A task scheduling algorithm based on genetic algorithm and ant colony optimization in cloud computing

Chun-Yan LIU
Department of Information Engineering
Wuhan university of technology HuaXia college
Wuhan, China
liuchunyan210@126.com

Cheng-Ming ZOU,Pei WU
School of Computer Science and Technology
Wuhan University of Technology
Wuhan, China
zoucm@hotmail.com

***Abstract*—An efficient approach to task scheduling algorithm remains a long-standing challenge in cloud computing. In spite of the various scheduling algorithms proposed for cloud environment, those are mostly improvements based on one algorithm.And it's easy to overlook limitations of the algorithm itself. Aiming at characteristics of task scheduling in cloud environment, this paper proposes a task scheduling algorithm based on genetic-ant colony algorithm. We take the advantage of strong positive feedback of ant colony optimization (ACO) on convergence rate of the algorithm into account.But the choice of the initial pheromone has a crucial impact on the convergence rate. The algorithm makes use of the global search ability of genetic algorithm to solve the optimal solution quickly, and then converts it into the initial pheromone of ACO. The simulation experiments show that under the same conditions, this algorithm overweighs genetic algorithm and ACO, even has efficiency advantage in large-scale environments. It is an efficient task scheduling algorithm in the cloud computing environment.**

Keywords-cloud computing; task scheduling; genetic algorithm; ant colony optimization

I. INTRODUCTION

Cloud computing is an emerging technology where information technology resources are provisioned to users in a set of a unified computing resources on a pay per use basis[1].Cloud computing using virtual technology parts the huge computing task into a number of small tasks through the network, which are next allocated into the huge system consisting of multiple servers by some allocation methods, then returning the results to the user after computing[2-5]. Thus, key points and difficulties of cloud computing are how to reasonably carry out the task scheduling and resource allocation. It is an important topic that designs an excellent performance scheduling algorithm to improve quality of service in cloud environment.

Task scheduling in cloud computing has attracted great attentions. Many researchers have proposed different scheduling algorithms which run under the cloud computing environment. However, most task scheduling algorithms that have been proposed are based on an improved algorithm. Here, we review the most relevant research works done in the literature for scheduling algorithm.A bandwidth-aware algorithm is proposed for dicisible task scheduling in could computing environment ,which is on the basis of the optimized allocation scheme[6]. the Jianfeng LI,Jian PENG design a task scheduling algorithm based on double fitness genetic algorithm. It improves the efficiency of cloud computing, through setting two optimization goals. One is total task completion time and the other one is average task completion time[7]. Liangliang FENG, Tao Zhang, Zhen-hong Jia, Xiao-yan Xia, Xi-zhong Qin propose a task scheduling algorithm based on improved particle swarms, considering the total task completion time and the total cost to complete tasks[8].The pricing and peak aware scheduling algorithm for cloud computing is proposed in 2012,which demonstrated feasibility of interactions between distrebutors and one of their heavy use customers in a smart grid environment[9].Xia-yu Hua, Jun Zheng, Wen-xin Hu introduce a cloud computing resource allocation method based on ACO, which is fully taking inherent properties of computing resources and the node's load into account, in order to predict the execution speed[10]. Jing-zhao ZHANG,Tao JIANG propose an improved adaptive genetic algorithm, to a certain extent, solve the traditional genetic algorithm "premature" issue, and accelerate the convergence rate[11]. Zong-bin ZHU,Zhong-jun DU propose improvements GA cloud computing task scheduling algorithm, considering two elements, the time and cost of the task scheduling[12]. Jian-ping LUO,Xia LI,Min-rong CHEN address the problem of resource scheduling based on shuffled frog leaping algorithm, combining with the tasks and resources, and propose two types of network coding structures on the basis of which make merits of individual choice according to the value of Qos[13]. Ming-hai XU,Yuan ZI propose a network selection based on ant colony optimization, where the feedback mechanism is introduced into the network selection algorithm to choose the right path using pheromone concentration[14]. What's more,there has an improved differential evolution algorithm based on the proposed cost and time models on cloud computing environment ,and this algoruthm can optimize task scheduling and resource allocation[15].Analyzing the above task scheduling algorithms, we can find that it is easy to overlook the inherent limitations of algorithm itself, as optimization ability of genetic algorithm at late stage is poor and prone to premature degradation, where colony algorithm's searching is inefficient. Combining the

978-1-4799-4169-8/14 $31.00 © 2014 IEEE
DOI 10.1109/DCABES.2014.18

advantages and disadvantages of various intelligent algorithms and taking global search capability of genetic algorithm and high accuracy of ACO into account, the paper proposes a genetic-ant colony scheduling algorithm (GA-ACO algorithm) integrating the global search capability of genetic algorithm and high accuracy of ACO, which shows its good performance through simulation experiments.

II. PROBLEM OF TASK SCHEDULING IN CLOUD COMPUTING

In cloud computing, task scheduling policy directly affects the efficiency of the user's tasks and the efficient usage of resources under the cloud environment. Hence, how to achieve optimal allocation of user's tasks is the key issue of task scheduling in cloud computing. The process of task scheduling under the cloud environment is as follows.Firstly, tasks and resources will be mapped according to current task and resource information in accordance with certain strategy. Then follow the mapping between the resources allocated to the implementation of the task to ensure the efficiency of the task and the quality of service requirements of users. Finally the summary of the results is excuted to the submitting user.

The current cloud computing environment is mostly built according to MapReduce programming model, which is an efficient task scheduling model especially for the generation and processing of large data sets[16-20]. The specific implementation process was shown in Fig.1.

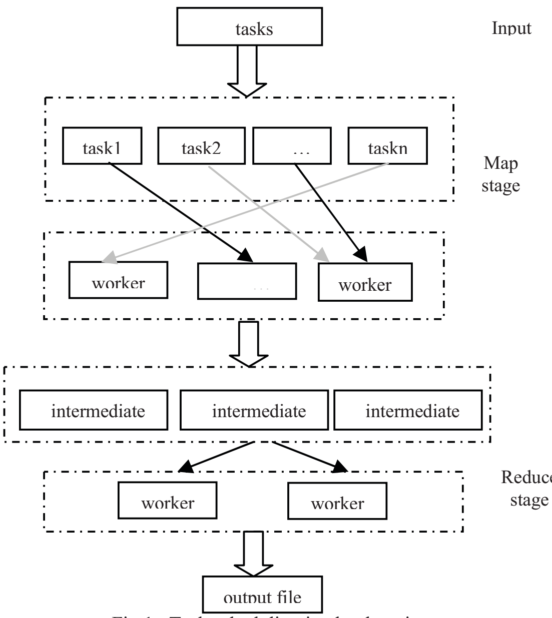

Fig 1 Task scheduling in cloud enviroment

Fig.1 illustrates the proposed model for task scheduling under cloud environment which consists of two stages, namely, the map and the reduction. The core idea of MapReduce is to divide a parallel processing task execution stage into Map and Reduce stages. In the Map stage user's task is divided into smaller sub-M tasks by MapReduce function, which allocates them to multiple workers, and then there will output an intermediate file . In the reduce stage ,results will be outputted after the treatment of the map pooled analysis of the pre-result

III. TASK SCHEDULING ALGORITHM BASED ON GENETIC-ANT COLONY ALGORITHM IN CLOUD COMPUTING

This study aims at task scheduling problem in the cloud computing.In order to get the best result of task scheduling and takes less time,an integrate of the genetic algorithm and ACO effectively can be made r,eferring the MapReduce model.Accordingly, there proposes a GA-ACO algorithm.

A. Design Ideas

The main idea of the GA-ACO algorithm are as follws. In the early stage of task scheduling ,it takes advantage of genetic algorithm's global search ability, and forms chromosome by indirect encoding.Then choose reciprocal of task completion time as the fitness function.After selection, crossover and mutation, generate the optimal solution and convert this solution into ACO's initial pheromone, and form optimal solution of task scheduling through the feature of positive feedback and efficiency.

B. Rules of Genetic Algorithm

1) Chromosome Encoding and Decoding

To solve the problem of the task scheduling under cloud environment, we should encode scheduling scheme into chromosomes where each chromosome represents a particular scheduling scheme. This paper applies an indirect encoding method. Specific operation is as follows: Each task occupying the resource is encoded. The length of the chromosome equals to the total number of sub-tasks. The number of each bit position represents the gene sub-tasks' number and the value of Gene-bit represents the number of the occupied resource.

The amount of sub-tasks is calculated using (1):

$$subtaskNum = \sum_{t=1}^{m} taskNum(t) \qquad (1)$$

Where:
m = number of tasks
t = the order of task
$taskNum(t)$ = the number of sub-tasks assigned to task t

For example, it assumes that there are 3 tasks ,then m=3.And 3 resources means n=3 and 3 tasks are divided into 3,4,2 sub-tasks, meaning that $taskNum(1)$=3, $taskNum(2)$=4, $taskNum(3)$=2,so $subtaskNum$=9.It means that the length of the chromosome is 9. Setting the value scope of the gene to be (1, 3). Applying method of indirect encoding to generate a set of chromosomes: {2, 3, 1, 2, 3, 1, 2, 1, 1}. The first sub-task is assigned to the second resource, the second sub-task is assigned to the third resource, and so on. Then decode these chromosomes, and obtain the distribution of various resources on tasks, w1:{3,6,8,9} w2:{1,4,7} w3:{2,5}

2) The Objective Function and the Fitness Function

Calculate the execution time to perform tasks for each resource using decoded sequence and ETC matrix. It follows that the total time to complete the task of resource scheduling,as in:

$$F(x) = \max_{r=1}^{n} \sum_{i=1}^{w} work(r,i) \quad (2)$$

Where:

Work(r, i) = the time spent by the resource *r* performing sub-task *i* which is on this resource.

w= the quantity of the sub-tasks assigned to the resource.

Equation (2) is defined as objective function.

The fitness function is used to evaluate chromosomes' pros and cons. The value of the function is bigger, then the chromosomes' survivability is stronger and the function's solution is better. Since the value of the fitness function is the reciprocal of objective function, and the time is shorter, and the fitness value is bigger, and the probability of being selected is larger.

The fitness function is defined as:

$$f(x) = 1/F(x) \quad (3)$$

3) Genetic Manipulation

Genetic manipulation of genetic algorithm including selection, crossover and mutation. And through these operations it continues to generate new individuals so as to search out the optimal solution.

a. selection

Probability of selection for each individual is calculated based on the value of fitness function. Equation (4) illustrates how the probability of selection is computed:

$$P(i) = \frac{f(i)}{\sum_{j=1}^{SCALE} f(j)} \quad (4)$$

b. crossover

This paper chooses adaptive crossover methods. Larger crossover probability exchange some bit between individuals, so that it can avoid the occurrence of premature. In the latter part of the algorithm, as crossover probability decreases, it is easier to generate new good individual and accelerate the convergence rate.

c. mutation

This paper adopts single point mutation to change some individual bits in groups for smaller probability, like "1"to "0","0"to "1".

In actual operation, it eliminates the new individuals whose value of fitness function is less than the average value after several recursive iterations, and gets the optimal solution of certain groups as a basis for obtaining a pheromone ACO.

C. Exact Solution based on ACO

1) Combining Genetic Algorithm and ACO

Evaluate the chromosome population according that successive five generations' evolutionary rates are small.Then the genetic algorithm can be terminated and enter the ACO.

When genetic algorithm is terminated, sort individuals in the population according to the size of the fitness function values, from which the top 10% of individuals are selected as an optimization solution and converted to initial pheromone. The specific initialization rule is shown in (5):

$$T_i^G(0) = \rho S_n \quad (5)$$

Where:

ρ = self set constant

S_n = as the genetic algorithm optimization solution

Through the operation of genetic algorithm we can obtain the distribution of pheromone. The initial value of resource pheromone is set in (6):

$$T_i(0) = r_i + T_i^G(0) \quad (6)$$

Where:

r_i = processing capacity of the resource

$T_i^G(0)$ =the pheromone value transformed from the optimal solution when the current Genetic Algorithms is terminated.

2) Path Selection

Each ant determines the probability of the next resource according to the information of the current resource.

$$P_{k,}(i,j) = \frac{[T_j(t)]^{\alpha}[\eta_j]^{\beta}}{\sum_{\in U}([T_u(t)]^{\alpha}[\eta_u]^{\beta})} \quad (7)$$

Where:

$T_j(t)$ = the value of the pheromone in resource j at the moment t

η_j = processing capacity of the resource j

α or β = the importance of the pheromone

3) Update Pheromone

By comparing the performance of ACO, the method of global updating the pheromone can improve the convergence efficiency. That is, when an ant successfully completes a resource selection , the pheromone will change. Pheromone update rule is shown in (8):

$$T_j^{new} = \rho T_j^{old} + \Delta T_j \quad (8)$$

Where:

ρ = termination condition of ACO

When the cycle counter N reaches the maximum number of iteration's range, the current value is the optimal scheduling scheme, and then the ACO terminates.

IV. RESULTS AND DISCUSSION

In order to verify the feasibility and effectiveness of GA-ACO algorithm, we need to simulate it from the performance of its task scheduling. Subsequently, experimental results are compared to genetic algorithm and ACO under the same environment.

A. Parameter Settings

For advantages of each searching and solving by genetic algorithm and ACO with repeated experiments, various numbers of scenarios with different parameters values are taken into consideration during simulation. Table 1 summarizes the simulation parameters used in these experiments.

TABLE I. THE PARAMETERS SETTINGS

Algorithm	Parameter	Value
GA	Number of population	100
	Crossover rate	0.6
	Mutation rate	0.1
ACO	Number of ants	100
	α	1
	β	1
	ρ	0.3
	a	1
	b	0.8

B. *Experimental Results and Analysis*

All algorithms are implemented and tested on the platform of clouds IM simulator. In the experiments carried out in this study, Fig.2 depicts the results under the same environment where the number of tasks are 50, and compares the success rate of searching the optimal solution and the number of iterations among GA-ACO algorithm, GA and ACO.

From the Fig.2, it can be noted that success rate of searching the optimal solution reaches up to 98% when the number of iteration is 28 in GA-ACO algorithm experiment, but when the number of iteration is 50, GA algorithm's rate reaches up to 63% and ACO algorithm's rate reaches up to 95%.

From the Fig.2, it can also be concluded that GA-ACO algorithm requires less iterations to find the optimal solution. And it's solving efficiency is better than GA and ACO significantly. This is because the GA-ACO algorithm converts several optimization solution generated by GA into pheromone of ACO, greatly shortening the time to collect pheromone.

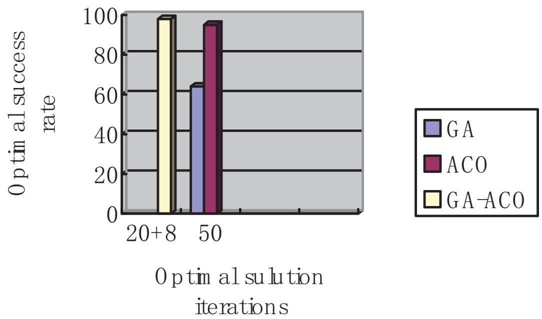

Fig.2 Comparison of optimal success rate

In another experiment, we used CA-ACO algorithm, GA and ACO respectively to test the performance of schedule tasks. It is sampled data once in each task number 50, 100, 200,300,400,500. The simulation effect diagram by the three Algorithms is as follow.

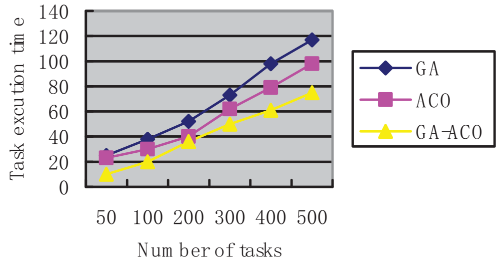

Fig 3 Comparison of task execution time

As we can see from Figure 3, it illustrates the observation of the execution time with the increasing number of tasks for each algorithm. From the figure, it is clear that when the number of tasks is smaller, the resources are more adequate in cloud environment. As for the task execution time ,all these three algorithms relatively cost little.And among the three algorithms, GA-ACO algorithm is slightly better than ACO and GA, though the gap is not obvious. With increasing number of tasks, the increasing trend of the execution time spent by CA-ACO is significantly less than the other two algorithms. What's more, the performance's improvement is obvious. This is main due to that the increasing number of tasks results in a high load for each algorithm which leads to extend the execution time. However, at a larger number of tasks, GA-ACO algorithm makes use of its own advantages, avoiding the defects of GA's local searching and ACO's lacking initial pheromone.

V. CONCLUSIONS

This paper makes some researches on task scheduling under cloud environment, aiming at solving the slow convnergence problem caused by the lack of initial pheromone of ACO .Then there introduces the GA-ACO(the integration of genetic algorithm and ACO), which uses the strong global search capability of GA to get better solution, and then converts it into the initial pheromone of ACO, and finally gets optimal scheduling through positive feedback of ACO. Based on the simulation results, it shows that the integration of GA and ACO is beneficial to be used in cloud computing to solve the task scheduling, as it effectively improves the searching efficiency of algorithm.

ACKNOWLEDGMENT

We would like to thank to the organizers of the committee of DCABES 2014 and the editor of the IEEE CPS who offer us the opportunity and posibility to have our paper published.Also we want to thank to the professors who give us many suggestions to this paper.The paper is supported by "the Fundamental Research Funds for the Central Universities (WUT:2014-Ⅶ-027)

REFERENCES

[1] LeiLa Ismail,Rajeev Barua.Impementation and performance evalution of a distributed conjugate gradient method in a cloud computing environment,Software-practice and Experience,2013,pp.281-304.

[2] Abadi D J. Data management in the cloud:Limitations and opportunities.Bulletion of the IEEEComputer Society Technical Committee on Data Engineering,2009,32(1):3-12.

[3] FOSTER I,ZHAO Y,RAICU I, et al.Cloud computing and grid computing 360-degree compared[C].In Proc of IEEEGrid Computing Environments Workshop,2008:1-10.

[4] Peng Liu. Cloud computing[M].BeiJing: Electronic Industry Press,2007:2-13

[5] Rodrigo N.Calherios,Rajiv Ranjan,Anton Beloglazov.CloudSim:a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms,Software Practice and Experience,2011,41(1),pp.23-50.

[6] Chen Liang,James Z.Wang,Rajkumar Buyya.Bandwidth-aware divisible task scheduling for cloud computing,Software-practice and Experience,2014,44:163-174.

[7] Jian-Feng Li,Jian Peng.Task scheduling algorithm based on improved genetic algorithm in cloud computing environment[J].Jounal of Computer Applications. 2011,31(1):184-186.

[8] Liang-liang Feng, Tao Zhang, Zhen-hong Jia, Xiao-yan Xia, Xi-zhong Qin.Task schedule algorithm based on improved particle swarm under cloud computing environment[J].Computer Engineering, 2013,39(5):183-186.

[9] Xuan Li,Jine-Chung Lo.Pricing and peak aware scheduling algorithm for cloud computing,2012 IEEE PES Innovative Smart Gird Technologies,pp.1-7.

[10] Xia-yu Hua, Jun Zheng, Wen-xin Hu. Ant colony optimization algorithm for computing resource allocation based on cloud computing environment[J].Journal of East China Normal University(Natural Science), 2010,2(1):127-134.

[11] Jing-zhao ZHANG,Tao JIANG.Improved adaptive genetic algorithm[J].Computer Engineering and Applications,2010,46(11):53-55.

[12] Zong-bin ZHU,Zhong-jun DU.Improved GA-based task scheduling algorithm in cloud computing[J]. Computer Engineering and Applications,2013,49(5):77-80.

[13] Jian-ping LUO,Xia LI,Min-rong CHEN.Guaranteed QoS resource scheduling scheme based on improved shuffled frog leaping algorithm in cloud environment[J].Computer Engineering and Applications,2012,48(29):67-72.

[14] Ming-hai XU,Yuan ZI.ACO-based network selection algorithm[J].Computer Engineering and Applications,2012,48(5):84-88.

[15] Jinn-Tsong Tsai,Jia-Cen Fang,Jyh-Horng Chou.Optimized task scheduling and resource allocation on cloud computing environment using imprived differential evolution algorithm,Computers & Operations Research,2013,pp.0305-0548.

[16] J.Dean,S.Ghemawat.Mapreduce.Simplified data processing on large clusters,Sixth Symposium on Operating System Design and Implementation,San Francisco,CA,USA,December 2004

[17] Bhandarkar,M.MapReduce programming with apache Hadoop,Parallel&Distributed Processing(IPDPS),2010 IEEE International Symposium on Digital Object Identifier,2010.

[18] J.Dean and S.Ghemawant,"MapReduce:Simplified Data Processing on Large Clusters,"In Communications of the ACM,Vol.51,Issue 1,2008,pp.107-113.

[19] X.H.Zhang,Z.Y.Zhong,S.Z.Feng,B.B.Tu,and J.P.Fan."Improving Data Locality of MapReduce by Scheduling in Homogeneous Computing Environments," In IEEE 9th International Symposium on Parallel and Distributed Processing with Applications,2011,pp.120-126,doi:10.1109/ISPA.2011.14.

[20] Kristi Morton,Magdalena Balazinska, and Dan Grossman,"Paratimer: a progress indicator for mapreduce DAGs,"In SIGMOD'10:Proceedings of the 2010 international conference on Management of data,pp.507-518,New York,NY,USA,2010.ACM.