# An Energy Efficient Server Load Balancing Algorithm

Rima M. Shah [1], Dr. Priti Srinivas Sajja [2]

[1] *Assistant Professor in Master of Computer Application,ITM Universe,Vadodara, India*

[2] *Professor at Post Graduate Department of Computer Science, Sardar Patel University, V.V.Nagar, India*

[1] *rimammshah@gmail.com*

[2] *priti@pritisajja.info*

***Abstract:-*** *The theory of Cloud computing has considerably changed the field of parallel and distributed computing systems. Cloud computing enables a wide range of users to access storage, hardware, software and any type of services in distributed, virtualized and scalable manner through Internet. Load balancing is a method to distribute request for SAAS, IAAS and PAAS from multiple clients to multiple computers connected in network to achieve optimal resource utilization. Virtualization maximize throughput in minimum response time, and avoid overload. With recent invention rather than investing lot on highly compatible server, why not to use cluster of computer which can be work as client as well as server when needed. Rather than single server facilitate all request a cluster of normal computer are used to serve request in efficient manner. With recent advent of technology, resource control or load balancing along with efficient energy usage in cloud computing is main challenging issue. A few existing scheduling algorithms can maintain load balancing and provide better strategies through efficient job scheduling and resource allocation techniques as well. Now there is a need for efficient and environment savvy algorithm to reduce carbon foot prints and to save energy. The objective of this paper to present environment savvy algorithm with effective resource utilization.*

***Key words:*** *SAAS, PAAS, IAAS, Virtualization, cloud computing*

## I. INTRODUCTION

Load balancing is probably the most commonly used term for describing a class of processes that attempt to

optimize system performance. System performance is optimized by attempting to best utilize a group of processing elements, typically a CPU, or storage elements, such as memory of disk, or some other resource that are interconnected in a distributed network. The process of Load Balancing may also be known as Load Sharing, Load Distribution, Parallel Programming, Concurrent Programming, and Control Scheduling. Although these processes can be quite different in their purpose, the processes all have a common goal. This goal is to allocate logical processes evenly across multiple processors, or a distributed network of processing elements, so that collectively all the logical processes are executed in the most efficient manner possible [1].

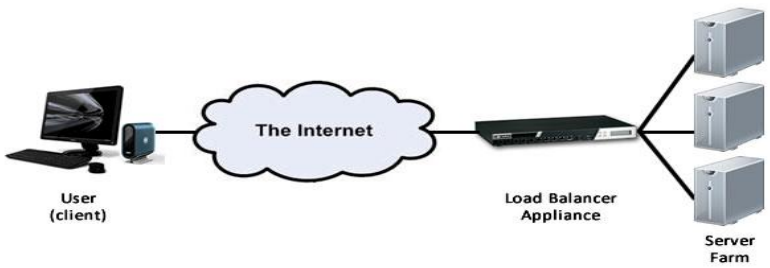


**Figure 1: network-based load balancing**

In below method the dynamic cloud computing environment is used, The intermediate node is used to monitor the load of each VM in the cloud pool. In this approach the user can send the request to the intermediate node. It is responsible for transfer the client request to the cloud. Here, the load is consider as in terms of CPU load with the amount of memory used, delay or Network load [2]

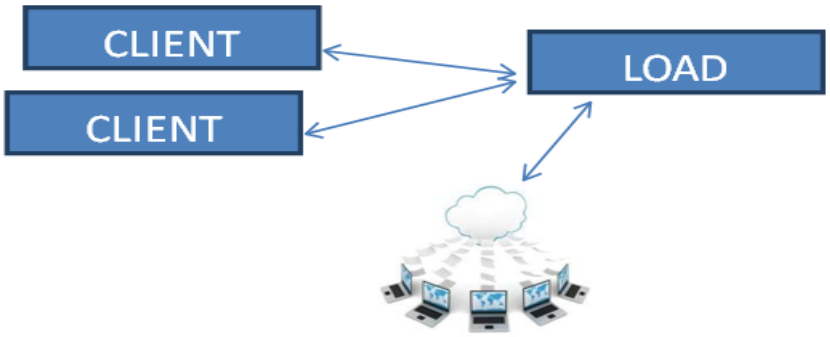


**Figure 2: Cloud with intermediate node [5]**

## II. DIFFERENT LOAD BALANCING ALGORITHM

**Round Robin Load balancing algorithm:**

Round robin is a simple method of load balancing. In a cluster with three servers, round robin load balancing passes the first request to Server A, the second request to Server B and the third request to Server C, before starting over at Server A with the forth request. In figure 1, requests are generated by single client and spread across three different web servers by means of round robin load balancing. This method works well if all your servers are the same although this might not be good strategy if we are trying to balance load across different servers with different requirements. [3].

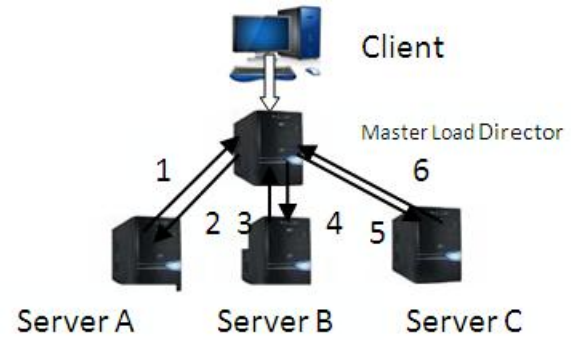


**Figure 3: Round Robin Load Balancing Algortithm**

**Equally Spread Current Execution Load Algorithm [4]:**

In this technique load balancer spread the load of the job into multiple virtual machines. The load balancer maintains a queue of the jobs that need to use and are currently using the services of the virtual machine. The balancer then continuously scans this queue and the list of virtual machines. If there is a VM available that can handled request of the node/client, the VM is allocated to that request [6]. If however there is a VM that is free and there is another VM that needs to be free of the load, then the balancer distributes some of the tasks of that VM to the free one so as to reduce the overhead to the VM manager [7]. The load also maintains a list of the
jobs, there size and the resources requested. The balancer selects the job that matches the criteria for execution at the present time. The working principle of ESCEL is shown below:

1. Find next available VM
2. Check for all current allocation count is less than max length of VM list allocate the VM
3. If available VM is not allocated create a new one.
4. Count the active load on each VM
5. Return the id of those VM
6. The UMLoadBalancer will allocate the request to one of the VM
7. If a VM is overloaded then the UMLoadBalancer will distribute some of its work to the VM having least work so that every VM is equally loaded.
8. The data center controller receives the response to the request sent and then allocates the waiting requests from the job pool/queue to the available VM and so on.
9. Continue from step 2.

## III. PROBLEM DEFINITION

In load balancing algorithm when load increases the capacity of virtual machine than new virtual machine is created to serve the user request. If more request come than as per the need new virtual machine are created. In the existing cloud computing [5], if client requests are increases above 100%, then Load balancer is create a new virtual machine to serve the new request to manage the load. It will not merge the load of virtual machines into one if no more virtual machines are actually required.

If suppose summation of newly created two virtual machine's load is less than 50% then there is no need to keep two virtual machines for serving the requests, because only one virtual machine is capable to handle the load as per defined its capacity to handle that load. But in prevailing algorithm, more than one virtual machines are used to handle requests whereas there is no need to keep all one. So, it is just wastage of energy and resources as lightly loaded virtual machine will consume the same energy as it is consumed by fully loaded virtual machine.

## IV. PROPOSED METHOD FLOW

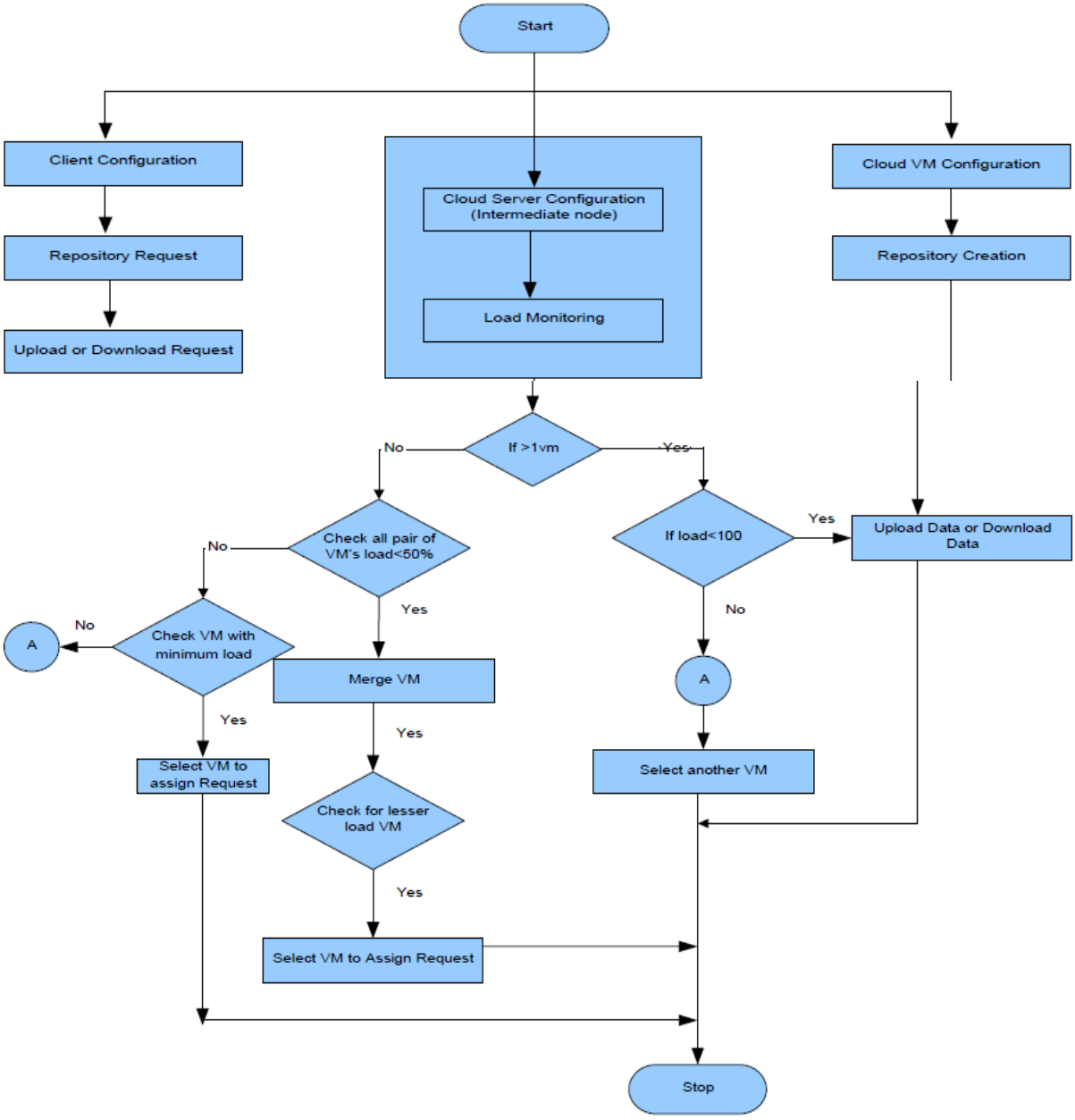


**Figure 4: Proposed Flowchart**

Our algorithm will start when there is a request from client in cloud server. In this algorithm first we check for one virtual machine to serve client request. If we do not have any virtual machine to serve client request than new virtual machine will be created. If there is one VM than the load balancer node check the CPU utilization if the CPU utilization is less than 100% then it will accept request otherwise VM load balancing Algorithm check for other virtual machine with less than max load or instantiates a new virtual machine on the compute node with the lowest utilization number.

The algorithm is to identify the reliable VM and process the client request. If there is more than one VM than check for load on all VM pairs. If the load of all VM pairs is < 50% than transfer the VM load in to any one VM which is less loaded. Then it will check pairs of VM load is <100% than check for less loaded VM and transfer request to that VM. If load of all VM is >100% than create new VM.

## V. DETAILED LOAD BALANCING ALGORITHM-MODIFIED APPROACH

For each VirtualMachine VM: vm1, vm2, vm3, _ _ _ _ vmi; For each task request T: t1, t2, t3, _ _ _ _, tj
Maximum load handled by each virtual machine VM(max)= 100% Set TimeDuration TD: Any Constant

```
1.  Begin
2.  while T ≠ ∅ Do
3.  select task Request(t)
4.      if (vm< 1) then
5.          create: vm1
6.          allocate vm to request t
7.          break;
8.      elseif  (vm1(load) < 100%) then
9.          allocate vm to request t
10.         break;
11.     elseif
12.         check for all (vmi(load)<50) then  // check load of each virtual machine
13.         search vm with least load
14.         merge all vm into one vm
15.                 if(vm<100%) then
16.                   Allocate vm to request t
17.                 Else
18.                   Create new vm to serve request t
19.                 end
20.     end
21.  end while
22.  End
```

## VI. CONCLUSION

A Modified algorithm has been formulated for dynamics of a distributed computing system in the context of load balancing. Initially it will try to create new virtual machine if cpu is overloaded. If load increases on virtual machine it will create new virtual machine. When virtual machines start serving user request load on virtual machines decreases. Many times it happens that many virtual machines are running carrying very low load with them. Lightly loaded virtual machine can efficiently serve user request but it results into wastage of energy. Running many lightly loaded virtual machines consumes more energy than one fully loaded virtual machine. This is attributable to the fact that new dynamic load balancing policy achieves a higher success, in comparison to the previously used load balancing techniques in reducing energy consumption and efficient request handling. Our future work considers the implementation and evaluation of the complexity of the modified approach for load balancing.

## REFERENCES

[1]. A.L. Blais "Evaluation of Load Balancing Algorithms and Internet Traffic Modeling For PerformanceAnalysis"
[2]. N.Chandrakala Dr. P.Sivaprakasam, "Analysis of Fault Tolerance Approaches in Dynamic Cloud Computing", International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 2, February 2013 ISSN: 2277 128X
[3]. Book: Linux Clustering Building and Maintaining Linux Clusters by Charles Bookman
[4]. Book: Handbook of Research on Securing Cloud-Based Databases with Biometric Applications.

Author: Ganesh Chandra Deka(Ministry of Labour and Employment, India), Sambit Bakshi (National Institute of Technology Rourkela, India)

[5]. D. Verma, R. K. Somani, "Tune the Resource Provisioning & Virtual Machine Migration for Cloud Environment", International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 3, Issue 6, November 2014, ISSN: 2319-5967 ISO 9001:2008

[6]. S. Mohapatra, Smruti K Rekha, S. Mohanty, "A Comparison of Four Popular Heuristics for Load Balancing of Virtual Machines in Cloud Computing," International Journal of Computer Applications ,Volume 68 - Number 6, Year of Publication: 2013

[7]. Kaushal Chandra, Sanjeev Kumar Sharma et.al. "Load Balancing in Wireless Mobile AD -HOC Networks" International Journal of Computing (IIJC) Vol2|issue 4|Oct 2012, pp 787-790. International Journal of Computer Science Issues IJCSI, paper id IJCSI-2012-9-5-4003, volume 9, issue 5, September 2012.