Research on the Task Scheduling Algorithm for Cloud Computing on the Basis of Particle Swarm Optimization

Tao Jiang *, Jia Li

Chongqing Technology and Business Institute
Chongqing 400052, China

Abstract — **This paper explores the task scheduling algorithm for cloud computing on the basis of Particle Swarm Optimization (PSO). Based on task scheduling problems of the cloud computing, first of all, this paper detailed introduction to cloud computing, task scheduling of cloud computing, particle swarm optimization algorithm and ant colony optimization algorithm. On this basis of the above, task scheduling algorithm based on particle swarm optimization and ant colony optimization for cloud computing (PSO-ACO) is presented in this paper. The scheduling algorithm absorbs the fast convergence ability of particle swarm optimization algorithm and the optimization ability of ant colony algorithm, and considers the cost factor. The scheduling algorithm improves the speed of the cloud computing task scheduling and optimization performance. Finally, we evaluate the task scheduling algorithm in cloud environment which was simulated by C1oudSim. In the meantime, the result of the first experiment and the second experiment show that the scheduling algorithm can not only quickly find the optimal solution, but also minimize the processing time and costs of all tasks.**

Keywords - The Task Scheduling Algorithm; Cloud Computing; Particle Swarm Optimization (PSO); Optimization Performance

I. INTRODUCTION

The concept of cloud computing comes from the concept of grid computing, parallel computing and distributed computing. It is a new business computing model. Cloud computing commercial value is huge and has a huge impact on the TT operation pattern, so cloud computing was researched by business organizations and scientific research institution both at home and abroad. In the cloud computing environment, users are numerous, the amount of task which the system to deal with is very huge, and cloud computing system structure is very complex [1]. So in order to make the cloud computing system is fast enough to deal with the service request, meet the users' quality of service, how to efficiently scheduling tasks in the "cloud" so as to realize the global optimization, become the key points and difficulties of cloud computing research.

Heterogeneous multiprocessor system consists of a set of processors with different processing capacities. Task scheduling is a crucial factor in improving the efficiency of this system. It needs to resolve the problem of how to allocate tasks to different processors so that the system can obtain the highest performance. Traditional scheduling algorithms face new challenges because of the heterogeneousness, complexity and flexibility of the heterogeneous multiprocessor system.

Therefore it is very important and realistic to put forward a better scheduling algorithm, which can make full use of all kinds of resources and improve the throughput and resource utilization of heterogeneous multiprocessor system based on analyzing the existing algorithms [2].

Particle Swarm Optimization algorithm is a new kind of modern heuristics algorithm and it is well characterized by its self-organizing, self-learning, self-adaptive characteristics and the implicit parallelism and guided search, which is often used to solve different kinds of NP-complete problems and complex task scheduling problems. Some simulation experiments have confirmed [3] that Particle Swarm Optimization algorithm has more advantages compared with traditional scheduling algorithms in dealing with task scheduling problem.

(1) Considering the issues of independent tasks matching and scheduling of the heterogeneous multiprocessor system, an improved Particle Swarm Optimization (IPSO) algorithm is presented to enhance the ability of searching optimal solution. When calculating the fitness function, the paper makes rounded operation of the value of particle location to make PSO algorithm better apply to discrete areas. A performance index is established by analysing the computation ability of each processor. Adjusting method on inertia weight is presented to improve the global convergence and overcome the defect that the searching ability of particle is decreasing during the later stage of iteration. The results show that our improved algorithm is able to find better schedule quality in a shorter time than other PSO algorithms [4].

(2)Considering the issues of directed acyclic graph tasks matching and scheduling in heterogeneous multiprocessor system, a hybrid particle swarm optimization (HPSO) algorithm is presented to enhance the ability of searching optimal solution. An optimization mathematical model of multiprocessor scheduling problem is established in this paper. The concept of Swap Operator is introduced to construct a kind of special particle swarm optimization algorithm, which makes PSO algorithm apply to discrete areas [5-6]. Then Hill-climbing algorithm is presented to overcome the defect of its precocious convergence and bad local optimization ability, which can improve the solution quality and accelerate the convergence of the algorithms. Compared with TPSO and Genetic Algorithm, our improved algorithm is able to find better schedule quality in a short time and is especially useful in solving the heterogeneous multiprocessor scheduling problem with a number of tasks and processors. The task scheduling problem is NP-complete problem. It can shorten the completion time and improve the efficiency of the heterogeneous multiprocessor system with the characteristics of parallelism and global solution space when using the particle swarm optimization algorithm to solve the task scheduling problem. The research result of this thesis is valuable for spreading the application of particle swarm optimization algorithm.

II. THE BASIC FRAMWORK AND BASIS OF PSO

The particle swarm optimization (PSO) is a population-based algorithm that was invented by Kennedy and Eberhart, which was inspired by the social behavior of animals such as fish schooling and bird flocking. Similar to other population-based algorithms, such as evolutionary algorithms, PSO can solve a variety of difficult optimization problems but has shown a faster convergence rate than other evolutionary algorithms on some problems. Another advantage of PSO is that it has very few parameters to adjust, which makes it particularly easy to implement. It was pointed out that although PSO may outperform other evolutionary algorithms in the early iterations, its performance may not be competitive as the number generations are increased [7-9]. Recently, several investigations have undertaken to improve the performance of standard PSO (SPSO). Lobjerg et al. presented a hybrid PSO model with breeding and subpopulations Kennedy and Mendes investigated the impacts of population Structures to the search performance of SPSO. Other investigations on improving the performance of SPSO were undertaken using cluster analysis and fuzzy adaptive inertia weightier.

The foundation of SPSO is based on the hypothesis that social sharing of information among conspecifics offers an evolutionary advantageous. The SPSO model is based on the following two factors: (1) The autobiographical memory, which remembers the best previous position of each individual in the swarm; (2) The publicized knowledge, which is the best solution found currently by the population. The PSO algorithm has shown its robustness and efficiency in solving function value optimization problems and many other research fields, such as practices of neural network, controls of fuzzy system, etc. However, similar to other global optimization algorithms such as genetic algorithms, PSO may be trapped in local minima, especially in optimizing complex multimodal functions. Enlarging population size can help to improve the performance but fail to settle thoroughly this kind of problem, which is so-called premature convergence. In this paper, a new PSO with mutation operator (PSOMO) is presented based on the standard PSO. By introducing mutation operation, the algorithm shows higher calculation efficiency and a faster convergence rate than PSO.

The cloud-computing or virtualization model gives computer users access to powerful computers and software applications hosted by remote groups of servers. Lee and Zomaya [10] investigate the effectiveness of rescheduling using cloud resources to increase the reliability of job completion. Li, Huai, Hu, and Zhu [11] propose a secure collaboration service, called PEACE-VO, for dynamic virtual organizations management. The federation approach based on role mapping has extensively been used to build virtual organizations over multiple domains. Peebler [12] provides information about cloud computing, where consumers have the ability to leverage computer resources that they do not own, manage, or house Paquette, Jaeger, and Wilson [13] discusses the current use of cloud computing in government, and the risks-tangible and intangible-associated with its use Liu, Qiao, Yu, and Jiang [14] discusses the challenges of cloud computing systems using limited bandwidth networking connections. It suggests a combined appropriation of network and computing resources in a consolidated computing system directly established on a wavelength-division multiplexing (WDIV) network. Cloud computing is an emerging new for users use the personal computers or notebooks. We program the system and software with Extensible Mark-up Language (XML) and C sharp language. If the users begin to search, the kernel safety computing paradigm for delivering computing services [15]. Webley, Dehn, Lovick, Dean, Bailey, and Valcic discuss on how information of the ash cloud such as location, particle size and concentrations, could be used as VATD model initialization. Xie present a high-accuracy method for fine registration of two partially overlapping point clouds that have been coarsely registered. Neumann propose heuristic and pricing schemes find an interesting match between scalability and strategic behavior Tabu search and ant colony perform better for large-sized problems, whereas simulated annealing is optimal for small-sized problems and it is therefore essential that a maintenance scheduling optimizer can incorporate the options of shortening the maintenance duration and/or deferring maintenance tasks in the search for practical maintenance schedules. A two-stage assembly flow-shop scheduling problem with a weighted sum of makes span and mean completion time criteria, known as bicriteria is prompted. The learners and lecturers agree that style-based ant colony systems can provide useful supplementary learning paths.

Ant colony intelligence (ACI) is proposed to be combined with local agent coordination in order to make autonomous agents adapt to changing circumstances, thereby yielding efficient global performance. This indicates that the ACO algorithm is an optional compromise strategy between preferable phase unwrapping precision and time-consuming computations.

III. THE BASIC MODEL AND ALGORITHM OF PSO

PSO is a population-based optimization algorithm. The population of PSO is called a swarm and each individual in the population of PSO is called particle. The ith particle at iteration k has the following two attributes:

(1) A current position in an N-dimensional search space $X_i^k = \left(X_1^k, ..., X_n^k, ..., X_N^k\right)$ where $X_n^k \in \left[l_n, u_n\right]$,

DOI 10.5013/IJSSST.a.17.04.11

ISSN: 1473-804x online, 1473-8031 print

$1 \le n \le N$, l_n, u_n is lower and upper bound for the nth dimension, respectively.

(2) A current velocity V_i^k, $V_i^k = \left(V_1^k, ..., V_n^k, ..., V_N^k\right)$ which is bounded by a maximum velocity $V_{\max}^k = \left(V_{\max,1}^k, ..., V_{\max,n}^k, ..., V_{\max,N}^k\right)$, and a minimum $V_{\min}^k = \left(V_{\min,1}^k, ..., V_{\min,n}^k, ..., V_{\min,N}^k\right)$.

In each iteration of PSO,the swarm is updated by the following equations:

$$V_i^{k+1} = \omega^{V_i^k} + C_1 r_1 \left(P_i^k - X_i^k\right) + C_2 r_2 \left(P_g^k - X_i^k\right) \quad (1)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (2)$$

Where P_i is the best previous position of the ith particle (also known as pbest). According to the different definitions of P_g there are two different versions of PSO. If P_g is the best position among all the particles in the swarm (also known as gbest), such a version is called particles the global version. If P_g is taken of the population (also known as best), from some smaller number of adjacent such a version is called the local version.

The r1 and r2 are elements from two uniform random sequences in the range (0, 1): r1 ~ U (0, 1); r2 ~ U (0, 1). The variables c1 and c2 are acceleration constants, which control how far a particle will move in a single iteration .In general, the acceleration constants c 1 and c2 are both 2 .The variable w is an inertia weight, which is initialized typically in the range of [0,1]. It is regarded that a larger inertia weight facilitates global exploration and a smaller inertia weight tends to facilitate local exploration to fame-tune the current search area. Thus, Shi and Eberhart made a significant improvement in the performance of the PSO with a linearly varying inertia weight over the generations, which linearly vary from 0.9 at the beginning of the search to 0.4 at the end. Therefore, the inertia weight factor cc can be defined as follows.

$$\omega = \omega_{\max} - iter \times \frac{\omega_{\max} \left| \omega_{\min} \right.}{iter_{\max}} \quad (3)$$

Where $\omega_{\max}$ and $\omega_{\min}$ is the maximum and minimum of ω and iter is current iteration while itermax is maximum iteration.

It had been discovered aggregated at one or particles among the swarm would tend to be particular places as the PSO algorithm kept on running. In order to describe quantitatively the convergence status of the particle swarm, the variance of the population's fitness (σ^2) is introduced and defined as follows:

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^{n} \left[\frac{f\left(X_i - favg\right)}{f_d} \right]^2 \quad (4)$$

$$E_{TX}(l,d) = \begin{cases} l*E_{elec} + l\varepsilon_{fs} d^2 (d<d_0) \\ l*E_{elec} + l\varepsilon_{amp} d^4 (d>d_0) \end{cases} \quad (5)$$

$$E_{RX}(l) = l*E_{elec} \quad (6)$$

In the formula, $d_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{amp}}}$.

l bit data is integrated and forwarded, and computational formula of consuming by a cluster head node is as follows:

$$E_{da-fu}(l) = l*E_{DA} \quad (7)$$

$$T = t_{work} + t_{sleep} \quad (8)$$

$$T_s = t_{s1} + t_{s2} \quad (9)$$

Therefore，the procedure of the PSOMO algorithm can be described as follows.

Step 1. Randomly initialize positions and velocities of all particles.

Step 2. For each particle, set P_i to the current position X_i, and set P_g to the current best position of the swarm.

Step 3. For each particle in the swarm

Step 3.1. Update velocities Vi and positions Xi using Eq. (1) ~ (3);

Step 3.2. Calculate the fitness value of current particle: f (Xi);

Step 3.3. Compare the fitness value of P with f (Xi).If f (Xi) is better than the fitness value of P_i, then set P_i to the current position Xi.

Step 3.4. If f (Xi) is better than the fitness value of P_g, then g is set to the position of the current Xi.

Step 4. Calculate the variance of the population's fitness σ^2 using Eq. (4) and (5).

Step 5. Calculate the mutation probability prob according to Eq. (6).

Step 6. Randomly generate a number r $\in$ [0, 1], if r $\le$ prob, update P_g .

Figure 1 shows the framework of Cloud system and figure 2 shows the application framework. The foundation of SPSO is based on the hypothesis that social sharing of information among conspecifics offers an evolutionary advantageous. The cloud-computing or virtualization model gives computer users access to powerful computers and software applications hosted by remote groups of servers. Figure 3 shows the single cloud and Figure 4 shows the relationship of provider and client.

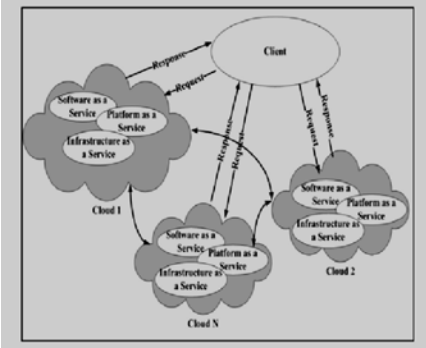

Figure 1. Framework of Cloud system

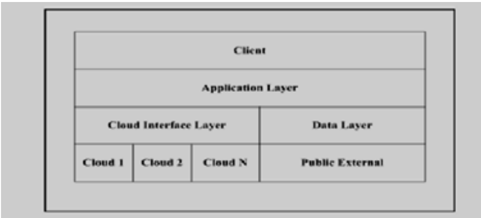

Figure 2. Application Framework

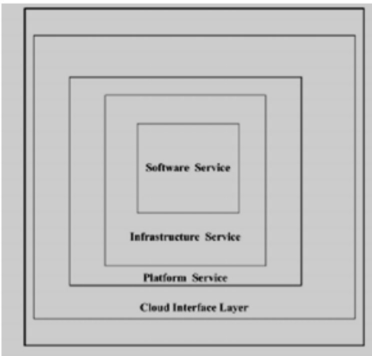

Figure 3. Single Cloud

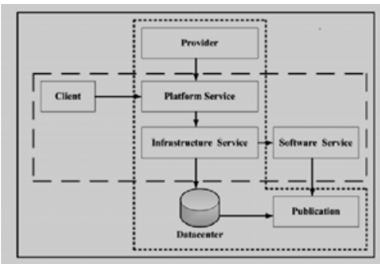

Figure 4. Relationship of provider and client

IV. EXPERIMENTAL STUDIES

Cloud Administration Centre is the access interface to Internet and also the management, scheduling and monitoring center of the resources within the cloud. The administration center of one public cloud accepts the resources request from the Internet users and creates the demanded resources, e.g. virtual machine and storage resources, and configures them, then return the resources to the users.

Cloud computing Resources Centre is composed by the physical computing resources. To one cloud platform, the physical resources will be used as the host machines to be administrated by the cloud administration center. The scheduling server will select the optimal resources according to the user demands to create virtual machines. In general, multiple cloud computing resource centers access the administration center with agent servers which can also be used to support the monitoring and scheduling of the computing resources.

In our experimental studies, a set of four benchmark functions was employed to evaluate the PSOMO algorithm in comparison with PSO.

Sphere's function:

$$R = \sum_{i=1}^{n} x_i^2 \qquad (10)$$

Rastrigrin's function:

$$R = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi - x_i) + 10] \qquad (11)$$

Griewank's function:

$$R = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(x_i / \sqrt{i}) + 1 \qquad (12)$$

Ackley's function:

$$\begin{aligned} &20 + e - 20\exp(-0.2 sqrt(\frac{1}{n}\sum_{i=1}^{n} x_i^2)) \\ &-\exp(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi - x_i^2)) \end{aligned} \qquad (13)$$

The above benchmark functions can be grouped as unimodal (function 1) and multimodal functions (functions 2~4) where the number of local minima increases with the problem dimension. The dimension of each function n is set to 20, feasible solution space. The parameters Xi $\in$ [-100, 100]. our experiments are set as follows. The acceleration constants c1 = c2 =2, the maximum Amax and minimum wmin of w is 0.9 and 0.1, θ_d^2 =1, λ =0.4, the precision of convergence δ = 10-100, the population size M=40, maximum iteration itermax=1~70 .Both algorithms are programmed using MATLAB6.5 and all experiments are repeated for 30 runs. The experimental results for each algorithm on each test function are listed in the figure 5 and figure 6.

From figure 5 and figure 6, PSOMO outperformed the standard PSO algorithm significantly for all benchmark functions .For unimodal function the convergence rate is more important than the final results of optimization as there are other methods such gradient-based search methods that are designed specially to optimize unimodal functions.

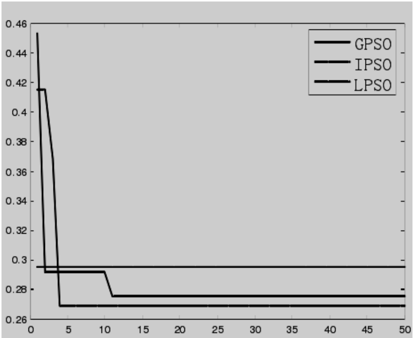

Figure 5. The experiment result A.

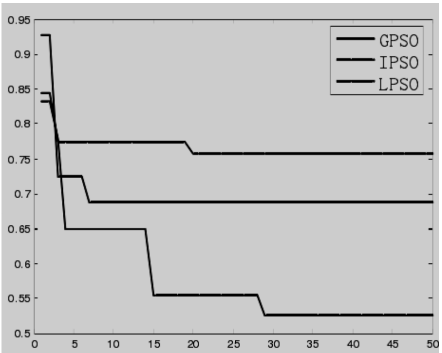

Figure 6. The experiment result B.

V. CONCLUSIONS CONFLICT OF INTEREST ACKNOWLEDGMENT

In cloud computing environment there are a large number of users which lead to huge amount of tasks system. In order to make the system complete the service requests efficiently how to schedule the tasks becomes the focus research. A task scheduling algorithm based on PSO and ACO for cloud commuting is presented in this paper. First to be processed by of cloud computing the algorithm uses particle swarm optimization algorithm to get the initial solution quickly and then according to this scheduling result the initial pheromone distribution of ant colony algorithm is generated. Finally the ant colony algorithm is used to get the optimal solution of task scheduling. The experiment simulated on cloudSim platform shows that the algorithm has good efficient in real-time performance and optimization capability. It is an effective task scheduling algorithm.

In this paper, a new PSO with mutation operator is presented based on the standard PSO. Only a few elementary calculations are included in the PSOMO algorithm that is very easy to implement due to its simple algorithmic structure .Experimental results show that PSOMO not only outperforms PSO in terms of accuracy and convergence rate but also avoids effectively being trapped in local minima.

ACKNOWLEDGMENT

This work is supported by Chongqing Natural Science Foundation with grant No. cstc2014jcyjA40010 and Scientific Project of Chongqing Technology & Business Institute with grant No. ZD2014-05.

REFERENCES

[1] Kennedy J., Eberhart R. C.,"Particle swarm optimization", Proc IEEE International Conference on Neural Networks, IV Piscataway, NJ:IEEE Service Center, vol. 12,pp. 1942-1948, 1995.

[2] Kennedy J., Eberhart R. C., Swarm Intelligence, Morgan Kaufmann Publishers, vol. 15, pp.45-82, 2001.

[3] Angeline P., "Evolutionary Optimization 'Versus Particle Swarm Optimization: Philosophy and Performance Difference " , In: Proceedings of the Evolutionary Programming Conference, San Diago, USA,vol. 45, pp.789-798, 1998.

[4] Lelbjerg M., Rasmussen T.K., Krink K., "Hybrid Particle Swarm Optimizer with Breeding and Subpopulations", In: Proceedings of the Third Genetie and Evolutionary Computation Conference (GECCO-2001), vol. 11, pp. 469-476, 2001.

[5] Kennedy J., Mender R., "Population Structure and Particle Swarm Performance", In: Proceedings of the 2002 Congress on Evolutionary Computation, IEEE Press, vol. 24, pp. 1671-1676, 2002.

[6] Kennedy J., "Stereotyping: Improving Particle Swarrn Performance with Cluster Analysis,In: Proceedings of the IEEE International Conference on Evolutiona-ry Computation, vol. 23,pp. 1501-1512, 2000.

[7] Shi Y, Eberhart R.C.,"Fuzzy Adaptive Particle Swarm Optimization", In: Proceedings of the IEEE International Conference on Evolutionary Computation, vol. 14,pp. 101-106, 2001.

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

[9] 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, vol. 41, no.1, pp.23-50,2011.

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

[11] Zong-bin ZHU,Zhong-jun DU.Improved GA-based task scheduling algorithm in cloud computing. Computer Engineering and Applications, vol. 49, no. 5, pp.77-80,2013.

[12] Jian-ping LUO,Xia LI,Min-rang CHEN.Guaranteed QoS resource scheduling scheme based on improved shuffled frog leaping algorithm in cloud environment.Computer Engineering and Applications, vol. 48, no. 29,pp.67-72,2012.

[13] Ming-hai XU,Yuan ZLACO-based network selection algorithm.Computer Engineering and Applications, vol. 48, no. 5, pp. 84- 88,2012.

[14] 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, vol. 23, pp. 305-548, 2013.

[15] J.Dean,S.Ghemawat.Mapreduce.Simplified data processing on large clusters,Sixth Symposium on Operating System Design and Implementation,San Francisco,CA,USA, pp.125-132, 2004.