as follows: population size Size=100, Gm=500, PC=0.85, variation probability Rate Pm=0.25.

Because in the FBGSA algorithm the method of the random generate population will affect the results of the experiment.Therefore, the following experiment will be run independently 10 times, the experimental record to take 10 times mean value of the test.

Table 1 we know the fairness of FBGSA algorithm is minimal, which indicates that the actual execution time of the task is closer to expected execution time is applied to the FBGSA algorithm.

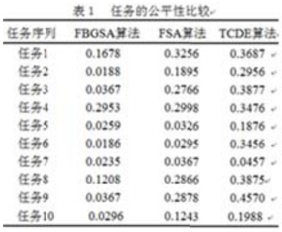

表1　任务的公平性比较

任务序列	FBGSA算法	FSA算法	TCDE算法
任务1	0.1678	0.3256	0.3687
任务2	0.0188	0.1895	0.2956
任务3	0.0367	0.2766	0.3877
任务4	0.2953	0.2998	0.3476
任务5	0.0259	0.0326	0.1876
任务6	0.0186	0.0295	0.3456
任务7	0.0235	0.0367	0.0457
任务8	0.1208	0.2866	0.3875
任务9	0.0367	0.2878	0.4570
任务10	0.0296	0.1243	0.1988

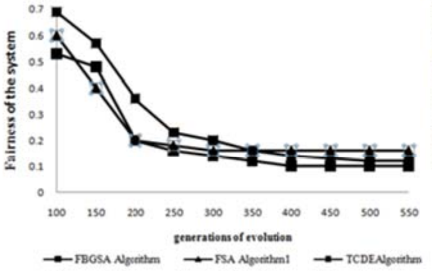

Figure 2-The comparision ofthe fairness of the system

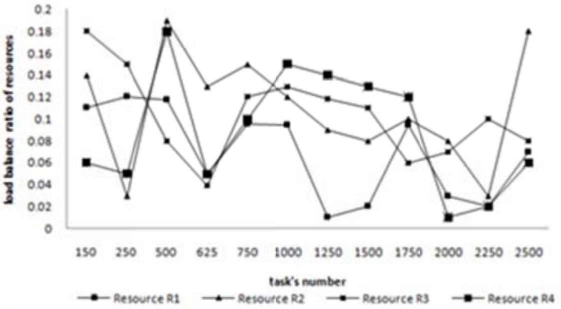

Figure 3-the comparision of load balance ratio of 4 resources based on the FBGSA Algorithm

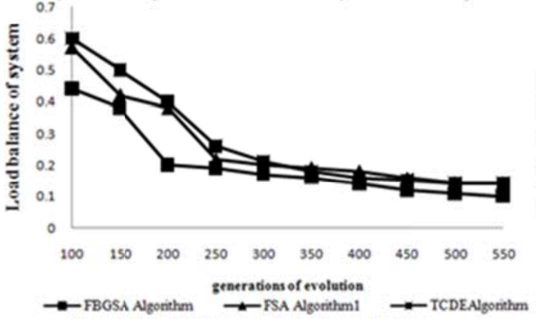

Figure 4-The comparision of the load balance of the system

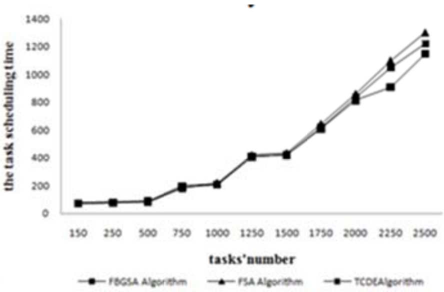

Figure 5-the camparision of task scheduling time

Figure 2 shows fairness obtained by the FBGSA algorithm in system has the best performance in the vast majority states.Figure 3 shows load balance ratio of the 4 resources based on the FBGSA algorithm.In it, the load balance of the 4 resources fluctuates between0.01 and 0.2,with the increase of the number of tasks in the system, the fluctuates of load balance of the four kinds of resources in the area of the load balance is still small.Figure 4 shows load balance of the system. Figure 5 shows that the gradual increase of the number of tasks in the system, the task scheduling time of the three algorithms is gradually increased.

The simulation experiments show that the algorithm can not only satisfy the fairness of the task and the resources in the system, and can effectively achieve the load balance of system resources.

VI　Conclusion

According to the results of the experiment test of the LB-ECT algorithm and the FBGSA algorithm, we can guess and infer to that the FBEGSA algorithm is correct, which means that the FBEGSA algorithm can not only effectively reduce the total completion time, but also can satisfy the fairness of the system resources and tasks and effective resource load balancing.In the future research, we will combine with other more factors base on the FBEGSA algorithm.

References

[1] Hom J, Nafpliotis N, Goldberg D E A Niched Pareto Genetic Algorithm for Multi-objective Optimization[c]//Michalewicz Z,ed.Proc of the 1st IEEE Conf.on Evolutionary Computation. Piscateway:IEEE Press,1994:82-87.

[2]Coello Coello C A,Lechuga M S.MOPSO: A proposal for multiple objective particle swarm optimization[C]//Evolutionary Computation, 2002.CEC'02.Proceeding of the 2002 Congress on. IEEE,2002,2:1051-1056.

[3]Zhang Aike.Xie Cuilan. Task Scheduling Algorithm in Cloud Computing on Fairness and Load

Balance. DOI:10.3969/j.issn.1000 386x.2015.02.065(in Chinese).

[4]Wang Guoan,Yang Huan. The research of algorithm of task scheduling in Cloud based on load balance. 2012,28(12)Fujian Computer.

[5]Gong M,JiaoL, Du H, et al. Multiobjective immune algorithm with neighbor-based selection[J].Evolutionary Computation,2008,16(2);255-255.