© 2018 IJSRST | Volume 4 | Issue 9 | Print ISSN : 2395-6011 | Online ISSN : 2395-602X
Themed Section: Science and Technology

Improved Effective Load Balancing Technique for Cloud

Vividha Kulkarni, Sunita, Shilpa B Kodli
Department of Studies in Computer Applications (MCA), VTU Centre for Post Graduation Studies Kalaburagi, Karnataka, India

ABSTRACT

Cloud technology provides services over the network on demand and request. In cloud computing, the cloud user may get access to the data and applications from anywhere at any time. Users have the advantage of paying only for what they are using satisfying their fluctuating demands. Managing such a large amount of user requests coming from all over the globe is a tedious task, thus the cloud service provider should take efficient steps to handle them. This involves the concept of load balancing. Load Balancing basically helps in even distribution of loads among all the nodes such that no node is overloaded or under loaded. Considering the growing users and importance of cloud, finding new ways which can prove efficient in balancing the load among various processing nodes will always be an area of concern and research. Various algorithms already exist for load balancing, each having its own concept but have the same goal to reduce the response time to client problems. The goal of this paper is to propose an algorithm for Load balancing The algorithm proposed in this paper aims to give higher importance to requests with higher priorities.

Keywords : Broker, Burst time , Cloudlet, Context switch, Data center, Priority queue, Queue, Threshold ,Time quantum, Virtual Machines.

I. INTRODUCTION

In cloud computing, if users are increasing load will also be increased. The increase in the number of users will lead to poor performance in terms of resource usage if the cloud provider is not configured with any good mechanism for load balancing and also the capacity of cloud servers would not be utilized properly. This will confiscate or seize the performance of heavy loaded node. If some good load balancing technique is implemented, it will equally divide the load (here term equally defines low load on heavy loaded node and more load on node with less load now) and thereby we can maximize resource utilization. One of the crucial issue of cloud computing is to divide the workload dynamically. Load balancing is the process of improving the performance of the system by shifting of workload among the processors. Workload of a machine means the total processing time it requires to execute all the tasks assigned to the machine. Balancing the load of virtual machines uniformly means that anyone of the available machine is not idle or partially loaded while others are heavily loaded. Load balancing is one of the important factors to heighten the working performance of the cloud service provider. In the IT industry, the main reason behind the development of new technologies is to reduce the response time to client problem & gain more & more client satisfaction. Load balancing in cloud computing does the same work here, request after reaching the data center (where information are stored in bulk) are distributed among the virtual machine following certain load balancing algorithm. The way the data are selected, virtual machine is created, tasks are allocated & other similar things are done, are depended on policies of the cloud made by the cloud service providers. Load balancing is a complicated task in cloud computing

because in a network we have computers with varying capacities & users with different requirements which make the process of assigning task to different nodes complex.

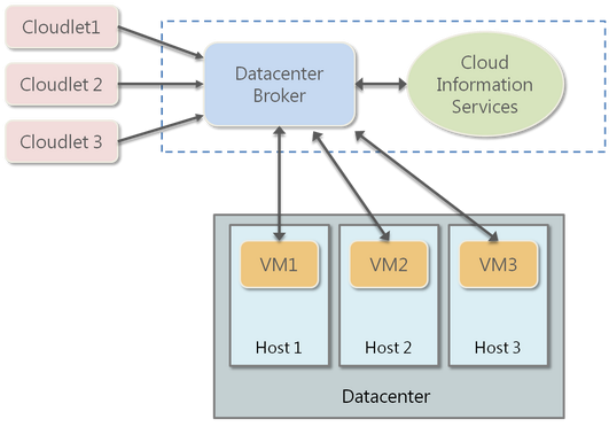

Figure 1. load balancing in cloud computing

In the fig1 cloudlet is the first one it receive the user request from the users and send to the broker. As the users request cloudlet will be created. The load balancing schedule the task it will send to data centre. The data centre send to virtual machine Id .Cloud information system check the virtual machine how many processer are busy and how many are ideal that information gives to broker.

II. LITERATURE SUREVEY

In [1] author Ajit Singh et.al discussed round robin scheduling which help to improve the CPU efficiency in real time and time sharing operating system. There are many algorithms available for CPU scheduling. But we cannot implemented in real time operating system because of high context switch rates, large waiting time, large response time, large turn around time and less throughput. The proposed algorithm improves all the drawback of simple round robin architecture. The author have also given comparative analysis of proposed with simple round robin scheduling algorithm. Therefore, the In [1] author strongly feel that the proposed architecture solves all the problem encountered in simple round robin architecture by decreasing the performance parameters to desirable extent and thereby increasing the system throughput.

Cloud computing is a fast growing area in computing research and industry today. With the advancement of the Cloud, there are new possibilities opening up on how applications can be built and how different services can be offered to the end user through Virtualization, on the internet[2]. There are the cloud service providers who provide large scaled computing infrastructure defined on usage, and provide the infrastructure services in a very flexible manner which the users can scale up or down at will. The establishment of an effective load balancing algorithm and how to use Cloud computing resources efficiently for effective and efficient cloud computing is one of the Cloud computing service providers' ultimate goals.

In [3] author **Jasmin James** et.al. proposed a new VM load balancing algorithm and implemented for an IaaS framework and Simulated cloud computing environment; i.e. 'Weighted Active Monitoring Load Balancing Algorithm' using CloudSim tools, for the Datacenter to effectively load balance requests between the available virtual machines assigning a weight, in order to achieve better performance parameters such as response time and Data processing time.

Central Processing Unit (CPU) scheduling plays a deep-seated role by switching the CPU among various processes. As processor is the important resource, CPU scheduling becomes very important in accomplishing the operating system (OS)design goals. The intention of the OS should allow the process as many as possible running at all the time in order to make best use of CPU. The high efficient CPU scheduler depends on design of the high quality scheduling algorithms which suits the scheduling goals. In [4] authors proposed an algorithm which can handle all types of process with optimum scheduling criteria.

In [5] author **Dr.Bhupendra Verma** et.al proposed an algorithm to serve all types of job with optimum scheduling criteria. The treatment of shortest process in SJF scheduling tends to result in increased waiting time for long processes. And the long process will never get served, though it produces minimum average waiting time and average turnaround time. This problem can be overcome by the proposed algorithm. It is recommended that any kind of simulation for any CPU scheduling algorithm has limited accuracy. The only way to evaluate a scheduling algorithm to code it and has to put it in the operating system, only then a proper working capability of the algorithm can be measured in real time systems.

III. PROPOSED SYSTEM

The proposed algorithm uses the concept SJF and WRR .The tasks arrive along with their priority values as well as with their burst time. The tasks are kept into priority queues according to their priority values and a separate time quantum is given to each priority queue. The proposed algorithm uses the concept SJF and WRR combine together created the MOSA.

MOSA(Modified optimal scheduling algorithm)

In this paper we have proposed the new scheduling algorithm using the Shortest Job First algorithm and Weighted Round Robin algorithms combined together as an Effective Modified optimal scheduling algorithm which handles the multiple user requests to optimize the load to the server and balancing the overload of the servers based on the requests. In the proposed system the two algorithms are carried out based on the time difference in the subsequent processes by comparing with the threshold value of the system. The proposed system consumes the less time compare to other scheduling SJF and WRR algorithms. Priorities are assigned in the beginning after analysing there source requirement ,time or memory requirement or user preference.
matching priority.

Algorithm:

1. Start the process.
2. Initializing the main queue Q. All the incoming processes are stored in the main queue.
3. The main queue is divided into five priority queues depending upon their priorities namely (q1, q2, q3, q4, q5). Highest priority q5 Lowest priority q1
4. Each priority queue is assigned a time quantum .
5. Any new request for resource (containing its priority value), goes to the main queue Q first.
6. The request is assigned to a priority queue with the matching priority.
7. A threshold value is decided initially, which is used to compare the time difference

If (time-diff>threshold)
SJF will be used
Else
WRR will be used

Synonyms

Processes will arrive along with their priority values.

Table 1

Process	Burst Time	Priority
P1	10	5
P2	6	4
P3	8	3
P4	4	3
P5	5	5

Q5

P1	P5

Time quantum for q5= floor((10+5)/2)= 8ms

Q4

P2

Time quantum for q4=6 ms

Q3

P3	P4

Time quantum for q3=floor((8+4)/2)= 6ms

Threshold value = 4

Round 1.

In q5 Difference in execution time of process P1 and P5 is 5 which is greater than 4(threshold value), thus WRR will be used and process P5 will be executed using the time quantum of q5 In q4 Process P2 will be executed using the time quantum of q4.

In q3 Difference in execution time of Process P3 and P4 is 4 which is equal to 4(threshold value), thus SJF will be used and process P4 will be executed.

P5

P5	P2	P4

5 11 15

Round 2.

In q5 Process P1 will be executed.

In q3 Process P3 will be executed.

P1	P3	P1	P3

15 23 29 31 33

The overall gantt chart is :

P5	P2	P4	P1	P3	P1	P3

0 5 11 15 23 29 31 33

Waiting time and turn around time:

Process	Waiting time	Waiting time
P1	21	31
P2	5	11
P3	25	33
P4	11	15
P5	0	5

Average Waiting time=(21+5+25+11+0)/5= 12.6

Average turn around time=(31+11+33+15+5)/5=19

IV. RESULTS AND ANALYSIS

Table 2

Parameters	WRR	SJF	MOSA
Data Centres	5	5	5
V.M	50	50	50
Response time(in seconds)	2187	3187	1187
Total cost	2818	2885	2808

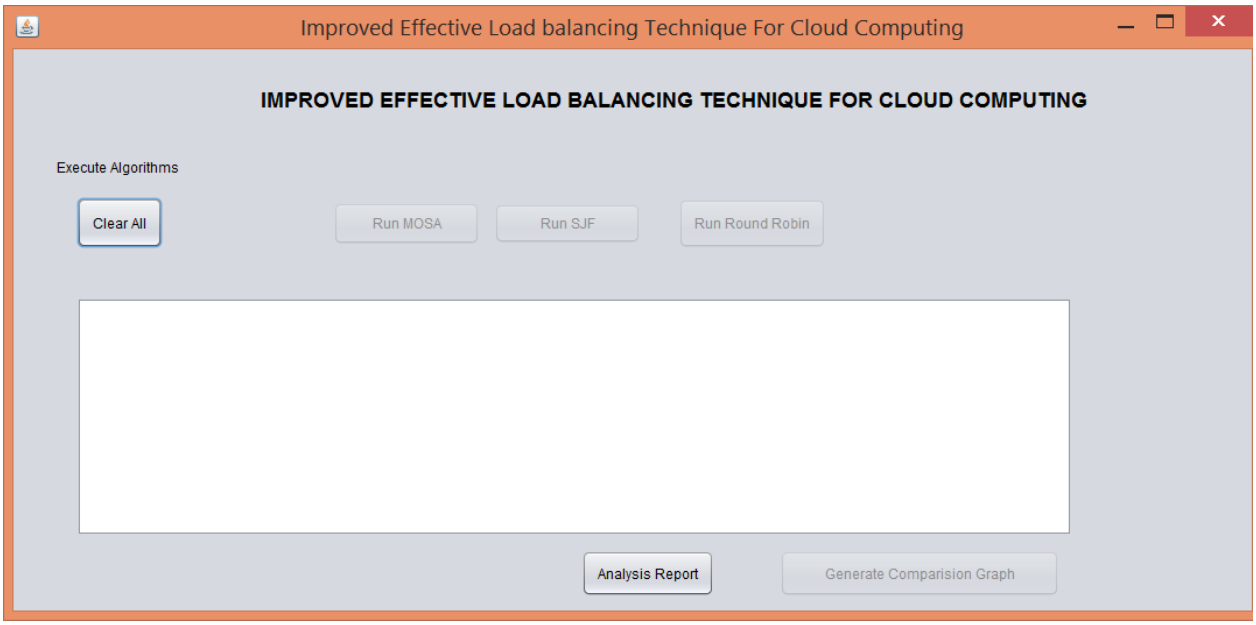

Figure 1

This is the user UI(user interface) screen. This is the first screen in the application. The screen where you can run algorithms and run the simulation.

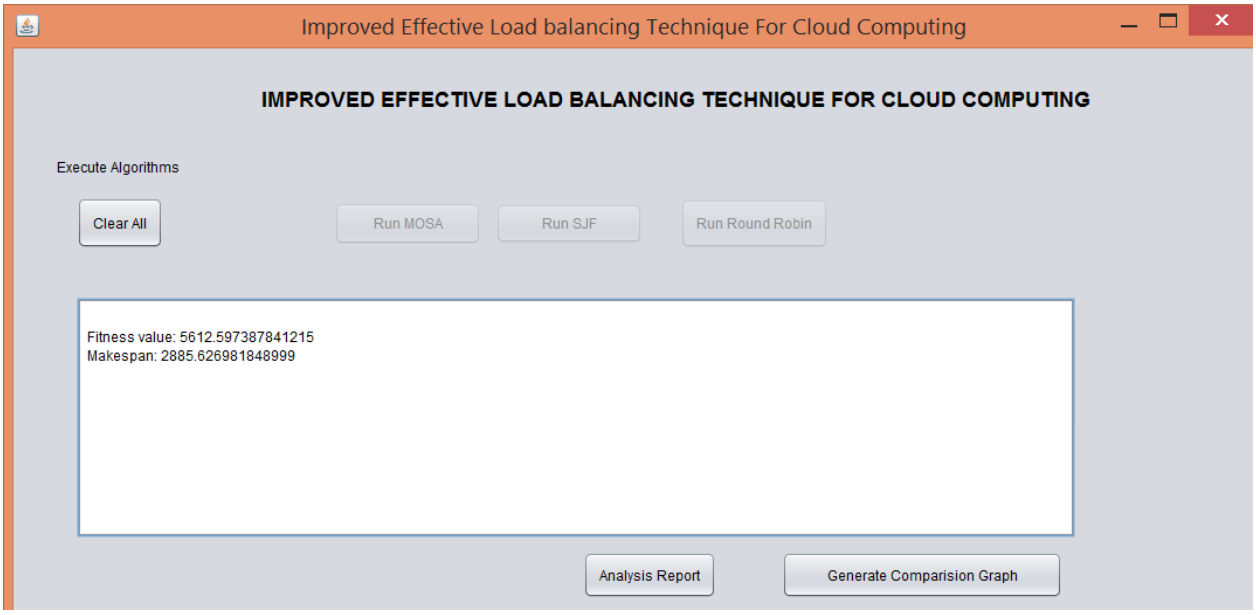

Figure 2

In this screen the text area it will display the fitness value and Makespan of the new proposed algorithm i.e MOSA.

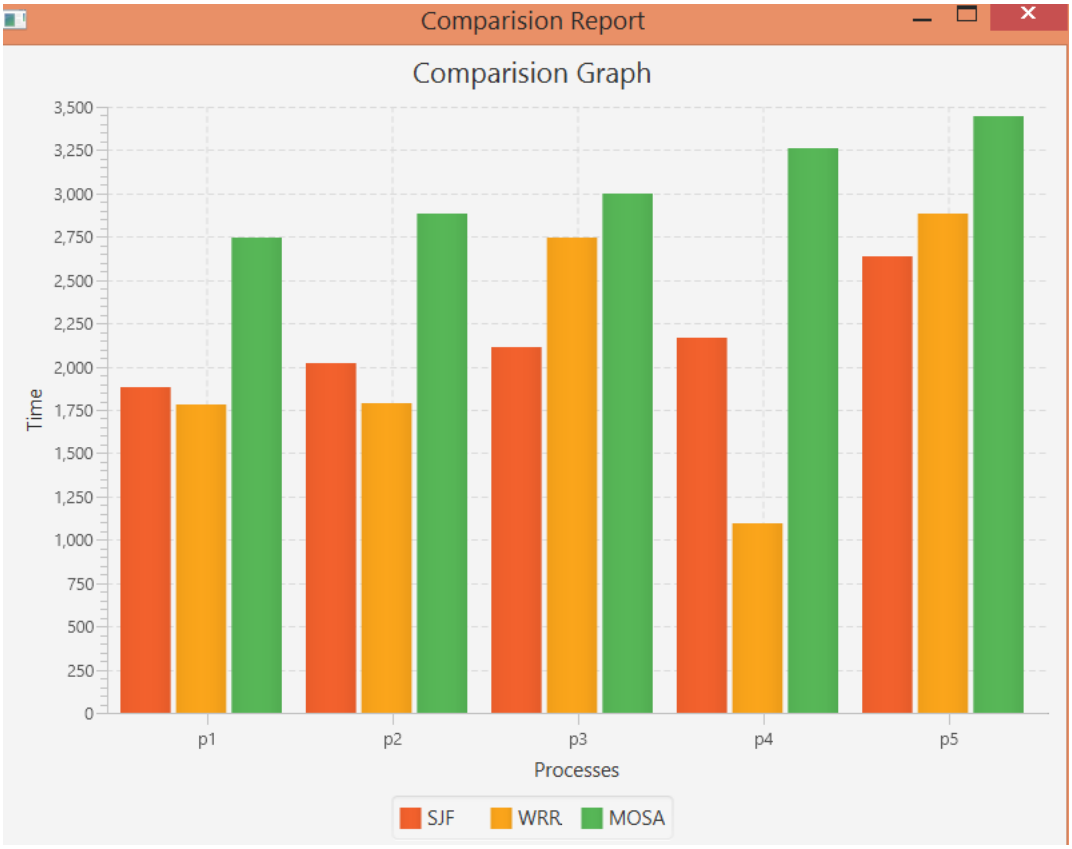

Figure 3

In this screen user can see the comparison of process execution in deferent algorithms used in this project.

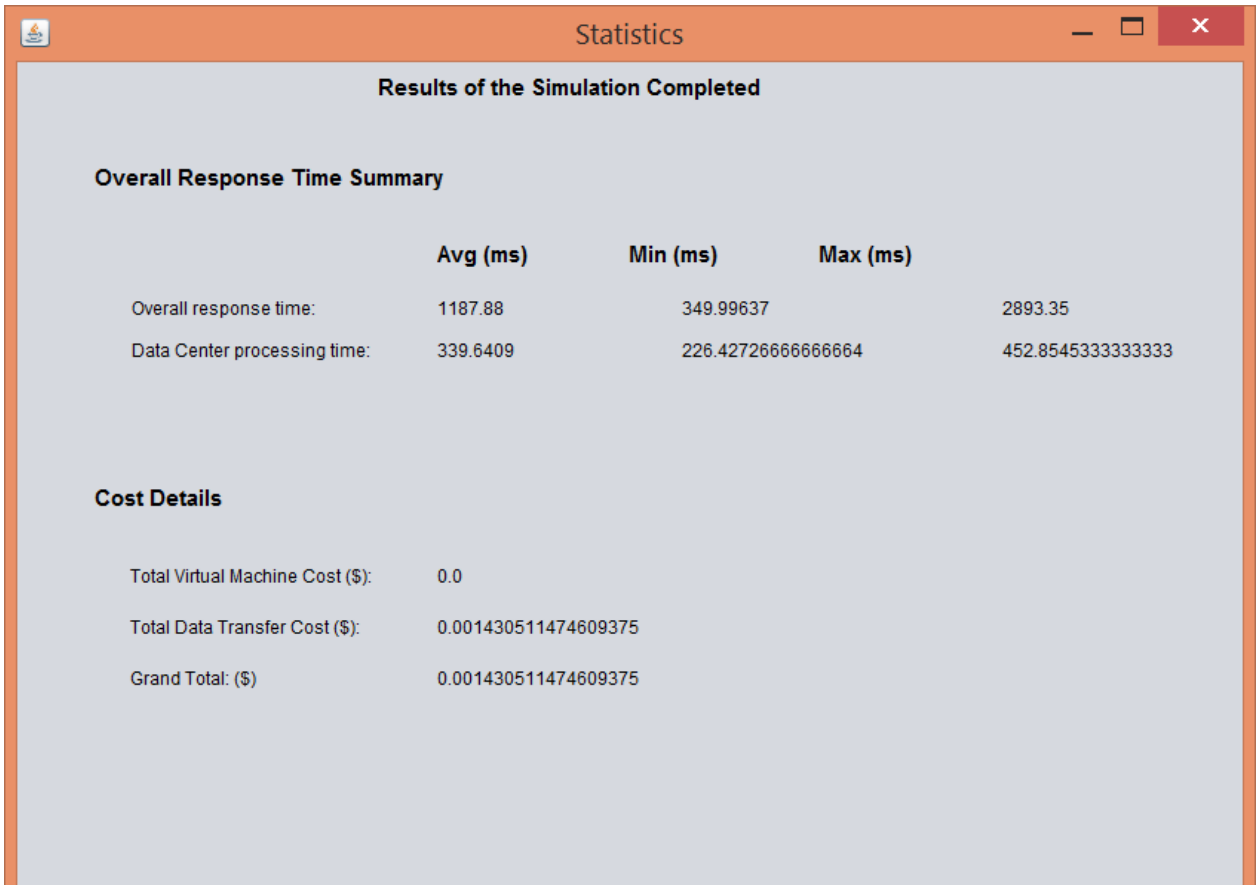

Figure 4

In this screen user can see the statistics/analysis about the new proposed algorithm that is MOSA

V. CONCLUSION

In this paper we have proposed the new scheduling algorithm using the Shortest Job First algorithm and Weighted Round Robin algorithms combined together as Modified optimal scheduling algorithm which handles the multiple user requests to optimize the load to the server and balancing the overload of the servers based on the requests. In the proposed system the two algorithms are carried out based on the time difference in the subsequent processes by comparing with the threshold value of the system. The proposed system consumes the less time compare to other scheduling algorithms.

VI. REFERENCES

[1]. Resource Management and Scheduling in Cloud Environment -Vignesh V, Sendhil Kumar KS, Jaisankar N.School of Computing Science and Engineering, VIT University Vellore, Tamil, Nadu, India – 632 014

[2]. Cloud Computing Bible by Barrie Sosinsky

[3]. "An Implementation of Load Balancing Policy for VirtualMachines Associated With a Data Center-"B.SANTHOSH KUMAR Assistant Professor, Department Of Computer Science, G.Pulla Reddy Engineering College. Kurnool-518007, Andhra Pradesh India. DR.LATHA PARTHIBAN Department of Computer Science, Community College Pondicherry University.

[4]. Abraham Silberschatz, Peter Baer Galvin, Greg Gangne;"Operating System Concepts", edition-7, 2005, John Wiley and Sons, INC..

[5]. "Optimized Scheduling Algorithm " LalitKishor Dinesh Goyal, Rajendra Singh ,Praveen Sharma Suresh GyanVihar University, Jaipur, Rajasthan, India. Proceedings published by International Journal of Computer Applications® (IJCA) , International Conference on Computer Communication and Networks CSICOMNET- 2011

[6]. Ajit Singh, PriyankaGoyal, SahilBatra," An Optimized Round Robin Scheduling Algorithm for CPU Scheduling", (IJCSE)International Journal on Computer Science and Engineering Vol. 02, No. 07, 2383-2385, 2010.

[7]. Mr. Sindhu M, Mr. Rajkamal R and Mr. Vigneshwaran P, "An Optimum Multilevel CPU Scheduling Algorithm". 978076954058-0/10 $26.00 © 2010 IEEE

[8]. GhalemBelalem, SamahBouamama and LarbiSekhri, "AnEffective Economic Management of Resources in Cloud Computing", March 2011, JOURNAL OF COMPUTERS, Vol. 6, No. 3, page no: 404-411.