Task Scheduling Algorithms with Multiple Factor in Cloud Computing Environment

Nidhi Bansal, Amit Awasthi and Shruti Bansal

Abstract Optimized task scheduling concepts can meet user requirements efficiently by using priority concepts. Increasing the resource utilization and reducing the cost, both are compulsory factors to be compromise in task scheduling algorithms of cloud computation for executing many tasks. With updating the technology many new features in cloud computing introduced such as fault tolerance, high resource utilization, expandability, flexibility, reduced overhead for users, reduced cost, required services etc., this paper discussed task scheduling algorithms based on priority for virtual machines and tasks. This algorithm performs good results with balance the load, but it's not effective with cost performance. Secondly comparative study also has been done in this paper between various scheduling algorithms by CloudSim simulator.

Keywords Cloud computing · Task scheduling · Cost · Load balancing

1 Introduction

In these days all the application runs via virtual machines and resources are allocated to all virtual machines. All applications are independent, unique in technology and has no connection to each other. Resources are compromised on every event or activity performed with all individual units of products and service.

N. Bansal (✉)
Department of Computer Science and Engineering, Vidya College of Engineering, Meerut, India
e-mail: nidhi18jul@gmail.com

A. Awasthi
Department of Computer Science and Engineering NITTTR, Chandigarh, India
e-mail: awasthi.amit1989@gmail.com

S. Bansal
Department of Electronic & Communication, Shobhit University, Meerut, India
e-mail: shruti_mrt@rediffmail.com

S.C. Satapathy et al. (eds.), *Information Systems Design and Intelligent Applications*, Advances in Intelligent Systems and Computing 433,
DOI 10.1007/978-81-322-2755-7_64

The concept of scheduling is performing very important role in cloud computing with requirement of users in the market. 'Mapping the tasks' is the basic concept of scheduling. This is a necessary condition in successful working of cloud as many factors must be examined for useful scheduling. The feasible resources should be appropriate for execution in task scheduling (Tables 1, 2, 3 and 4).

The mechanism of task scheduling can not only satisfy to the user, but also increase the utilization for resources [1]. Load balancing factor must be calculated to acquire more resource utilization. The process of load balancing is distributing the load between various nodes to enhance utilization of resource and task response time while also neglecting the condition where nodes are fully occupied while many other distinct nodes are free or performing for limited work. Load balancing assure that processors in the setup or all node in the rooted network connection does approximately the uniform amount of execution at any present of time.

To get the complete cost of every user's applications, all individual service of resources (like Processor cost, Internal memory used, Input/Output cost, etc.) need to be calculated. When the complete cost of all resources has been identified, factual cost and output dissection depend on it can be retrieving, related to all of the traditional concepts of scheduling. Traditional concept leading the use of absolute tasks for users and exceeds the overheads in applications of cloud computation. It may be true that any distinct tasks may not the reason of exceeding costs for

Table 1 FCFS Vs VM-Tree

Cloudlets	FCFS	VM_Tree
50	5668.944	5894.572
70	5512.491	5602.98
100	5486.416	5794.572

Table 2 FCFS Vs PSO

Cloudlets	FCFS	PSO
50	5668.944	2473.44
70	5512.491	2864.571
100	5486.416	2929.76

Table 3 QoS of FCFS

Cloudlets	FCFS	QoS
50	5668.944	4481.248
70	5512.491	4298.72
100	5486.416	4389.984

Table 4 FCFS Vs ABC

Cloudlets	FCFS	ABC
50	5668.944	2473.44
70	5512.491	2864.571
100	5486.416	2929.76

resources in traditional way. The result is that exceeds in estimate and raises the cost. To compete towards market, some organization has had to reduce the cost or prices of expansive items or products. But they have been capable to get huge mark-ups on less expensive tasks. Minimization in cost or price has raised marketing for individual units/item but degrades the complete strength and good mark-ups on specific tasks or product didn't equal the down in the confine output of expensive items [2].

However, Load Balancing and Allocation Cost are the primary issues for task scheduling algorithms of user's applications in cloud computation. Presently the complete cost and proper resource utilization of the scheduling issue has encouraged researchers to recommend multifarious cost related task scheduling algorithms. More advanced algorithm designed on these factors are introduced by innovators or researchers such as Activity Based Cost, Particle Swarm Optimization, DLA (Double Level Priority) and Balancing the Load [1–3, 6–13] etc.

This paper proposes an implementation view for scheduling techniques with simulated outcome in CloudSim3.0 simulator by taking the factor cost and balancing the load.

The remaining part of this research paper is implemented as followed: Sect. 2 explains/methodology. Section 3 derives the implementation by simulation and analysis. Section 4 shows the conclusion.

2 Various Task Schedulings

This research paper examined with traditional method i.e. First Come First Serve and optimized scheduling methods i.e. VMT, PSO, QoS, ABC. Brief introduction for every scheduling is described here:

Generally First Come First Serve algorithm is take it as by default scheduling to explain any traditional concept of scheduling because there is no priority for any parameter and its very simple in implementation without any error.

Now comes to optimized or prioritized scheduling concepts. First optimized method is explained a tree form hierarchy named Virtual Machine_Tree for proper running of input tasks with including the concept of priority for machines and for cloudlets/tasks also. Depth First Search is modified according to scheduling concept to get effective output [4].

Second optimization technique introduces an algorithm which relates the method of small position value (SPV). Particle Swarm Optimization (PSO) is used to reduce the execution cost for task scheduling in cloud computation [3].

QoS-driven is the third method used in this paper that proposed a concept with considering many task attributes like privileges of application, expectation, length of executed task and the time awaiting in series to execute and sort applications by the priority [5].

ABC (Activity Based Costing) is the fourth one that introduces an optimized method based on priority in terms of benefit for SP (service provider). The

Fig. 1 Basic model for implementing task scheduling

traditional concept of task processing cannot fulfill the user's requirement. Activity-based method measures the every event or activity cost for all objects and the outcome is better than traditional method in Cloud Computation [2].

2.1 Methodology

Figure 1 At step first, all optimized methods and traditional scheduling algorithm have been compared using simulator CloudSim3.0. Comparison shows that optimized algorithms (priority concepts) are always perform better than the traditional methods. Load balancing and allocation cost parameter are calculated in these comparisons. At the final step, outcome of this research paper is very able to find an effective technique that executes or performs good to get more resource utilization and reduce cost.

3 Simulation

CloudSim3.0 simulator is used to simulate the all these task scheduling algorithms explained above. To compare the effective performance under many distinct parameters. An open environment is considered with two host node to implement the scheduling techniques with thirty autonomous tasks. It can be dynamically changed throughout the simulation. This simulation mainly focuses on the load balancing and cost factors between these scheduling methods.

To estimate the effective performance of scheduling methods, datacentres, virtual machines and many cloudlets based on user's requirement are created in simulator. Now time to schedule the tasks based on all entire scheduling methods for example, virtual machine tree [4], particle swarm optimization [3] and QoS-driven [5] activity based costing [2].

3.1 Performance Metrics

We have evaluated the scheduling techniques using allocation cost and load balancing metrics and compared with traditional scheduling algorithm. Parameters are:

(1) Size of virtual machine is 10,000 with 512 memory allocation, 250 instructions per seconds, 100 BandWidth.
(2) Length of task is 40,000 and the file size is 300.
(3) Memory allocated for host is 16,384, 1,000,000 for Storage, 10,000 BandWidth.

The structure designed for the CloudSim tool is includes two datacenters, thirty virtual machines, two hosts in each DC and 4 Processing Element (PE) or CPU cores for each host. Implementation has been done using 50 tasks to analyze the algorithms for much number of tasks/cloudlets. As the cloudlets (applications) are submitted by the user it is the task of the cloud broker (Behalf of client, Cloud broker works and search the best virtual machine to execute the application, the VM is selected by measuring the various parameters, for example size, bandwidth, cost) to allocate those tasks to the VM and then Virtual Machine Manager selects the host on which this VM should be worked based on the allocation policy of virtual machines. When VM is assigned to the host then VM starts for execution. Every VM has a virtual processor called PE (processing element) in CloudSim. The VM can have much processors or process elements which simulates the real multi-core CPUs.

Cost is measured by:

$$\begin{aligned}\text{Cost} = {} & \text{datacenterhost.costPerStorage} * \text{vm.size} + \text{datacenterhost.costPerRam} * \text{vm.ram} \\ & + \text{datacenterhost.costPerBw} * \text{vm.bw} \\ & + \text{datacenterhost.costPerMips} * (\text{vm.mips} * \text{vm.numberOfPes});\end{aligned}$$

Load balancing is measured by:

$$\text{AL} = \text{VmL}/\text{n}$$

AL represents average load.
VmL represents Load of virtual machines is calculated by the load average of the cloudlets that execute on it.
n represents number of virtual machines.

3.2 Simulation and Results

3.2.1 Load Balancing Parameter

With designing the environment, Fig. 2 illustrates 10–50 input tasks and five virtual machines are taking for calculate the load balancing parameter in traditional method

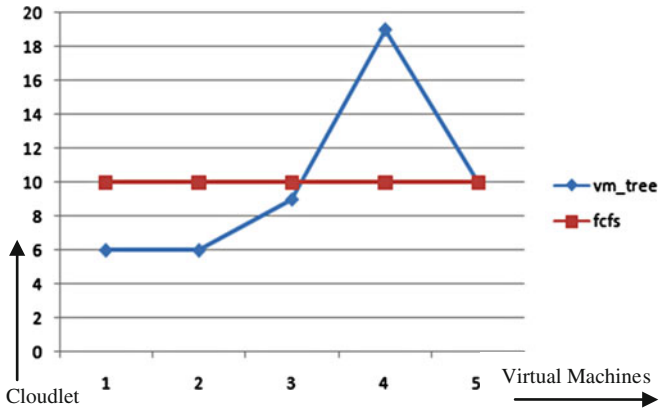

Fig. 2 Comparison with load balancing factor between VM_Tree and FCFS method

and optimized method. By introducing the figure, much number of input cloudlets runs on that machines have high memory, instructions and bandwidth in VM_tree optimization algorithm. While load are same for all VM's in FCFS algorithm.

3.2.2 Cost Parameter

With designing the environment, 10–100 cloudlets and thirty virtual machines are taking and calculate the cost factor with four all optimized methods illustrated below.

Figure 3, explains the comparison between VM_Tree and FCFS scheduling algorithms with allocation cost metrics against the number of cloudlets. It identifies that allocation cost is increased in VM_Tree optimized scheduling algorithm.

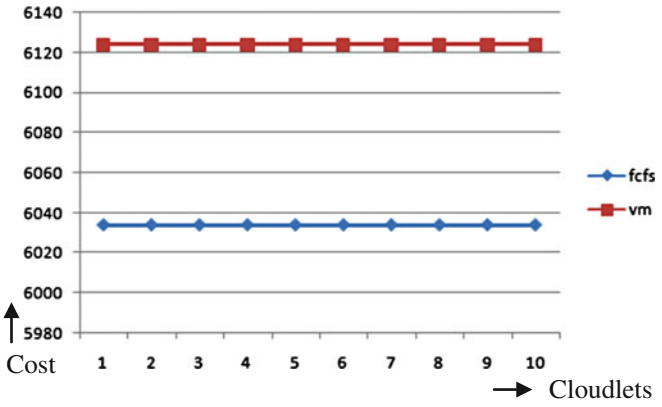

Fig. 3 Variation in cost factor between VM_Tree and FCFS scheduling

Hence proved that VM_Tree task scheduling method is perform good in load balancing factor only.

This approach can also be justified by rest of the algorithms with cost parameter only.

Figure 4, illustrates the comparison between PSO and FCFS task scheduling algorithm with parameter allocation cost against the number of cloudlets.

Figure 5, measuring the allocation cost is against the number of cloudlets and shows the comparison between QoS and FCFS (First Come First Serve) task scheduling algorithm. An optimized method i.e. Qos gives better performance from the traditional one.

Figure 6, illustrates the comparison between ABC and FCFS task scheduling algorithm based on cost against the number of cloudlets. It clearly identifies that allocation cost is reduced in optimized scheduling algorithm.

Results of this study proved that all the optimized task scheduling algorithms are very efficient.

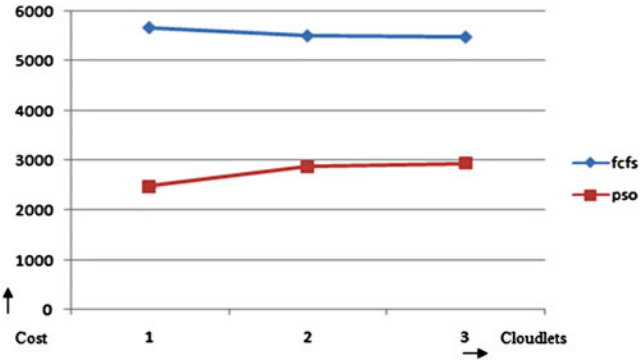

Fig. 4 Variation in cost factor between PSO and FCFS scheduling

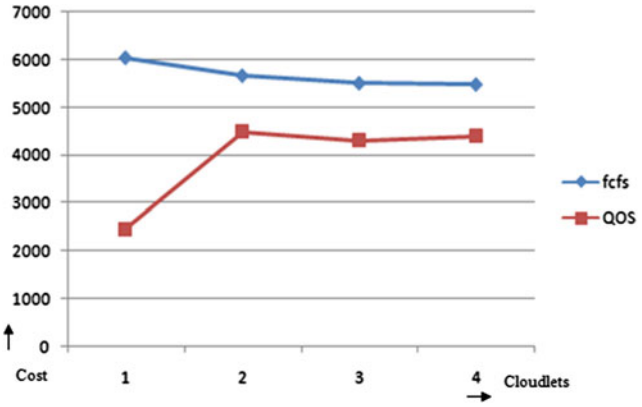

Fig. 5 Variation in cost factor between QoS and FCFS scheduling

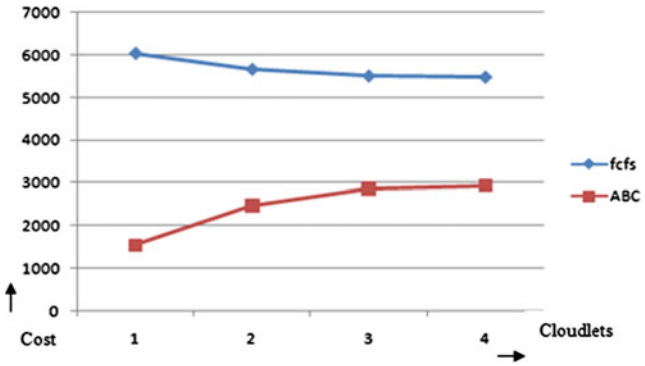

Fig. 6 Allocation cost variation in ABC and FCFS scheduling

4 Conclusion and Future Work

The traditional way of task scheduling in cloud computing is to execute or schedule the task is very difficult. For proper utilization of resources and optimal solution with task scheduling algorithms in cloud computing is very important according to its hike fame day by day. With considered the load balancing and cost parameter for virtual machine tree optimized task scheduling algorithm, proved that cost parameter is not so efficient with this algorithm. Apart from this result other comparisons are also implemented in this paper with cost parameter in cloud computing environment. Many more users' requirements will consider in future, such as resource reliability and availability to further enhance the scheduling techniques.

References

1. Yiqiu Fang, Fei Wang and Junwei Ge, "A Task Scheduling Algorithm Based on Load Balancing in Cloud Computing" Springer, Lecture Notes in Computer Science Vol. 6318, 2010.
2. Qi Cao, Zhi-Bo Wei and Wen-Mao Gong, "An Optimized Algorithm for Task Scheduling Based On Activity Based Costing in Cloud Computing", IEEE The 3rd International Conference on Bioinformatics and Biomedical Engineering 2009.
3. Lizheng Guo, Shuguang Zhao, Shigen Shen and Changyuan Jiang, "Task Scheduling Optimization in Cloud Computing Based on Heuristic Algorithm", ACADEMY PUBLISHER, Journal of Networks, Vol. 7, No. 3, pp. 547–553, 2012.
4. Raghavendra Achar, P. Santhi Thilagam, Shwetha D, Pooja H, Roshni and Andrea, "Optimal Scheduling of Computational Task in Cloud using Virtual Machine Tree", Third International Conference on Emerging Applications of Information Technology, IEEE 2012.

5. Xiaonoan Wu, Mengqing Deng, Runlian Zhang and Shengyuan, "A task scheduling algorithm based on QoS-driven in Cloud Computing", First International Conference on Information Technology and Quantitative Management, Procedia Computer Science, Vol. 17, pp. 1162–1169, ELSEVIER 2013.
6. R. Buyya and M. Murshed, "GridSim: A Toolkit for the Modelling and Simulation of Distributed Resource Management and scheduling for Grid Computing", The Journal of Concurrency and Computation: Practice and Experience (CCPE), Vol. 14, No. 13–15, Wiley Press, November 2002.
7. Calheiros, R.N., Ranjan R, De Rose, C.A.F. and Buyya R., "CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services" in Technical Report, GRIDS-TR-2009-1, Grid Computing and Distributed Systems Laboratory, The University of Melbourne, Australia, 2009.
8. Rajkumar Buyya, Rajiv Ranjan and Rodrigo N. Calheiros, "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities", High Performance Computing & Simulation, pp. 1–11, June 2009.
9. GAO Zhong-wen; ZHANG Kai, "The Research on cloud computing resource scheduling method based on Time-Cost-Trust model" IEEE 2012 2nd International Conference on Computer Science and Network Technology (ICCSNT).
10. Cory Janssen : http://www.techopedia.com/definition/23455/first-come-first-served-fcfs.
11. Wang Zong jiang; Zheng Qiu sheng, "A New Task Scheduling Algorithm in Hybrid Cloud Environment" IEEE 2012, (CSC) International Conference on Cloud Computing and Service Computing.
12. Hu Wu, Zhuo Tang, Renfa Li, -A Priority constrained scheduling strategy of Multiple Workflows for Cloud Computing‖, ISBN 978-89-5519-163-9 ICACT 2012.
13. Shachee Parikh and Richa Sinha, "Double Level Priority based Optimization Algorithm for Task Scheduling in Cloud Computing" International Journal of Computer Applications Vol. 62, No. 20, 2013.