Abstract: Cloud computing provides the services either in the form of software application or hardware infrastructure on the basis of pay per use over the internet. It is the collection of heterogeneous resources that contain the characteristics of on demand self-service access to the users through a shared pool of cluster computing resources, scalability (scale-out and scale-up), resource pooling, broad network access, rapid elasticity and higher availability. However, maintaining the stability of processing several tasks in the cloud environment is a complex issue. Hence, it requires a load balancing technique that allocates the task to the Virtual Machines (VMs) without affecting the performance of the system. In this work, dynamic scheduling algorithm has been developed in order to balance the workability of all virtual machines. First, we have found the number of overloaded and underloaded Virtual machines using computational formula. Then assigning overloaded machines to underloaded machines in order to enhance the performance of cloud environment, the task to meet the deadline ratio as compared to other existing algorithms and reduce the makespan time and response time. However, the algorithm has been tested on variable number of tasks to achieve better scalability.

Keywords: Cloud computing, Virtual machines, dynamic scheduling algorithm

I. INTRODUCTION

Cloud computing [7] [12] is very vast and virtualized technology. Cloud computing is the delivery of on-demand computing resources over internet on a pay-per-use basis. It is a new computing technology in which renting of storage infrastructure and computing services is achieving. This new technology allows individual to access computing services without burden of installing, maintaining and upgrading computer software.

Cloud computing is a new computing technology in which renting of storage infrastructure and computing services is achieving.

- **Load Balancing**

Load Balancing is a mechanism which distributes the workload on the resources of a node to respective resources on the other node in a network without eliminating any of the running tasks [11]. Load balancing is ensures that no node in the cloud environment is over loaded or under loaded for any instant of time. The responsibility of load balancing algorithm to balancing the load between nodes in system so that overall performance of system is improved as well as efficient utilization of resources.
Datacenter to improve the utilization of services and resources, provide high user satisfaction. Resource overloading degrades the performance of system whereas resource underloading is the causing of wastage of resources. Load balancer avoids the overloading and underloading of resources by distributing the work across all nodes in Datacenter according to their capacity.

II. LITERATURE SURVEY

Komal Singh Patel et al. (2012) presented methods for VM allocation among multiple cloud providers, they have presented two algorithms. First one allocates the resources to VM, so that it improves the cloud service provider’s profit. Second one allocates VM in order to balance the load among the multiple datacenters in federated cloud environment [2].

Yi Han et al. (2014) compare three basic VM allocation policies from a security perspective, and model their performance in the presence of this kind of attack. They find that the Most VM policy performs the best when the servers are appropriately configured, and oversubscription is enabled [16].

B. Mallick et al. (2015) proposed a dynamic algorithm which will manage the load incoming by focusing on their present status at cloudlet for all free VMs to be used at request assignment and will take more requests that are dynamic in nature. The response time has been improved efficiently [9].

Raja Benali et al. (2016) presented an evaluation study of some VM placement and migration policies. To simulate a distributed cloud environment, they have used CloudSim toolkit. They have implemented some extensions in order to take into consideration several aspects such as the VMs communication between different DCs and the inter-DCs migration [8].

Mohit Kumar et al. (2017) modified the architecture of cloud resource broker and developed an efficient dynamic algorithm for task scheduling, which is based on the last optimal k-interval that not only minimizes the makespan time of tasks but also increase the ratio of tasks to meet the deadline and fulfill the objective of elasticity in cloud environment [5].

Yun-Yao Chen et al. (2017) proposed a new cloud load balancing mechanisms by comparing previous studies. The proposed new paradigm CLB for load balancing architecture and an algorithm can be applied to both virtual web servers and physical servers [13].

Mainak Adhikari et al. (2018) proposed a new heuristic-based scheduling and load balancing algorithm for IaaS cloud, referred as HBLBA. The proposed algorithm is divided into two phases, namely server configuration, and task-VM mapping. They have developed a new and efficient strategy to find the best feasible VM configurations [3].

Mohit Kumar et al. (2018) developed a load balancing algorithm to achieve minimum makespan time and increase the ratio of tasks meet to the deadline in cloud environment that considered deadline as QoS parameter in their paper [10].

S.C. Sharma et al. (2018) modified the architecture of cloud resource broker and developed an efficient dynamic algorithm for task scheduling, which is based on the last optimal k-interval that not only minimizes the makespan time of tasks but also increase the ratio of tasks to meet the deadline and fulfill the objective of elasticity in cloud environment [4].

Xun Shao et al. (2018) present a virtual node based, decentralized load-balancing method for range-queriable cloud storage systems. In their method, they partition physical nodes into multiple virtual nodes, and organize the virtual nodes with a range-queriable P2P network [15].

III. STEPS USED FOR THE IMPLEMENTATION OF ALGORITHM

We have proposed a dynamic load balancing algorithm whose objective is to reduce the makespan time and increase the average resource utilization ratio in cloud environment as shown in below implementation steps [5].

Symbol Notation

- UVM [] =under loaded virtual machine array, BVM [] =balanced virtual machine array, OVM [] =overloaded virtual machine array, UM=under loaded machine, OM=overloaded machine.

Task scheduling Operation

Step 1: Generate N number of task and M number of virtual machine along with different properties of processing power in terms of MIPS, RAM etc.

Step 2: Length of each task should be generated randomly; this range lie between 20000MI to 40000 MI.

Step 3: Then sort the task and virtual machine in descending order based on task length and processing speed.

Step 4: Allocation of task has been started to virtual machine in first come first serve order.

Step 5: Each allocated Id of VM has been contained in array.

Step 6: Once the assignment of virtual machines done, then load balancing operation has been started and find number of task, length of each task assigned to individual virtual machine.

Step 7: Compute the load on each VM, total load on datacenter at a particular time t.
Step 8: Then check (upcoming virtual machine < capacity of datacenter)
   (a) Load balancing is not possible, in case of condition false
   (b) load balancing is possible in existing cloud infrastructure; in condition of true

Load Balancing Operation
1. Apply loop in order to perform load balancing operation starting from i=0.
   For(i<VmSize&&cloudeletSize)
2. Compute both Load and capacity of VM; where L refers to Load and C refers to Capacity
3. If(Load<Capacity); then there is no possibility of load balancing. Use the concept of elasticity
4. Otherwise start to monitor each VM.
5. Find out the number of under loaded, balanced and overloaded VM by using the following formula:
   \[ \text{UM} = 0.25 \times C_{VM} \]
   \[ \text{OM} = 0.8 \times C_{VM} \]
6. Now transfer task has been operated.
7. Sort OM in decreasing order and UM in increasing order

Task Transfer operation
8. While OM!isempty&& UM!isempty
   a. For loop for OM
   b. If OM & UM is exist then transfer the task from OM to UM
   c. TI \to VMj \] Until load at VMj \leq OM || VMj \geq UM
   d. Also calculate transfer time of task (TLi / bandwidth).
   After the task transfer check the status of each VM, if any virtual machine is still overloaded condition, repeat load balancing operation again.

V. RESULTS
Following are the graphs that show the average utilization of number of overloaded machine in basic load balancing algorithm with respect to improved load balancing algorithm. Also, transfer time has also been improved in proposed load balancing algorithm.

<table>
<thead>
<tr>
<th>Number of overloaded machine</th>
<th>Basic Load Balancing</th>
<th>Improved Load Balancing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Load Balancing</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>Improved Load Balancing</td>
<td>5</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transfer Time of allocating overloaded machine to underloaded machine</th>
<th>Basic Load Balancing</th>
<th>Improved Load Balancing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Load Balancing</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Improved Load Balancing</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
iv. conclusion

There are numerous types of load balancing algorithms that exist in cloud environment. However, these algorithms are being capable to enhance the workability of virtual machines, besides; the major disadvantage is that the proper utilization of cloud resources. In this work, dynamic load balancing algorithm has been implemented which aims to depreciate both the makespan time and waiting time. Dynamic Load Balancing algorithm is allowed to cloud environment to assign overloaded virtual machines to underloaded virtual machines. Calculation of both underloaded and overloaded machines depend on the computational formula. After this, another algorithm named as Improved Load Balancing strategy has been implemented to reduce the response time, transfer time, and number of overloaded machines. This is done by assigning free processing elements. Experimental results shows that the improved load balancing algorithm increases the overall average resource utilization ratio as compare to basic load balancing algorithm.

References

http://en.wikipedia.org/wiki/Cloud_computing

IJSRCSAMS

Volume 7, Issue 5 (September 2018)