Journal of Theoretical and Applied Information Technology
10th May 2013. Vol. 51 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SCHEDULING RESEARCH BASED ON GENETIC ALGORITHM AND QOS CONSTRAINTS OF CLOUD COMPUTING RESOURCES

[1]GUANG LIU, [2]CHEN YANG, [3]DAOGUOLI
[1]Doc., Hangzhou Dianzi University, Hangzhou,China
[2]Master, Hangzhou Dianzi University, Hangzhou,China
[3] Prof., Hangzhou Dianzi University, Hangzhou,China
E-mail: [1]dhulg@126.com, [2] yangchen870825@163.com,[3] ldgyq2003@yahoo.com.cn

ABSTRACT

Cloud computing is a new business model based on Internet and aims to provide information service for users. Resource scheduling is one of the key technologies of Cloud Computing. Analyzing the representative achievement of cloud computing resources scheduling, and then according to the problem that there are large gap between most algorithm task design and actual service needs, we present a task scheduling algorithm on the basis of multi-QoS constrains and genetic algorithm. It supports different users choose different scheduling goals according to their own needs. By doing experiment on cloud computing simulation platform called CloudSim, the results shows that this algorithm can satisfy the QoS constrains, at the same time, it can guarantee system load and improve the efficiency of the task scheduling.

Keywords: *Cloud Computing, Qos Constrains, Genetic Algorithm, Task Scheduling*

1. INTRODUCTION

With the development of Internet technology and electronic commerce, cloud computing as a new type of business computing model has become the research hot spot. Cloud computing is a rapid developing area of modern computing science [1][2][3]. It provides services of basic facilities, platform and software, consumer can order these services in go-to-pay mode [4][5]. Cloud computing is developed by distributed computing, parallel computing and grid computing [6]. In the development of distributed computing, cloud computing can split huge processing program into many little subroutines through network, then submit them to huge system composed by multiple serves, after searching and analyzing calculation, it returns the handling results to users. However, the Internet users have different needs to different tasks, cloud computing should integrate heterogeneous network resources and consider different needs of users, so its resources must be distributed reasonably. To a specific task, first it is necessary to decompose the task into several subtasks, and then choose appropriate node to execute each subtask. The quality of service (QoS) is the satisfaction standards of using cloud computing service, so how to coordinate the various resources and optimize resource scheduling then solving large scale problems is one of the key technologies which cloud computing should study on.

In cloud computing environment, the user's QoS target is different from each other. Some users wish the execution time of their application be shorter, they will choose high quality resources to serve, so they can shorten the execution time and finish the task as soon as possible. Accordingly, the high quality resources' cost is more expensive. Some users are concerned about the cost, they hope the price of using cloud resource be as low as possible. For the cloud service providers, they pay close attention to system load. If resource scheduling is unreasonable, the system load may imbalance and cause failure. Based on the multiple QoS constraint environment, the essence of cloud computing task scheduling is allocating the decomposition subtasks to appropr- ate resources, and under the premise of meeting users' needs, assurances the system load balance. Resource scheduling algorithm in cloud computing environment is a NP complete problem, some scholars use genetic algorithm to study the resource scheduling in parallel environment [7]. In recent years, considering users' satisfaction as a factor for resource scheduling caused many scholars' attention, it becomes a focus in cloud computing study fields. Therefore, combine with QoS constraints to study resource scheduling and then choose reasonable resource allocation scheme is of most significance. and under the premise of coordinating users' QoS constraints, guarantee system load balance.

According to multiple QoS constraints resource sche- duling, this paper presents a multiple-target function model, designs the genetic algorithm for task scheduling.

2. QOS MODEL

Resources in cloud computing environment are heterogeneous. To integrate the resources effectively and schedule users' tasks is the key problem. The goal of resource scheduling is to coordinate various resources and manage them reasonably, optimizing resource scheduling according to task demands submitted by users. Resource scheduling consider- ing QoS constraints from the aspect of user and system load balance can make both of them be satisfied. Quality service is a measure of cloud users' needs [8], QoS model is a extending vector, it can be described from many aspects such as complete time, cost, extension, throughput etc. They estimate QoS from different aspects. In this paper, multiple QoS constrains can be divided into three parts, they are complete time, cost and load balance. The front two parts are constraint indexes of user, the third one is system constraint index.

Establish QoS constraints index. QoS constraints index have positive index and reverse index. For positive quality index, index of the higher the value, the quality is better, for rever-se quality index, index of the value is smaller, the better the quality. This paper considers three indexes for reverse quality index. Each of these indexes can be subdivided into specific index, they are showed as below.

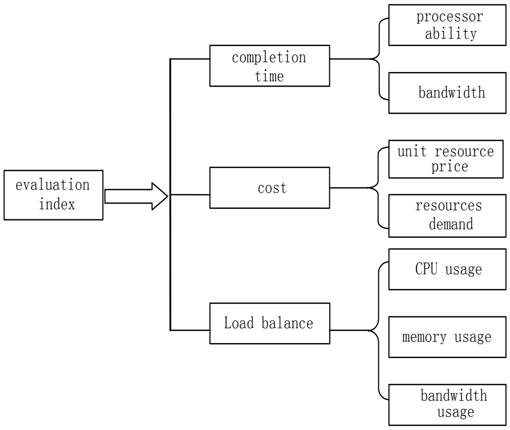

Figure1. Index System

Assume that the comprehensive index is M (x) and it is the weighted summation of completion time, cost and load.

$$M(x) = \omega_1 * Time + \omega_2 * Cost + \omega_3 * Load \qquad (1)$$

In the above formula, $\omega_1 + \omega_2 + \omega_3 = 1$. $\omega_1, \omega_2, \omega_3$ represent respectively the weight coefficient of each indicator. Time refers to completion time, Cost refers to the price of used resource, and Load refers to the system load.

Time is a two-dimensional vector, including two son indexes, respectively processor ability (r_cpu) and bandwidth (r_comm). Time_exec is the execution time of task, and it closely related to processing ability Time_comm is the transmission time of task, closely related with bandwidth.

Time=Time_exc+Time_comm (2)

Time_exc=rq_instruction_count/r_cpu (3)

Time_comm=rq_size/r_comm (4)

Among them, the rq_instruction is the length of the task, and the task of rq_size is data file size. Cost is a two-dimensional vector, including two son index, respectively resource unit price (r_cpu_cost, r_mem_ cost, r_comm_cost) and the demand for resources (rq_cpu, rq_mem, rq_stor).

Cost=r_cpu_cost*rq_cpu+r_mem_cost*rq_mem+
r_comm_cost*rq_comm (5)

Load is a three-dimensional vector, including three son index, respectively CPU utilization (Load_cpu), memory utilization (Load_mem) and bandwidth utilization ratio (Load_br). Through the experimental test, result shows that the weight coefficient ratio of CPU, memory, and bandwidth is 4:3:15. Comprehe- nsive load calculation in punishment type substitution iterative method and the specific formula is as follows.

$$Load = 1 - \prod_{k=1}^{3}\left(1 - Load_k\right)^{\omega_{Lk}} \qquad (6)$$

There are various resource types in cloud computing environment, through the technology of virtualization that physical resource can be virtualiz- ation into resource with properties. Each calculation node (server, and personal PC machine and so on) can be virtualization into index as CPU calculation ability, memory size, data storage capacity, etc. This kind of index can be called hard index, for only these indicators have been fulfilled, can search the right resources scheduling according to other aspects of task requirements. Such as the resource nodes need to meet the demand for task.

Resources information refers to calculation node information the task will choose for scheduling, including CPU calculation ability, memory size, data storage space, communication ability and the price and load capacity of above resource. Calculation node can be expressed as RI={r_cpu, r_mem, r_stor, r_ comm, r_cpu_cost, r_mem_cost, r_stor_cost, r_comm _cost}.The r_cpu refers that node can provide CPU calculation ability, the unit is one million instructions per second (MIPS); r_mem refers that node could provide memory size, the unit is MB; r_stor express node can provide data storage space size, the unit is GB; r_comm means that node can provide data transfer capacity, the unit is MB/S; r_cpu_cost refers to CPU unit price, its standard is every one million instruction, r_mem _cost for memory unit price to 1024 MB as calculation reference; r_stor_cost for data storage unit price to 100 GB as calculation reference; r_comm_cost for bandwidth unit price to 1 MB/S as the benchmark.

Task information refers to demand resources to complete a task. It can be expressed as rq={rq_cpu, rq_mem, rq_stor, rq_comm, rq_instruction_count, rq_size}.

3. DESIGN OF GENETIC ALGORITHM

3.1 Problem Description

Resources in cloud computing environment are dynamic, different calculation node has different resource attribute value. Task need to scheduling has uncertain attribute value. To a set of submitted tasks, according to different demand user can input the expectations of corresponding weights about completion time, cost and load, then cloud computing scheduling system allocation right resources for each task according users' needs and system load, make full use of resources in cloud computing environment. Finally, obtain the satisfactory distribution plan, allocate each task to appropriate calculation node.

Generate several computing nodes random, each node has its own attribute value. RI={r_cpu, r_mem, r_stor,r_comm, r_cpu_cost,r_mem_cost,r_stor_cost,

r_comm_cost}. Random generate several tasks, because of the number of calculation node in cloud computing environment is very large, so system can find several appropriate scheduling nodes for a task. Therefore, in this paper we assume that the number of required scheduling task is less than the number of computing node. Each task has its own attribute values, rq={rq_cpu, rq_mem, rq_stor, rq_comm, rq_instruction_ count, rq_size}. Start cloud computing task scheduling simulation in CloudSim environment, in order to get reason -able resource nodes distribution scheme.

3.2 Problem Analysis

In cloud computing environment, there are a lot of calculation node meeting the task scheduling target, and each calculation node also has its own attribute value. First, according to hard conditions we choose some computing nodes which meet the requiremen-ts as initial population of genetic algorithm. Then establish a target function according to completion time, cost and load. Fitness function is the reciprocal of the objective function. By using fitness funct- ion to selection operation, in order to make the selected node meet the needs of shorter execution time, lower cost and load balance. Finally, to selected node for crossover and mutation operation, making the population diversity.

The algorithm should be able to make user input the weight of each index, according to their own needs to choose reasonable nodes for tasks. Otherwise, user need to input the number of popsize, and the number of nodes and tasks generate random. Task generation needs to meet certain conditions, due to the size of task and bandwidth is positive correlation, therefore the grater the task, the grater the bandwidth required. In the process of task attribute generation, this condition should be satisfied.

3.3 Genetic Algorithm

Genetic operation includes selection, crossover and mutation. When realizing selection operation, this paper uses a roulette wheel method, it is a kind of group members of the selection method. The grater individual fitness value, the bigger probability of selection is. The probability of chromosome was chosen as follows:

$$p_s(i)=\frac{f(i)}{\sum_{i=1}^{popsize} f(i)} \tag{7}$$

Assumptions are established.

In cloud computing environment, resources have dynamic heterogeneous characteristics, the task is more complex. Different users submit tasks of different types. Therefore, in the design of algorith-m, we put forward the following hypothesis conditions, and under the premise of meeting the conditions, design algorithm so as to obtain a satisfactory resource allocation scheme.

H1 denotes a large calculation procedure been decomposed into several subtasks, they are independent form each other.

H2 denotes calculation ability of resources given and computational power refers to the process ability of scheduled tasks.

H3 denotes the number of tasks that is less than the number of resource nodes, as well as the number and size of tasks given.

H4 represents the real-time state of resources given.

H5 represents each resource nodes at the same time that can only deal with a task, which resources are exclusive and non premptive.

The code is indirect coding mode and chromosome length equal to the task quantity, each bit in chromo- some is positive integer, each position Numbers represent task Numbers, the positive integer on each position representative resources Numbers task occupied.

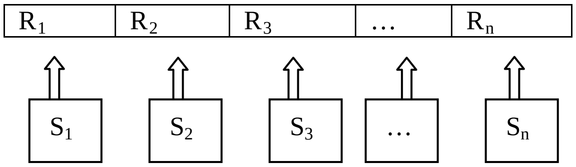

Figure 2. Chromosome's Indirect Coding

The Si is representative of task Numbers, Ri is representative of resources Numbers, n is representative of the length of the chromosome.

In the algorithm, there are some important parameters can affect scheduling results. Details are as foll -ows:

popsize：The population scale, the general value range is from 10 to 150;

Ps: choice probability of Individual i

Pc: hybrid probability, range from 0.4 to 0.99;

Pm：Mutation probability range from 0.001 to 0.1;

Ξ: Termination evolution algebra, range from 100 to 1000

Fitness function: the reciprocal of objective function, M (x) for refers to the objective function.

$$f(x) = 1/M(x). \quad (8)$$

Algorithm steps as follows.

Step 1: According to task demand for resource, through the judgment of resource attribute value calculation node can provide to choose some nodes meet hard conditions as the initial population of genetic algorithm, the number of nodes is popsize.

Step 2: According to multiple QoS target constraint indexes to get a comprehensive index function, namely the objective function, its bottom as the fitness function of the genetic algorithm. Through the fitness function to evaluate fitness of each eligible calculate node.

Step 3: Calculate choice probability of each individual in the group.

Step 4: Random generate a number r = random [0, 1], 0 < r < 1.

Step 5: If r≤Ps, executive selection operation, the two father chromosome insert to the new group without change.

Step 6: Setting cross probability Pc, if Pc≥r, executive hybrid operation, and insert two offspring into a new community; Otherwise, father generations don't need to cross.

Step 7: Setting the probability of variation Pm, if Pm≥r, executive variation operation, and insert mutated individual into a new community; otherwise, father generations don't need to variation.

Step 8: Make t = t + 1, when t < Ξ, return to step 4. Otherwise, return to the optimal solution. (t respective for iterations)

Step 9: Get resource allocation scheme of each task.

4. SIMULATION AND COMPARISON

4.1 Experiments and Parameter Specification

In order to test the genetic algorithm based on QoS constraints this paper proposed, we realized the algorithm in the cloud computing simulation platform called CloudSim [9] , and compare the experimental results with minimum execution time algorithm (Min - Min algorithm). In the simulation system, we can randomly generate several calculation nodes and task, input weight coefficient of three indexes called completion time, cost and system load. First of all, we choose the number of calculation nodes and tasks need simulation, the weighting coefficient can be inputted according to their own needs, as long as the guarantee of the three indexes weights for the sum of 1, and the weight values are between 0 and 1. Population size of algorithm can be inputted by user, determined by the calculation nodes and the size of task (in value range is allowed). Algorithm set the probability of crossover and mutation is 0.8 and 0.1, the iterative terminated number is 600 generations.

In the calculation of the expenses, we mainly make reference to four aspects of cost include CPU, memory, hard disk and bandwidth. The cost of the CPU in accordance with the time, consumption 1 s of CPU time needs a unit cost. Memory for each occupy 1024 MB memory needs to pay one unit cost, hard disk for each occupy 100 GB data space needs to pay one unit cost, bandwidth for each take 1 MBPS needs to pay one unit cost.

4.2 The Analysis of Experimental Results

Through the simulation experiment, the results show that the algorithm can get satisfactory distribution scheme.

It can choose appropriate node to run the task according to user's demand. With the increase of number of tasks, time scheduling needs to spend is gradually increasing.

The experimental results show that genetic algorithm scheduling time is slightly higher than Min - Min algorithm, the result is shown in Fig.3. But for cost and comprehens-ive objective function value comparison, genetic algorithm is smaller than Min - Min algorithm.

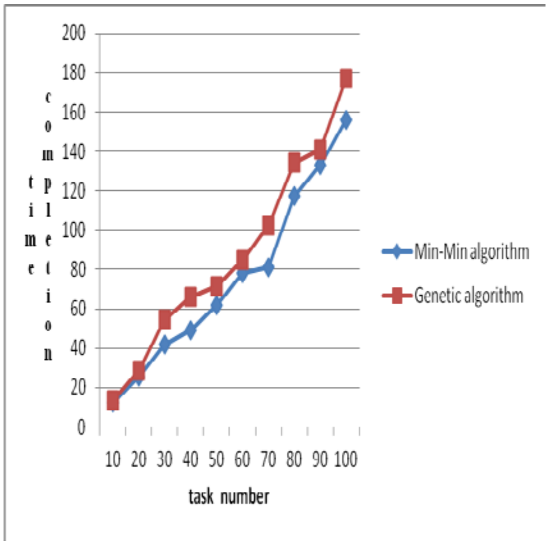

Figure 3. Completion Time

Fig.4 shows that with the growing number of tasks, the scheduling objective function value increases gradually, as can be seen from the graph, when the task quantity becomes large, the objective function of the genetic algorithm value is smaller than Min - Min algorithm. Genetic algorithm has more obvious advantages in overall performance.

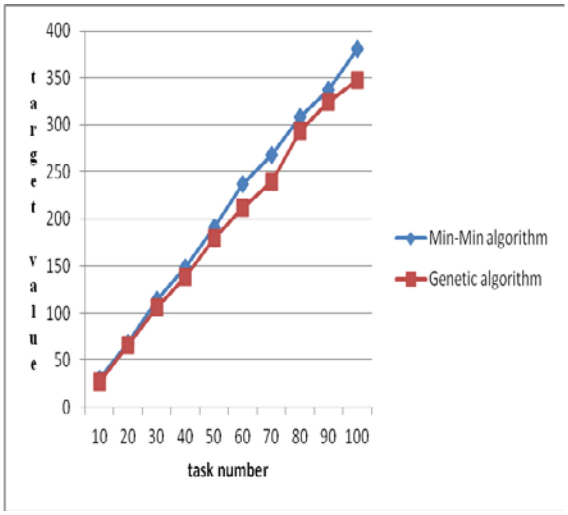

Figure 4. Target Value

Cloud computing users want to use less cost to obtain satisfactory service. Compare the algorithm presented in this paper with the Min-Min algorithm, the results show that with the increase of number of tasks, cost of both algorithms is more and more high, but to overall cost the former is superior to the latter. The results are shown in Fig.5.

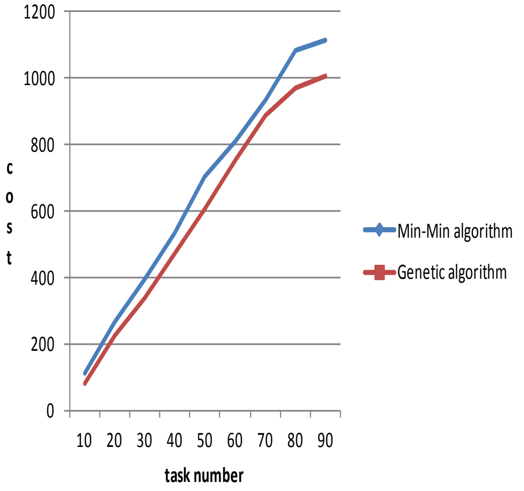

Figure 5. Scheduling Cost

System load is a constraint in the cloud computing, which can be established by setting weight coefficient of load in order to select appropriate calculation node for algorithm. The experimental results show that with increasing the number of task, load value will also increase. The load value of two algorithms varies with different weight coefficient. But load of Min-Min algorithm is obtained according to the formula, the algorithm does not take into account the needs of users. Therefore, from the point of comprehensive analysis, genetic algorithm can better satisfy different QoS requirements.

5. CONCLUSION AND FUTURE WORK

Based on the traditional scheduling algorithm, combine with QoS demand, this paper studies resources scheduling algorithm in cloud computing environment. The algorithm concerns both user demand and system performance, considers various constraint conditions as the task completion time, cost and load balance and so on. Finally, in the cloud computing simulation platform Cloudsim, we realize the algorithm.

The result shows that the algorithm can give satisfied allocation scheme according to different QoS requirements. The proposed algorithm has a advantage than the traditional minimum execution time algorithm in cost and comprehensive performance. But for the algorithm different parameter setting will affect simulation results. If we try to research different values for the parameters setting, find the reasonable value and improve it, scheduling efficiency can be higher. According to the defects of genetic algorithm, if improve the algorithm, we may make better scheduling result, so the other heuristic algorithm or improvement of the existing algorithm is the focus of the next step of work.

ACKNOWLEDGMENT

The authors thank the scholars whose related research achievements inspired me. This work was supported by a grant from the Social Sciences Project of Zhejiang Province (RWSKZD01-201203),the Humanities and Social Sciences project of the Ministry of education (09YJC630053), the natural sciences fund of zhejiang province(Y6110764, LY12G01002).

REFRENCES:

[1] J. Arshad, P. Townend, and J. Xu, “An Automatic Intrusion Diagnosis Approach for Clouds”, *International Journal of Automation and Computing*, Vol. 8, No. 3, 2011,pp. 286–296.

[2] Y. K. Guo, and L. Guo, “IC cloud: Enabling Compositional Cloud”,*International Journal of Automation and Computing*, Vol.8, No. 3,2011, pp. 269–279.

[3] B. Li, B. Q. Cao, K. M. Wen, and R. X. Li, “Trustworthy Assurance of Service Interoperation in Cloud Environment”, *International Journal of Automation and Computing*, Vol.8, No. 3,2011,pp. 297–308.

[4] M. Armbrust, A. Fox, R. Grifth, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D.Patterson, A.Rabkin,I. Stoica, and M. Zaharia, “Above the clouds: A berkeley view of cloud computing”,*Electrical Engineering and Computer Sciences*, Vo15, No.4,2009, pp. 259–268.

[5] R. Buyya, S. Pandey, and C. Vecchiola, “Cloudbus toolkit for market-oriented cloud computing”,*Proceedings of the 1st International Conference on Cloud Computing, ACM, Berlin, Germany*, 2009, pp. 24–44.

[6] Q.N.Deng ,and Q.Chen, “Cloud Computing and Key Technology”, *Computer Application*, Vol. 29,No 9,2009,pp. 2562-2567.

[7] V. D. Martino, and M. Mililotti, “Sub Optimal Scheduling in a Grid Using Genetic Algorithms”,*Parallel Computing*, Vol. 30, No. 5–6 , 2004, pp. 553–565.

[8] X.Pu, and X.L.Lu, “Grid Task Scheduling Based on the Improved Genetic Algorithm and QoS Constraints”, *Journal of Electronic Science and Technology University*, Vol.39,2010, pp. 54-60.

[9] Buyya R, Ranjan R, and Calheiros RN, “Modeling and Simulation of Scalable Cloud Computing Environments and CloudSim Toolkit: Challenges and Opportunities”, *Proceedings of the Conference on High Performance Computing and Simulation (HPCS)*,Leipzig, Germany. IEEE Press: New York, U.S.A., June 21–24, 2009,pp. 1–11.