Improved Throttling Load Balancing Algorithm With Respect To Computing Cost and Throughput For Cloud Based Requests

Nandkishore H Patel[1], Jasmine Jha[2]

[1]P.G. Student,[2] Assistant Professor

[1,2]Department Of Computer Engineering

[1,2]L.J Institute Of Engineering And Technology, Ahmedabad, Gujarat, India

Abstract

Cloud computing is a developing computing paradigm that has inclined every other entity in the digital industry, it may be government sector or the personal sector. Taking into account the mounting significance of cloud, finding new ways to advance cloud services is an area of concern and research center. Usually clouds have powerful data centers to handle large number of users. Cloud is a platform providing dynamic pool of resources and virtualization. To properly manage the resources of the service contributor, load balancing is required for the jobs that are submitted to the service contributor. Load balancing is a tactic to share out workload across many virtual machines in a Server over the network to achieve optimal resource consumption, least data processing time, least average response time, and avoid overload. In the present work, a local throttling load balancing approach is proposed for distributing of incoming jobs uniformly among the servers or virtual machines. Further, the performance is analyzed using CloudAnalyst simulator and compared with existing Throttled algorithms.

Keyword: *Load balancing, Cloud Computing, Virtual machine, CloudAnalyst*

1. INTRODUCTION

The present computing time is that of cloud computing or cloud. In the remaining sections of the paper the terms cloud computing is also been used as cloud. Irrespective of the developments in the IT business, the one growing prototype that has influenced every other entity in the digital industry, whether it is in the government sector or the private sector, is cloud computing. "The USA administration's budget for the 2013 Fiscal Year (FY) is clear that the deployment of cloud computing solutions will remain in precedence for U.S. Government Departments and Agencies moving ahead". Now management agencies are also supporting the growth of the cloud computing. The newest trend shows that a growing number of small and medium scale businesses are shifting to cloud. The number of service providers are raising and the cost of services are declining. Considering the rising significance of cloud, discovering new ways to advance cloud services is an area of concern and research focus.[1]

Cloud computing possess distributed technologies to satisfy a variety of applications and user needs . Sharing resources, software, information via internet are the main interest of cloud computing with an aim to reduced capital and operational cost, better performance in terms of response time and data processing time, maintain the system constancy and to accommodate future adaptation in the system.So there are various technical challenges

that needs to be addressed like Virtual machine relocation, server consolidation, fault tolerance, high availability and scalability but central issue is the load balancing , it is the mechanism of spreading the load among various nodes of a distributed system to improve both resource deployment and job response time while also avoiding a situation where some of the nodes are having huge amount of load while other nodes are doing nothing or idle with very little work. It also ensures that all the processor in the system or each node in the network does approximately the equal amount of work at any instant of time.[5]

2. RELATED WORK

In this section, we briefly summarize the load balancing algorithms used in the cloud computing environment. The main focus is on the assignment of all incoming jobs among the available virtual machines with minimal response time. Load balancing is defined as a process of making effective resource utilization by reassigning the total load to the individual nodes of the collective system and thereby minimizing the response time of the job. Brototi Mondal et al. have developed the Stochastic Hill Climbing algorithm for balancing the load. Stochastic Hill Climbing is one of the incomplete approaches for solving such optimization problems. A stochastic and Local Optimization algorithm is simply a loop that continuously moves in the direction of increasing value, which is uphill. It stops when it reaches the peak value where no neighbor has a higher value. This variant chooses at random from among the uphill moves and the probability of selection can vary with the steepness of the uphill move. Thus it maps assignment of values to a set of other values by making only minor changes to the original value. The best element of the set is made the next assignment. This basic operation is repeated until either a solution is found or a stopping criterion is reached. The results are quite encouraging when compared to Round Robin and FCFS algorithms.[2]

3. EXISTING LOAD BALANCING ALGORITHMS FOR CLOUD COMPUTING PROPOSED SYSTEM

To distribute workload among multiple network links, to achieve maximum throughput, minimize response time and to avoid overloading. We use three algorithms to distribute the load. And check the performance time and cost.[5]

A. Round Robin Algorithm(RR): It is the simplest algorithm that uses the concept of time quantum or slices Here the time is divided into multiple slices and each node is given a particular time quantum or time interval and in this quantum the node will perform its operations. The resources are assigned to the customer by the service provider on the basis of this time quantum. In Round Robin Scheduling the time quantum plays a very important role for scheduling, since if time quantum is extremely large then Round Robin Scheduling Algorithm is same as the FCFS Scheduling.[8] If the time quantum is too small then Round Robin Scheduling is called as Processor Sharing Algorithm and number of context switches is very high. It selects the load on random basis and leads to the situation where some nodes are heavily loaded and some are evenly loaded. However the algorithm is extremely simple but there is an additional load on the scheduler to decide the size of quantum and it has longer average waiting time, elevated context switches higher turnaround time and low throughput.[4]

B. Equally Spread Current Execution Algorithm (ESCE): In this technique load balancer makes effort to preserve equal load to all the virtual machines connected with the data centre. This load balancer maintains an index table of Virtual machines as well as number of requests currently assigned to the Virtual Machine (VM). If the request comes from the data centre to allocate the new VM, it scans the index table for least loaded VM.[7] In case there are more than one VM is found than first identified VM is selected for handling the request of the client/node, the load balancer also returns the VM id to the data centre controller. The data centre communicates the request to the VM identified by that id. The data centre revises the index table by increasing the allocation count of identified VM. When VM completes the assigned task, a request is communicated to data centre which is further notified by the load balancer. The load balancer again revises

the index table by decreasing the allocation count for identified VM by one but there is an additional computation overhead to scan the queue again and again.[3]

C. Throttled Load Balancing Algorithm(TLB): In this algorithm the load balancer maintains an index table of virtual machines as well as their states (Available or Busy). The client/server first makes a request to data centre to find a suitable virtual machine (VM) to perform the recommended job. The data centre queries the load balancer for allocation of the VM. The load balancer scans the index table from top until the first available VM is found or the index table is scanned fully. If the VM is found, the VM id is send to the data centre. The data centre communicates the request to the VM identified by the id. Further, the data centre acknowledges the load balancer of the new allocation and the data centre revises the index table accordingly. While processing the request of client, if appropriate VM is not found, the load balancer returns -1 to the data centre . The data centre queues the request with it. When the VM completes the due task, a request is acknowledged to data centre, which is further apprised to load balancer to de- allocate the same VM whose id is already communicated. The total execution time is estimated in three phases. In the first phase the formation of the virtual machines and they will be idle waiting for the scheduler to schedule the jobs in the queue, once jobs are allocated, the virtual machines in the cloud will start processing, which is the second phase, and finally in the third phase the cleanup or the destruction of the virtual machines. The throughput of the computing model can be estimated as the total number of jobs executed within a time span without considering the virtual machine formation time and destruction time.[1]

4. PROPOSED WORK

In modified throttled algorithm hear the VM's are used in serial manner, Instead of that I can try to implement it in parallel formation so that response time can be improved and by changing the data structure of the index table we can get quick allocation of available VMs. And the sorting of the assigned load to VMs will be done loadwise (the VM with less load will be on the top and greater load will be at bottom). After that the VMs are compared with the throttled capacity so afterwads we get the VMs with the greater remaining space at the top. So the next allocation of the request is directly assigned to the VM at the top.And accordingly the task allocation and task completion table is maintained.

The algorithm of the proposed system is as follows:

Input : No. of task; T1, T2, T3,….., Tn.

Initialize virtual machine VM.

Time; t1, t2, t3,…,tn

1: Monitor the initialization time.

2: Arrange the VM in parallel .

3:This throttled algorithm identifies VM by its load count. Sort the VM loadwise.

4: Pass the VM list to the load balancer.

5: Compare VM with throttled capacity, then balancer will allocate the task.

6: Calculate the throughput;

Throughput = no. of task completed / no. of total task supplied

= f(Tcomp) / Tn

7: Maintain the task allocation and completion table.

8: Calculate the cost.

The flowchart for the proposed system is shown in figure 1.

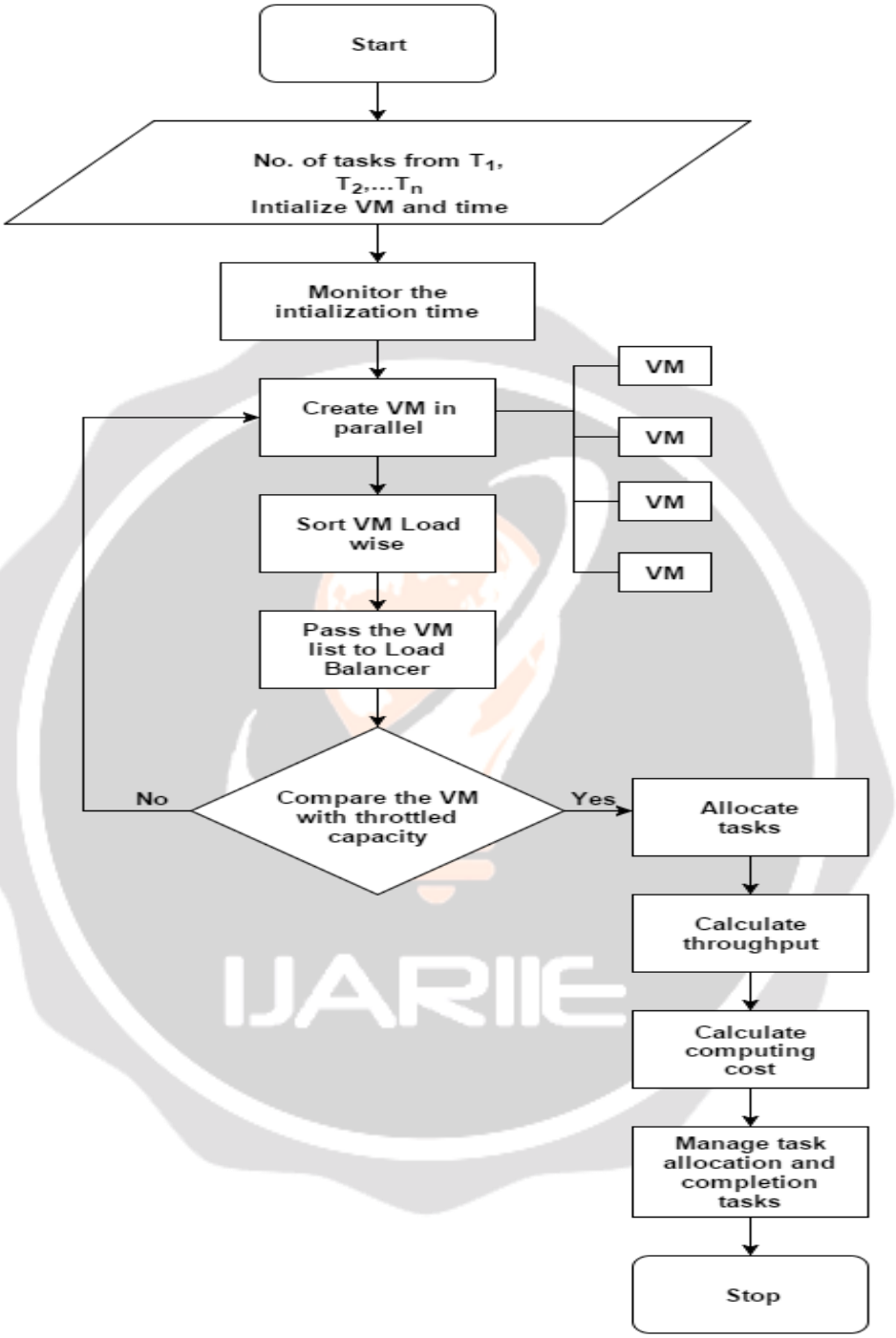

Figure 1 Proposed System Flowchart.

5. EXPERIMENTAL RESULT

For generating results, the Cloudsim tool is to be installed. And for simulation the cloud analyst is used. Here we compare the proposed system with existing throttled algorithm.

For response time,

Response Time		
User Base	Existing Throttle Algorithm	Proposed Throttle Algorithm
UB1	61.127	60.611
UB2	62.623	62.62
UB3	61.364	61.364
UB4	62.872	60.87
UB5	61.355	61.12
UB6	61.872	61.12
UB7	61.122	60.619
UB8	61.124	61.127
UB9	60.623	60.619
UB10	61.127	61.127

Table 1

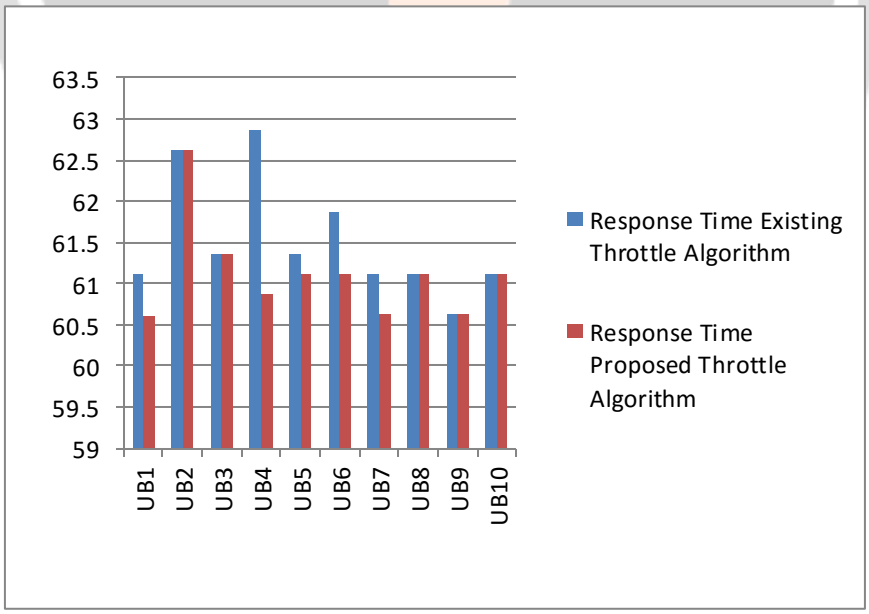

Figure 2: comparison of the existing and the proposed system.

Response Time		
User Base	Existing HoneyBee Algorithm	Proposed HoneyBee Algorithm
UB1	61.127	60.611
UB2	62.623	62.62
UB3	61.364	61.364
UB4	62.872	60.87
UB5	61.355	61.12
UB6	61.872	61.12
UB7	61.122	60.619
UB8	61.124	61.127
UB9	60.623	60.619
UB10	61.127	61.127

Table 2

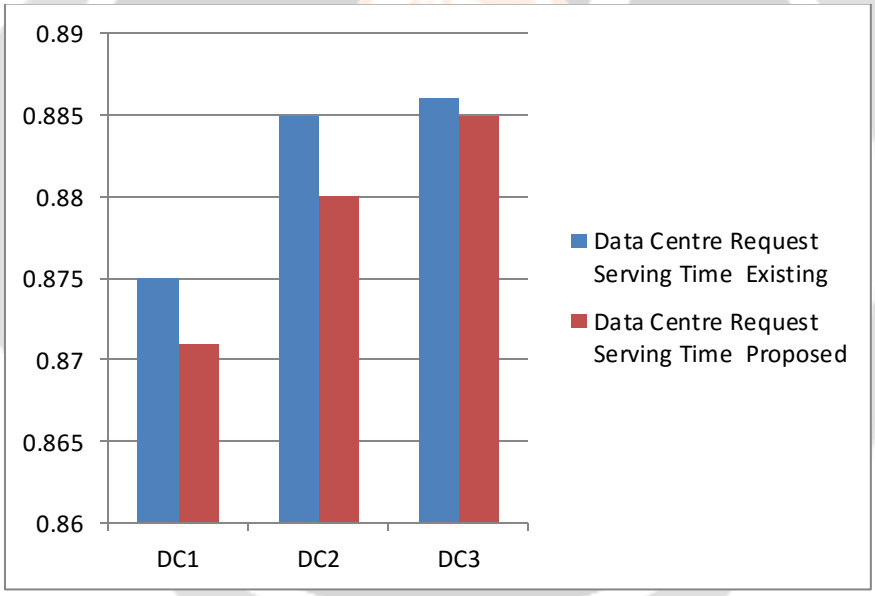

Figure 3: The comparison for data centre serving time.

6. CONCLUSION

In this paper, throttled algorithm for load balancing is used. Energy efficient cloud computing technology is very recent technology, and there been many areas which is still not explore fully as compared to other technologies, and thus there is a huge prospective for growth and new innovations. The implementation results show that the proposed system shows better results than the existing one. The parameters compared were response time and the data centre serving time.

7. REFERENCES

[1] Hamid Shoja, Hossein Nahid, Reza Azizi,"A COMPARATIVE SURVEY ON LOAD BALANCING ALGORITHMS IN CLOUD COMPUTING",IEEE,5th ICCCNT 2014

[2] Shridhar G. Domanal, G. Ram Mohana Reddy, "Load Balancing in Cloud Computing Using Modified Throttled Algorithm", IEEE, International conference. CCEM 2013.

[3] Priyadarashini Adyasha Pattanaik, Sharmistha Roy and Prasant Kumar Pattnaik," Performance Study of Some Dynamic Load Balancing Algorithms in Cloud Computing Environment", International conference on signal processing(SPIN),IEEE, April 2015

[4] Lee Badger,TimGrance, Robert Patt-Corner, JeftVoas, "Cloud Computing Synopsis and Recommendations", National Institute of Standards and Technology, Special Publication 800-146, , May 2012.

[5] Gulshan Soni , Mala Kalra "A Novel Approach for Load Balancing in Cloud Data Center" In Advance Computing Conference (IACC), 2014 IEEE International, 2014, DOI:10.1109/IAdCC.2014.6779427.

[6] B. P. Rima, E. Choi, and I. Lumb, "A Taxonomy and Survey of Cloud Computing Systems", Proceedings of 5th IEEE International Joint Conference on INC, IMS and IDC, Seoul, Korea, August 2009, pages 44-5l.

[7] Sandeep Sharma, Sarabjeet Singh, Meenaksshi Sharma, "Performance Analysis of Load Balancing Algorithms", World Academy of Science, Engineering and Technology, 2008.

[8] Hisao Kameda, EL-Zoghdy Said Fathyy and Inhwan Ryuz Jie Lix, "A Performance Comparison of Dynamic vs Static Load Balancing Policies in a Mainframe, Personal Computer Network Model",Proceedings Of The 39th IEEE Conference on Decision &Control,2000.

[9] Nidhi Jain Kansal, Inderveer Chana, "Cloud Load Balancing Techniques: A Step Towards Green Computing", IJCSI, Vol. 9, Issue1,January2012.