TELKOMNIKA, Vol.11, No.11, November 2013, pp. 6664~6672
e-ISSN: 2087-278X

Research and Simulation of Task Scheduling Algorithm in Cloud Computing

Hong Sun*[1,2], Shi-ping Chen[1], Chen Jin[1,2], Kai Guo[1,2]
[1]University of Shanghai for Science and Technology
[2]Shanghai Key Lab of Modern Optical System, No. 516, Jungong Road, Yangpu District, Shanghai, China 200093
Corresponding author, e-mail：sunhong_sh@sohu.com

Abstract

This paper focuses on the task scheduling algorithms based on comprehensive QoS and constraint of expectation. Under the environment of dynamic cloud computing, efficiency improving of task scheduling and load balancing are eternal problems. For users, however, it's more important to meet their requirements of QoS. This paper relates to a benefit-fairness algorithm based on new Berger's model under the environment of dynamic cloud computing. According to the different type of task scheduling, we describe the priority of fairness, efficiency and the Balance between benefit and fairness respectively. We recompile the CloudSim and simulate the three task scheduling algorithms above on the basis of extended CloudSim respectively. The experimental results indicate that this algorithm dose not only meet the principle of giving priority to benefit with due to consideration to fairness, but also meet users' needs of synthesized QoS.

Keywords: *cloud computing, new Berger’s model, task scheduling, quality of service (QoS), general expect constraint*

1. Introduction

As an important part of cloud computing, task scheduling is a mapping mechanism from users‘ tasks to the appropriate selection of resources and its execution. Compared with grid computing, cloud computing has many unique features including virtualization and flexibility. By using the technology of virtualization, all physical resources are virtualized and transparent for users. All users have their own virtual device, these devises do not interact with each other and they are created based on users' requirements. In addition, one or more virtual machines can run on a single host computer so that the utilization rate of resources has been effectively improved. The independence of users' application ensures the system's security of information and enhances the availability of service [1]. Supplying resources under the cloud computing environment is flexible, we increase or reduce the supplying of resources depends on users' demand. Because of these new features, grid computing, the original task scheduling mechanism, can’t work effectively in cloud computing environments [2].

1.1. The Features of Task Scheduling in the Cloud Computing Environment

In the cloud computing environment, task scheduling and resource assignment have been unified managed by providers through virtualized technology. They have been used to hide and complete users’ tasks transparently. Task scheduling becomes more complex because of the transparent and dynamic flexibility of cloud computing system, and the different needs for recourses of different applications. Task scheduling strategies only focus on equity or efficiency will increase the cost of time, space, throughput and improve the quality of service of the entire cloud computing at the same time. The characteristics of the task scheduling in the cloud computing environment are as follows:

1. Task scheduling caters to a unified resources platform.

As cloud computing using the virtualized technology, we abstracting the underlying physical resources (all types of hosts, workstations or even PC, etc.) as a unified resource pool, and shielding heterogeneous, supply the upper use. It mainly distributes in a large number of distributed computers, and supply the use of resources in the form of a data center.

Received May 5, 2013; Revised June 23, 2013; Accepted July 25, 2013

2. Task scheduling is global centralized.

As cloud computing is a computing model which supply the centralized resource by the mirror service to multiple distributed applications, and this mirroring deployment can make heterogeneous procedures' executing of interoperate become easier, which used to be difficult to deal with. Therefore, virtualized technology and mirroring services make the task scheduling of cloud computing achieve a global centralized scheduling.

3. Each node in the cloud is independent.

In cloud computing, the internal scheduling of every cloud node is autonomous, and the schedulers in the cloud will not interfere with the scheduling policy of these nodes.

4. The scalability of task scheduling

The scale of resources supply from cloud provider may be limited in early stages. With the addition of a variety of computing resources, the size of the abstract virtual resources may become large, and the application demand continue increasing. In the cloud, task scheduling must meet the scalability features, so that the throughput of the task scheduling in the cloud may not be too low.

5. Task scheduling can be dynamically self-adaptive

Expanding and shrinking applications in the cloud may be necessary depend on the requirement. The virtual computing resources in cloud system may also expand or shrink at the same time. The resources are constantly changing, some resources may fails, new resources may join in the clouds or restart.

6. The set of task scheduling

Task scheduling is divided into two parts: one is used as a unified resource pool scheduling, and primarily responsible for the scheduling of applications and cloud API; the other is for the unified port resource scheduling in the cloud, for example, MapReduce task scheduling. However, each scheduling consists of two two-way process: scheduler leases resource from cloud, scheduler callbacks the requested resources after use. The former process is scheduling strategy and the latter one is callback strategy [3, 4]. The combination of the scheduling and callback resource strategy is the set of task scheduling [5].

1.2. The Target of Task Scheduling in Cloud Environment

The task scheduling goals of Cloud computing is provide optimal tasks scheduling for users, and provide the entire cloud system throughput and QoS at the same time. Specific goals are load balance, quality of service (QoS), economic principle, the optimal operation time and system throughput [4, 5].

1. Load balance

Load balancing and task scheduling has close contacts with each other in the cloud environment, task scheduling mechanism responsible for the optimal matching of tasks and resources [6]. Because of the pertinency of task scheduling algorithm, load balancing become another important measure in the cloud. Since load balancing state level two load in task scheduling under cloud computing environment: the first stage is the virtual machine load, the second one is the resource layer load [4].

2. Quality of Service

The cloud is mainly to provide users with computing and cloud storage services, resource demand for users and resources supplied by provider are performing in the form of quality of service. When task scheduling management comes to task allocation, it is necessary to guarantee the resources' QoS.

3. Economic Principles

Cloud computing resources are widely distributed throughout the world. These resources may belong to different organizations. They have their own management policies. As a business model, cloud computing, according to the different requirements, provide relevant services. So the demand charges is reasonable. Market economy drives task scheduling and resource management, we must make sure their benefit both (consumer and provider) so that the cloud computing can move more and more further [3, 5].

4. The best running time

Primarily for applications, tasks can be divided into different categories according to the needs of users, and then set the best running time on the basis of different goals for each task. It will improve the QoS of task scheduling indirectly in a cloud environment.

5. The throughput of the system

Mainly for cloud computing systems, throughput is a measure of system task scheduling optimizing performance, and it is also a target which have to be considered in business model development. Increase throughput for users and cloud providers would be benefit for them both. [3-5].

2. Strategy for Task Scheduling Algorithm

Within the internal cloud MapReduce mode, task scheduling is essentially to assign n independent task to m virtual machine resource in order to achieve the target of full use of resources in minimum finishing time [1]. We define FTI as the finish time of the task i, define the span as FTmax= max{ FTi, i=1, …, n }. The scheduled task is to find the optimal collection that make the spans FTmax and ∑FTI minimum in the 2m subset of the possible resource space. So, cloud computing, the multi-task scheduling problem, is a NP-hard problem. Nowadays, more research on task scheduling algorithm is as follows [3].

2.1. Intelligent Scheduling-Heuristic Task Scheduling

2.1.1. GA Task Scheduling-Genetic Scheduling (Genetic Algorithm)

GA simulate solving process of problems by chromosomes, GA find the optimal algorithm at the basis of the optimal chromosome. The main idea is: in the environment of question assumptions, according to the way you adapt to survive, electing chromosomes which have stronger power to survive and then copying them with continuous variation. Finding evolution individuals with more vitality and resilience, and ultimately finding the best individual which is the best solutions to problems. We represent the correspondence of task and processor by ProElm, and represent the order value of task priority by Pri. Chromosomes express in a form of a priority table ultimately. In fact, tasks are ranged in accordance with the rise of the order of the PRI in most cases. The goal of GA is to reduce the scheduled time of task and the number of the priority tables. In this way, even there are many tasks replication without any function waiting to be copied, the impact will still very satisfactory [7].

2.1.2. Task Scheduling based on Ant Algorithm (AA Ant Algorithm)

Ant algorithm is a kind of new heuristic algorithm which based on the behavior of ants. It's idea is: On their way of walking around in searching food, it release pheromones on the way they passed, so pheromones will increase very quickly in the shorter path. And the number of pheromones on each path, at the same time, indirectly reflects the probability that other ants' choices. Finally, all ants will choose the shortest path. According to experiences, the scheduler can get the forecast results more quickly and easily.

Comparative analysis of two heuristic algorithms: Genetic algorithm possesses fast search capabilities, while it is not conducive to use feedback information in an entire system. When the solution is in redundant case, the exact solution is obtained inefficiently. Ant algorithm possesses global search capabilities which are parallel and distributed. The efficiency is very low when it is lack of information in early time. By analysis, when designing task scheduling algorithm, the fusion of the two algorithms can be used. As a result, you can get the accumulation of information and can search quickly at the same time, then get a better scheduling effect by complementing advantages and disadvantages.

2.2. Agent Task Scheduling

The whole idea of task scheduling based on agent: Each node of the resource information will be sealed into a proxy so that we can regard cloud systems as a set of systems which have many layers of agent. The high performance in computing applications of the entire cloud system are generally provided by the underlying agent system, which means provided by different computer cluster (resources information node distributed in different places). Therefore, the agent task scheduling has the feature of strong expansion, suitable for applying in a lot of infrastructure resources. In fact, the entire agent schedule is how to assign tasks among multiple agents, and conduct the secondary allocation based on complete status of the agent processing task [3, 7].

2.3. Task Scheduling Algorithm based on the PETRI Network

Mainly used to describe asynchronous, concurrent computer system model. The PETRI network has strict mathematical formulation. It also has very intuitive graphical portrayed. Therefore, this model is not only suitable for static structure, but also for the dynamic processing. Overall, the PETRI network design the whole task flow chart: Data flow, task status, control information and other expressions to the allocation and the using of resource (concurrent style, synchronization and resource contention type) can be presented on this model in a formal, comprehensive way [7, 8].

2.4. Task Scheduling based on the Cost Standard

The scheduling which regard the economy as the measure standard is called cost scheduling. The idea of the cost scheduling is to combine a variety of different resources (processor, bandwidth, storage space) which are used in the cloud systems through the translation of the virtual machine. In the process of scheduling, we can use different cost functions according to different situation when we use virtual machine. We can take the minimum total cost of the machines to complete the scheduling strategy [7].

3. Resource Management and Task Scheduling in Cloud Computing Environment

Cloud computing technology provides the method of sharing basic infrastructure. It bring the computing resources and storage resources in different geographical positions into a resource pool through virtual technology. Users need to apply before using it, and we need to release resources after using it so that the resources can be reused. In this way, the cloud computing center can provide high-performance computing resources and huge storage resources which are simple and low cost.

With further development and maturity of cloud computing technology, the features of efficient, flexible, and customizable provide a new way to solve the problems encountered in the operation process of the scientific workflow. When using the cloud platform, researchers need to upload resources data sets to the cloud computing platform. Because the scale of resources set may be very large, there are bandwidth limitations between different parts and some set of resources can only be stored in the specified resource center, researchers can not upload all the set of data resources to the same resource center or upload all the set of data resources to every resource center. While they need to upload different set of data resources to different data resource centers, so that the tasks of scientific workflow can be executed in parallel. Because there are strong data resource dependencies between scientific workflow, their implementation often requires frequent transfer and access to the resource center. Unreasonable resource data placement and task scheduling strategies will lead to the excessive of transmission volume and traffic volume easily. On one hand, it increases the user fees for the use of cloud resources, on the other hand, it seriously impact the implementation efficiency of the scientific workflow. So studying an effective and fair task scheduling algorithm in a cloud environment is not only important in resources transmission and reducing the transmission of user fees, but also important in enhancing the implementation of the performance and user satisfaction.

3.1. Questions

The cloud computing environment is famous for its business model. A variety of tasks that users demand, a unified solution, task scheduling remains key issues. There are analogies between the allocating resources in cloud computing system and distribution of social income wealth: The resources provided by foundation facilities manufacturers are equivalent to overall social wealth. The demands of different users are said of different task forms, which can abstractly as social individual. The resources volume paid by user can be seen as reward that social individual gets by labor. On the basis of labor differences, they distribute different wealth. In cloud computing, we know New Berger model of justice distribution and efficiency (time and cost game) which is mainly based on following several points:

1. Cloud computing offers available computing and resources storage for a variety of users and enterprises. Therefore, the cloud computing requires the general resource-allocation policy to meet the needs of different users for the allocation of resources, and achieves higher quality of service. In cloud computing environments, by introducing New Berger fair and justice

distribution theory of the actual demand, resource allocation can be assigned as social distribution with the nature of automatic regulation.

2. Traditional task scheduling algorithm focuses on efficiency or cost. It has pertinency for tasks of Specific style and targets for specific types of tasks such as to target the least finishing time, the most optimum availability, and the least cost. These scheduling policies have better efficiency or better cost advantages, but can cause uneven loading, unilateral advantages of efficiency and cost. The expectations of enterprises integrated QoS are not balanced, that means service requirement quality of task scheduling in cloud environment cannot meet the expectations of users. Therefore, it's more important that the efficiency and cost of task scheduling model are balanced. Task scheduling in cloud computing environment should not only meet the balance between efficiency and cost, but also meet the equitable resources distribution. New Berger model emphasizes these two aspects both [11, 12].

3. Divide users' tasks according to the QoS, users have a clear direction of resources service. The game between efficiency and cost is based on meeting users' benefits or fee requirement and then seeking optimal value or equilibrium point. And finally achieve double-win in user efficiency and cost. We call this process efficiency optimization

4. Cloud computing uses virtualized technology to pack resources and then supply for users. These new traits require us to establish a link between users and virtual resources. And we need to develop new applicable task scheduling and resource mapping mechanism.

5. The level of QoS: To enhance the overall QoS is same as improve customer satisfaction, the main method is using the strategy which is fair and satisfactory. So benefit comes first, and we consider the fairness and costs at the same time [11-13].

3.2. The Analysis of Problem Model

New Berger game theory model apply the theory of social distribution and game theory on task scheduling in cloud computing environment. We hope to find new breakthrough research on task scheduling in cloud environments.

C and c are object features in the structure of ideal and the reality. GO and go are allocation values of resources between the ideal and the reality. In a task scheduling model under cloud environment, c is features of users' task, go is the real resources in accordance with the task allocation. In the referenced structure, c is the QoS features of users' tasks, PRi is the standard of equilibrium point of the previous task data. According to the PRi, we can calculate the user's expectation value of the resources Ei indirectly and avoid user's subjectivity on the allocation of resource, and reduced the influence of user QoS or C. GO, as reasonable resource allocation standard, is a general expectation. The relationships of different parts in the figure: Ei, as the reference in the analogy of C and c, can determine the relationship between Ei and go during task scheduling. Finally, Ei make go and GO convergence, and tend higher integrated QoS. That is to say, PRi to Ei to achieve efficiency and cost game equilibrium constraint, Ei to GO to achieve fairness and justice fair distribution. Between GO and QoS constraints, there is comprehensive QoS constraint, which is also called general expectation constraint. In other words, constraint relationship formed in the pathway of PRi-Ei-GO-go is used in the process of high integrated QoS resources selection constraints during task scheduling; We define the fair evaluation function between go and GO for judging fairly during task scheduling.

4. Simulation Design and Analysis of the Results

4.1. Cloud Computing Simulator

This simulation framework has the following characteristics:

1. It supports the simulation and instantiation of large-scale cloud computing infrastructure on single physical Compute Node.

2. It provides an independent platform, it's main function are data centers, service agents, and the scheduling policy.

3. It can provide a virtualized engine and independent virtualization services in a data center node.

4. It can switch the virtualized services flexibly between Core allocation strategy of shared space and shared time.

4.2. Analysis of Experimental Data

We use three types of task scheduling in the experiments to simulate general expectation constraint scheduling, task parameters are showed In Table 1 and detailed performance information of the virtual machine in Table 2 (mainly in memory and bandwidth, the product of numbers of CPU).

Table 1. Task Parameters
(1 is minimum expected completion time, 2 is expected cost, 3 is expected bandwidth)

Task id	Type	Length	Size of input	Size of output	Expected bandwidth	Expected time	Expected cost
0	1	2500	1200	600	1500	400	500
1	1	800	300	300	3000	200	100
2	1	3000	2000	600	1200	500	500
3	2	2000	800	500	2000	1000	250
4	2	5000	5000	3000	1800	400	50
5	2	2000	800	400	1500	1000	100
6	3	3000	2000	500	1000	500	150
7	3	3000	2500	600	1000	600	150

Table 2. The Table of Virtual Machine Parameters

Virtual id	CPU	Memory	Bandwidth	Performance
1	2	1024	3000	6144000
2	2	1024	1500	3072000
3	1	512	1200	614400
4	4	2048	1200	9830400

4.3. Achievement of the Three Scheduling Algorithms

CloudSim is open source. We can run it on the Windows or Linux operating system. It provides users a series of extended entities and methods. In addition, it help users realize their own scheduling and allocation strategy and conduct related performance tests by expanding these interfaces. This paper aims at expanding CloudSim by using the task scheduling strategy. CloudSim provides a very good simulation platform for cloud computing task scheduling algorithm. It requires appropriate API according to the general expected constraints of users' tasks. This paper mainly achieves a custom scheduling, and simulates the scheduling algorithm experiment by using bindCloudletToVm(int cloudletId, int vmId) which provided in class DatacenterBroker.

1. Equitable distribution algorithm strategy

Assigning a set of tasks to a group of virtual machines. The first virtual machine will reallocate tasks after all virtual machines running tasks. This method is to ensure that each virtual machine runs the same number of tasks to flat load as fair as possible. It doesn't consider demand and the difference between different virtual machine. Assigning task sets to DatacenterBroker sequentially and adding method bindCloudletsToVmsSimple() into this class.

2. Efficiency-first algorithm strategy

It means complete within a minimum of time, the configuration of tasks and virtual machine can not be exactly the same. In general, the efficiency-first strategy is adopted without regarding fairness. Hope all tasks are completed within the minimum time, and only consider the difference of two parameters (instruction length (MI) and the virtual machine execution speed (MIPS)) between tasks. By analyzing CloudSim's default operation mode, the full time of a task is equal to that the task's instruction length divided by the speed of execution of the virtual machine. The task execution time is only related with MI and MIPS. On this premise, the following conclusions can be drawn:

(1) Due to task instruction length and execution speed of the virtual machine is certain, so is the task completion time. it doesn't relate to time-sharing or space contribution between tasks and virtual machine.

(2) If the execution time of a task in a virtual machine is the shortest. then the execution time in other virtual machine either.

(3) If the speed of a virtual machine executed is the fastest, then it is faster than any other virtual machines no matter what task it runs.

Defining a matrix, time [i] [j] stands for the execution time that task i needs in the virtual machine j. Obviously, time [i] [j] = MI [i] / MIPS [j]. Before initializing the matrix time, you need to sort tasks in descending order by the size of MI, and then on the contrary, sort the virtual machine in the ascending order by the size of MIPS.

Pay attention that in the reordered matrix, the line number and the task id are not one-to-one corresponding any more. The corresponding relations between the column numbers and virtual machine id also change accordingly.

After initialization, all of the elements in each line and each column are in descending order. Then do greedy to time.

The greedy strategy that we choose is: each time try to assign the task whose line number is 0 from the matrix to the corresponding virtual machine in the last column. It will complete the assignment if the choice is optimal relative to other options. Otherwise, task will be assigned to the virtual machine for the best results.

Meanwhile, if there are a variety of distribution methods that can make the best results, task will be allocated to the virtual machine which runs the least tasks so that we can achieve a simple load balancing. This reflects that the more complex the task is, the faster virtual machine you need to handle so we can deal with the bottleneck caused by the complex tasks, and reduce the total execution time of all tasks.

3. general expectation constraint algorithm strategy

The optimal efficiency and fair allocation algorithm. In this strategy, efficiency and cost can achieve the optimal equilibrium point, named the game equilibrium point. Then fair allocation strategy will be used when there are more than two dispatches on the basis of the first scheduling. Users submit multiple tasks, and set the corresponding parameters in benefit function. It constructs benefit function between the deadline D and the budget b: B={a, t<bD; a-c (t-bD), t>=bD}; With the description of time and the budget consumption in the benefit function, the change of benefit to users can be very clear. That is to say, higher QoS is the target of task scheduling. System will adopt different scheduling strategies according to the value of these two: if advantages lie in the time efficiency, cost optimization scheduling will be used in the system. On the contrary, if advantages lie in the budget cost, system will adopt scheduling algorithm based on time optimization.

4.4. The Process of Simulation and the Achievement of Coding

1.Initialization

Submit the number of tasks, dispatch records, the order of priority and expected resources to the CIS users. CloudSim.init(num_user, calendar, trace_flag, exclude_from_file, expect resource, exclude_from_processing, report_name);

2. Create a cloud services layer

In the simulation platform, HOST and data center in the cloud services layer is composed of one or more machines, and a machine may be made up by one or more CPU. The experimental procedure create three data centers, a data center consists of two machines, and each machine is made up of 2 CPU. In addition, the two data centers are single-CPU and four-CPU [1, 21].

3. Create DBroker // Create Data Agent

DBroker dataBroker = create DBroker();

int dataBroker Id = dataBroker.get_id();

4. Create VM

vmList=new VMList();

VM vm0 = new VM(new VMCharacter(vm_id, vm_ size, dataBroker Id, mem, bandwide, cpus, pri, vmm, VMsharedTimeScheduling()));//specify dataBroker IdCreated for the virtual machine when create it.

vmList.add(vm0); //Put the virtual machine into the corresponding list.

dataBroker.submitToVMList(vmList); // Assign the virtual machine list vmList to dataBroker and register for it.

5. Creat cloud task

cloudtaskList = new CloudtaskList();

Cloudtast cloudtask0 = new Cloudtast (cloudtask_ id, cloudtask_ length, cloudtask_ inputsize, cloudtask_outputsize);

//Specify task id, task length and the size of task input and output file when you create a

cloud task.
//Sort cloudtaskList at the basis of the priority of the cloud task
The main binding code: //input parameters are the task list and the virtual machine list which will be assigned
6. Start the simulation.
CloudSim.startCloudSim();
7. Statistical results
New cloudtaskList =dataBroker.getcloudtaskList();
CloudSim.stopCloudSim();
printCloudletList(newList);//Print task list.
datacenter0.printDebts(newList)；//Statistics of Users' cost on the data center.
datacenter0.printTimes(newList)；//Statistics of Users' finish time on data center

4.5. The Analysis and Comparison of the Experimental Results

Virtual machine allot task through space shared strategy, so the tasks running on the same virtual machine must be accomplished in turn. Table 5 show an general expectation task scheduling Algorithm based on benefit and fairness. Considering the time, users' cost and comprehensive satisfaction, this algorithm is a good algorithm of splitting the difference.

Table 3. The Simulation Results of Scheduling Algorithm on the Priority of Equity

CloudletId	ClassType	STATUS	VMID	StartTime	FinishTime	Time	budget
4	2	SUCCESS	1	69.76	135.82	66.06	400
0	1	SUCCESS	1	0	69.76	69.76	400
5	2	SUCCESS	2	172.45	241.94	69.49	400
1	1	SUCCESS	2	0	172.45	172.45	400
3	2	SUCCESS	4	0	229.02	229.02	200
2	1	SUCCESS	3	0	211.38	211.38	800
6	3	SUCCESS	3	211.38	290.87	79.49	800
7	3	SUCCESS	4	229.02	467.76	238.73	600

Table 4. The Simulation Results of Scheduling Algorithm on the Priority of Efficiency

CloudletId	ClassType	STATUS	VMID	StartTime	FinishTime	Time	budget
7	3	SUCCESS	1	0	75.28	75.28	400
3	2	SUCCESS	3	0	154.5	154.5	800
1	1	SUCCESS	3	0	60.45	60.45	800
6	3	SUCCESS	1	154.5	224.71	70.22	400
2	1	SUCCESS	4	0	160.12	160.12	200
5	2	SUCCESS	2	60.45	148.48	87.93	400
0	1	SUCCESS	4	75.28	142.48	67.2	200
4	2	SUCCESS	3	142.48	201.23	58.75	400

Table 5. The Simulation Results of Scheduling Algorithm on General Expectation

CloudletId	ClassType	STATUS	VMID	Start Time	FinisTime	Time	budget
4	2	SUCCESS	2	0	60.15	60.15	400
6	3	SUCCESS	4	0	72.45	72.45	200
5	2	SUCCESS	2	0	90.49	90.49	400
2	1	SUCCESS	1	0	162.46	162.46	400
7	3	SUCCESS	1	0	229.12	229.12	400
0	1	SUCCESS	3	0	60.06	60.06	800
1	1	SUCCESS	3	0	70.12	70.12	800
3	2	SUCCESS	4	0	150.73	150.73	200

Experimental result above leads to a conclusion in this paper. Task scheduling algorithm based on the benefits and fair is superior to the previous two algorithms in regards to the cost and the total time. It's obvious that fair allocation strategy has many advantages in the load balance of virtual machine strategy. In the case that the completion time and fairness perform very general and there is no big difference between multitasking sizes and virtual machine resources, fair allocation strategy can be our first choice. Efficiency-first strategy is also called as minimum completion time strategy which is widely used now. If we just take task size and virtual machine performance into consideration, but ignoring communications and other factors. It can not be met and QoS comprehensive expectation will not be high enough in some tasks that have high requirements in reliability and costing. Under this circumstance, we put forward the algorithm of general QoS expectation that based on the Berger game model. Although in the real-time it is worse than efficiency-first strategy, but we can find a compromise which make sure not only users' comprehensive QoS, and also the load balance of relative resource allocation in regards to various users' demands.

5. Conclusion

With the great development and application of cloud computing technology, cloud computing has bring a bold change of traditional software, business model and enterprise management. And it's advantages become more obviously.

It can provide needed services according to users' demands while CIS (cloud information server cloud information service center) takes the management and implementation of task and resources information.

The nature of resources that dynamic and unsure under the cloud environment makes the task scheduling problem more complicated. Due to the shortcomings of past task scheduling problem, the introduction of benefit-fair algorithm under the cloud environment has many good points. For example, it can meet the balance of cost and performance, and up to the requirements of the load balancing fairness. At the same time it can reduce the task execution time and increase the chance of success, and also improve users' comprehensive QoS significantly.

Acknowledgements

The National natural Science Foundation of China (61170277), Supported by innovation Program of Shanghai Municipal Education Commission (12zz137), Top Discipline Construction Projects of Shanghai (S1201YLXK)

References

[1] YANG Zhexi, XUE Huacheng. Informatization Expectation with Cloud Computing in China. *TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2012; 10(4).
[2] Jing Liu, Xing Guo Luo, Bai Nan Li, Xing Ming Zhang, Fan Zhang. An Intelligent Job Scheduling System for Web Service in Cloud Computing. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(6).
[3] Hong Luo, Dejun Mu, Zhiqun Deng, et. Research of Task Scheduling in Grid Computing. *Application Research of Computers*. 2005; (5): 16-19.
[4] Xindong You, Guiran Chang, Xueyao Deng, et. Grid Task Scheduling Algorithm Based on Merit Function. *Computer Science*. 2006; 33(6).
[5] Wen sheng Yao,et al. Genetic Scheduling on Minimal Processing Ele-mentsinthe Grid. *Springer-VerlagHeidelberg*. 2002.
[6] Chunyan Zhao. Research and Implementation of Job Scheduling Algorithm Under Cloud Environment. Master's Thesis of Beijing jiaotong University. 2009; 06.
[7] Yinfeng Zhang, Yulin Li. Grid Computing Resources Management Scheduling System Based on Evolutionary Algorithm. *Computer Engineering*. 2003: 29(15): 110-175.
[8] Congyi Yuan. Petri net theory. Beijing Publishing House of Electronics Industry, 1998.
[9] Yan Jiang, Kai Hu, Zhibin Yang, et. Research of random DAG's parallel scheduling algorithm based on extention. *Computer Science*. 2008; 35(7).
[10] GuiyiWei, Athanasios V Vasilakos, Yao Zheng et.al. A Game-theoretic Method of Fair Resource allocation for cloud computing services. *Springer Science+Business Media*, LLC. 2009
[11] Yiqiu Fang, Fei Wang, Junwei Ge. A Task Scheduling Algorithm Based on Load Balancing Cloud Computing. *Lecture Notes in Computer Science*. 2010; 6318: 271-277.
[12] Peng Wang, Jingyi Dong. Research of a realization method of cloud computing architecture. *Computer Engineering and Science*. 2009; (31) A1.
[13] Zhijia Liu, Tirong Zhang, Xiongchen Xie. Research of "Our customers expect" task scheduling algorithm based on cloud computing. *Popular Science*. 2011; (4): 75-77.
[14] Xingyi Liu, Xiaoling, Wei. Research of the nearest improvement algorithm based on Euclidean distance. *Guangxi University's Journal of science*. 2010; 26(4): 409-411.
[15] Marcos Dias de Assunção, Alexandre di Costanzo, Rajkumar Buyya. A Cost-benefit Analysis of Using Cloud Computing to Extend the Capacity of Clusters. *Cluster Comput*. 2010; 13: 335–347.