ICIC Express Letters
Volume **13**, Number **6**, June **2019**

ICIC International ©2019 ISSN 1881-803X
pp. 499–504

IMPROVED GREEDY STRATEGY FOR CLOUD COMPUTING RESOURCES SCHEDULING

Hongwei Dai, Yu Yang, Jie Yin, Hongan Jiang and Cunhua Li

School of Computer Engineering
Huaihai Institute of Technology
No. 57, Cangwu Road, Lianyungang 222005, P. R. China
hongweidai@hotmail.com

Received November 2018; accepted February 2019

Abstract. *Cloud computing, a high performance computing environment with a large scale, heterogeneous collection of autonomous systems and flexible computational architecture, has drawn enormous attention due to its wide applications in both scientific and business areas. One of the most important problems of cloud computing resource management is resource scheduling. Optimized scheduling of tasks on the cloud virtual machines is an NP-hard problem and many algorithms have been presented to solve it. We proposed a novel algorithm extending the traditional greedy strategy that optimizes the scheduling of workflows on the cloud. The improved algorithm is implemented and embedded within the workflows simulation toolkit and tested in different simulated cloud environments with different parameters. The simulation experimental results show that the proposed algorithm can effectively enhance the quality of service of the cloud environment.*

Keywords: Cloud computing, Improved greedy strategy, Resources scheduling

1. **Introduction.** With the rapid development of modern network and the increasing demand and service of the public, the service based on cloud computing has developed rapidly. Cloud computing provides on demand services to the client. There are three different service levels including SaaS (Software as a Service) where application software and database access provided to the end user pay per use basis, IaaS (Infrastructure as a Service) where virtual machine is provided to the client using virtualization of physical machine which includes processing power, storage and other resources, PaaS (Platform as a Service) where cloud provider provides a computing platform which includes OS, programming language execution platform and web server [1,2].

Due to the introduction of virtualization technology, resource scheduling in cloud computing environment has different characteristics from traditional grid computing. Resource scheduling is an important issue which greatly influences the performance of cloud computing environment. Scheduling is the process of mapping different tasks to available resources (such as virtual machines, VM) on the basis of tasks' characteristics and requirements. Efficient scheduling is essential requirement for the cloud, as it can optimize the use of the cloud resources to provide highly efficient services with high quality [12].

The problem of cloud computing resource allocation is an NP difficult problem. Its scheduling goal is to improve the utilization of resources as much as possible and ensure the balance of the system load on the premise of completing the task. Generally, cloud computing resource allocation problem [4] is studied from the aspects of QoS, architecture optimization and performance. Liu and Liu proposed a cloud computing resource scheduling model based on QoS and maximization utility [5]. The simulation results show

DOI: 10.24507/icicel.13.06.499

that the algorithm has less average execution time and execution cost, and has some advantages. In the aspect of architecture optimization, the master and subordinate model (master-slave) is more used, that is, the task is divided effectively by the main node master. The sub tasks after the division are assigned to each slave node, and each sub task is completed by the assigned slave node, such as MapReduce [6] technology. However, in more cases, the performance of the cloud computing resource allocation algorithm with the shortest time and the lowest cost as the task is the focus and hot spot of [7-9].

Based on the analysis of the shortcomings of traditional greedy strategy, this paper proposes an improved greedy optimization strategy based on task execution time span. The simulation results show that the improved greedy algorithm can detect the resource allocation results better than the sequential allocation strategy and the traditional greedy strategy.

The rest of this paper is organized as follows. Section 2 gives a brief overview of the resources scheduling and optimization goals. Section 3 introduces two traditional methods, sequential allocation and greedy allocation strategy. Improved greedy allocation method and simulation process are shown in Section 4. Simulation results are presented in Section 5. Section 6 draws conclusions.

2. Concept of Cloud Computing Resources Scheduling and Optimization Goals.

2.1. Cloud computing resources scheduling.

The resource scheduling of cloud computing is an important research topic in cloud computing. It refers to the process of distribution of resources to different users in a specific cloud environment, in a specific time, according to the rules of the use of certain resources [10].

The allocation of resources is based on the objectives that need to be optimized, and the current strategies are almost all carried out through the virtual machine level, and the goals that need to be optimized include the completion time of the task, the cost and the utilization of resources and so on. Different applications of cloud computing require different resources. If we want to put forward a suitable resource scheduling strategy, we need to consider many factors. In the previous study, it can be found that the most outstanding questions of resource scheduling in cloud computing are low resource utilization, unequal allocation of tasks and processors, and more time and energy consumption, so the first need to solve this problem is to solve the problem. It is precisely these problems that will bring maximum benefits to reduce execution time, energy consumption, increase efficiency and optimize the allocation of resources. Therefore, resource scheduling management should not only meet the needs of users, but also improve the utilization of resources at the same time, so as to maximize the benefits of cloud computing service providers and users.

2.2. Scheduling optimization goals.

Based on the former introduction, it can be easily found that the goal of cloud computing resource scheduling is to achieve the user's task requirements through a certain strategy to achieve the goal of increasing the utilization of resources and maximizing the benefit of the provider [5]. Its main indicators are: service quality, optimal span, load balance and economic principle. The optimization objective of this paper is execution time span.

The optimal span refers to the length of the user's task scheduling. Its value can be represented by the time from the beginning of the first task to the total time of the end of the last task. The shorter the time, the smaller the index, which represents the better scheduling strategy of the cloud computing resource used. What users expect is that the cloud computing system can process the tasks submitted by users at the fastest speed.

However, the scheduling problem is an NP-complete problem which has been extensively studied for other paradigms, such as grid and cluster computing. It is noteworthy that if there are n tasks in a workflow and m available virtual machines, then there exist m^n different ways in which the tasks can be mapped to the VM pool. For a large value of n and m, finding an optimal solution by brute force approach is computationally very expensive. Therefore, many approaches have been used to solve this problem.

3. Sequential Allocation Strategy and Greedy Allocation Strategy.

3.1. **Sequential allocation strategy.** Sequential allocation strategy is the simplest allocation method in cloud computing resource allocation strategy. The strategy assigns [11] according to the order of tasks. Assume that there are n tasks, m virtual machines, then assign them to the virtual machine one by one according to the task order of $1 \sim n$, and the virtual machine is also dealing with these tasks in the order of $1 \sim m$. If $n > m$, then all the virtual machines have been assigned tasks, then they will be allocated from the beginning or in the order of $1 \sim n$. Because of the difference in the processing efficiency of each virtual machine and the size of each task, this method may lead to a low efficiency virtual machine to handle large tasks, while the efficient virtual machine is assigned small tasks, resulting in the waste of time and the increase of cost.

3.2. **Greedy allocation strategy.** Although the sequential allocation strategy is simple and convenient, there are also many problems, such as the waste of time, and the large cost. It is mainly due to the reason that the task is allocated to the inappropriate virtual machine, and it does not take account of the difference between each task and each virtual machine, and the length and size of the task are different. The performance of the simulator is different, so this study adopts a greedy strategy to optimize it in order to achieve the purpose of executing all the tasks in the shortest time.

Through the research on resource scheduling in CloudSim cloud platform emulator, it is found that the main factors affecting the execution time of the task are the task command length (MI) and the virtual machine execution speed ($MIPS$), and it can be found through experiments that if a task runs fastest on a virtual machine, then this task is in each of the tasks. A virtual machine is the shortest in time; if a virtual machine runs the most efficiently, then the virtual machine performs the fastest tasks.

Greedy strategy is to allocate the longest and most complex task to the most efficient virtual machine, so as to quickly solve complex and difficult problems and shorten the total execution time of all tasks.

Combined with the above conclusion, the task is represented by i, and j represents the virtual machine. Then the matrix $time[i][j]$ is the execution time of task i in virtual machine j, and the steps are as follows:

(1) sorting: descending order MI, ascending order $MIPS$;

(2) initialization matrix $time[i][j]$;

(3) carry out the greedy strategy.

4. Improved Greedy Allocation Strategy and Simulation Process.

4.1. **Improved greedy strategy.** In greedy strategy, if the current virtual machine has not allocated tasks, it is better to compare whether the current task is allocated to the virtual machine. However, this greedy strategy is often partial or temporary optimal, and it will not be able to cope with complex tasks with large changes in demand. Therefore, this paper puts forward the strategy of improving greedy algorithm. Choose the greedy strategy by a certain probability.

The algorithm is implemented as follows.

Step one: set up a set of virtual machines and tasks, and define the matrix $time[i][j]$.

Step two: initialize the matrix $time[i][j]$ and set the following four parameters:

(1) the total execution time of the task on a virtual machine is $vmLoad$;

(2) the number of tasks running on a virtual machine: $vmTasks$;

(3) the optimal value of current task allocation: $minLoad$;

(4) the virtual machine that corresponds to the optimal assignment of current tasks: idx.

Step three: assign the task of $i = 0$ to the maximum number of columns, that is, m's virtual machine ($m = vmNum - 1$).

Step four: when the size of i has not yet reached the number of tasks, execute the loop, assign tasks to $vmNum - 2$, $vmNum - 3$, ... until the first virtual machine.

In this step, the following operation is performed with a certain probability p: if the current virtual machine has not assigned the task and the virtual machine is optimal, then the loop is allocated and withdrawn; otherwise, a simple load balancing is realized.

Step five: after the comparison is completed, you can sort the tasks in descending order and ascending the virtual machine in ascending order.

4.2. **Simulation process of cloud resource allocation based on CloudSim.** CloudSim is a cloud computing simulator. CloudSim developed by researchers at the Melbourne University in Australia, is developed on the basis of the GridSim model, providing the features of cloud computing, supporting the resource management and scheduling of cloud computing [12].

The simulation process is shown in Figure 1.

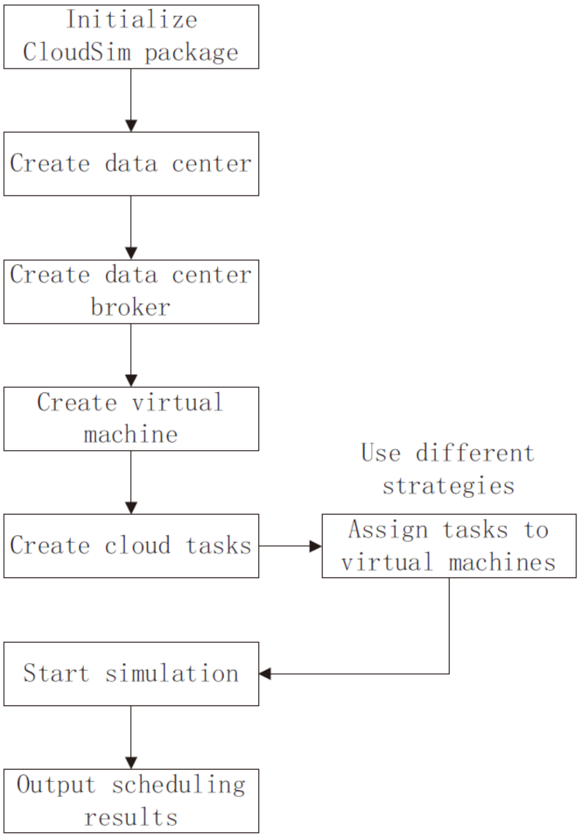
Initialize CloudSim package
Create data center
Create data center broker
Create virtual machine
Use different strategies
Create cloud tasks
Assign tasks to virtual machines
Start simulation
Output scheduling results

FIGURE 1. Simulation process of CloudSim

5. **Experiment and Analysis.** In order to accurately evaluate the performance of the algorithm, different resource allocation strategies are analyzed and compared based on the CloudSim simulation platform.

10 tasks are $T[10] = 19365, 49809, 30218, 44157, 16754, 18336, 20045, 31493, 30727, 31017$, and the processing speed of the 5 virtual machines is $VM[5] = 278, 289, 132, 209, 286$. Table 1 shows the results of cloud resource allocation under different policies. In this table,

Table 1. Cloud resource allocation results with different policies

	T_1		T_2		T_3		T_4		T_5		T_6		T_7		T_8		T_9		T_{10}		T_{sum}
Sequential allocation strategy	1	0.10	2	0.10	3	0.10	4	0.10	5	0.10	1	69.76	2	172.45	3	229.02	4	221.38	5	58.68	467.60
		69.76		172.45		229.02		211.38		58.68		135.71		241.80		467.60		358.39		167.13	
Traditional greedy allocation strategy	2	172.45	2	0.10	3	0.10	5	0.10	5	224.58	4	148.50	5	154.49	1	0.10	1	113.38	4	0.10	283.16
		239.45		172.45		229.02		154.49		283.16		236.23		224.58		113.38		223.91		148.50	
Improved greedy allocation strategy	2	172.45	2	0.10	1	113.38	5	0.10	1	222.08	4	148.50	5	154.49	1	0.10	3	0.10	4	0.10	282.35
		239.45		172.45		222.08		154.49		282.35		236.23		224.58		113.38		232.87		148.50	
	5	154.49	2	0.10	3	0.10	5	0.10	5	222.20	4	148.50	2	172.45	1	0.10	1	113.38	4	0.10	280.78
		222.20		172.45		229.02		154.49		280.78		236.23		241.81		113.38		223.91		148.50	
	4	148.50	2	0.10	3	0.10	5	0.10	5	218.60	5	154.49	2	172.45	1	0.10	1	113.38	4	0.10	277.18
		241.16		172.45		229.02		154.49		277.18		218.60		241.81		113.38		223.91		148.50	
	5	154.49	2	0.10	2	172.45	5	0.10	4	148.50	1	185.49	1	113.38	1	0.10	3	0.10	4	0.10	277.01
		222.20		172.45		277.01		154.49		228.67		251.44		185.49		113.38		232.88		148.50	
	2	172.45	2	0.10	5	154.49	5	0.10	4	148.50	1	185.49	1	113.38	1	0.10	3	0.10	4	0.10	260.15
		239.45		172.45		260.15		154.49		228.66		251.44		185.49		113.38		232.88		148.50	

T_i means different tasks, and T_{sum} is the longest time span for all tasks under each allocation strategy. Each unit in the table contains three parts, the left integer is the virtual machine index assigned to the current task, the right top is the start time of the task, and the right down is the end time of the task. Based on the simulation results, we can confirm that the improved greedy strategy can give a more reasonable allocation scheme, so that the completion time of tasks can be effectively shortened.

6. **Conclusions.** Because of the introduction of virtualization technology, cloud computing resources scheduling is different from traditional grid computing. In order to solve the problem of low resource utilization and uneven distribution of resources in cloud computing, based on the CloudSim cloud computing simulator, this paper studies the optimization scheduling strategy based on the task execution time span, and proposes an improved greedy optimization method. The improved greedy optimization method can overcome the shortcomings of traditional algorithm and provide better allocation scheme for cloud computing resource allocation. Compared with the simulation results, the improved greedy algorithm proposed in this paper can generate the optimization results better than the sequential allocation strategy and the traditional greedy strategy. In the future work, we would like to use heuristic and meta-heuristic based algorithms to solve resources scheduling problem in cloud computing.

Acknowledgements. This work was supported by the Prospective Joint Research of University-Industry Cooperation of Jiangsu under Grant No. BY2016056-02, the Haiyan Projects of Lianyungang, the 521 Project of Lianyungang under Grant No. LYG52105-2018030, the Science and Technology Plan Projects of HHIT under Grant No. Z2015005.

REFERENCES

[1] A. V. Lakra and D. K. Yadav, Multi-objective tasks scheduling algorithm for cloud computing throughput optimization, *Procedia Computer Science*, vol.48, pp.107-113, 2015.

[2] J. Luo, J. Jin, A. Song and F. Dong, Cloud computing: Archiecture and key technologies, *Journal on Communications*, vol.32, no.7, pp.3-21, 2011.

[3] S. Elsherbiny, E. Eldaydamony, M. Alrahmawy and A. E. Reyad, An extended Intelligent Water Drops algorithm for workflow scheduling in cloud computing environment, *Egyptian Informatics Journal*, vol.19, pp.33-55, 2018.

[4] X. Wang, *Resource Allocation Algorithm Research in Cloud Computing Environment*, Master Thesis, Chongqing University, 2015.

[5] B. Liu and Q. Liu, Resource scheduling model of cloud computing based on QoS and utility, *Computer Measurement and Control*, vol.22, no.3, pp.826-828, 2014.

[6] G. S. Sadasivam and D. Selvaraj, A novel parallel hybrid PSO-GA using MapReduce to schedule jobs in Hadoop data grids, *The 2nd World Congress on Nature and Biologically Inspired Computing (NaBIC)*, Fukuoka, pp.377-382, 2010.

[7] Y. Liu, J. Cheng and J. Lin, Research on cloud computing resource scheduling based on the Gauss mutation artificial firefly algorithm, *Application Research of Computers*, vol.32, no.3, pp.834-837, 2015.

[8] Z. Sun, A multi task scheduling algorithm based on genetic algorithm in cloud computing environment, *Computer Science and Applications*, vol.6, no.6, pp.317-322, 2016.

[9] C. Li, B. Dai, Z. Kuang, X. Wu and R. Sun, Research on multidimensional constrained task scheduling based on improved genetic algorithm in cloud computing environment, *Journal of Chinese Computer Systems*, vol.38, no.9, pp.1945-1949, 2017.

[10] X. Wang, Research and applications of CloudSim cloud computing simulation tools, *Microcomputer Applications*, vol.29, no.8, pp.59-61, 2013.

[11] P. Liu, *Cloud Computing*, 2nd Edition, Publishing House of Electronics Industry, 2011.

[12] R. N. Calheiros, R. Ranjan, A. Beloglazov et al., 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.