© 2016 IJSRSET | Volume 2 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

A Shared Approach of Dynamic Load Balancing in Cloud Computing

Snehlata Mishra, Ritu Tondon

TRUBA College of Engineering and Technology, Indore-Rao Bypass Road, India

ABSTRACT

Load balancing play import role in cloud computing in terms of performance. Cloud computing performance more efficient and accurate with good load balancing for user satisfaction. Functionality of load balancing is divided into two function first will be allocation of resources and second provisioning of resources along with task scheduling among distributed system. Many load-balancing algorithms are used for balancing load of cloud computing such as , FCFS ,Round Robin ,Throttled ,Equally spread current execution etc. Each algorithm has some disadvantage. We proposed new hybrid algorithm that is combination of Throttled and equally spread current execution algorithm. Proposed algorithm overcomes the drawback of throttled and equally spread current execution algorithm and improve efficiency of load balancing in cloud computing. We implement hybrid algorithm with help cloud sim simulator using java language.

Keywords: Cloud computing, load balancing, Task Scheduling, Round Robin, Throttled, Equally spread current execution.

I. INTRODUCTION

Cloud computing comes throughout focus development of grid computing, virtualization in addition as net technologies. Cloud computing is typically the world wide net based mostly computing that presents infrastructure as service (IaaS), platform as service (PaaS), package as Service (SaaS). Throughout SaaS, package form is typically created shown through the cloud supplier. PaaS an honest application development platform for the developer to produce a net based mostly application. Within IaaS computing infrastructure may be sent to be a facilitate towards the requester. In your current form associated with Virtual Machine (VM).These model typically square measure developed seeable from an honest subscription basis utilizing value equally you-use model to be able to customers, regardless concerning their location. Cloud Computing still below within their development stage and conjointly has quite an few issue additionally to challenges out of a many queries in cloud programming plays terribly vital role with policies to be able to management your order involving operates for you to presumably be vital role within determinative your current effective execution. programming refers for the set connected performed by associate computing system. There has been totally different individuals related to programming algorithmic program existing throughout distributed system, beside job programming can be single of them. the main advantage-involving job programming algorithmic program is going to be in order to accomplish an honest high performance computing and conjointly the simplest method outturn. Programming manages handiness involving processor memory and smart programming policy provides most utilization of resource.

The various objectives of optimization criteria can be [2]:

- **CPU Utilization:** The total percentage of time for which CPU was utilized or used i.e. was not idle.
- **Throughput:** Total no. of tasks executed (or requests served) per unit time.
- **Response Time:** The time spent by a request in the waiting queue till it gets the first time to use the CPU.
- **Waiting Time:** The total time spent by the request waiting in the ready queue after the first response from CPU.
- **Turnaround Time:** The total time taken by a request to get completely served, including its response time, waiting time and service time.
- **Fairness:** The principle that states that every

request should get equal share of CPU time.

- **Resource Cost:** The total cost of the resources acquired or used for the servicing of requests by various cloud consumers. The main goal is to maximize the CPU Utilization, maximize the Throughput, minimize the Response Time, minimize the Waiting time, minimize the Turnaround Time, minimize the Resource Cost and obey the Fairness principle.

II. METHODS AND MATERIAL

A. Literature Survey

In Current Scenario, with an environment of mobile cloud the task is divided and disseminated into same size of small jobs i.e. Cloudlets. These Cloudlets as well as Virtual Machines are scheduled according to the various scheduling policy for e.g. FCFS, Round Robin etc. Generally in Cloud Computing scenario user submit the task to be performed / executed. Cloud Coordinator (CC) [2] divides the task into equal sized cloudlets and passes it to Datacenter (DC).Normally it takes a lot of time because the cloudlets are processed one at a time in FCFS manner as and when they reach to VM. VM executes the cloudlets present in the queue as they reach the VM's. Basically this default job scheduled policy is extremely Time- Consuming, Cost insensitive and inefficient. Cloud services are currently among the top-ranked high growth areas in computer services and seeing an acceleration in enterprise adoption with the worldwide market predicted to reach more than$140b in 2014[14],[3]. For secure cloud bursting and aggregation, the author uses encryption scheme of 64-bit cipher. But they are fail to apply this concept in real environment check for the real time simulations on different platforms.

In paper [7] a brand new VM fill up Balancing Algorithm is actually Weighted Active Monitoring populate Balancing Algorithm applying CloudSim tools, due to the Datacenter to help efficiently load balance requests between ones exhibited virtual devices assigning the weight, in order to achieve far better performance parameters. Here VMs associated with different processing powers along with the tasks/requests usually are designated or perhaps issued on the all-powerful VM and then on the lowest so on. In paper [8] author proposed a good algorithm can be ant colony optimization that random optimization search approach is usually obtained pertaining to allocating your current incoming jobs on the virtual machine.

In 2014[5], the problem of load balancing in cloud bursting was discovered and various scheduling algorithms are proposed to solve the problem but the practical implementation of that solutions is still remaining. In 2015[19], automatic cloud bursting was suggested which allows computer resources to be dynamically reconfigured to meet users‟ demands but the launching of virtual machines on commercial cloud takes much more time than on fermicloud. Hence further more tests in virtual machines‟ performance on commercial cloud needed to be done in the future.

B. Components of Cloud System

A typical Cloud modeled applying CloudSim involves after four entities Datacenters, Hosts, Virtual m/c in addition application form along with system Software which are shown in figure1.

1. Datacenter: Datacenter asset of hosts . This can be responsible regarding managing virtual models (VMs) (e.g., VM provisioning). It behaves similar to a IaaS provider from finding requests with regard to VMs via brokers.

2. Datacenter Broker: This class represents the broker acting on behalf of a user. It modifies a couple of mechanisms: ones mechanism for submitting VM provisioning requests to be able to data centers and mechanism with regard to submitting tasks to VMs.

3. Host: Host executes actions regarding management of VMs (e.g., creation along with destruction) and update task processing to be able to VMs. a good host possesses the defined policy to provisioning memory, processing elements, and also bandwidth to virtual machines. a good host is associated for you to the data center. The idea can host virtual machines.

4. VM: This represents the software implementation of a machine that executes applications called virtual machine (VM) which functions to be a physical machine. Each virtual machine divides your own resources received by the host among tasks working from it.

5. Cloudlet: The cloudlet class can be also known as being a task. CloudSim represents your complexity of the application in relation to their computational requirements. The class is managed through the

scheduling policy that will be implemented inside Datacenter Broker Class.

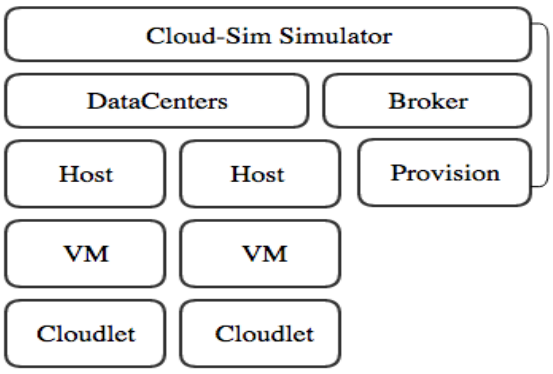

Figure1. Block diagram of Components of Cloud System

C. LOAD Balancing Algorithms:

Since the job arrival pattern is not predictable and the capacities of each node in the cloud differ, for load balancing problem, workload control is crucial to improve system performance and maintain stability. Load balancing schemes depending on whether the system dynamics are important can be either static or dynamic. Static schemes do not use the system information and are less complex while dynamic schemes will bring additional costs for the system but can change as the system status changes. A dynamic scheme is used here for its flexibility.
Few exiting load-balancing algorithms are as follows:

1. Token Routing
2. Round Robin
3. Randomized
4. Central Queue
5. Throttled
6. Equally spread current execution

We work on throttled and equal load sharing algorithms.

Throttled : The method 1st starts by maintain a listing of all the VMs every row is on an individual basis indexed to hurry up the search process. If a match is found on the idea of size and handiness of the machine, then the load balancer accepts the request of the consumer and allocates that VM to the consumer. If, but there's no VM accessible that matches the factors then the load balancer returns -1 and therefore the request is queued.

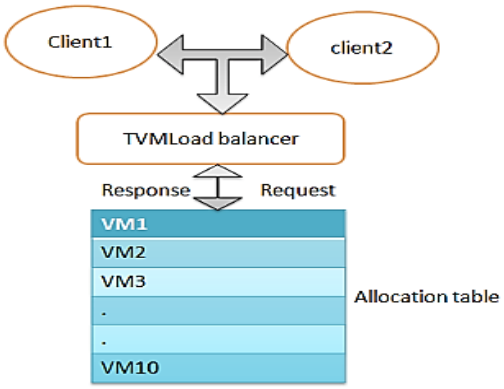

Figure 2. Block diagram of Throttled Algorithm

Equally Spread Current Execution : This algorithmic rule distributes the load indiscriminately by first checking the dimensions of the method and so transferring the load to a Virtual Machine that is gently loaded. The load balancer spreads the load on to completely different nodes, and hence, it's referred to as unfold spectrum technique. The load balancer maintains a queue of the roles that require to use and are presently mistreatment the services of the virtual machine. The balancer then unendingly scans this queue and therefore the list of virtual machines.

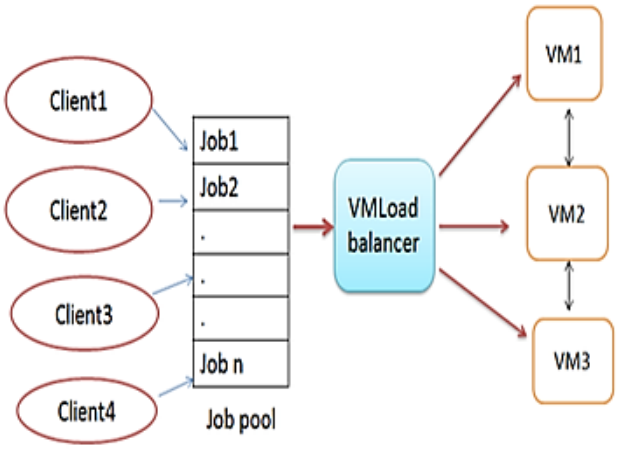

Figure3. Block diagram of Equally Spread Current Execution Algorithm

D. Problem in existing system

Load balancing schemes depending on whether the system dynamics are important can be either static or

dynamic. Static schemes do not use the system information and are less complex. Considering that the job arrival pattern isn't predictable and the capacities of each one node in differ, for load managing problem, workload control is important to improve system performance and keep stability. A dynamic scheme can be used here for its freedom.

- Cloud-computing environment can be a very complex problem along with load balancing receiving.
- The job arrival pattern isn't predictable and the capacities of each one node in the differ, for load managing problem, workload control is important to improve system performance and keep stability.

III. RESULTS AND DISCUSSION

A. Proposed Algorithm

Figure4:Proposed Hybrid load balancing Algorithm

Algorithm: **Hybrid Algorithm(DC,VMs,Cloutlets)**

1. *It maintain an index table of VMs & the state of VM (BUSY/FREE).*
2. *At the start all VMs are FREE.*
3. *Data center controller receives a new request.*
4. *Data center controller queries the VM load balancer for the next allocation.*
5. *VM load balancer check for all current allocation count is less than max length of VM list allocation the VM.*
6. *If VM is not allocation create a new one.*
7. *Count the active load on each VM.*
8. *Return the id of those VM which is having least load.*
9. *The VM load balancer will allocation the request to one of VM.*
10. *If a VM is overloaded then the VM load balancer will distribute same of its work to the VM having least work so that every VM is equally loaded.*
11. *When the VM finishes processing the request & the data center controller receives the response cloud let, it notifies the VM load balancer of VM reallocation.*
12. *The data center controller check if there are any waiting request in the queue, if there are, it continues from step4.*
13. *Continue from step3.*

We roposed new hybrid load balancing algorithm, which is combination of throttled and equally spread current execution load balancing algorithm. The method 1st starts by maintain a listing of all the VMs every row is on an individual basis indexed to hurry up the search process. If a match is found on the idea of size and handiness of the machine, then the load balancer accepts the request of the consumer and allocates that VM to the consumer.

The load balancer spreads the load on to completely different nodes, and hence, it's referred to as unfold spectrum technique. The load balancer maintains a queue of the roles that require to use and are presently mistreatment the services of the virtual machine. The balancer then unendingly scans this queue and therefore the list of virtual machines.

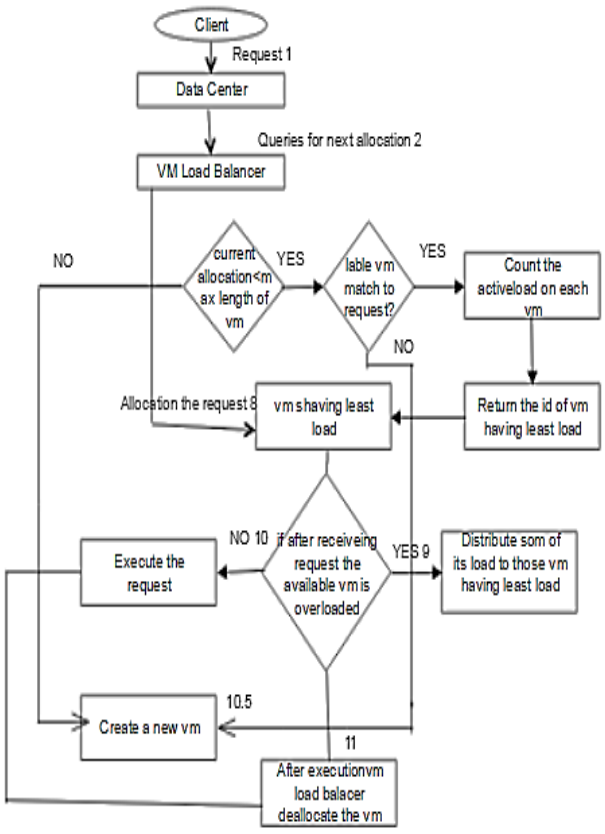

Figure 5 : Flow Chart of Hybrid load balancing algorithm

B. Implementation and Result Analysis

We implemented hybrid algorithm, which is combination of throttled, and equally spread current execution balancing algorithm on NetBeans using advanced JAVA. Cloud simulator is simulated for simulation with different configuration. Before simulation we configure many parameters like number of datacenters, number of cloudlets, VM configuration, bandwidth and MIPS. We implemented three algorithm of load balancing are:

- Throttled
- Equally spread current execution
- Hybrid

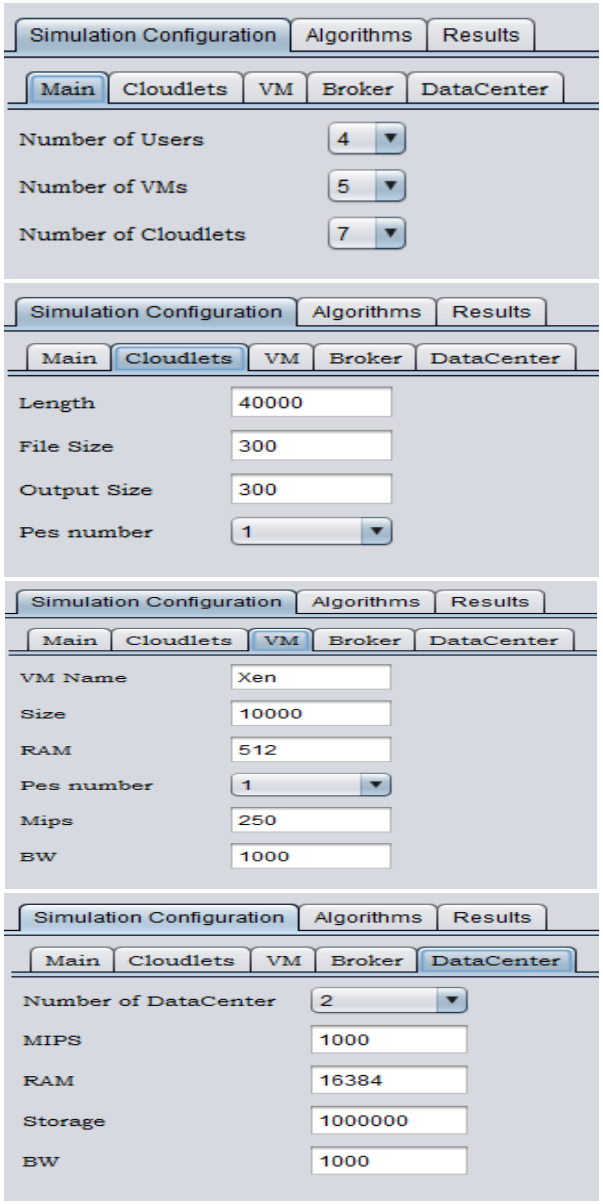

Figure 6 : Configuration Details of Cloud-Sim Simulator

Below diagrams 7,8 and 9 show that execution cloudlets, amount of time needed for execution, it is also showing that which cloudlets assign on which datacenter and virtual machine.

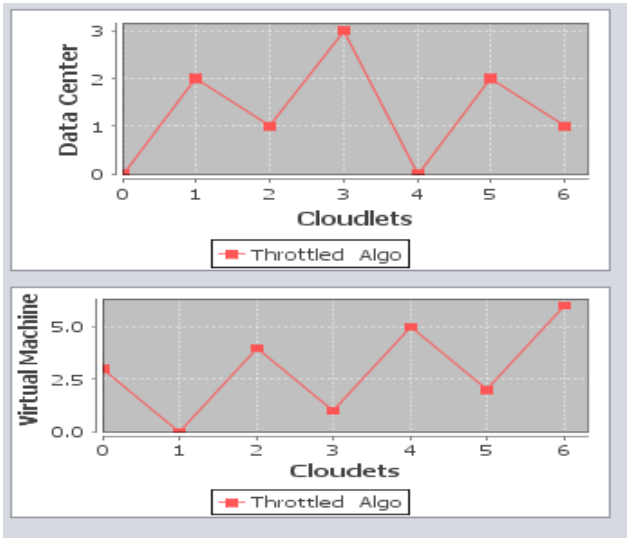

Figure 7 : Data Centers and VM Allocation using Throttled Load Balancing Algorithm

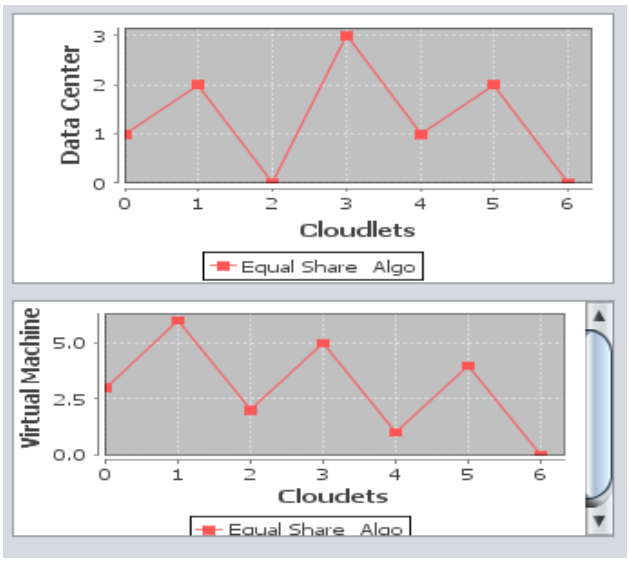

Figure 8 : Data Centers and VM Allocation using Equally Load Shared Load Balancing Algorithm

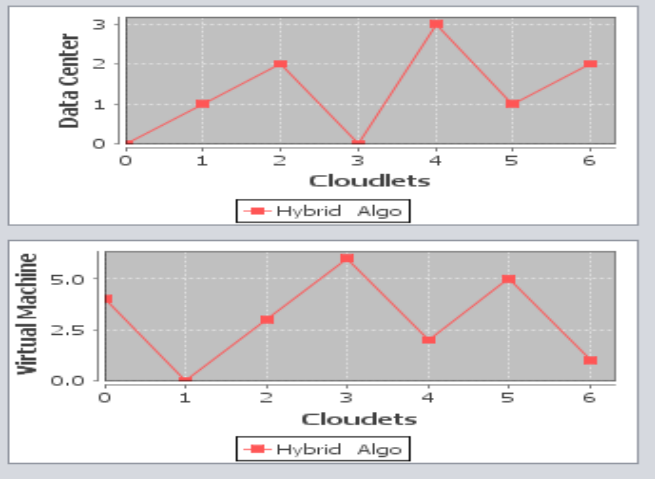

Figure 9 : Data Centers and VM Allocation using Hybrid Load Balancing Algorithm

C. Cloudsim Simulator

CloudSim [12] is the many efficient tool you can use with regard to modeling regarding Cloud. during your current lifecycle of an Cloud, CloudSim allows VMs for you to be managed coming from hosts that will inside turn are usually managed by datacenters. CloudSim offers architecture inside four uncomplicated entities. These types of entities offer consumer to set-up the basic cloud computing environment as well as measure your effectiveness involving fill up Balancing algorithms.

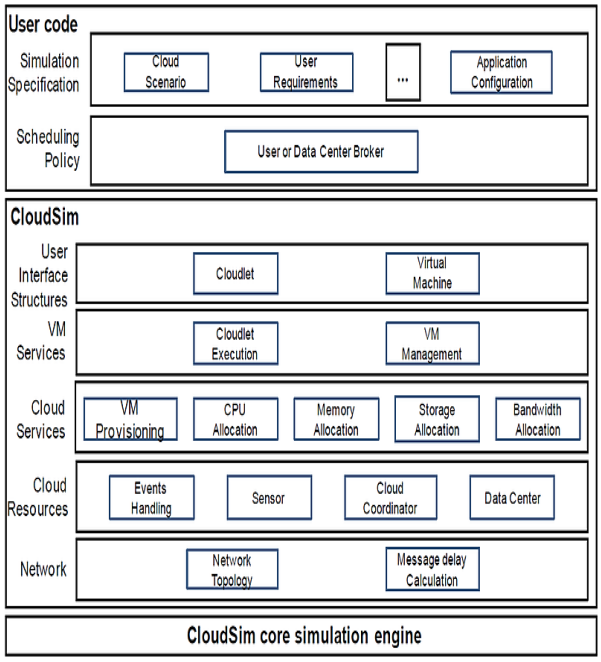

Figure 10. CloudSim Architecture

Datacenters entity features the responsibility of providing Infrastructure level solutions for the Cloud Users. They act as a home to help a lot of Host Entities or maybe a lot of instances hosts' entities aggregate to help application form the solitary Datacenter entity. Hosts with Cloud are usually Physical Servers The idea have pre-configured processing capabilities. Host is actually responsible regarding providing Software level SERVICE towards Cloud Users. Hosts have their particular storage and memory. Processing features regarding hosts is usually expressed throughout MIPS (million instructions per second)

IV. CONCLUSION

We study many static load balancing algorithms and dynamic load balancing algorithms. We have implemented three dynamic load balancing algorithms are Throttled , Equally spread Current Execution and hybrid that is combination of both throttled and equal load sharing for execute user cloudlets in cloud computing. Here we used 10 cloudlets, 4 data centers and 4 VMs for simulation. All three algorithms simulation performed with the help of cloud-sim simulator on java language. Compare data center service time, allocation of data center ,and allocation of VM, response time, total cost of each data center of throttled, equal load share and hybrid algorithm. Found that hybrid algorithm data center allocation and cost of each data center is more accurately than throttled and equal load sharing algorithm. Future work of our project is improve efficiency and reduce cost of data centers and VM using new adaptive algorithm. And also improve it in future to work well in heterogeneous environment such as big data.

V. REFERENCES

[1] GaochaoXu, Junjie Pang, and Xiaodong Fu "A Load Balancing Model Based on Cloud Partitioning for the Public Cloud"Tsinghua Science and Technology, February 2013

[2] Abhijeet G Purohitet. "Load balancing in public cloud by division of cloud based on the geographical location"International Journal of Research in Engineering and Technology] Volume:03/2014 http://www.ijret.org

[3] Ms.ShilpaD.Moreet."Reviews of Load Balancing Based on Partitioning in Cloud Computing" International Journal of Computer Science and Information Technologies, Vol. 5 (3) , 2014

[4] Pooja and Mishra2, "Analysis of Variants in Round Robin Algorithms for Load Balancing in Cloud Computing", (IJCSIT) International Journals of Computer Science and Information Technologies, Volume 4 (3), 2013, pg. no. 416-419.

[5] Kunal Mahurkar1, Shraddha Katore2 and Suraj Bhaisade3, Pratikawale4, "Reducing Cost of Provisioning in Cloud Computing", International Journal of Advance in Computer Science and Cloud Computing, Volume- 1, Issue- 2, nov.- 2013, pg. 6- 8.

[6] Dr. Rakesh Rathi1, Vaishali Sharma2 and Sumit Kumar Bole3, "Round Robin Data Center Selection in Single Region for Service Proximity Service Broker in Cloud Analyst" , International Journal of Computer & Technology, Volume 4 no. 2, March- April 2013, pg. no. 254- 260.

[7] Bhatiya Wickremansinghe1, Rodrigo N. Calheiros2and Dr. Rajkumar Buyya3, "CloudAnalyst: A CloudSim- based VisulModeller for Analysing Cloud Computing Environments and Applications", IEEE Computer Society, 2010, pp. 446-452.

[8] Jaspreet Kaur, "Comparison of load balancing algorithm in a Cloud", International Journal of Engineering Research and Applications (IJERA), vol. 2, Issue 3, May- June 2012, pp. 1169- 1173.

[9] Syed Tauhid Zuheri1, Tamanna Shamrin2 and Rusia Tanbin3, Firoj Mahmud4, "An Efficient Load Balancing Approach in Cloud Environment by using Round Robin Algorithm", International Journal of Artificial and Mechatronics, volume 1, issue 5, 2013, pp 96-99.

[10] B. Santosh Kumar1 and Dr. Latha Parthiban2, "An Implementation of Load Balancing Policy for Virtual Machines Associated with a Data Centre", International Journal of Computer Science & Engineering Technology (IJCSET), volume 5 no. 03, March 2014, pp. 253- 261.

[11] Sonika Matele1, Dr, K James2 and Navneet Singh3, "A Study of Load Balancing Issue Among Multifarious Issues of Cloud Computing Environment", International Journals of Emerging Technolog Computational and Applied Science (IJETCAS), volume 13- 142, 2013, pg. 236- 241.

[12] B. Adler, Load balancing in the cloud: Tools, tips and Tools, tips and techniques, http://www.rightscale. com/info center/whitepapers/ Load-Balancing-in-the-Cloud.pdf, 2012

[13] Z. Chaczko, VenkateshMahadevan, ShahrzadAslanzadeh, Christopher Mcdermid. "Availability and Load Balancing in Cloud Computing" 2011 International Conference on Computer and Software Modeling. x

[14] K. Nishant, P. Sharma, V. Krishna, C. Gupta, K. P. Singh,N. Nitin, and R. Rastogi, Load balancing of nodes in cloud using ant colony optimization, in Proc. 14thInternational Conference on Computer Modeling and Simulation (UKSim), Cambridgeshire, United Kingdom, Mar. 2012, pp. 28-30.

[15] D. Grosu, A. T. Chronopoulos, Non cooperative Load balancing in distributed system , Journal of Parallel and Distributed computing ,2005

[16] P. Mell and T. Grance, The NIST definition of cloud computing, http://csrc.nist.gov/ publications/nist pubs/800-145/SP800-145.pdf, 2012.

[17] S. Penmatsa and A. T. Chronopoulos, Game-theoretic static load balancing for distributed systems, Journal of Parallel and Distributed Computing, vol. 71, no. 4, pp. 537-555, Apr. 2011.