Third International Conference on Recent Trends in Computing (ICRTC 2015)

Cost performance of QoS Driven task scheduling in cloud computing

Nidhi Bansal\textsuperscript{a}, Amitab Maurya\textsuperscript{a}, Tarun Kumar\textsuperscript{a}, Manzeet Singh\textsuperscript{a}, Shruti Bansal\textsuperscript{b}

\textsuperscript{a}Department of Computer Science and Engineering, Vidyav College of Engineering, Meerut, INDIA
\textsuperscript{b}Department of Electronics and Communication Engineering, Shobhit University, Meerut, INDIA

Abstract

Till now many parameters considered in QoS driven like makespan, latency and load balancing. But allocation cost parameter is not considered in QoS-driven scheduling algorithm. Minimizing the total allocation cost is an important issue in cloud computing. In this paper, the cost is calculated of QoS-driven task scheduling algorithm and compare with traditional task scheduling algorithm in cloud computing environment. The experimental results based on cloudsim3.0 toolkit with NetBeans IDE8.0 shows that QoS-driven achieves good performance in cost parameter.

© 2015 The Authors. Published by Elsevier B.V.

1. Introduction

In these days, all the companies are using the concept of cloud computing. It is a new technology. It works based on Pay on Demand model. This model means, in which quantity user required the resources for a specific time to complete a task, user have to pay some money only for that much of time. Due to this technology many benefits are applicable to get the better profit in the market. Profit in terms of time, cost, load balancing, storage and so on. In this technology, all applications run on a virtual platform and all the resources are distributed among these virtual machines. Each and every application is different and is independent.

Scheduling is another concept in cloud computing with ongoing days its hike in cloud fame. Basic means of scheduling is that mapping of processing tasks to the resources on the basis of tasks’ attributes. This all happened with the care of all available resources should be utilized properly.

To measure the total allocation cost of each applications, every individual use of resources (like CPU cost, memory cost, I/O cost, etc.) must be calculated. When the total allocation cost of each individual resources are calculated, accurate cost and profit analysis based on it can be obtained, compared to those of the traditional scheduling methods. Due to traditional scheduling concept over budgeting and over pricing is obtained in the market.
The task scheduling concept is not only complete user's satisfaction, but also get better resource utilization [12].

However, Allocation Cost is the major challenging problem for scheduling of tasks in cloud computing. Many researchers' works on this parameter. Many improved algorithm based on this problem are introduced by researchers such as ABC, PSO, Double Level Priority [1, 2, 5] etc.

This paper completes the QoS-driven scheduling algorithm with considering cost parameter.

The rest of the paper is introduced as follows: Section 2 explains methodology. Section 3 describes the simulation. Section 4 shows the conclusion.

2. Task Scheduling Algorithm

With the collected material, according to the requirements of all tasks, we introduced cost parameter in developed QoS strategy to deal in a market place with more profit to the user.

In this research paper, we deal with two algorithms that are traditional scheduling algorithm FCFS and optimized scheduling algorithm QoS driven.

The brief descriptions for these scheduling methods are described here.

First method that is FCFS (first come first serve). This scheduling algorithm is very common in method. It is a simple method to develop. It uses non-pre-emptive method for scheduling in which all processes are automatically situated in the queue and go for processing according to user request.

Second method proposed a scheduling algorithm based on QoS-driven in Cloud Computing with considering execution time, load balancing and latency. This algorithm based on user requirements, user rights, process expectation, process length and the execution time to compute the task priority [3].

2.1 Methodology

Figure 1 shows the basic model for execution of scheduling algorithm using Cloudsim. In cloud computing system first of all cloudsim simulator package is initialized then datacenter then broker then virtual machines and then cloudlets or tasks. Then broker communicates with the virtual machines and scheduling executes. Finally display the results.
Figure 2 shows the comparison between the optimized scheduling algorithm and traditional scheduling algorithm. The outcome of this comparison is that this optimized algorithm is better than the traditional one in cost parameter also.

Figure 2. Shows the comparison process

3. EXPERIMENTAL EVALUATION

Cloud computing provide an open source concept to simulate the scheduling algorithms. It contains many datacenters and all type of resources by using virtualization to deliver the services to the customer with many benefits.

To analyse the performance of the scheduling algorithms, cloudsim3.0 toolkit is used to simulate cloud environment. All the tasks are non-pre-emptive which means processor cannot leave the task until the task has completed its execution, it is necessary so that the processor can be efficiently used for the remaining unprocessed tasks.

3.1 Performance metrics and Simulation configuration

In this paper, we tested the above two scheduling methods using a common metric i.e. allocation cost.

The configuration of the cloudSim tool includes 2 datacenters, 30 VMs, 2 hosts in each DC and 4 Processing Element (PE) or CPU cores for each host. Implementation has been done using 50 cloudlets for simplicity and also analyzed the algorithms for more cloudlets. As the cloudlets (applications) are submitted by the user it is the task of the cloud broker (Cloud broker works on behalf of client and finds out the best VM to run the application, the VM is decided by looking at different parameters like size, bandwidth, cost of VM) to assign those tasks to the VM and then Virtual Machine Manager (VMM) decides the host on which this VM should be allocated based on the VM Allocation policy. After VM is assigned to the host the VM starts running the cloudlets i.e. applications. Here CPU scheduling algorithms come into existence. Every VM has a virtual CPU called PE in CloudSim. The VM can have one PE or more which simulates the original multi-core CPUs. Analyses have been done between above two algorithms that are FCFS and QoS driven.

The total cost of scheduling is calculated by:

$$\text{Double cost} = \text{datacenterhost.costPerStorage} \times \text{vm.size} + \text{datacenterhost.costPerRam} \times \text{vm.ram} + \text{datacenterhost.costPerBw} \times \text{vm.bw} + \text{datacenterhost.costPerMips} \times (\text{vm.mips} \times \text{vm.numberOfPes})$$

Simulated parameters are set as follows.

(1) Set the virtual machine with size 10000, RAM 512, MIPS 250, BandWidth 100.
(2) Set the task with Length 40000, file size 300.
(3) Set the configuration for a host with RAM 16384, Storage 1000000, BandWidth 10000.

3.2 Experiments and Results analysis

(1) Variation in allocation cost parameter

Under the design environment, the averages of allocation cost with two different scheduling methods with
10 ~ 100 tasks or cloudlets and 30 virtual machines are illustrated in Figure 3.

![Figure 3. Allocation cost variation in QoS and FCFS scheduling algorithm](image)

Results of this paper is that QoS optimized scheduling algorithms is very efficient with the traditional one. Tabulated representation of results is also mentioned here in Table 1.

**Table 1. Analysis of scheduling algorithms**

| Cloudlets | FCFS     | Qos       |
|-----------|----------|-----------|
| 50        | 5668.944 | 4481.248  |
| 70        | 5512.491 | 4298.72   |
| 100       | 5486.416 | 4398.984  |

**4. Conclusion and Future Scope**

There are many task scheduling algorithms which give good performance, good resource utilization and are cost effective. But according to user satisfaction this parameter is also important to consider. This paper analysed that QoS scheduling algorithm is very efficient algorithm with makespan, latency, load balancing and cost factor. This algorithm efficient with first three parameters except cost is introduced in paper [3]. But this paper completes this algorithm with the result of cost factor also.

Cloud computing is growing day by day in the market. This paper discussed the results with cost factor only with a specific QoS algorithm. In future, the implementation can be done for further improvements with more factors measuring in real time with many other algorithms.

**Table 2. Performance of Qos Driven scheduling algorithm before simulation this paper**

| Algorithm  | Makespan | Latency | Load Balancing | Cost |
|------------|----------|---------|----------------|------|
| Qos Driven | Yes      | Yes     | Yes            | No   |

**Table 3. Performance of Qos Driven scheduling algorithm after simulation this paper**

| Algorithm  | Makespan | Latency | Load Balancing | Cost |
|------------|----------|---------|----------------|------|
| Qos Driven | Yes      | Yes     | Yes            | Yes  |
References

[1] 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.

[2] 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.

[3] 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.

[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] 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.

[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] R. 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] 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.

[10] S. Liu, G. Quan and S. Ren, “On-line Scheduling of Real-time Services for Cloud Computing”, In Proceedings of 6th World Congress on Services (SERVICES-1), pp. 459-464, 2010.

[11] Juan M. Fernández-Luna, Juan F. Huete, Ramiro Pérez-Vázquez, Julio C. Rodríguez-Cano and Chirag Shah, “COSME: A NetBeans IDE plugin as a team-centric alternative for search driven software development”, ACM 2010.

[12] 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.