A Virtual Machine Migration Based Load Balancing Model for Optimizing the Profit for the Cloud Provider

Shivani Goel[1]
er.shivani25@gmail.com[1]

Tripti Arjariya[2]
tripti.beri@gmail.com[2]

Varsha Namdeo[3]
exambhabha@gmail.com[3]

***Abstract*-** **Cloud computing is one of the well developing fields in Computer Science and Information Technology. It is industry model where owner of the cloud known as cloud provider want to get maximum profit from their existing infrastructure. To gain more and more profit they used the concept of virtualization which boosts the utilization of the physical resources by dividing the physical resources. Proper distributions of the load on physical resources also help the provider to gain maximum profit. This paper presents another approach which increases the profit of the provider by scheduling the VM effectively. This approach places the VM according to the time of execution and amount of resource used during the execution. CloudSim is use as a simulation tool to measure the performance of the proposed approach. Experiment result says that proposed approach gives better results.**

Key Terms- Distributed Computing, On Demand Resources, Cloud Computing, Virtualization.

I. INTRODUCTION

In the cloud computing, the computing resources are provided to the client through virtualization, on the internet. The large scale computing infrastructure is established by cloud providers to make availability of online computing services in flexible manner so the user find easiness to use the computing services [1]. According to NIST cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources.

The computing resources include networks, servers, storage, applications, and services. In cloud computing, the shared pool of computing resources can be rapidly provisioned and released [2]. The management effort or service provider interaction for cloud user is also minimized to increase easiness. This cloud model is basically composed of five essential characteristics, three types of service models, and four deployment models [3, 4].

Figure 1: Cloud Computing

One of the important technologies of the cloud is the virtualization [5, 6] which divides the hardware resources into the several type according to the user demand. Hence virtualization is the backbone of the cloud technology. Proper distribution of the load on the physical hardware is the prime concern of the provider because resource utilization plays a significant role in minimizing the number of active server and increasing the resource utilization. This paper gives the overview of some existing load balancing approach with their constraints.

II. RELATED WORK

It is necessary to do scheduling of cloud tasks in cloud environment to complete the client job. It should be decided which process need to allocated to which host system. If the computing resources are sufficient in current host system to complete the execution of process within deadline then no need to move or share the resources of other host. In this paper we have focused on various task scheduling techniques used in hybrid cloud environment with the aim of cost minimization and optimization for cloud resources. Some of the basic techniques are next part of the paper.

1.1 A Task Scheduling Algorithm Based on Load Balancing in Cloud Computing [7]

Cloud Computing Architecture includes three layers, application layer, platform layer and infrastructure layer. The application layer is oriented to users; it implements the interaction mechanism between user and service provider with the support of platform layer. Users can submit tasks and receive the results through the application layer in the task scheduling process. The infrastructure layer is a set of virtual hardware resources and related management function. Furthermore, the platform layer is a set of software resources with versatility and reusability, which can provide an environment for cloud application to develop, run, manage and monitor.

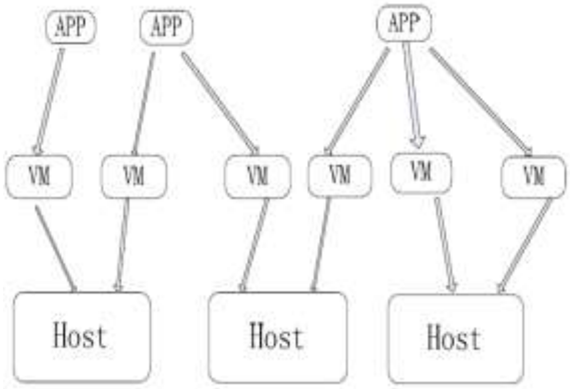

Figure 2: Job Allocation Policy in Cloud Server

So according to the above architecture, they are using two levels scheduling model. The first level scheduling is from the users' application to the virtual machine, and the second is from the virtual machine to host resources.

In this two levels scheduling model, the first scheduler create the task description of a virtual machine, including the task of computing resources, network resources, storage resources, and other configuration information, according to resource demand of tasks. Then the second scheduler find appropriate resources for the virtual machine in the host resources under certain rules, based on each task description of virtual machine.

Algorithm is divided into two levels scheduling, one is the mapping from task to a virtual machine, another is mapping from the virtual machine to host resources. Only task response time and the demand for resources are considered in this paper. At the same time, because tasks are dynamic, they may arrive randomly. If the tasks arrive at same time, they will be sorted ascending according to the resource applied by users. And if they arrive at different time, they will be sorted according to the time sequence arrived.

In the above algorithm, the virtual machine is scheduled to the host with lightest load each time. The advantage is to avoid overloading for the host hold more resources. Recent studies [2] show that on average an idle server consumes approximately 70% of the power consumed when it is fully utilized. And it only task response time and the demand for resources are considered in this paper.

1.2 A Load Balancing Model Based on Cloud Partitioning for the Public Cloud [8]

There are several cloud computing categories, with this work focused on a public cloud. A large public cloud will include many nodes and the nodes in different geographical locations. Cloud partitioning is used to manage this large cloud. A cloud partition is a subarea of the public cloud with divisions based on the geographic locations.

In this paper they are using global and local resource manager. Global resource manager (Main controller) installed in the main server and local resource manager (Partition controller) installed into the local host. When a job arrives at the system, Global resource manager decide which cloud partition should receive the job.

The partition load balancer (local resource manager) then decides how to assign the jobs to the nodes. When the load status of a cloud partition is normal, this partitioning can be accomplished locally. If the cloud partition load status is not normal, this job should be transferred to another partition.

The load balance solution is done by the main controller and the balancers. The main controller first assigns jobs to the suitable cloud partition and then communicates with the balancers in each partition to refresh this status information.

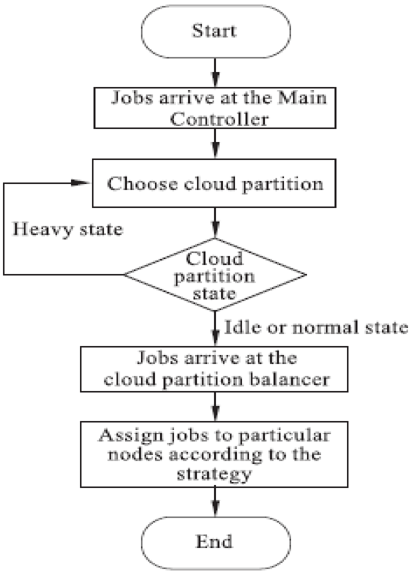
Start
Jobs arrive at the Main Controller
Choose cloud partition
Heavy state
Cloud partition state
Idle or normal state
Jobs arrive at the cloud partition balancer
Assign jobs to particular nodes according to the strategy
End

Figure 3: Job assignment strategy

When a job reached to the scheduler, the first job of the main controller is to select the correct partition. Based on the load cloud partition status categorized into three types:

(1) Idle: When load on the cloud is more than α Then PM is considered as an idle.
(2) Normal: When the percentage load on the cloud is more than β then that partition is considered as a normal load cloud partition.
(3) Overload: When the percentage of the overloaded nodes exceeds γ, change to overloaded status.

The parameters α, β and γ are set by the cloud partition balancers.

When this job reaches to the local controller it places it. If this location's status is idle or normal, the job is handled locally. If not, another cloud partition is found that is not overloaded. The node load degree is related to various static parameters and dynamic parameters. The static parameters include the number of CPU's, the CPU processing speeds, the memory size, etc.

Dynamic parameters are the memory utilization ratio, the CPU utilization ratio, the network bandwidth, etc. In the above algorithm, the virtual machine is scheduled to the cloud partition with lightest load each time. The advantage is to avoid overloading for the host hold more resources. Recent studies show that on average an idle server consumes approximately 70% of the power consumed when it is fully utilized.

1.3 Compare And Balance Algorithms [9]

This algorithm uses the concept of compare and balance approach to reach equilibrium condition and manage resource management. In this algorithm, work load of each host server is calculated firstly and then sum of total load on the all host of the systems is calculated. After that any host with the probability of p[k] (where p[k] is the probability of a host having minimum virtual machine) is selected. If the load of selected host server is less then as compare to current node than the load of current host server is transferred to chosen host.

The working approach is presented in algorithm 1. On the basis of probability (number of virtual machine running on the current host and number of virtual machine running in the whole cloud system) each host server randomly selects a host server until it find appropriate host system to transfer their extra load. This process execute continuously so it consume large number of processing power. The above model is only applicable for load balancing of cloud system.

1.4 Adaptive Threshold Based Energy Efficient Consolidation of Virtual Machines in Cloud [10]

The target system, designed by this approach, is an IaaS environment, represented by a large-scale data center consisting of N heterogeneous physical nodes. This approach focuses on multi-core CPU architectures, as well as consideration of multiple system resources, such as memory and network interface, as these resources also significantly contribute to the overall energy consumption.

In order to evaluate the proposed system in a real Cloud infrastructure, besides the reduction in infrastructure and on-going operating costs, this work also has social significance as it decreases carbon dioxide footprints and energy consumption by modern IT infrastructures.

1.5 A Novel Approach Towards Improving Performance of Load Balancing Using Genetic Algorithm in Cloud Computing [11]

This approach gives the different solution to the provider to enhance their profit. They argue cloud is the utility model where user need to pay only for the used resources just like a electricity bill. Amount of the bill generated by the provider to client id propositional to the size of the requested VM and the time which is required by the VM to execute the user job. Cost of the bill is calculated by the following equation:

$$\zeta = w1 * \alpha(NIC \div MIPS) + w2 * L$$

Where $w_1$and w_2 are the weighting coefficient
NIC shows the number of instruction in code
MIPS is the million instruction per second
α is the cost of the resources
L is the cost of delay.

Now to place the VM first they find the value of ζ then place the VM according to the value of ζ. higher value of ζ represent that VM gives more profit to the provider so must be place before the VM which has lower value of ζ. This approach not provides any the solution for the migration.

III. PROPOSED WORK

Cloud is the model where the main concern of the provider is to increase their income with the existing infrastructure. For this purpose provider use the concept of the virtualization which increase the hardware utilization by sharing the single resources to multiple user. This approach increases the income of the provider by serving the maximum number of user.

In other hand, M. Pilavare et al. proposed the other solution to boost the provider income. They think that provider can increase their income by placing those VM which usage more resource for long time because these user gives more revenue to the provider. But this approach not uses the concept of virtualization. In our proposed work we also use the same concept to place the VM. Moreover we also imply the concept of migration to balance the load on the physical node and minimize the service level agreement (SLA) violation.

Invoice generated by the provider is proportional to the time and size of the VM. If size of the VM is V and cost of the VM is C/sec. Then the cost of the VM is VC. How much time is taken by the VM to complete the user process is depends on the number of lines in code. So first we find the value of time for which VM uses the cloud services. Following equation are used to measure time VM want for completing the user task

$$T = \frac{Numbeer\ \ of\ lines\ in\ code}{MIPS\ of\ the\ VM} + D$$

Where- D is the delay time. Above equation can also be written as:

$$T = \frac{Numbeer\ \ of\ lines\ in\ code}{Requested\ \ MIPS\ of\ the\ VM}$$

Where T shows the time for which user use the cloud resources. Now we find the size of all available VM and then calculate the amount paid by the user for using the cloud resources. Then assign the VM first which produce more profit to the provider.

T_i avoid the circumstance where lower profit VM will never get their chance to place we will increase their priority by 1 so that after some time this VM will also place in the PM. When the load on the PM is below the lower bound and higher than the upper bound or threshold then PM is unbalance and VM migration is started which migrates all VM from the unloaded PM and lower priority or lower profit VM will be migrated in case when the PM is overloaded.

For placing the VM income of the VM is calculated then place the VM according to the income or profit. If t is the time for which VM is used and C_{cpu}, C_{ram} and C_{bw} are the cost of the VM then following equation is used to find the value of the cost

$$C= (t^*\ C_{cpu})^*\ (t^*\ C_{ram})^*\ (t^*\ C_{bw})\ \text{-------------(1)}$$

Following algorithms are used to select and place the VM to the PM.

Algorithm for the VM Selection

```
pmList ← List of available PM
    for each PM in the pmList do
        $PM_{load}$ ← Load of the PM
        if $PM_{load} < 20$
            vmList ← List of all VM in the underloded PM
                Migrates all VMS
            end if
        if $PM_{load} > 80$
            Select the VM with lower priority for the migration
        end if
    end for
```

Algorithm for the VM Placement

```
vmList ← List of all VM need to be placed
pmList ← List of available PM
for all VM in vmList do
    Calculate the priority of the VM
end for
for all VM in vmList do
    Select VM with higher priority
        if VM = new do
            for all PM in pmList
                select the largest PM
            end for
        else
            for all PM in pmList
                $PM_{load}$ ← Load of the PM
            if $PM_{load} > 20$ && $PM_{load} < 80$
                Add PM to the pmList1
            end if
            end for
        for all PM in the pmList1 do
            Select the largest PM
        end for
    end for
```

IV. RESULT ANALYSIS

Implementation of the load balancing approach is really difficult in real environment because they required large infrastructure to share the resource among multiple user. Due to this reason we use the simulation tool to evaluate the performance of the proposed approach. There numbers of tools are available for this purpose like CloudSim simulator [13], Cloud Analysis etc. But we are using CloudSim simulator for performing our work.

To form the cloud setup we create the 20 PM. These PM are heterogeneous in nature and having 1000, 2000 and 3000 of CPU capacity which is measured in MIPS. These PM have 10000MB RAM and 100000 bits/sec bandwidth. Several VM are created during the experiment like 30, 35, 40 and 45.

These VM can request 250, 500, 750, 1000 MIPS and 128 MB RAM and 2500bits/sec bandwidth. To measure the efficiency it is compare with already present VM scheduling approach [13] and check the number of PM required to place the VM and energy consumption approach. Our approach gives improved results because we place the VM to the largest PM whereas base approach schedule VM to the first PM which has the sufficient resource to place the VM. So our approach reduces the number of server which needs to place the VM.

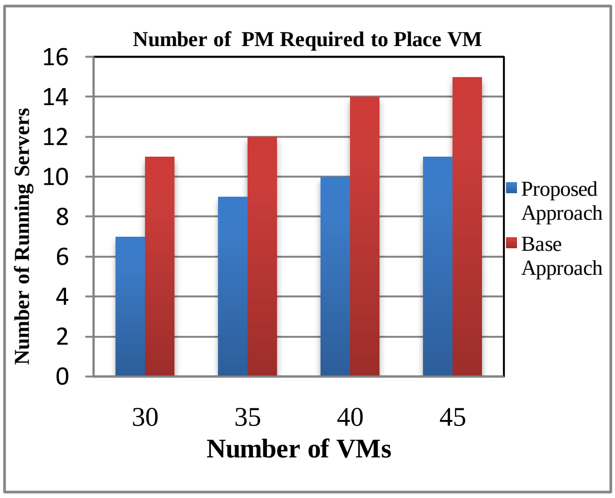

Figure 4: Number of Required PM to Place VM

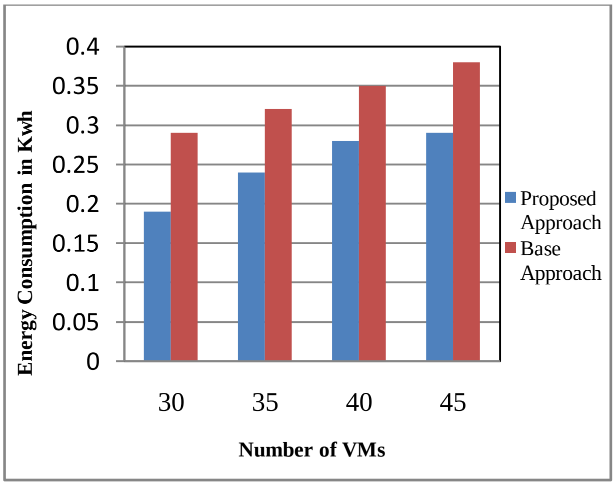

Figure 5: Energy Consumed by the Data Center

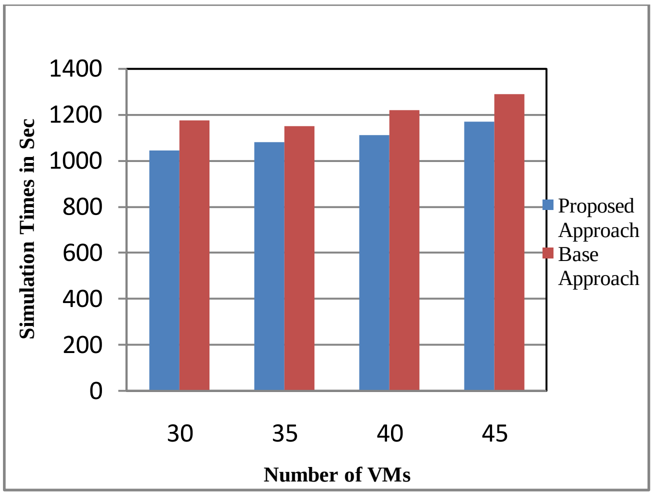

Figure 6: Total Simulation Time

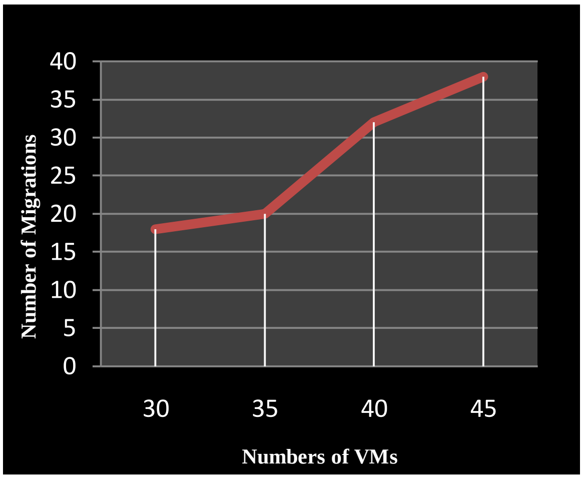

Figure 7: Number VM Migrations

V. CONCLUSION AND FUTURE WORK

Cloud computing have large number of resources to distributes their resources on demand. When the user requests for the resources, virtual machine manager creates the VM and assign to the user.

VM works similar to the PM. Each PM can have number of VM. So the proper placement of these VM is very challenging task due to the unpredictable nature of the VM. Since load on the VM can change dynamically, so there is a need of an effective dynamic load balancing algorithms that place the VM effectively. Several load balancing approaches was introduced in the past few years which increase the resource utilization.

This paper proposed an approach which helps the provider to increase their income with the existing infrastructure. Experiments result says that our approach optimize the performance of the cloud services by minimizing the total simulation time.

This approach is implemented in CloudSim simulator, so real private cloud can also be used for evaluating the performance of the proposed approach.

REFERENCES

1. R.K. Gupta and R.K. Pateriya, “A Complete Theoretical Review on Virtual Machine Migration in Cloud Environment”, International Journal of Cloud Computing and Services Science (IJ-CLOSER), Vol.3, No.3, June 2014, pp. 172-178.
2. Peter Mell,Timothy Grance, "Cloud Computing" by National Institute of Standards and Technology - Computer Security Resource Center-www.csrc.nist.gov.
3. R.K.Gupta and R.K. Pateriya,” Survey on Virtual Machine Placement Techniques in Cloud Computing Environment”, International Journal on Cloud Computing: Services and Architecture (IJCCSA) ,Vol. 4, No. 4, August 2014, pp. 1 -7.
4. R. Santhosh and T. Ravichandran, “A Survey on Cloud-Based Scheduling Algorithms”, International Journal of Communications and Engineering (IJCE), Volume 01– No.1, Issue: 01 May2013.
5. “Hypervisor,” [Online] available: http://en.wikipedia.org/wiki/ hypervisor, June 2014
6. “Xen, virtual machine manager in Cloud computing,” [online] available: http://www.xen.org, April 2013.
7. Y. Fang, F. Wang, and J. Ge, “A Task Scheduling Algorithm Based on Load Balancing in Cloud Computing”, Web Information Systems and Mining, Lecture Notes in Computer Science, Vol.6318, 2010, pages 271-277.
8. G. Xu et al. “A Load Balancing Model Based on Cloud Partitioning for the Public Cloud” in the proceeding of IEEE conference on TSINGHUA SCIENCE AND TECHNOLOGY, pp. 34-39, 2013..
9. Yi Zhao, Wenlong Huang, "Adaptive Distributed Load Balancing Algorithm based on Live Migration of Virtual Machines in Cloud" Fifth International Joint Conference on INC, IMS and IDC, IEEE, 2009, pp: 6-09.
10. Anton Belaglozav, R. Buyya, “Adaptive Threshold-Based Approach for Energy-Efficient Consolidation of Virtual Machines in Cloud Data Centers”, Proceedings of the 8th International Workshop on Middleware for Grids, Clouds and e-Science, ACM 2010.
11. Mayur S. Pilavare and Amish Desai, " A Novel Approach Towards Improving Performance of Load Balancing Using Genetic Algorithm in Cloud Computing", IEEE Sponsored 2nd International Conference on Innovations in Information Embedded and Communication Systems ICIIECS’15, pp. 1-4 , March 2015
12. K. Dasgupta et al.,“A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing“ in Proc. of Elsevier, Procedia Technology 2013.
13. R. Calheiros, R Ranjan, César A. F. De Rose, R. Buyya, “ CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services” , 2011.