Volume 3, Issue 7, July 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Zero Proof Authentication and Efficient Load Balancing Algorithm for Dynamic Cloud Environment

Nayandeep Sran*
CSE Department
PEC University of Technology
Chandigarh, India

Navdeep Kaur
Assistant Professor, IT Department
PEC University of Technology
Chandigarh, India

Abstract- Cloud Computing, an Internet-based development in which dynamically scalable and often virtualized resources are provided as a service over the Internet has become a new research topic. Cloud Computing is to utilize the computing resources (service nodes) on the network to facilitate the execution of complicated tasks that require large-scale computation. In this paper, we have developed a Load Balancer Algorithm that controls the flow of payload based on the safety thresholds, which may be static or dynamic in nature, depending on the available machines and bandwidth as well. We had analyzed the existing algorithms of Load Balancing such as Round Robin, Throttled, Equally Spread and Biased Random Sampling and we have proposed a new algorithm which will meliorate the existing Load Balancing Approach, by decreasing the overall requesting time and processing time as compared to the existing algorithms and hence will decrease the cost which is proved through rigorous simulation study. Our Proposed Algorithm will also provide security to the data in cloud during Load Balancing process by using Zero Proof Algorithm.

Keywords: Cloud Computing, Load Balancing, Virtualization, Zero Proof Algorithm, Cloud Analyst

I. Introduction

Cloud Computing means different things to different people. Cloud computing is an internet-based model of computing, where the shared information, software and resources are provided to computers and other devices upon demand. The prominent attributes are on-demand scalability of highly available and reliable pooled computing resources, secure access to metered services from nearly anywhere, and dislocation of data from inside to outside the organization. While aspects of these characteristics have been realized to a certain extent, Cloud Computing remains a work in progress. Cloud Computing is a service oriented architecture [1], which is provided via internet. Many companies are trying to implement and introduce clouds, due to its simple and flexible architecture. Although clouds are bifurcated in public, private, community and hybrid models but still problem of reliability may arise in these. Cloud models used virtualization technology; this technology helps in slicing a single data centre or high power server to act as multiple machines. It depends on the hardware configuration of the data centre or server in how may virtual machine they can be divided [4].

As the Cloud Computing is a new style of computing over internet. It has many advantages along with some crucial issues to be resolved in order to improve reliability of cloud environment. These issues [7] are related with:

- Security
- Efficient Load Balancing
- Performance Monitoring
- Consistent & Robust Service abstractions
- Resource Scheduling
- Scale and QoS management
- Interoperability & Portability
- Requires a constant & speedy Internet connection

Central to these issues lays the establishment of an effective Load Balancing Algorithm. The load can be CPU load, memory capacity, and delay or network load [2]. Load Balancing is the phenomenon of distributing the load among various nodes of a distributed system so as to improve both resource utilization and job response time whilst also avoiding a situation where some of the nodes are heavily loaded while other nodes are not doing any work. The important things to consider while developing such algorithm are: estimation of load, comparison of load, stability of different system, performance of system, interaction between the nodes, nature of work to be transferred, CPU utilization, selecting of nodes and many other ones [7]. This load considered can be in terms of CPU load, amount of memory used, delay or network load.

II. Literature Review

Due to the recent emergence of Cloud Computing research in this area is in the preliminary stage. Jiyin et.al,[3] have proposed a resource allocation mechanism with preemptable task execution which increases the utilization of clouds. They have proposed an adaptive resource allocation algorithm for cloud system with preemptable tasks but their approach does not pertain to cost optimization and time optimization. Load Balancing in Cloud Computing system[2] Ram Prasad Padhy, P Goutam Prasad Rao discussed basic concepts of Load Balancing and studied some existing Load Balancing Algorithms, which can be applied to clouds. In addition to that, the closed-form solutions for minimum measurement and reporting time for single level tree networks with different Load balancing strategies were also studied.

Authors in [8] have recommended Load Balancing in a three–level Cloud Computing network, by using a scheduling algorithm which combines the features of Opportunistic Load Balancing (OLB) and Load Balance Min-Min (LBMM) which can utilize better executing efficiency and maintain load balancing of the system. The objective is to select a node based for executing the complicated tasks that needs large-scale computation. The scheduling algorithm proposed in this paper is not dynamic and also there is an overhead involved in the selection of the node and only one factor is considered. However, in the cloud environment it is necessary to achieve optimal solution. Henceforth, it is imperative to consider all parameters which influence the realization of minimal completion time of tasks.

III. Purpose Of Research/Motivation

The random arrival of load in heterogeneous environment can cause some server to be heavily loaded while other server is idle or only lightly loaded. Equally load distributing improves performance by transferring load from heavily loaded server. Efficient scheduling and resource allocation is a critical characteristic of cloud computing based on which the performance of the system is estimated. Load Balancing is used to achieve a high user satisfaction and resource utilization ratio, making sure that no single node is overwhelmed, hence improving the overall performance of the system. The considered characteristics have an impact on cost optimization, which can be obtained by improved response time and processing time.

IV. Our Proposed Model

Flowchart of Proposed Algorithm is shown in Fig. 1. that depicts the detailed view of the algorithm. There are two types of policies for VM: VM consolidation and VM migration. By default VM works on consolidation policy in which the Cloud Sim dynamically assigns VM's to each task. In our thesis we are working on VM migration policy by giving priority to VM's on the basis of resources available to it. Firstly, the algorithm will check that whether the CPU utilization of VM(Virtual Machine) is equal to, greater than or less than 80%. We had chosen that value to prevent the machine fron being overloaded.

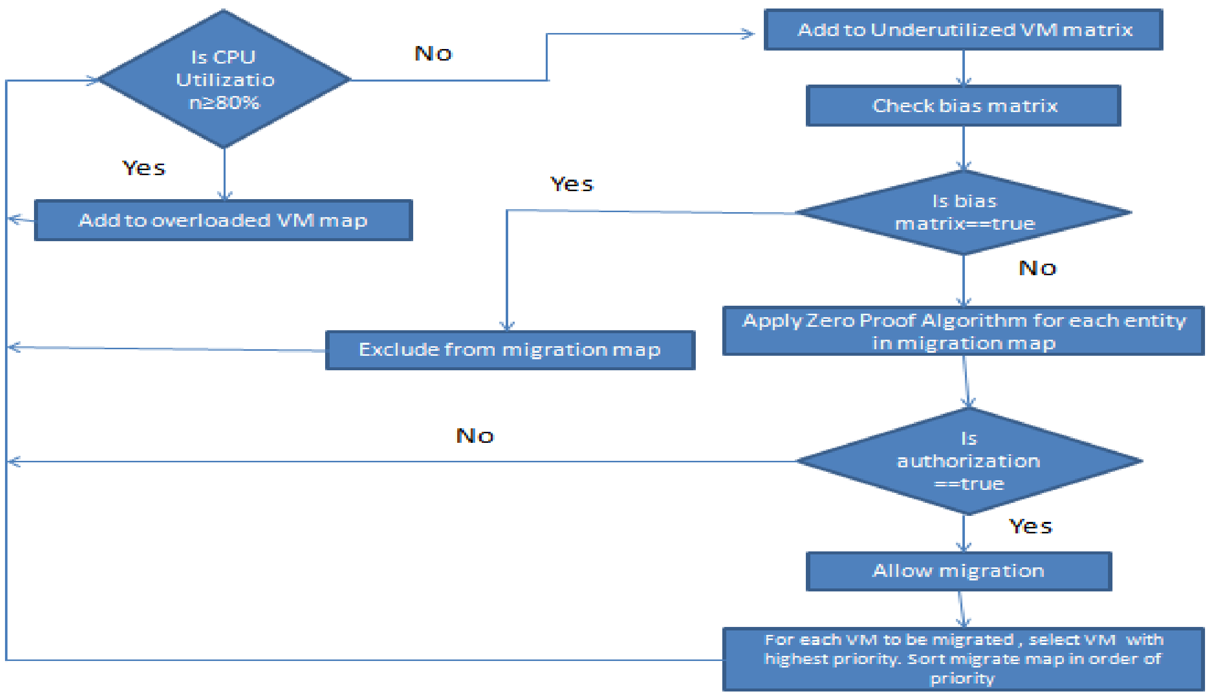

Fig. 1 Flowchart for Proposed Algorithm

If the utilization of VM is greater than or equal to 80% then that machine is added to overloaded VM matrix. If CPU utilization is less than 80% then that machine is added to underutilized VM matrix. Then the algorithm will check the bias matrix; which is used to maintain the list of VM's, which are reserved for particular task. If the value of bias matrix== true, then that machine is excluded from the migration map, means the machine will not participate.

The next step of algorithm is to apply Zero proof Algorithm for each entity in migration map. The working of Zero proof Algorithm is explained in Fig. 2.

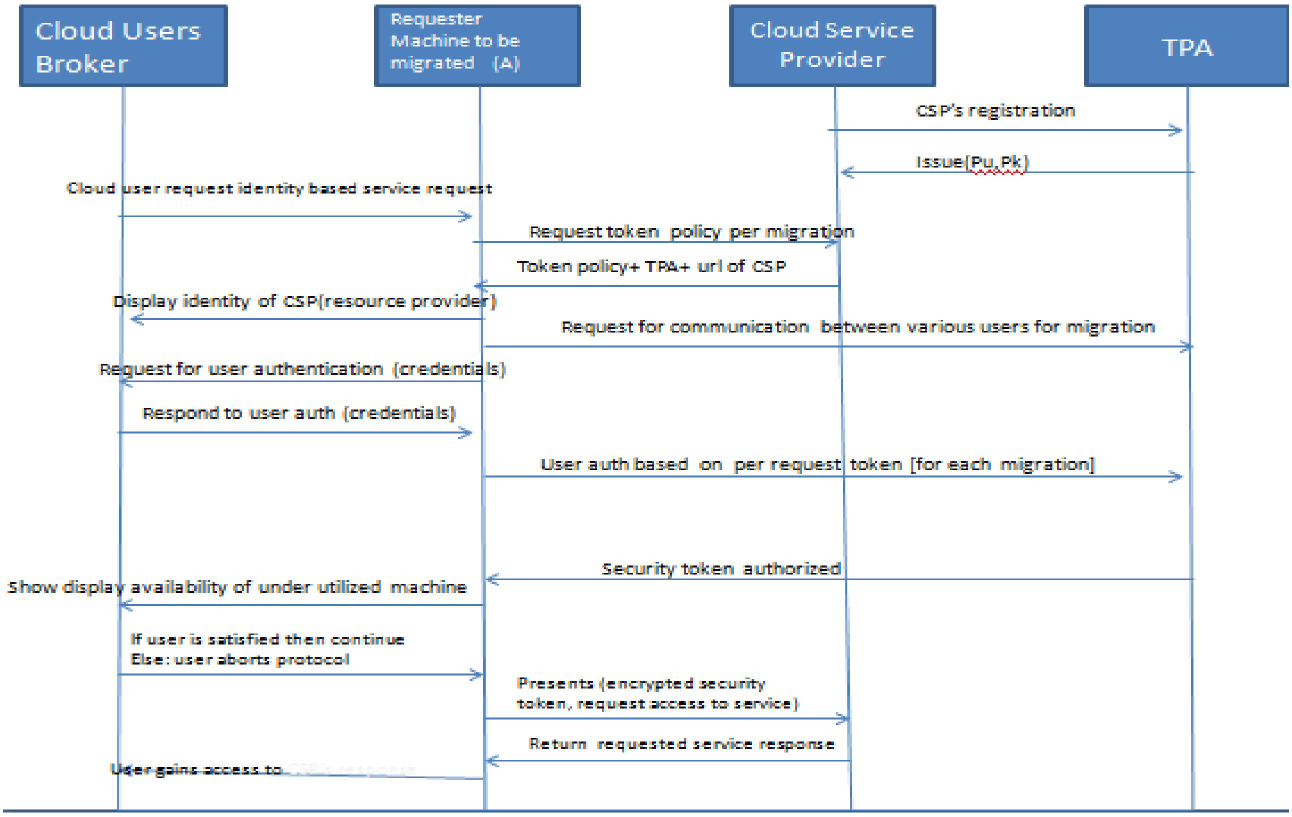

Fig. 2 Zero Proof Algorithm

When the process of Zero Proof Algorithm gets completed then the algorithm will check that if authorization is granted or not. If the Zero Proof Algorithm provides authorization only then Proposed Algorithm will allow migration. And the final step of algorithm is to give priority to VM, for each VM to be migrated, select VM with highest priority and sort the map in the order of priority.

V. Simulation Setup And Performance Analysis

Experiments had been performed using the Cloud Analyst simulation which helps in testing the outputs with respects to the virtual machine. Cloud Analyst is equipped with an easy to use graphical user interface as shown in Fig. 3. that enables users to set up experiments quickly and easily.

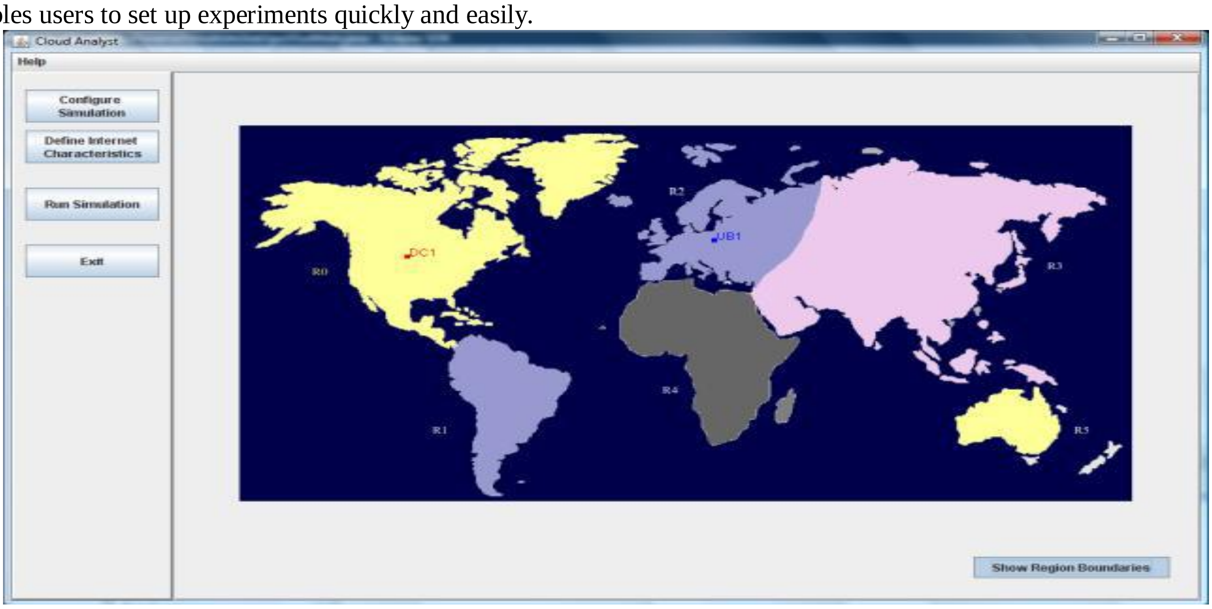

Fig. 3 Cloud Analyst GUI [5]

In this simulation we have took seven data centers and eighteen user bases. DC1, DC2 & DC7 use five VM's, while DC3, DC4 & DC6 use fifty VM's. For this configuration we had run the simulation five times for each of the Load Balancing algorithms (Round Robin, Equally Spread, Throttled, Biased Random Sampling and our Proposed Algorithm). The time for simulation is eight hours. Cloud Analyst GUI for running this configuration is shown in Fig. 4.

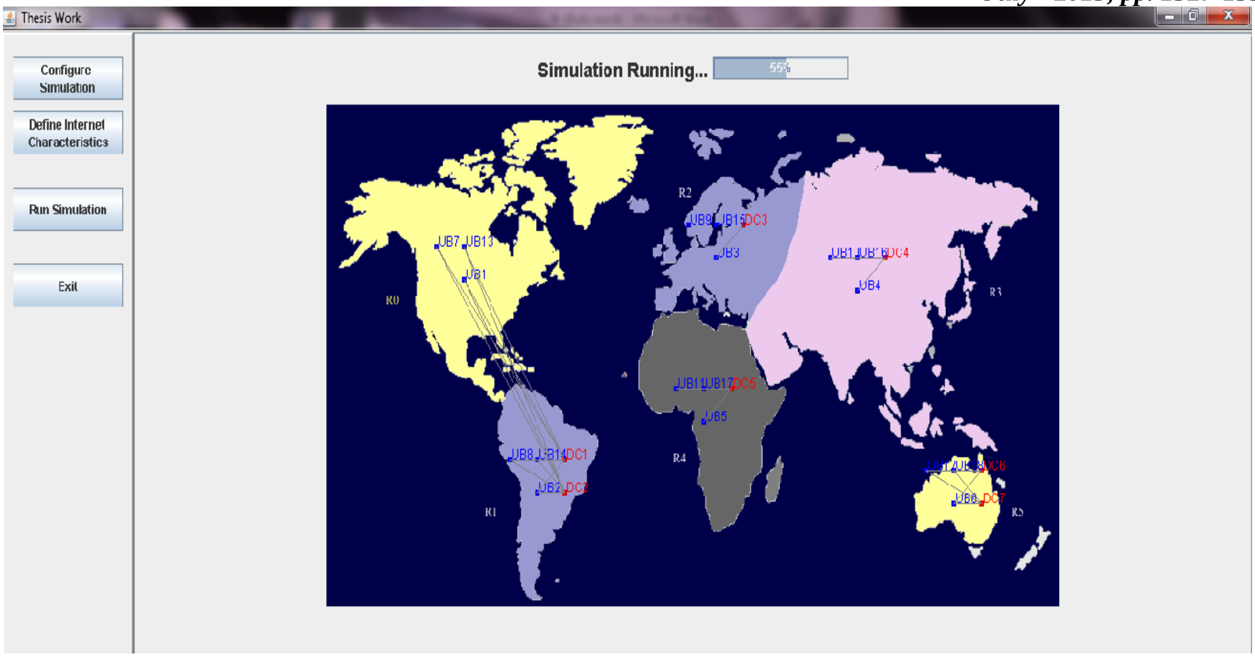

Fig.4. Cloud Analyst GUI for 7 data centers & 18 user bases (running simulation)

VI. Performance Analysis Results

In Fig. 4 we have shown user base and data center configuration showing positions of data centers and user base in different regions. Figure 4 also shows the selection of different VM's. In Fig. 5 we have shown the output window for this configuration. Here we had shown the execution of VM's, which are shown by red circles in Fig. 6.

Name	Region	Requests per User per Hr	Data Size per Request (bytes)	Peak Hours Start (GMT)	Peak Hours End (GMT)	Avg Peak Users	Avg Off-Peak Users
UB1	0	60	100	3	9	1000	100
UB2	1	60	100	3	9	1000	100
UB3	2	60	100	3	9	1000	100
UB4	3	60	100	3	9	1000	100
UB5	4	60	100	3	9	1000	100
UB6	5	60	100	3	9	1000	100
UB7	0	60	100	3	9	1000	100
UB8	1	60	100	3	9	1000	100
UB9	2	60	100	3	9	1000	100
UB10	3	60	100	3	9	1000	100
UB11	4	60	100	3	9	1000	100
UB12	5	60	100	3	9	1000	100
UB13	0	60	100	3	9	1000	100
UB14	1	60	100	3	9	1000	100
UB15	2	60	100	3	9	1000	100
UB16	3	60	100	3	9	1000	100
UB17	4	60	100	3	9	1000	100
UB18	5	60	100	3	9	1000	100

Data Center	#VMs	Image Size	Memory	BW
DC1	5	10000	512	1000
DC2	5	10000	512	1000
DC3	50	10000	512	1000
DC4	50	10000	512	1000
DC5	50	10000	512	1000
DC6	50	10000	512	1000
DC7	5	10000	512	1000

Fig.5. Showing regions & VM's for user base and data center configuration for Proposed Algorithm

With the help of rectangle we had shown the priority of VM in which they are executed, VM 25 is executed before 24 for DC4 because priority is more.

Results of cost for the Proposed Algorithm is Shown in Fig. 7. Formula used for cost calculation is:

$$\text{Total cost}=\sum_{i=1}^{24} x_1+\sum_{i=1}^{24} x_2+\sum_{i=1}^{24} x_3+\sum_{i=1}^{24} x_4$$

Where, x_1 = cost of process, x_2 = cost of using memory by resources, x_3 = cost per storage, x_4= cost per bandwidth

Fig.6. Showing output window for order of running of VM's for Proposed Algorithm

Fig.7. Showing cost of Proposed Algorithm for eight hours simulation

The comparison of total cost for second configuration with Proposed Algorithm is also shown in Fig. 8. In our current research work, we have developed a Load Balancer Algorithm that controls the flow of payload based on the safety thresholds, which may be static or dynamic in nature, depending upon the available machine, bandwidth as well. In case of round robin, the VM simply take turns to balance the load, therefore, the total cost for round robin after running simulation of eight hours is 232.17 $ and for Random Bias, where Load Balancing is done based on understanding the migration policy and on the load safety parameter and on which machine's having more bias or weightage or preference. So in case of Random Bias cost for the simulation is 229.18$. Throttled Algorithm controls the flow of load of data in such a manner that whenever overload or adversities crosses a threshold, it need to move the work load to next available underutilized machines, so its cost for is 231.34$. Since the main factor of cost depends upon the bandwidth and the quantum of load, and the times the request to reallocate the resources again and again. It is apparent from analysis of Load Balancing Algorithm that by keeping the bandwidth constant for all algorithms and quantum of workload for our Proposed Algorithm we were able to reduce the average response time as the flow of work is now sequenced based on priority which is calculated based on the metric score for that reallocation request by Load Balancer, therefore the average response time had less load time and cost for provisioning etc.

VII. Conclusion

In this thesis we proposed a new Load Balancing Algorithm which efficiently balances load by reducing the overall requesting time. It is clear from the results that the proposed method in thesis is able to balance the load to a greater extent than the analyzed algorithms (Round Robin, Equally Spread, Throttled and Biased Random Sampling). A comparative study is done in this thesis with the Proposed Algorithm. This Load Balancing algorithm aims at providing dynamic, on- demand, balance of resources available to the resources required to accomplish the task. The Load Balancing policy comprises of two major parts which include: 1.) Default allocation of resources and 2.)How resources need to be reallocated when there is either adversity or overload of work. Since, this algorithm involves reallocation of resources involving VM and data centers. Therefore, every time whenever reallocation occur reauthentication of the resource allocation is also conducted along with reallocation of Load Balancing. So, this algorithm also provides better security, by focusing on the concept of not disclosing the personal details of the user to cloud provider i.e. identity based security. It is shown in our results that the algorithm is better because it reduces the requesting time and cost as compared to other algorithms (Round Robin, Throttled, Equally Spread and Biased Random Sampling). However, the Proposed Algorithm does not handle certain cases like the failure of nodes but it works on various parameters such as CPU utilization, data centers in different regions, response time and processing time, which is the biggest vantage of our Proposed Algorithm.

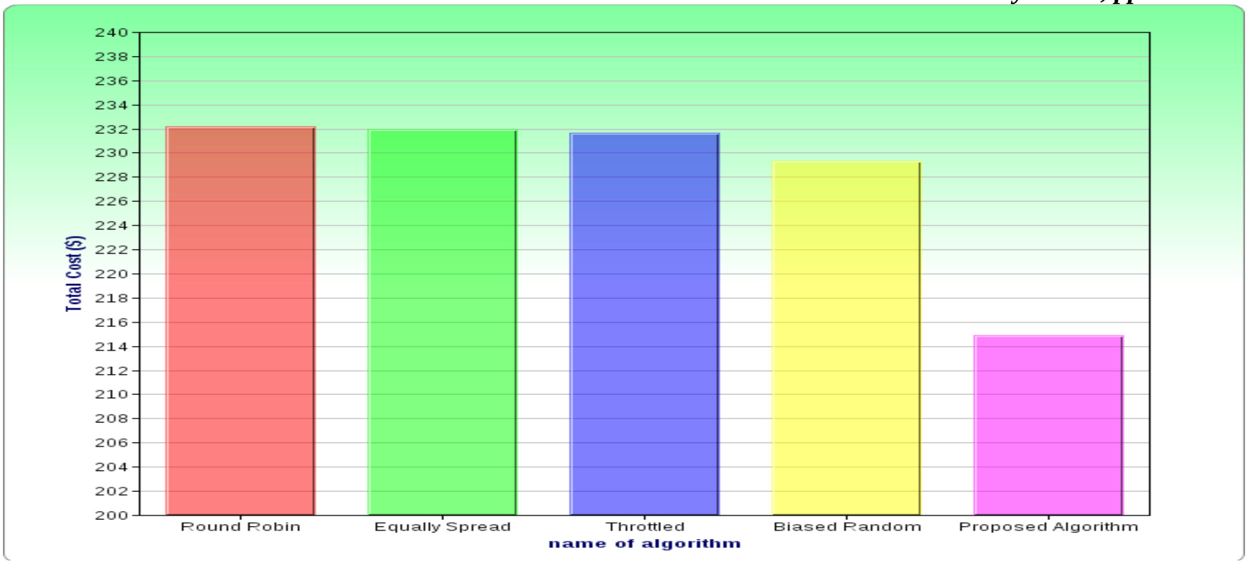

Fig.8. Comparison of total cost for second configuration with our Proposed Algorithm

VIII. Future Work

This scheduling algorithm can be extended to suit other cloud platforms also. Here, the load factor above which the highest priority node is selected is kept constant (which in our case is 80%). Further extension to this algorithm can be done by varying the maximum load factor, above which the priority of a node decreases, dynamically by setting it to an optimum value based on the present conditions or by further enhancement in algorithm other than CPU utilization. However a great amount of effort has been made to achieve this proposed work, but there is also lot of work to be done on Cloud that can elevate Cloud Computing at its apex by enhancing more in Load Balancing by biasing the random sampling toward specific nodes. Hence, selection can be based on a predefined criterion (e.g. computing power or communication latency) rather than selecting the last node in the walk. And security can be further increased by adding more cryptographic mechanisms. By better implementation of architecture of Cloud Computing, cloud providers can give better cloud services to customers at very low prices. Concept of Green Cloud can also be enhanced by cloud providers which will also save environment by saving lots of energy which is being used by cloud users.

Acknowledgement

I would like to express my gratitude to those who have made this research possible. First and foremost, I offer my sincerest gratitude to my supervisor, Assistant Professor Mrs. Navdeep Kaur, who supported me throughout my research with her patience and guidance whilst allowing me the room to work in my own way. I would also like to express my sincere thanks to my colleagues and friends. Last but not the least, I would like to dedicate this research to my parents and the people I love and who love me. Without their continuous supports, I cannot complete the research alone.

References

[1] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, *Above the Clouds: A Berkeley View of Cloud Computing*, University of California, Berkeley, Technical Report No., UCB/EECS, February 10, 2009.

[2] A.Khiyaita and M.Zbakh, *Load Balancing Cloud Computing: State of Art* IEEE, pp. 106-109, May 2012.

[3] Ram Prasad Padhy and P Goutam Prasad Rao *Load Balancing in Cloud Computing Systems* B.Tech Dessertion, National Institute of Technology, CSE dept., Rourkela, May 2011.

[4] Jiyin Li, Meikang Qiu, Jain-Wei Niu, YuChen, Zhong Ming *Adaptive Resource Allocation for Preeemptable Jobs in Cloud Systems*. IEEE International Conference on Intelligent Systems Design and Applications, pp. 31-36, 2010.

[5] Bhathiya Wickremasinghe, Rodrigo N. Calheiros, and Rajkumar Buyya *CloudAnalyst: A CloudSim-based Visual Modeller for Analyzing Cloud Computing Environments and Applications* from http://www.cloudbus.org/cloudsim/

[6] Martin Randles, David Lamb and A. Taleb-Bendiab *A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing* IEEE, 24th International Conference on Advanced Information Networking and Applications Workshops, pp. 551-556, May2010.

[7] Shu-Ching Wang, Kuo-Qin Yan, Wen-Pin Liao, Shun-Sheng Wang, *Towards a Load Balancing in a Three-level Cloud Computing Network*, 2010 IEEE, pp. 108-113.

