"Implementation of Load Balancing in Cloud Computing thorough Round Robin & Priority using CloudSim"

Aabha Sahu[1], Dr. Durgesh Kumar Mishra[2]
[1]*M.Tech Scholar,* [2]*Professor*
[12]*Department of Computer Science & Engineering, SAIT, Indore*

Abstract- Load balancing in the cloud computing environment has an important impact on the performance. Good load balancing makes cloud computing more efficient and improves user satisfaction. Load balancing with cloud computing provides a good efficient strategy to several inquiries residing inside cloud computing environment set. Complete balancing must acquire straight into account two tasks, one will be the resource provisioning as well as resource allocation along with task scheduling throughout distributed system. Round Robin algorithm can be the easiest algorithm shown to help distribute populate among nodes. Because of this reason it is frequently the first preference when implementing a easy scheduler. One of the reason for it being so simple is that the only information is required is a list of nodes. The proposed algorithm eliminates the drawbacks of implementing simple round robin architecture in cloud computing by introducing a concept of assigning different time slices to individual processes depending on their priorities.

Keywords- Cloud Computing, Load Balancing, Virtual Machine, Round Robin, Datacenter Broker, Host, Cloudlets, Cloud Coordinator.

I. INTRODUTION

Load balancing in the cloud computing has an important impact on the performance. Good load balancing makes cloud computing more efficient and improves user satisfaction. This article introduces a better load balancing model for the public cloud based on the cloud partitioning concept with a switch mechanism to choose different strategies for different situations. The algorithm applies the game theory to the load balancing strategy to improve the efficiency in the public cloud environment. A typical cloud model applying CloudSim involves after four entities. Datacenters, Hosts, Virtual Machine in addition application form along with system software. *Datacenter* is a set of host. This can be responsible regarding managing virtual models. It behaves like IaaS provider from finding request with regard to virtual models via brokers. *Datacenter Broker* represents the broker acting on behalf of the user. It modifies a couple of mechanism: Ones mechanism for submitting virtual machine provisioning requests to be able to datacenters and mechanism with regard to submitting tasks to virtual machine. *Host* executes actions regarding management of virtual machines (e.g. creation along with destruction) and update task processing to be able to virtual machines. A good host is possesses the defined policy to provisioning memory, processing elements and also bandwidth to virtual machines. A good host is associated for you to the datacenter. The idea can host virtual machines. *Virtual Machine* 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. *Cloudlet* can be viewed as a datacenter in a box whose goal is to bring the cloud closer. The class is managed through the scheduling policy that will be implemented inside datacenter broker class.

II. LITERATURE REVIEW

Authors proposed Equally Spread Current Execution (ESCE) scheduling algorithm to solve the load distributing problem on various nodes of a distributed system to improve both resource utilization and job response time while also avoiding a situation where some of the nodes are heavily loaded while other nodes are idle or doing very little work.[2]

This research paper concentrated on various VM load balancing algorithm(Throttled Load Balancing Algorithm and Active Monitoring Load Balancing Algorithm) and proposed a new VM load balancing algorithm after modifying Throttled Load Balancing algorithm in virtual machine environment of cloud computing in order to achieve better response time, processing time, and cost.[3]

This research paper concentrated on dynamically optimized cost based task scheduling which combines cost based task scheduling beneficial to user and dynamically optimized resource allocation strategy beneficial to service provider. They also prove computation ratio and utilization of available resources by grouping the user tasks before resource allocation. [4]

Authors proposed Optimal Cloud Resource Provisioning (OCRP) algorithm which is used to obtain optimal solution. It uses Bender decomposition, stochastic programming model, sample average approximation and deterministic equivalent formulation. OCRP result in reduction of the cost for resource provisioning. [5]

Authors proposed an algorithm Round Robin (RR) to reschedule the CPUs. Here at first consumer's request submitted into the Service Accepter (SA) and SA search for free VMs. When it finds one it starts to serve the services to those VMs using RR. The work is done for ten servers and the procedure of the process scheduled is not dynamic. [6]

B. Santhosh Kumar and Dr. Latha Parthiban implemented an algorithm VMLeastFrequentlyUsed to distribute the load among virtual machine created in a data center. The algorithm mainly relies upon Least frequently used mechanism. When a task has to be submitted to a data center for processing choice is made upon the VM which has been assigned tasks least number of times. [7, 11]

Lipika Datta introduced modified Round Robin scheduling for processes with user defined external priorities so that the scheduling algorithm can be used to schedule processes of soft real time systems which reduces the average response time and less number of context switching. [8, 14]

This research paper suggests a new VM load balancing algorithm: “Round Robin VM Load Balancing”, which is implemented using round robin approach with previous allocation status of virtual machine to handle service request from the user base. [9]

Rajkumar Somani and Jyotsana Ojha, proposed a hybrid approach for virtual machine level load balancing using concepts of two algorithm for load balancing: Round Robin Algorithm and Throttled Algorithm. The Hybrid approach gave better results in terms of response time, data center request serving time and data center processing time. It is effective in case of same data size per request as well as for different data size per request. [10]

Ritu Kapur, introduced a new scheduling algorithm Cost Effective Resource Scheduling (CERS) algorithm. This algorithm leads to an efficient utilization of resources; it serves the aim of Green IT and makes a contribution towards a better future. [12]

Authors proposed an approach of Round Robin technique in a circular way and by this method try clarify the load balancing scenario of a cloud server during its execution. It helps to get an effective and fast execution environment of task assigned by the user which helps in to create an effective communication framework between broker and virtual machine to optimize the time and minimize the cost. [13]

III. PROBLEM DOMAIN

In current scenario, with an environment of 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) divides the task into equal sized cloudlets and passes it to Data Center (DC). Normally it takes a lot of time because the cloudlets are processed one at a time in FCFS manner and when they reach to virtual machine (VM). VM executes the cloudlets present in the queue as they reach the VM’s. Basically this default scheduling policy is extremely time consuming, cost incentive and inefficient.

IV. AIM

Our aim is to implement Round Robin with Priority scheduling policy for VM using CloudSim3.0. We will also implement combination of load balancing algorithms like Round Robin with Priority and less resources first. This synopsis aims towards the establishment of performance qualitative analysis on existing VM load balancing algorithm and then implemented in CloudSim and Java language.

V. EXISTING SYSTEM

Cloud computing is efficient and scalable but maintaining the stability of processing so many jobs in cloud computing environment is a very complex problem with load balancing, receiving much attention for researches. 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 system 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. For this we use load balancers.

The existing load balances are Round Robin, Throttled and Active Monitoring load balancer.

Round Robin Load Balancer:-

It is one of the simplest scheduling techniques that utilize the principle of time slices [9]. Here the time is divided into multiple slices and each node is given a particular time slice or time interval i.e. it utilizes the principle of time scheduling. Each node is given a quantum and its operation. They are provided to the requesting client on the basis of time slice by the service provider.

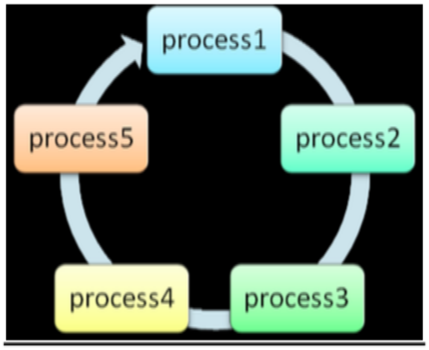


Fig.1: Round Robin Load Balancer

Throttled Load Balancer (TLB):-

This algorithm ensures that pre-defined number of cloudlets is allocated to a single VM at any given time. If there are more request groups are present than the number of available VM's at data centre allocate incoming request in queue basis until the next VM becomes available.

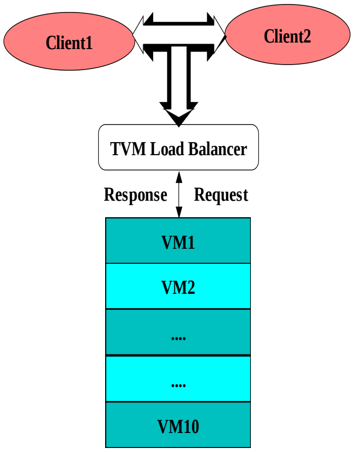


Fig.2: Throttled Load Balancer

Active Monitoring Load Balancer (AMLB):-

The AMLB is a load balancer which maintains information about each virtual machine and the number of request currently allocated to which virtual machine when a new virtual machine arrives. If there are more than one virtual machine, the first identified is selected. AMLB returns the virtual machine identification number to the data center controller. The data center controller sends the request to the virtual machine identified by that identification number. The data centre controller notifies the AMLB to new allocation and cloudlets is sent to it.

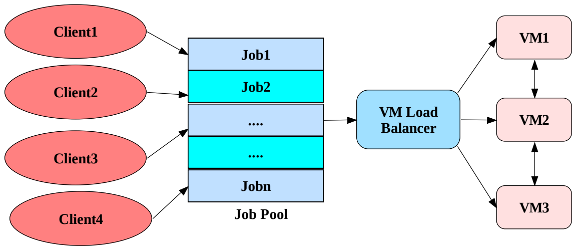


Fig.3: Active Monitoring Load Balancer

CloudSim:-

Figure 4 shows that the multilayered design of CloudSim software framework and its architectural components [1]. CloudSim is efficient tool which can be used with regard to Cloud modeling. CloudSim provides a generalized and extensible simulation framework that enables seamless modeling and simulation of app performance during current life cycle of a Cloud. CloudSim allows VMs for us to be managed coming from hosts that will inside turn are usually managed by datacenters. Architecture inside four uncomplicated entities is offered by CloudSim. 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. The responsibility of providing Infrastructure level solutions for the Cloud users is featured by Datacenters entities. 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 are usually Physical Servers. Software level Service is provided by the host to the Cloud Users. Hosts have their particular storage and memory. Processing features regarding hosts is usually expressed throughout MIPS (million instructions per second).

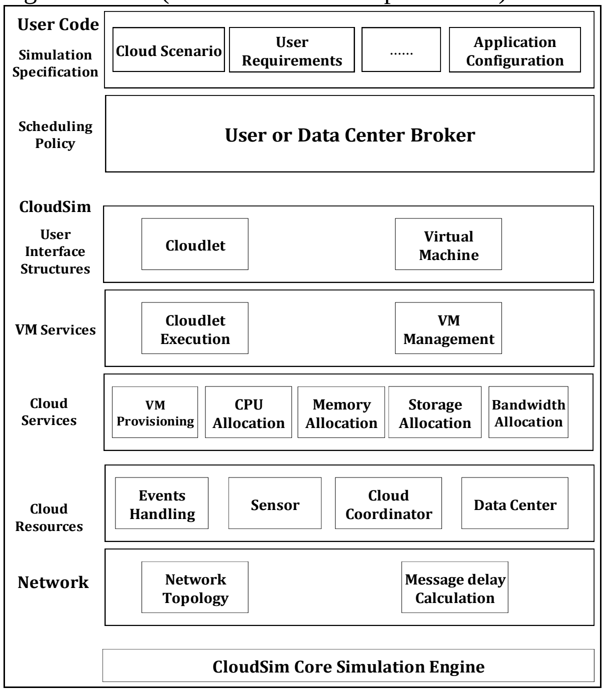


Fig.4: CloudSim Architecture

VI. PROPOSED SYSTEM

It is a static fill up balancing algorithm that does not take the previous fill up state of a node for the day involving assigning jobs. This makes use of round robin scheduling algorithm regarding allocating jobs.

The item selects your very first node arbitrarily and then allocates jobs for you to just about all additional nodes in a round robin manner. The actual algorithm is effective from random menus of the virtual machine. The datacenter controller allocates your request for you to a record of VMs with a good rotating basis. Your current primary obtain can be assigned to VM selected randomly by the group subsequently ones details center controller assigns your request in the circular order soon after your own VM is actually allotted your request, your own VM is usually shifted towards end of a record.

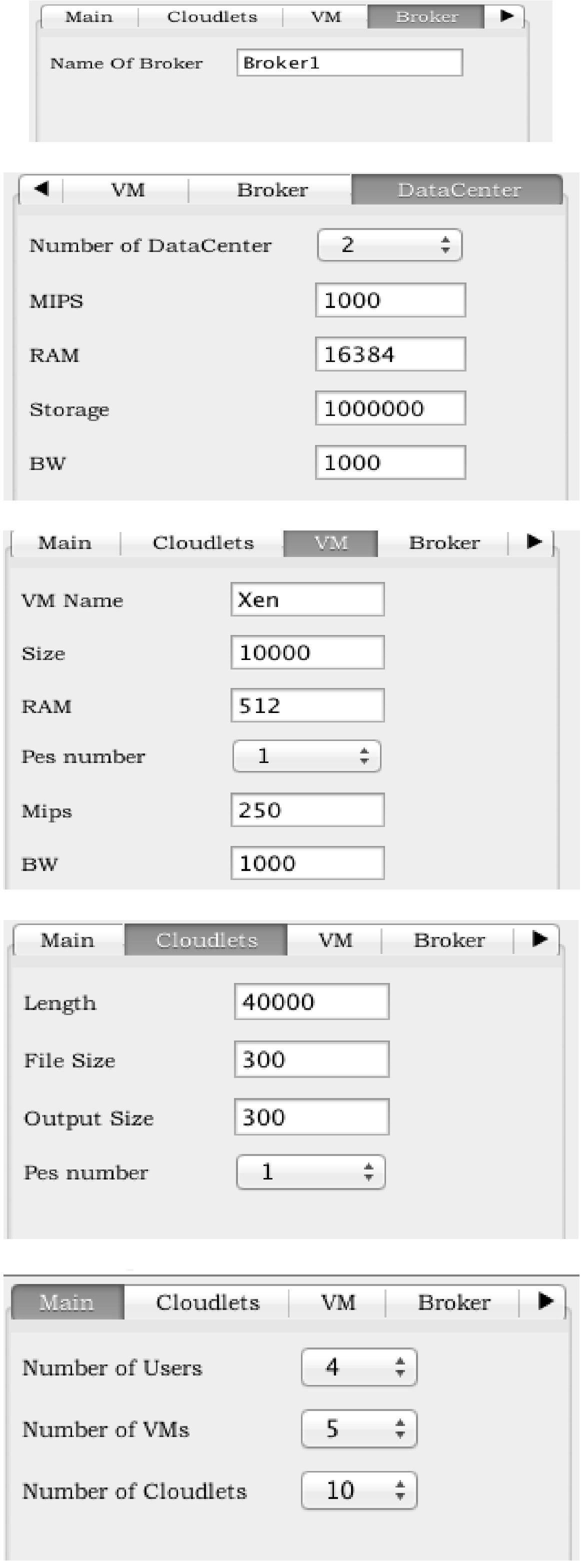
Modified Round Robin

The proposed algorithm eliminates the drawbacks of implementing a simple round robin architecture in cloud computing by introducing a concept of assigning different time slices to individual processes depending on their are priorities. User assigns the priority of a process externally. In the proposed architecture when a new process arrives in the system it is queued at a small processor. This small dedicated processor is used to calculate the time slices of each process, arranges the processes in ascending order of their burst times and then creates the ready queue for the main processor. This small dedicated processor is used to reduce the burden of the main processor. The processes then execute in the main processor according to round robin scheduling algorithm with their individual time slices. Whenever a new process arrives in the system ready queue, its time slice is calculated and enquired to the main processor's ready queue. Whenever a process completes its execution it is removed from both the system ready queue and the main processor ready queue. The process continues until the main processor ready queue becomes empty. I am assuming that lesser number implies higher priority

```
Algorithm: Modified Round Robin
The proposed round robin algorithm is as follows:
Step 1.
    Set all the VM allocation is zero and record of each VM index by Round Robin Load Balancer.
Step 2.
    a. User request/task/cloudlet receives by datacenter receivers.
    b. On the base of priority allocated virtual machine and calculate rangr (R)
       R=Max Burst Time + Min Burst Time
    c. Basis of range and priority, load balancer allocates the time quantum to user request.
Step 3.
    After the complete of task (cloudlets), VM are allocated to other user request.
Step 4.
    Checks new/ pending/ waiting requests in queue by datacenter controller.
```

VII. RESULT ANALYSIS

Proposed system implemented in 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 Round Robin evolution with following configuration which is shown below.



Name Of Broker: Broker1

Number of DataCenter: 2
MIPS: 1000
RAM: 16384
Storage: 1000000
BW: 1000

VM Name: Xen
Size: 10000
RAM: 512
Pes number: 1
Mips: 250
BW: 1000

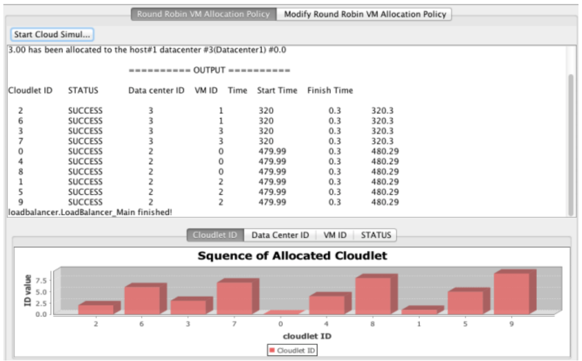
Length: 40000
File Size: 300
Output Size: 300
Pes number: 1

Number of Users: 4
Number of VMs: 5
Number of Cloudlets: 10

Fig.5: Configuration details of CloudSim simulator

VIII. EXPERIMENTAL RESULTS

We apply Round Robin algorithm on above configuration with the help of CloudSim simulator. In below show that execution cloudlets, amount of time needed for execution, it is also showing that which cloudlets assign on which datacenter and virtual machine. Response time of each cloudlets is calculate and shown in diagram.


Round Robin VM Allocation Policy
Modify Round Robin VM Allocation Policy
Start Cloud Simul...

```
3.00 has been allocated to the host#1 datacenter #3(Datacenter1) #0.0

                    ========== OUTPUT ==========

Cloudlet ID    STATUS       Data center ID  VM ID  Time  Start Time  Finish Time

  2            SUCCESS          3             1     320        0.3      320.3
  6            SUCCESS          3             1     320        0.3      320.3
  3            SUCCESS          3             3     320        0.3      320.3
  7            SUCCESS          3             3     320        0.3      320.3
  0            SUCCESS          2             0     479.99     0.3      480.29
  4            SUCCESS          2             0     479.99     0.3      480.29
  8            SUCCESS          2             0     479.99     0.3      480.29
  1            SUCCESS          2             2     479.99     0.3      480.29
  5            SUCCESS          2             2     479.99     0.3      480.29
  9            SUCCESS          2             2     479.99     0.3      480.29
loadbalancer.LoadBalancer_Main finished!
```

Fig.6: Response Time of cloudlets using Round Robin Load Balancing Algorithm

Above snapshots consist of the different Response time of different process including Start time and finish time. So we can easily identify the Response time of each process by above snapshot.

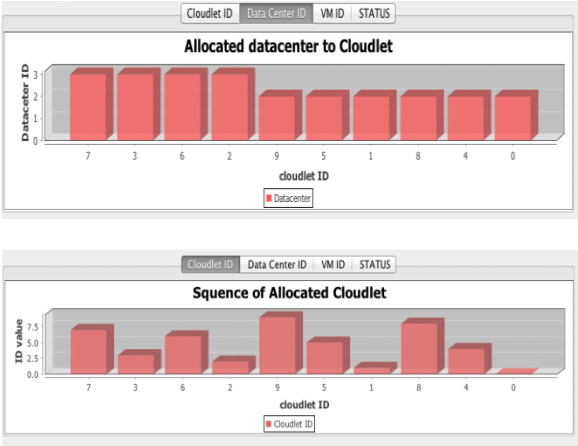
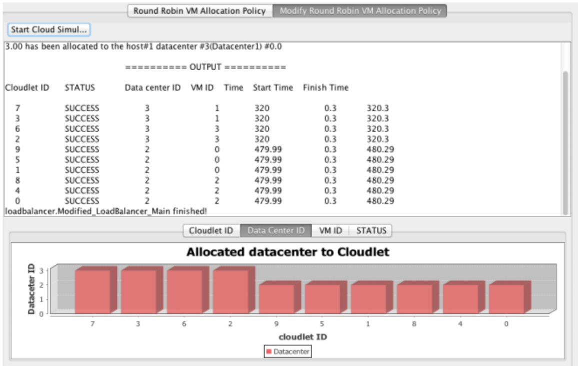


Fig.7: Data Centers and VM Allocation using Round Robin Load Balancing Algorithm


Round Robin VM Allocation Policy
Modify Round Robin VM Allocation Policy
Start Cloud Simul...

```
3.00 has been allocated to the host#1 datacenter #3(Datacenter1) #0.0

                    ========== OUTPUT ==========

Cloudlet ID    STATUS       Data center ID  VM ID  Time  Start Time  Finish Time

  7            SUCCESS          3             1     320        0.3      320.3
  3            SUCCESS          3             1     320        0.3      320.3
  6            SUCCESS          3             3     320        0.3      320.3
  2            SUCCESS          3             3     320        0.3      320.3
  9            SUCCESS          2             0     479.99     0.3      480.29
  5            SUCCESS          2             0     479.99     0.3      480.29
  1            SUCCESS          2             0     479.99     0.3      480.29
  8            SUCCESS          2             2     479.99     0.3      480.29
  4            SUCCESS          2             2     479.99     0.3      480.29
  0            SUCCESS          2             2     479.99     0.3      480.29
loadbalancer.Modified_LoadBalancer_Main finished!
```

Fig.8:Response Time of cloudlets using Modified Round Robin VM Allocation Policy

Comparison of Existing & Proposed Algorithm:-

We apply Modified Round Robin algorithm on above configuration with the help of CloudSim simulator. In below diagram 5.4 show that execution cloudlets, amount of time needed for execution, it is also showing that which cloudlets assign on which datacenter and virtual machine.

Cloudlets	**Existing Algorithm**	**Proposed Algorithm**
4	40	12
7	78	40
10	93	65
15	138	110
20	178	160
Average	**105**	**79**

Table (1): Waiting Time Comparisons between algorithms

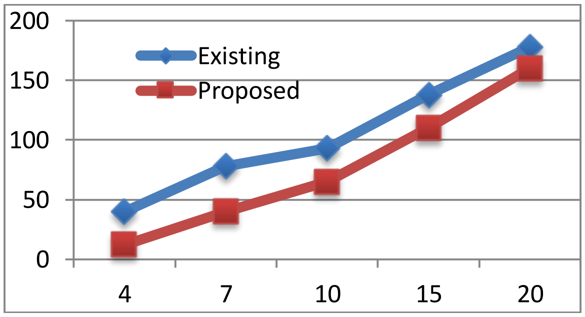
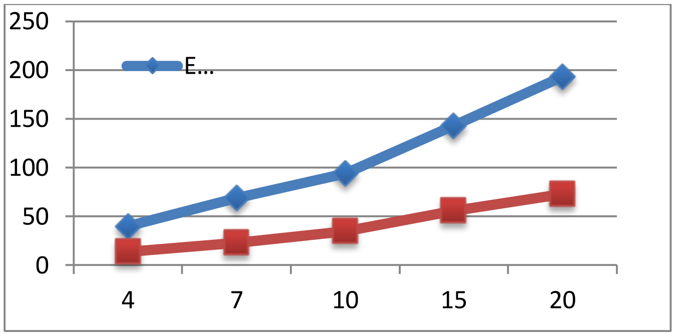


Fig.9: Waiting Time Comparisons between algorithms

Cloudlets	Existing	Proposed
4	40	14
7	69	23
10	94	35
15	143	56
20	193	73
Average	**107**	**41**

Table(2): Response Time Comparisons between algorithms



IX. CONCLUSION

This research focused on Cloud Computing along with research challenges in load balancing. It also focus on merits and demerits of the cloud computing. Major thrust is given on the study of load balancing algorithm, followed by a comparative survey of these above mentioned algorithms in cloud computing with respect to stability, resource utilization, static and dynamicity, cooperative or non-cooperativeness and process migration.

X. REFERENCES

[1]. Dr. Rajkumar Buyya, *"CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithm"*, Wiley Online Library, *24 August 2010, pp. 23- 50.*

[2]. Jaspreet Kaur, *"Comparison of load balancing algorithm in a Cloud"*, International Journal of Engineering Research and Applications (IJERA), *Vol. 2, No. 3, May- June 2012, pp. 1169-1173, ISSN:2248-9622.*

[3]. Prof Meenakshi Sharma1 and Pankaj Sharma, *"Performance Evaluation of Adaptive Virtual Machine Load Balancing Algorithm"*, International Journal of Advanced Computer Science and Applications(IJACSA), *Vol 3, No. 2, 2012, pp. 86-88*

[4]. Yogita Chawla and Mansi Bhonsle *"Dynamically optimized cost based task scheduling in cloud computing"*, International Journal of Emerging Trends and technology in Computer Science (IJETTCS), *Vol 2, No.3, May-June 2013, ISSN:2278-6856.*

[5]. Kunal Mahurkar, Shraddha Katore, Suraj Bhaisade and Pratikawale, *"Reducing Cost of Provisioning in Cloud Computing"*, International Journal of Advance in Computer Science and Cloud Computing, *Vol 1, No. 2, Nov- 2013, pp 6-8, ISSN:2321-4058.*

[6]. Syed Tauhid Zuheri1, Tamanna Shamrin, Rusia Tanbin and Firoj Mahmud, *"An Efficient Load Balancing Approach in Cloud Environment by using Round Robin Algorithm"*, International Journal of Artificial and Mechatronics, *Vol 1, No. 5, 2013, pp 96-99, ISSN:2320-5121.*

[7]. B. Santosh Kumar and Dr. Latha Parthiban, *"An Implementation of Load Balancing Policy for Virtual Machines Associated with a Data Centre"*, International Journal of Computer Science & Engineering Technology (IJCSET), *Vol 5 No. 3, March 2014, pp. 253- 261.*

[8]. Lipika Datta, *"Modified RR Algorithm with Dynamic Time Quantum for Externally Prioritized Tasks"*, International Journal on Recent and Innovation Trends in Computing and Communication. *Vol. 3, Issue 3, March 2014, ISSN:2278-0181*

[9]. Nusrat Pasha, Dr. Amit Agarwal and Dr. Ravi Rastogi, *"Round Robin Approach for VM load balancing algorithm in Cloud Computing Environment"*, International Journal of Advances Research in Computer Science and Software Engineering(IJRISE), *Vol 4, No. 5, May 2014, ISSN:2277-128X.*

[10]. Rajkumar Somani and Jyotsana Ojha *"A Hybrid Approach for VM Load Balancing in Cloud using CloudSim"*, Interntional Journal of Science, Engineering and Technology Research (IJSETR), *Vol 3, No. 6, June 2014*

[11]. B Preethi, Professor C. Kamalanathan, Dr. S.M Ramesh, S Shanmathi and P SathiyaBama, *"Optimization Of Resources in Cloud Computing Using Effective Load Balancing Algorithms"*, International Advanced Research Journal in Science, Engineering and Technology *Vol. 1, Issue 1, September 2014, ISSN:2393-8021.*

[12]. Ritu Kapur *"A Cost Effective approach for Resource Scheduling in Cloud Computing"* IEEE International Conference on Computer, Communication and Control *(IC4), 2015*

[13]. Soumen Santra and Dr. Kalyani Mali, "*A New Approach to Survey on Load Balancing in VM in Cloud Computing: using CloudSim"*, IEEE International Conference on Computer, Communication and Control (IC4-2015), 2015.

[14]. Shruti Tiwari, Shriya Prajapati and Nazish Ali Ansari, *"Modified Optimal Algorithm for Load Balancing in cloud Computing"*, International Conference on Computing, Communication and Automation (ICCCA), *2017, ISBN:978-1-5090-6471.*