Research on Balanced Scheduling Algorithm of Big Data in Network Under Cloud Computing

Lunqiang Ye(✉)

Southwest Minzu University, Chengdu, China
feiyun001b@163.com

Abstract. In order to find the optimal big data balanced scheduling scheme under cloud computing and reduce the completion time of the task, an improved ant colony algorithm based algorithm for large data equalization scheduling under cloud computing was proposed. Firstly, a balanced scheduling algorithm structure was established, then the equilibrium problem to be explored was described, finally, the ant colony algorithm was used to simulate the ant search food process to solve the objective function. And the local and global information deep update methods was introduced to improve, speed up the search speed, and finally the performance test experiments on CloudSim simulation platform was performed. The results show that compared with the discrete particle swarm optimization (DPSO), the algorithm not only greatly reduces the execution time of cloud computing tasks (2.5 s), but also solves the problem of unbalanced data load, and achieves the balanced scheduling of large network data under cloud computing.

Keywords: Cloud computing · Big data · Balanced scheduling · Ant colony algorithm

1 Introduction

The development of a series of new network applications, such as online social networking, search websites and business representatives, a large amount of Internet data integration and data aggregation, has caused a burst of data processing applications and demand, and the amount of business that the Internet has to analyze or process is becoming increasingly heavy [1]. For example, the central processing system of foreign social software Face hook receives an average of more than 3 billion messages and nearly 1000T network data per day; And China's Alibaba's Taobao. The emergence and application of a large amount of data has accelerated the arrival of the "big data era." For mass data services, how to effectively balance these data to provide convenient and fast network search services for network users has become an important issue facing the Internet. In this case, a new type of computing technology was born, which is the cloud computing technology [2]. The cloud computing platform such as Hadoop is the practical application result of cloud computing technology. In a cloud computing platform, the resource and job scheduling strategy is its core, which plays a vital role in

G. Gui and L. Yun (Eds.): ADHIP 2019, LNICST 301, pp. 197–206, 2019.
https://doi.org/10.1007/978-3-030-36402-1_21

the allocation of computing resources and the execution of the entire system. Therefore, the research of scheduling algorithms in cloud computing environment is of great significance. This study proposes an ant colony algorithm based algorithm for large data equalization scheduling in cloud computing. Ant colony algorithm is an intelligent algorithm, which is applied to cloud computing, which can shorten task execution time, improve load balance, better solve big data management and scheduling in cloud computing, and load balancing of virtual machines.

2 Balanced Scheduling Algorithm Based on Ant Colony Algorithm

Load balancing is applied in many professional fields. The load balancing of computers is accompanied by the demand for computing power and the emergence of clustering technology. It is the performance index of cluster systems and develops with the development of cluster technology [3]. The scheduling mechanism of the load balancing algorithm can solve the problem of unbalanced data load in the cluster environment, promote the reasonable distribution of data resources and improve the quality of service.

The early balanced scheduling algorithm only distributes the tasks evenly to the data nodes, so that the data is not idle and improves system efficiency. Excessive node load can lead to a large number of tasks queuing, directly affecting the task completion time and system throughput. The first phase is the application data phase when the task comes. In this phase, the task needs to be assigned to the data node with lower load. The second phase is to balance the load for the node, and transfer some tasks of the node to the node with low load when the data node load is too high during task execution.

Ant colony algorithm is similar to genetic algorithm, and it is a natural heuristic algorithm. The ant population is divided into foragers, followers and harvesters. In the process of foraging, workers are responsible for finding food sources. The rate of return from food sources depends on the quantity or quality of food and the distance from the nest. By observing the wagging dance of the follower, the follower decides which follower to harvest according to the rate of return, so the rate of return on the food source is proportional to the follower's probability of choosing the food source. When his followers arrived at the source of food, he became a harvester and began to harvest [4]. The flow chart of ant colony algorithm is shown in Fig. 1. The breeder is responsible for collecting the status information of the system resources and displaying it in the "ant nest". The followers of "ant nest" can understand the rate of return of different resources and take corresponding actions according to the rate of return of resources. When followers acquire resources, they become reapers and begin harvesting, and evaluate the solutions they have obtained.

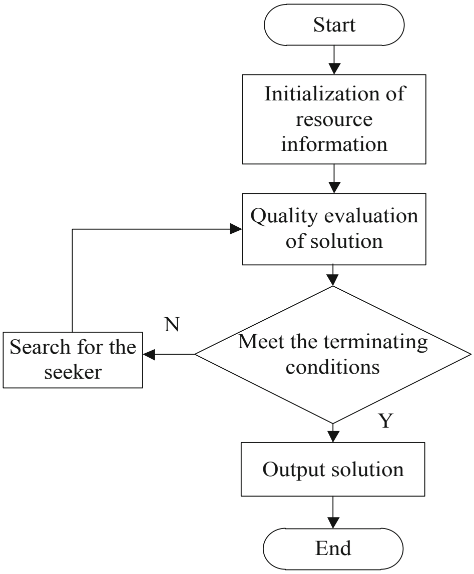

Fig. 1. Ant colony algorithm

Using ant colony algorithm to balance data scheduling is based on information collection of system load data and information broadcasting in honeycomb. This is a typical centralized control method. The follower can obtain detailed system load data information in the cell, and track the operation according to the profit margin of different nodes. Because of the high output, low-load nodes are easy to be chosen by followers [5].

2.1 Balanced Scheduling Algorithm Structure

In cloud computing, the system is usually divided into three layers as shown in Fig. 2, which are the application layer, the virtual layer, and the infrastructure layer. There are many hierarchical models of cloud computing, but all hierarchical models are based on these three layers, and the balanced scheduling algorithm model is also based on these three layers. The application layer is an interface for cloud computing to display services to users and users to apply for services to cloud computing. It is mainly responsible for submitting tasks to the virtual layer and collecting applications from users [6]. The virtual layer is all the resources that the application layer user can see. It is responsible for assigning the virtual machine resources to the specific application of the user, and handing over the tasks satisfying the operating conditions to the infrastructure layer for execution. The infrastructure layer is the actual server cluster.

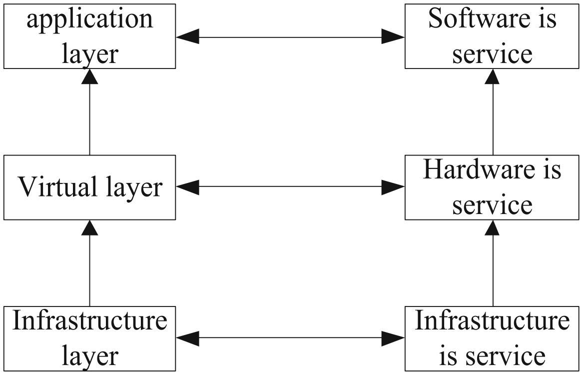

Fig. 2. Three-layer structure of the equilibrium scheduling algorithm under cloud computing

2.2 Equilibrium Problem Description

The current cloud computing system adopts the working mode of Map/Reduce, which is suitable for large-scale data processing, and has high work efficiency. The execution process is as shown in Fig. 3.

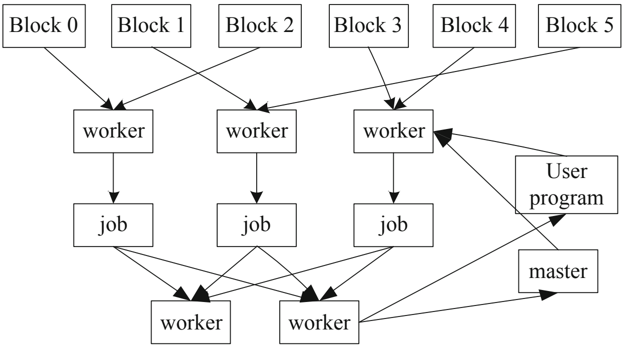

Fig. 3. Map/Reduce execution process

Map/Reduce mainly has two key stages, Map and Reduce, as shown below.

(1) The Map stage subdivides the task into multiple sub-tasks. The sub-tasks are generally in the range of 16 ~ 64 MB. Then the sub-tasks are assigned to the cloud computing data (worker) execution, and the results are fed back to the main node after execution [7].

(2) The Reduce phase aggregates the results of each worker in the Map phase and outputs the final result to R output data.

Big data equalization scheduling is a complex problem. Each task is divided into M, N subtasks of Map, Reduce stage, and the number of M and N is much larger than the number of workers. Each worker has to complete many subtasks, so it is reasonable to schedule tasks to ensure data load balancing becomes a key problem for task scheduling algorithms in cloud computing environments.

Equilibrium scheduling of big data in the network under cloud computing can be described by 3-tuple: {T, R, ETC}, T is a collection of tasks: t1, t2, …, tm, m is the number of tasks; R is a collection of resources, r1, r2, …, rn, n is the number of resources. The objective function of the equilibrium scheduling of big data of the network under cloud computing is to make the number of data on the node relatively balanced and ensure the task completion time is as short as possible under the condition that the task completion time meets the user requirements [8].

2.3 Standard Ant Colony Algorithm

Set the number of ants in the ant colony m. Starting from the starting node, the probability formula for the ant to select the next node according to the transition probability is

$$p_{ij}^{k}(t) = \frac{\vartheta_{ij}^{a} \cdot \theta_{j}^{b}}{\sum\limits_{j=1}^{i} \vartheta_{ij}^{a} \cdot \theta_{j}^{b}} \tag{1}$$

Where, θ_j is a heuristic information function associated with the edge (i, j); a and b are pheromones and heuristics; and ϑ_{ij} is a pheromone on the edge (i, j).

In order to prevent unrestricted accumulation of pheromones, a pheromone volatilization mechanism is introduced. When the ants complete a search from the start point to the end point, the pheromone on each side is updated, specifically

$$\vartheta_{ij}(t+1) = c\vartheta_{ij}(t) + \Delta\vartheta_{ij}(t) \tag{2}$$

$$\Delta\vartheta_{ij}(t) = \sum_{k=1}^{m} \Delta\vartheta_{ij}^{k}(t) \tag{3}$$

Where, $\Delta\vartheta_{ij}(t)$ and $\Delta\vartheta_{ij}^{k}(t)$ are the increments of pheromone on the side (i, j) of ant k and all ants; c is the pheromone volatilization coefficient.

$\Delta\vartheta_{ij}^{k}(t)$ uses the ant week system calculation method, specifically

$$\Delta\vartheta_{ij}^{k}(t) = D/\mathrm{E}_k \tag{4}$$

Where D is a constant; E_k is the path length of ant k.

The process of balancing the network big data using the standard ant colony algorithm is shown in Fig. 4.

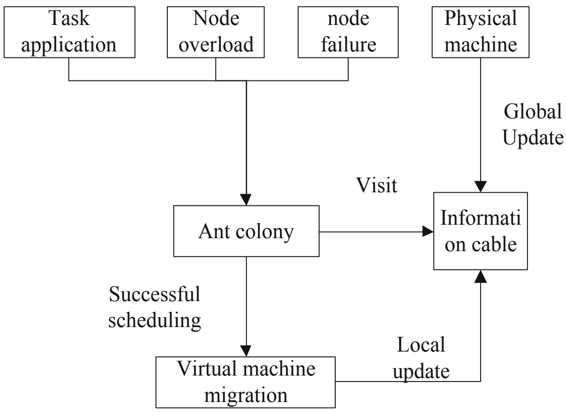

Fig. 4. Ant colony algorithm equilibrium scheduling structure

2.4 Improved Ant Colony Algorithm

In the ant colony algorithm, the value of the pheromone volatilization coefficient c has a great influence on the performance of the algorithm. In order to improve the adaptability of the algorithm, the pheromone volatilization coefficient of the basic ant colony algorithm is dynamically adjusted:

$$c(t) = \begin{cases} c_{\min} & \vartheta_{ij} \le \vartheta_{\min} \\ \vartheta(t)\vartheta_{\min} < \vartheta_{ij}(\mathrm{t}) < \vartheta_{\max} \\ c_{\max} & \vartheta_{ij}(\mathrm{t}) \ge \vartheta_{\max} \end{cases} \quad (5)$$

Where $c_{\min}$ and $c_{\max}$ are the maximum and minimum values of the volatilization coefficient; $\vartheta_{\min}$ and $\vartheta_{\max}$ are the minimum or maximum limit values for preventing pheromone.

Since the equilibrium scheduling of big data of the network under cloud computing is complex, in order to establish the mapping of tasks and data, the improved ant colony algorithm pheromone function will reward the global optimal path solved. That is, the pheromone concentration on the path it contains is strengthened, and at the same time, the solved local iterative worst path is punished. That is, the pheromone concentration on the path included is weakened, so that the difference between the optimal path and the worst path information concentration is further expanded, so as to achieve the purpose of searching the ant colony as close as possible to the optimal path [9]. The global pheromone concentration update method of the improved ant colony algorithm is

$$\vartheta_{ij}new = \begin{cases} (1-c)\cdot\vartheta_{ij}old + \sigma(L1 - L2) \\ \text{If the road segment } (i,j) \text{ belongs to the global optimal path} \\ \text{If the road segment } (i,j) \text{ belongs to the global worst path} \\ 0, \text{ otherwise} \end{cases} \quad (6)$$

Where $L1$ is the global optimal path, $L2$ is the iterative worst path, and σ is a parameter introduced in the algorithm.

After each ant completes the search for the path, it must perform a partial update on the path it has taken. The local update method refers to the basic ant colony algorithm model. The formula for partial update is:

$$\vartheta_{ij}new = (1 - \mathrm{c}) \cdot \vartheta_{ij}old + \Delta\vartheta_{ij}, c \in (0, 1) \tag{7}$$

In the formula, c is the information evaporation coefficient.

$$\Delta\vartheta_{ij} = \begin{cases} D/E_k, \\ 0, \text{ otherwise} \end{cases} \text{If the ant passes the path } (\mathrm{i}, \mathrm{j}) \tag{8}$$

In the formula, $E_{k,}$ is the path distance obtained by the k-th ant, and D is a parameter introduced in the algorithm. The process of equalizing the scheduling of network big data by using the improved ant colony algorithm is shown in Fig. 5.

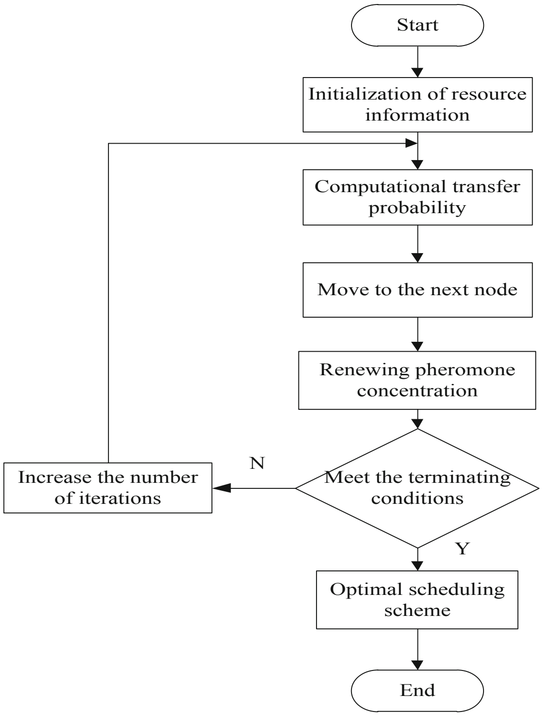

Fig. 5. Improved ant colony algorithm for the equilibrium scheduling solution process

3 Comparative Experiment

The scheduling algorithm is run on the cloud simulation software C1oudSim. The experiment is in the case where multiple users of different SLAs compete for the same resources. Comparing the nonSLA-MPS and SLA-MPS algorithms from the average response time of the task, the SLA default rate of the cloud service provider and the total revenue of the cloud service provider, the effects of the two algorithms on the interests of the cloud service providers are tested. The maximum number of virtual machines that are enabled in each data center is 30, and the total number of tasks specified in the experiment is 400, 600, 800, and 1000.

The ant colony algorithm based on cloud computing network completes the comparison of balanced scheduling time between large data balanced scheduling algorithm and discrete particle swarm optimization balanced scheduling algorithm, as shown in Fig. 6.

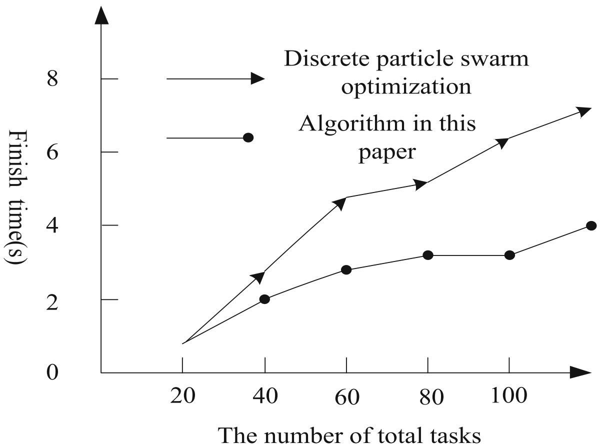

Fig. 6. Comparison of time required for equalization scheduling

As can be seen from Fig. 6, the execution time of the proposed algorithm is not much different from that of the discrete particle swarm optimization (DPSO) algorithm when the task load is small. However, with the increase of tasks, the execution time of the proposed algorithm is significantly shorter than that of the discrete particle swarm optimization (DPSO). When the number of tasks reaches 100, the execution time of the two algorithms is nearly 2.5 s apart. It can be seen that in a certain range, with the increase of the number of tasks, the time cost of allocating resources and executing resources of the two algorithms is increasing. Therefore, the proposed algorithm can balance the scheduling of large network data more effectively in cloud computing.

Ant colony algorithm based on cloud computing network completes the comparison of balanced scheduling errors between large data balanced scheduling algorithm and chaotic particle swarm optimization scheduling algorithm, as shown in Fig. 7.

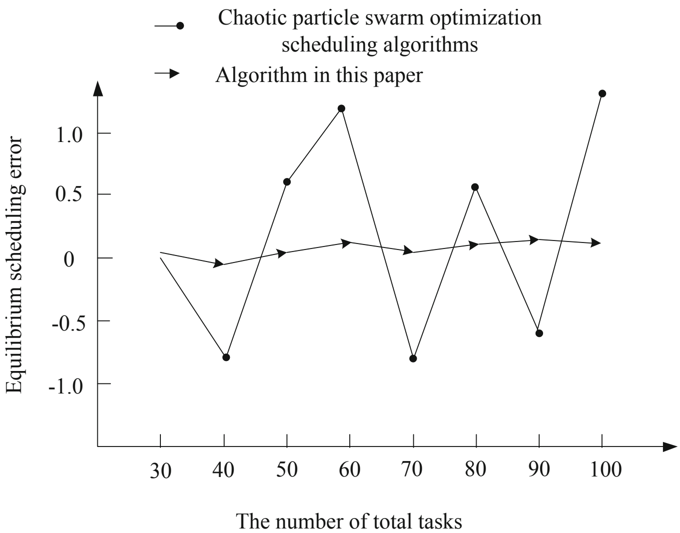

Fig. 7. Comparison of equilibrium scheduling errors

Analysis of Fig. 7 shows that with the increasing number of tasks, the equilibrium scheduling error of the chaotic particle swarm equilibrium scheduling algorithm is larger, and the error fluctuations constantly. The maximum scheduling error of this method can reach 2.0, which shows that the method has a high error in large data scheduling. The error of this algorithm is basically 0, which shows that this algorithm can schedule with zero error, and fully proves the overall effectiveness of this algorithm.

4 Conclusions

In summary, in the cloud computing environment, the balanced scheduling of data resources is the key to ensuring the quality of cloud computing services. In order to perform reasonable task assignment in the cloud computing environment, shorten the time to complete the total task, Each node achieves load balancing as much as possible, and applies the improved ant colony algorithm to the cloud computing data balanced scheduling. The simulation test shows that with the increase of the task amount, compared with the traditional data balance scheduling algorithm, the data scheduling time is significantly reduced, and the load balancing degree of the node is significantly improved.

Acknowledgements. Supported by Educational and Teaching Research and Reform Funds for Southwest Minzu University, “A Study on the Construction and Application of Teaching Resources in the SPOC Blended Teaching Mode” Project No.: 2018YB25.

References

1. Ying, T.J.: Multi source heterogeneous data scheduling algorithm under cloud computing. Sci. Technol. Eng. **21**(34), 268–272 (2017)
2. Jinfang, Z., Qingxin, W., Jiaman, D., et al.: A big data dynamic migration strategy in cloud computing environment. Comput. Eng. **42**(5), 13–17 (2016)
3. Luo Nan Super Cloud: The optimization of resource scheduling algorithm for balancing load under cloud computing. Sci. Technol. Eng. **16**(34), 86–91 (2017)
4. Xin, L.: Cloud computing communication network information download balanced scheduling optimization research. Comput. Simulation **33**(10), 162–165 (2016)
5. Xiaofeng, L.: Research and improvement of cloud resource scheduling method in cloud computing optical fiber network. Laser J. **37**(5), 99–103 (2016)
6. Junying, W., Xinrui, C.X.: Load balancing and efficient scheduling method for diversity resources in cloud computing environment. Bull. Sci. Technol. **33**(12), 167–170 (2017)
7. Kai, Z.: Research on cloud computing platform service resource scheduling. Comput. Simulation **34**(9), 424–427 (2017)
8. Xiaonian, W., Xin, Z., Mengchuan, et al.: Two-phase task scheduling algorithm for multi-objective in cloud computing. Comput. Eng. Des. **38**(6), 1551–1555 (2017)
9. Han, H., Peng, W., Kun, C., et al.: Task scheduling algorithm for cloud computing based on multi-scale quantum harmonic oscillator algorithm. J. Comput. Appl. **37**(7), 1888–1892 (2017)