Improved PSO-based Task Scheduling Algorithm in Cloud Computing

Shaobin Zhan*, Hongying Huo

Shenzhen Institute of Information Technology, Shenzhen 518172, China

Abstract

Job scheduling system problem is a core and challenging issue in cloud computing. How to use cloud computing resources efficiently and gain the maximum profits with job scheduling system is one of the cloud computing service providers’ ultimate goals. For characteristics of particle swarm optimization algorithm in solving the large-scale combination optimization problem easy to fall into the search speed slowly and partially the most superior, the global fast convergence of simulated annealing algorithm is utilized to combine particle swarm optimization algorithm in each iteration, which enhances the convergence rate and improves the efficiency. This paper proposed the improve particle swarm optimization algorithm in resources scheduling strategy of the cloud computing. Through experiments, the results show that this method can reduce the task average running time, and raises the rate availability of resources.

Keywords: Cloud Computing; Job Scheduling System; Simulated Annealing; Particle Swarm Optimization; Task Schedule

1 Introduction

Job scheduling system is one of the core and challenging issues in a cloud computing system. Traditional job scheduling systems in cloud computing only consider how to meet the QoS requirements for the resources users, they seldom consider how to make the maximum profits for the resource providers. Actually, a job scheduling system plays a very important role in how to meet cloud computing users’ job QoS requirements and use the cloud resources efficiently in an economic way. Usually, from the cloud computing resources users’ sides (we use CCU stands for cloud computing user), users always think which cloud computing resource can meet their job QoS requirements for computing (such as the due time of job finishing, the computing capacity etc.), how much money they must pay for the cloud computing resources. While, from the cloud computing service providers (we use CCSP stand for cloud computing Service Provider) side, the CCSP always think how they can gain the maximum profits by offering cloud computing resources, apart from meeting the CCU’s job QoS requirements. To make these two ends meet,
the job scheduling system must take efficient and economic strategies for CCU’s differentiated service QoS requirements. Focus on this issue, this paper put forward an optimistic differentiated service job scheduling system for CCSP and CCU.

Particle Swarm Optimization (PSO) is an adaptive searching algorithm based on group. Because of its merits of parallel distribution, scalability, easy to realize, strong Robustness, with high flexibility and robust in dynamic environments, PSO solves many combinational optimization problems successfully. Task scheduling problem can select a better one from various combinations distributed to task by resources. To solve the problem, PSO is very suitable to solve resource scheduling problem in cloud environment. Efficiency of scheduling algorithm will directly affect performance of the whole cloud environment. The paper deeply researches the principle of PSO, analyzes special properties of cloud computing environment, puts forward Simulated Annealing (SA) algorithms, improves the optimization and solution speed and advances scheduling efficiency.

2 Related Work

Job scheduling system is a hot and one of core research areas in cloud and grid computing. It plays a similar role in cloud and grid computing. Job scheduling system is responsible to select the best suitable resources in a cloud or grid for CCU’s jobs, by taking some static and dynamic parameters restrictions of CCU’s jobs into consideration. Most research work in grid computing can be used directly in cloud computing environment. Today, we can find many research work have done on job scheduling in grid computing. References [1-6] provided a board view for the roles of job scheduling in a grid computing environment. The presented topologies of job scheduling system in cloud or grid are classified into centralized and decentralized schedulers. Due to the implementation complexity of decentralized schedulers, most related works are on centralized schedulers. Reference [1] gave a brief description of a modeling and performance evaluation of hierarchical job scheduling, [2] showed an iterative scheduling algorithms on the grids. [3] presented a novel stochastic algorithm for QoS-constrained workflows job scheduling in a web service-oriented grid. [4] put forward a definition, modeling and simulation for a cloud computing scheduling system to get high throughput of computing etc.

In recent years, more and more academic researchers began to study the QoS of job scheduling system; we can see that references [5-9] put forward the approach of QoS performance analysis for cloud computing services with dynamic scheduling system. However, most research papers rarely mention the differential service-oriented QoS guaranteed job scheduling system in a cloud computing environment.

Apart from this, very a few papers care about for the how to make the maximum profits for CCSP. For, the conditions of existence for a cloud computing environment are that it must make profits for the CCSP with the lowest system costs. To meet the CCU’s job QoS requirement, job scheduling system should use the cloud computing resource as little as possible.

From a systemic viewpoint of a cloud computing environment, we can take a cloud computing environment as a very powerful server. This server will handle the CCU’s jobs (see Fig. 1). For each CCU may has different QoS requirement, usually, CCU’s jobs have different priorities to be processed. So we can classify the jobs priorities into several classes.
3 Mixed Scheduling Algorithm

PSO has fast speed, but low convergence accuracy. SA has strong commonality, easy to achieve, but long calculation, low efficient, and SA is easy to sink into local optima with serial search. This paper combines PSO and SA with characteristics of job scheduling to get the mixed scheduling algorithm.

3.1 Particle Swarm Algorithm

In PSO, a swarm of particles are used to represent the potential solutions, and each particle $i$ has two vectors, the velocity $V_i = [v_{i1}, v_{i2}, \cdots, v_{iD}]$ vector and the position $X_i = [x_{i1}, x_{i2}, \cdots, x_{iD}]$ vector. Here $D$ means that the solution is in $D$-dimension space. In the initialization, the velocity and position of each particle are set randomly within the search space [13-16]. During the evolutionary process, the particle $i$ is evaluated according to its present position. If the present fitness is better than the fitness of $pBest_{id}$, which stores the best solution that the $i$th particle has been explored so far, then the $pBest_{id}$ will be replaced by the current solution (include the position and fitness). At the same time of determining $pBest_{id}$, the algorithm selects the best $pBest_{id}$ of the swarm to be the globally best, which is regarded as $gBest_d$. Then, the velocity and position of each particle will be updated using the following two Equations (1) and (2).

$$
v_{id}(t+1) = w v_{id}(t) + c_1 r_1 (pBest_{id} - x_{id}(t)) + c_2 r_2 (gBest_d - x_{id}(t)) \tag{1}
$$

$$
x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \tag{2}
$$

where $c_1$ and $c_2$ are acceleration constants and, $r_1$ and $r_2$ are random values in the range [0, 1]. $v_{id}(t)$ and $x_{id}(t)$ represent the velocity and position of the $i$th particle with $d$ dimensions at iteration $t$. $pBest_{id}$ and $gBest_d$ are the best values of positions which are achieved respectively for the $i$th particle and all particles so far. The concept of PSO searching mechanism is illustrated in Fig. 2.

c_1$ and $c_2$ represent the cognitive and social component which lead each particle toward $pBest_{id}$ and $gBest_d$ position. Low values cause roaming far from the target regions before being tugged.
back and high value results in abrupt movement toward the target region. Hence, the acceleration constant $c_1$ and $c_2$ are set to 2 according to past experience [17-19].

The parameter $w$ in Eq. (1) is inertia weight that increases the overall performance of PSO. It is reported that a larger value of inertia weight encourage global exploration while a small one tends to promote local exploration. Suitable selection of inertia weight $w$ usually provides a balance between the global and local exploration and reduce the average number of iteration to locate the optimum solution. To achieve a high performance, we linearly decrease the value of inertia weight from about 0.9 to 0.4 during a run.

3.2 Particle Swarm-simulated Annealing (P-S) Algorithm

From evolutionary process of PSO, PSO has fast convergence speed in initial phase, but through several iterations, particles lose variety, tend to the same, convergence speed becomes slow, which lead to precocious. For overcoming precocious, PSO and SA can be combined, firstly, better swarm is got by fast searching ability of PSO, secondly, partly better individual is optimized by jumping ability of SA. This mixed algorithm is called particle swarm-simulated annealing (P-S) algorithm, which improves the probability and speed of convergence to the optimal solution with advantages of PSO and SA. The flow of P-S algorithm is shown in Fig. 3. SA algorithm takes partly excellent individual of the latest generation swarm of PSO as initial solution, to guarantee the variety of initial swarm, similarity between initial solutions should smaller than 0.9.

3.3 Improved Particle Swarm Optimization (IPSO) Algorithm

PSO algorithm has many advantages. Currently, there are many kinds of improved optimization algorithm of particle swarm, which have very good efficient. Because of strong randomicity of these algorithms, which are easy to sink into defects of local optima and low convergence rate when solving large scale optimization problem. This paper introduces simulated annealing, with its fast random global searching ability, adds it into every iteration of PSO, improves convergence rate, and guarantees solving accuracy of original algorithm. Meanwhile reversal variation strategy is introduced, which avoids sinking into local optima, keeps and increases population diversity.
Choose $M$ relatively optimal individuals as initial solutions from $pB_k$ and population similarity is less than 0.9

Set initial temperature and optima $s_0$, the minimum sampled length $U_{\text{min}}$ and the maximum $U_{\text{max}}$ sampled length

Output the optima $s_0$

Does it satisfy converging rules?

$Y$

$i = 0$

$Y$

Update the optima $s_i$, anneal, if $U_{\text{min}} < U_{\text{max}}$

set $U_{\text{min}} = U_{\text{min}} + 1$

$i < M$?

$N$

$i = i + 1$

Does it satisfy sampled stable rules?

$N$

Generate new state by current state $s_j$

Set new state by accepting new rules

Fig. 3: The flow of P-S algorithm

Based on the IPSO, cloud computing server cluster can fast realize resources discovery, resources matching, scheduling production, task execution.

In global PSO system, information is one-direction flow, $gB^k$ transfer information to other particles, other particles search near $gB^k$, the whole particle swarm evolve to the optima with $gB^k$. $gB^k$ has strong effect on optimal performance of PSO, poor searching ability of PSO for $gB^k$ is one of the main reasons for the prematurity of algorithm. To improve optimal performance of PSO, $gB^k$ can be sampled by SA after every iteration of particle swarm, whose result can be taken as new $gB^k$ of PSO system. Application of SA increase the searching ability of PSO for $gB^k$, so increase the probability of jumping out of local optima. The combined algorithm of PSO and SA is called Improved Particle Swarm Optimization (IPSO) algorithm. Fig. 4 is the flow of IPSO algorithm.

4 Simulation Analysis

CloudSim [20] is taken to stimulate cloud computing environment, for validating superiority of the algorithm, in the same condition, Genetic Algorithm (GA), Simulated Annealing (SA) algorithm, Ant Colony Optimization algorithm (ACO), Improved Particle Swarm Optimization algorithm (IPSO) are taken to solve the scheduling problem, and their performance and results
Select $N$ initial solutions randomly, initialize $w$, $c_1$, $c_2$ and inertia weight decrease coefficient $\lambda$, set initial temperature and annealing coefficient $\beta$, the minimum sampled length $U_{\text{min}}$ and the maximum sampled length $U_{\text{max}}$.

At the end of iteration?

If particle fitness is better than $pB_i^k$, set $pB_i^k = X_i^k$

If particle fitness is better than $gB_i^k$, set $gB_i^k = X_i^k$

$w = \lambda w$, $t = \beta t$
if $U_{\text{min}} < U_{\text{max}}$, set $U_{\text{min}} = U_{\text{min}} + 1$

$\forall i = 0$

Update $X_i^k$ and calculate fitness

Whether satisfy sampling stable standard?

Generate new state from $gB_i^k$

Set new state by accepting rules

Output the optima

Fig. 4: The flow of IPSO algorithm

are compared. Experimental parameters setting is shown in Table 1.

The execution time of each task is shown in Fig. 5. As a whole, GA algorithm and SA algorithm spends more time as the number of tasks increase. ACO algorithm execute task slowly at first, but at the later period its time increase is less than GA algorithm and SA algorithm with improved positive feedback. The execution time and efficiency of IPSO algorithm is better than the other algorithms.

5 Conclusions

For the defect of PSO algorithm which is shown at the time of solving large scale optimization, considering the characteristics of cloud environment, simulated annealing algorithm is added into PSO algorithm, and mixed scheduling algorithm is proposed, which not only increases convergence
Table 1: Parameters setting of the algorithm

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>GA</td>
<td>Population size</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Crossover probabilities ($P_c$)</td>
<td>0.8</td>
</tr>
<tr>
<td></td>
<td>Mutation probability ($P_m$)</td>
<td>0.02</td>
</tr>
<tr>
<td>SA</td>
<td>Operation times before temperature adjusting</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Temperature decrease factor</td>
<td>0.85</td>
</tr>
<tr>
<td></td>
<td>Controlling step vector</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Initial temperature</td>
<td>50</td>
</tr>
<tr>
<td>ACO</td>
<td>Ant number</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Hormone tracking weight $\alpha$</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Heuristic information weight $\beta$</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Hormone evaporation parameter $\rho$</td>
<td>0.2</td>
</tr>
<tr>
<td></td>
<td>Hormone updating constant $Q$</td>
<td>10</td>
</tr>
<tr>
<td>IPSO</td>
<td>Population size</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Self consciousness study factor $C_1$</td>
<td>1.49</td>
</tr>
<tr>
<td></td>
<td>Swarm consciousness study factor $C_2$</td>
<td>1.49</td>
</tr>
<tr>
<td></td>
<td>Inertia factor</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Fig. 5: Task average execution time contrast of the four models

speed of PSO, but also avoid sinking into local optima. The experimental results indicate that, improved particle swarm optimization algorithm shortens the average operation time of tasks, supplies proper resources to user task efficiently in the environment, increases utilization ratio of resources. In future work, we will lay stress on resource load balance of dynamic task scheduling in cloud computing, and consider other resource in cloud computing, for example, how internet quality of service and data distribution influence task scheduling.
Acknowledgements

This work is supported by the Natural Science Foundation of Guangdong Province (9151014002000002) & the SZIIT college key project (ZD201001).

References

[12] Li-ping Zhang, Huan-jun Yu, Shang-xu Hu, Optimal choice of parameters for particle swarm optimization, Journal of Zhejiang University SCIENCE, 6A (6), 2005, 528-534


