# Research on Task Scheduling Strategy Based on Improved Genetic Algorithm in Cloud Computing Environment

Jun Nie

Guangdong University of Science & Technology, Software department, Dongguan City , Guangdong Province, China.

## Abstract

By analyzing parallel scheduling model in cloud environment and studying how to allocate resources in cloud computing efficiently and schedule tasks efficiently, this paper proposes an improved genetic algorithm based on multiple constraints of quality of service. The simulation results show that this improved algorithm not only has higher convergence and the ability to search for the optimal solution, but also reflects the consistency between the scheduling results and user expectations, and provides an effective solution to the task scheduling problem in cloud environment.

## Keywords

Cloud computing; genetic algorithm; task scheduling.

## 1. Introduction

Cloud computing provides the infrastructure, platform and software services, it faces a huge number of computing tasks, so task scheduling and resource allocation issues determine the focus of cloud computing efficiency and difficulty. How to allocate and utilize resources in a cloud environment reasonably, schedule massive tasks submitted by users efficiently, and ensure load balancing of the cloud system becomes one of the focuses of cloud computing research. The task scheduling strategy has a direct impact on the execution time of the task, thereby affecting the performance of the entire cloud and user satisfaction. By analyzing the parallel scheduling model in cloud environment and studying how to allocate resources in cloud computing and scheduling tasks efficiently.This paper proposes an improved genetic algorithm based on multiple constraints of quality of service. Through simulation experiments, it can reflect the consistency between the scheduling result and the user's expectation, and provide an effective solution for the task scheduling problem in the cloud environment.

## 2. Distributed programming model in cloud environment

At present, the task scheduling in the cloud mostly uses the MapReduce distributed computing programming model proposed by Google, as shown in FIG. 1. The tasks submitted by the user is divided into multiple Map tasks and multiple Reduce tasks parallel processing, broadly divided into two phases: Map phase and Reduce phase. In the Map stage, the tasks submitted by the user are divided into M slices and distributed to multiple compute nodes for parallel execution, and then the processed files are output. In the Reduce phase, the results output in the Map stage are further summarized and processed, and the final processing result is output and presented to the user. Because of the large-scale and dynamic tasks in a cloud environment, the number of tasks and computing resources is very large. The system processes the tasks submitted by a large number of users every moment of the time. Therefore, in the MapReduce distributed computing programming Under the model, how to schedule a large number of tasks efficiently and efficiently is the key and difficult

point to decide the cloud computing efficiency. Improper task scheduling will increase the task execution time and reduce the performance and user satisfaction of the whole cloud.

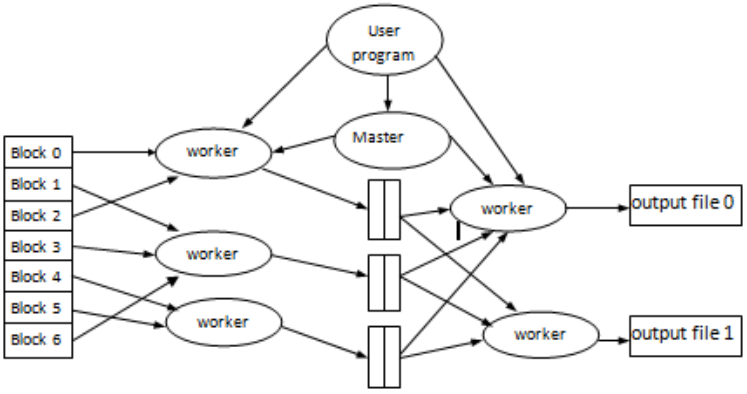


Figure 1 MapReduce distributed computing programming model

## 3. Cloud computing environment based on multi-objective constrained genetic algorithm design

In this paper, we regard the processor, memory and network in the cloud computing as the computing resource. By introducing the different needs of different users, the task scheduling can better meet the expectation of the users in the cloud computing environment.

The task scheduling in the cloud computing environment is described as follows: the total number of resources is P, the corresponding set R = {r1, r2, ..., rn}; the total number of jobs submitted by the user is M, corresponding to the set J = {j1, j2, Assuming that M jobs are divided into N tasks in total, the corresponding jobs T = {t1, t2, ... tN} and the Jn jobs are divided into Tnum (Jn) jobs, the total number of jobs corresponding to M jobs is:

$$N = T_{total}(M) = \sum_{n=1}^{M} T_{num}(J_m) \quad (1)$$

The problem of task height in cloud computing environment is to solve the problem of efficiently executing N interdependent parallel tasks on a limited number of P resources, and at the same time fully satisfy users' requirements on task completion time, occupied bandwidth, reliability and cost Expectations.

### 3.1 Encoding

Considering the assignment of jobs in cloud environment and the order of constraint between them, in order to improve the search ability of the algorithm, this paper adopts the method of allocation scheduling mixed coding. Chromosomes are divided into two parts, the left half of the sub-tasks that the distribution of resources, known as the distribution sub-string, the right half of the substring that the task scheduling order, called the scheduling substring, the length of each chromosome for the total number of tasks 2 times, that is 2L.

(1) Assign substring

Let Ak denote the kth chromosome in the current population and Bki is a chromosomal one, indicating that the ith task uses a resource numbered Bki. The first L genes of the chromosome are described as follows:

$$A_k = \{B_{k1}, \ldots, B_{ki}, \ldots, B_{kN}\} \quad (2)$$

If A = {00011}, it means that the first, second, and third tasks are assigned to resource 0, and the fourth and fifth tasks are assigned to resource 1.

(2) scheduling substring

Scheduling substrings correspond to the last L genes of the chromosomes, and the resulting chromosomes are further processed by assigning substrings. If the chromosome has a complete

encoding of 0001112345, it indicates that tasks 1, 2, and 3 are allocated to resources numbered 0 and the execution order is 1 → 2 → 3, tasks 4 and 5 are allocated resources of number 1, and the execution is performed The order is 45.

### 3.2 Initial population generation

In this paper, a random way to generate the initial population. In the process of generating the initial population, the invalidation scheme is excluded according to the constraint detection scheme. To check whether the scheme is effective, let the set Mf be a set of completed tasks, and then check the task Ti currently scheduled to be executed by each resource in turn. If the task Ti does not have a dependency or all dependent tasks already exist in the set Mf, it indicates that the task Ti can execute, execute Ti and put it into the set Mf, so as to perform the above process. If all the tasks can be executed, the solution is a valid solution, otherwise it is an invalid solution.

### 3.3 Fitness function

This paper combines the characteristics of cloud computing business model, select job completion time, bandwidth, cost and reliability of this goal to quantify the satisfaction of different users.

(1) job completion time

The user's demand for time can be mainly divided into the total job completion time, the start time, the latest completion time and so on. This article selects the total execution time of job execution as the criterion of time requirement. An N × P dimensional ETC (expected time to compute) matrix is provided, where tETC (I, j) represents the expected execution time of the i-th task on the j-th resource in the list, and the execution time of the Jth job It can be expressed as:

$$t(J_n) = \max_{j=1}^{p} \sum_{i=T_{total}(n-1)}^{T_{total}(n)} t_{ETC}(i,j) \tag{3}$$

The total time required to complete all jobs is:

$$t_{total} = \sum_{n=1}^{n} t(J_n) \tag{4}$$

Let texpt assign the expected completion time for the user of job Jn. According to the definition of the user satisfaction function, the job satisfaction of the job Jn is:

$$W_{time}(J_n) = \alpha L = \alpha \ln\left[\frac{t(J_n)}{t_{expt}}\right] \tag{5}$$

(2) Bandwidth

Let Bwn be the resource bandwidth of the cloud computing environment, Buser represents the expected bandwidth of the user-specified job Jm, and Bi represents the expected bandwidth of the task Ti divided by the job Jn.

$$B_{user} = \sum_{i=1}^{T_{num}(J_n)} B_i \tag{6}$$

The bandwidth user satisfaction function is:

$$W_{BW}(J_n) = \left[\frac{\alpha}{T_{num}(J_n)}\right] \sum_{i=T_{total}(n-1)}^{T_{total}9N0} \ln\left(\frac{B_{wn}}{B_i}\right) \tag{7}$$

(3) Reliability

The reliability of task completion is represented by the task completion rate. Assuming that the resource failure rate in the cloud computing environment is P and the task completion rate expected by the user is Psucc, the user satisfaction function of the job completion rate is:

$$W_{succ}(J_n) = \alpha \ln\left[\frac{1-P}{P_{succ}}\right] \tag{8}$$

(4) Cost constraints

In a cloud computing environment, users pay for the services they need, and cost constraints are one of the important components of the user. Suppose resources are charged by unit, Pi is the number of resources, Ccpu, Cmen, Cstor and CBW represent the prices of CPU, memory, storage and bandwidth resources, respectively, then the total cost of the task can be expressed as:

$$C_i = P_1 C_{cpu} + P_2 C_{men+} + P_3 C_{stor} + P_4 C_{BW} \quad (8)$$

Set Cuser for users to expect the cost, the cost of user satisfaction function is:

$$W_{cost}(J_n) = \alpha \ln\left(\sum_{i=T_{total}(n-1)}^{T_{total}(n)} \frac{cost_i}{C_{user}}\right) \quad (9)$$

(5) Fitness function

Job scheduling in a cloud environment must take into account the four goals listed above. For the user, the operation time and cost of the job is as low as possible. The larger the broadband value assigned to the job by the system, the better the system stability of the job is run. Therefore, the job adaptation function of job scheduling is as follows:

$$f = -\varepsilon 1 W_{time} + \varepsilon 2 W_{BW} - \varepsilon 3 W_{cost} + \varepsilon 4 W_{succ} \ (0 \le \varepsilon_i \le 1) \quad (10)$$

Where is the weight coefficient. According to different user preferences for different indicators can set different weight vector to measure user satisfaction with cloud computing services.

## 4. Genetic operation

### 4.1 Individual choice

This paper preclude the use of roulette method to choose, set the size of the population S, for a fitness chromosome for individual chromosomes, the choice of probability:

$$Q_i = f_i / \sum_{k=1}^{S} f_k \quad (11)$$

### 4.2 Cross operation

For the selection of crossover probability Pc, this paper adopts an adaptive way to prevent the possibility of damage to individual structure with high fitness due to too large Pc, and the slow search process due to too small Pc and stagnation . The adaptive adjustment formula of crossover probability is as follows:

$$P_c = \begin{cases} a_1(f_{max} - f)/(f_{max} - f_{avg}), & f \ge f_{avg} \\ a_2, & f \le f_{avg} \end{cases} \quad (12)$$

Where, is the average fitness value of the population, is the maximum fitness value of the population, and f is the fitness value of the greater fitness of the crossover.

### 4.3 Mutation operation

For mutation probability Pn selection, this article also uses an adaptive way. The adaptive adjustment formula of mutation probability is as follows:

$$P_n = \begin{cases} a_3(f_{max} - f')/(f_{max} - f_{avg}), & f' \ge f_{avg} \\ a_4, & f' \le f_{avg} \end{cases} \quad (13)$$

Where, is the average fitness of the population, is the maximum fitness of the population, and f is the fitness of the individual to be mutated.

## 5. Simulation test

In this paper, CloudSim is used as a simulation platform, and the comparison experiment is carried out by using the improved genetic algorithm and the traditional genetic algorithm in this paper. The initial experimental parameters are set as follows: the maximum number of iterations is 200, the number of resources is 30, the total number of jobs is 20, and the number of tasks that each job is divided into is in the range of [5,20], crossover probability 0.85 and mutation probability 0.25. Setting

(Since the simulation platform can not get the failure rate of resources, the weight vector parameter is set to 0. The fitness function can reflect the user's satisfaction degree with cloud service. Therefore, when user satisfaction is the base, A value of 0 indicates that the result of the job scheduling is consistent with the expected value of the user's satisfaction. When the value of the fitness is greater than 0, the result of the job scheduling is higher than the user's expected value; otherwise, when the value of the fitness is less than 0, the job scheduling Of the results do not meet the expectations of users.For the user, the first two cases are acceptable, but the last case can not meet the needs of users of the service.Against the two algorithms run, take the average of the results , The results of user satisfaction obtained are shown in Figure 2.

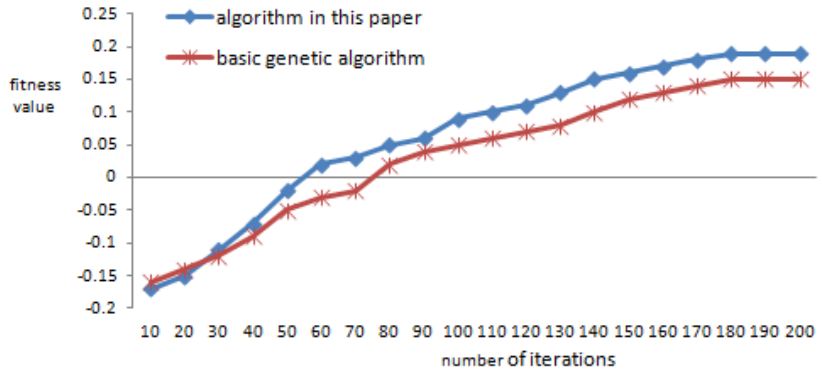


Figure 2 Average user satisfaction

## 6. Conclusion

In this paper, a task scheduling algorithm based on improved genetic algorithm is proposed. The fitness function model is established by using the user expectations of time, bandwidth, cost, and reliability constraints as criteria. The rule-constrained crossover and mutation operations improve the evolution The quality of individual individuals in the process. The simulation results show that this improved algorithm not only has higher convergence and the ability to search for the optimal solution, but also reflects the consistency between the scheduling results and user expectations, and provides an effective solution to the task scheduling problem in cloud environment.

## Acknowledgements

The work is supported in part by Department of Education of Guangdong Province under Grant 2015KTSCX162

## References

[1] Ren K,Wang C,Wang Q.Security challenges for the public cloud[J].IEEE Internet comput.
[2] Xiong Jinbo,Yao Zhiqiang. A Composite Document Model and Its Access Control Scheme in Cloud Computing[J]. Journal of Xi'an Jiaotong University, 48(2): 25–31(2014)
[3] Wang Yu-Ding,Yang jia-Hai.Survey on Access Control Technologies for Cloud Computing [J]. Journal of Software,26(5):1129-1150(2015)ing,2012,16(1):69-73.
[4] Feng Chao-sheng,Qin Zhi-guang.Key Techniques of Access Control for Cloud Computing[J]. Acta Electronica Sinica,02(43):312-319(2015).
[5] akabi H,Joshi J B D,Ahn G.Security and privacy chal-lenges in cloud computing environments [J]. Security &Privacy,IEEE, 8(6):24-31(2010).