IJCSE International Journal of Computer Sciences and Engineering Open Access
Survey Paper Vol.-6, Issue-12, Dec 2018 E-ISSN: 2347-2693

A ProperFit Virtual Machine Migration Approach for the Load Balancing in Cloud

[1*]Damodar Tiwari, [2]Shailendra Singh, [3]Sanjeev Sharma

[1]Dept. of Computer Science & Engineering Bansal Institute of Science and Technology Bhopal, India
[2]Dept. of Computer Engineering & Applications N National Institute of Technical Teachers Training and Research Bhopal, India
[3]Dept. of SOIT Rajiv Gandhi Proudyogiki Vishwavidyalaya Bhopal, India

**Corresponding Author: damodarptiwari21@gmail.com*

Available online at: www.ijcseonline.org

Accepted: 24/Dec/2018, Published: 31/Dec/2018

Abstract- With the invention of the cloud computing, the utilization of the physical resources has improved drastically. The main technology that enable the cloud computing is virtualization which allows to create several virtual machine (VM)onto the single physical machine (PM). It increased the utilization of the physical resources because single hardware resources are shared by the several users. Although virtualization technique optimize the server utilization but add new issue named load balancing that need to addressed for the effective utilization of the physical resource and maintain the quality of services (QoS). To deal with the load balancing VM migration approach is used which permit to travel the VM from physical machine (host) to another. Three stages are engaged with the relocation procedure i.e., source PM choice, VM selection and the last step is target PM selection. Plenty of work on the load balancing in cloud are presented in the last few decades and mostly they are differ in the VM selection and VM placement polices. After the study of previous work on the VM migration it can be says that choosing an appropriate VM is a non-trival task and the performance of the load balancing approach is mainly depends on the appropriate VM selection polices.

In this paper we select the three different types of virtual machine for the migration and then placed it to the physical machine where the load on the physical machine is between 20 to 50. CloudSim simulator is used to evaluate the performance of the physical.

***Keywords*:-**Migration, energy efficient, virtualization, VM selection, VM placement, SLA violation.

I. INTRODUCTION

Cloud computing take a new revolution IT industries. It offers all computing resources as a services that can used any geographical location in the word. One of the most beauty of the cloud is that these services are very user friendly and provide the power of super computer on a mobile also.Because of several feature it become a first choice of the user [1]. Figure 1 shows the services which are offers by cloud and deployment model of the cloud computing.

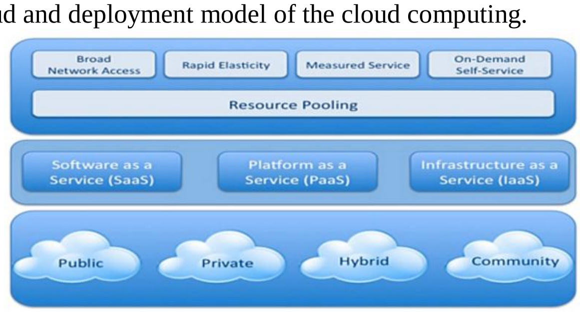

Figure 1: Cloud Computing Model

Cloud gives the figuring assets to the customer as an administration as virtual machine (VM). VM is the legitimate substance which is like the PM and executing the client application. The big scale computing infrastructure is established by cloud vendors to make availability of on-line computing services in bendy manner so that the user discover easiness to use the computing offerings [3]. Because the needs for the computing resources is growing, proper resource consumption of the physical resources is primary challenge for the provider. As cloud resources are shared by the numerous customers and the needs for the sources is trade regularly, so there may be a demand for a treasured load balancing method that enlarge the resource utilization and increase the performance of the cloud offerings.

If the load or the cloud application are not distributed properly than it will degrades the performance of the cloud services. For this purpose VM migration [5, 6] strategy is used which allow the movement of the VM from one PM to another. But VM migration is a challenging task because before triggering the VM, provider must known which VM required to be migrated and wherever migrated VM is boing be placed. In the past decade numerous load balancing approach have been proposed which uses different VM selection and placement polices.

II. RELATED WORK

A. Awad et al. [7], proposed Enhanced Particle Swarm Optimization approach for scheduling the task in cloud environment. For this purpose they design a mathematical model which used the Load Balancing Mutation for scheduling the task. This approach mainly considered the reliability and availability during the placement of the user task.

G.Shobana et al. [8], suggested aload balancing technique for cloud which is based on the preemptive task scheduling. This technique makes use of CPU and bandwidth as a selection metrics for calculation load at the VM. According this approach, load of the PM is equivalent to the summation of load of all VM running to the PM. In this approach VM are grouped into three types named Overloaded VMs (OVM), Under loaded VMs (UVM) and Balanced VMs (BVM) based on their load. At the point when the PM is over-burden organize all VM in OVM group in descending order whereas all VM in UVM arrange in ascending order. One task is remove from the OVM, VM is and place to the any VM in the UVM group. This approach migrate the task from one VM to another VM to balance the load. But this approach is not effective for the cloud because task is move from one VM to another VM. In this case load on the PM is remaining same hence PM is still overloaded.

A. Rabiatul et al. [9], suggested a load balancing method which uses VM migration approach to balance the load. This method set the value of lower and upper limit for the resource utilization of the PM are 10 and 90 respectively. When the load is above the upper limit larger VM is selected from the overloaded PM and put it to the host where resource utilization of the host is below 50. This approach seems good but may increases the number of migration due to setting higher value of the upper threshold.

Lei Xu et al.[10], proposed an load balancing approach for the cloud. This method provides the solution for the various steps involved in the VM migration i.e., when to trigger migration process, which VM is useful for the migration and where selected VM going to be places. It is an integrated framework where the primary goal is to deal with the various existing issue in cloud like load balancing, server consolidation and hotspot mitigation.

Y. Fang et al. [11], proposed task scheduling model for the VM in cloud. They proposed the two layer architecture for the VM placement. First layer gives the description of the VM and second layer assign the resource to the VM. VM is assign to the smallest PM. When the hot spot or load un balancing situation is occurring they select the smallest VM for the migration and place it to the lightest loaded PM. This approach selects the small VM for the migration which may increase the number of migration. In addition, they are also not focus on the server consolidation.

III. PROPOSED WORK

After the study of various load balancing approaches, it can be concluding that most of the load balancing approaches are differ in their VM selection and VM placement policies. Hence, these two steps play a vital role in the performance of the load balancing approach. Due to this reason primary goal of our proposed method is to find the appropriate VM after triggering the migration and located it to the suitable PM.
Whenever load of the PM is cross the certain threshold some VM need to be moved to the alternative PM in-order to balance the PM. The migration approach consists of three steps. These steps are selecting overloaded or underloaded PM, find the VM for the migration and the final step is to locate the appropriate PM to host this VM.

I. Select the Overloaded or Underloaded PM

To find the abnormal PM, lower and upper limits (thresholds) are set which recognized the overloaded or underloaded scenario. If the physical resource consumption is less than the lower limit then PM is consider as a underloaded PM and if the utilization is greater than the higher limit then PM is considered as a overloaded PM. In our approach we use 20 and 80 as a lower and upper threshold.

II. VM Selection

If the VM not utilizing the physical resources properly then the resources of the PM is wastage. So to increase the utilization of the physical resources these PM must release their load, so that PM can be shutdown. Due to this when the PM is under loaded then all VM operating on the PM are shifted to the other PM.

Since VM selection affect the overall performance in term of total migration time (TMT) and down time (DT), so which VM is migrated is the critical job when the PM is overloaded. It has been observed that choosing the larger VM selection may amplify the TMT and DT whereas choosing the smaller VM lead to increase the number of VM migration which tend to increase SLA violation. Hence the proper VM selection is the prime requirement of any load balancing approach. In our approach we select the different type of VM i.e., small size VM, large size VM and best fit size VM, for the migration and then check the effect of various VM selection.

Biggest VM Selection for the migration

1) HostList ← {Available physical machine or host list}
2) vmList ← {Available virtual machine list}

3) for all host in HostList
4) if ($host_{util}$ < T_L) then
5) Migrate all VM form the selected host
6) end if
7) if ($host_{util}$ > T_U) then
8) choose biggest VM from the selected host
9) end if
10) end for

Smallest VM Selection for the migration

1) HostList ← {Available physical machine or host list}
2) vmList ← {Available virtual machine list }
3) for all host in HostList
4) if ($host_{util}$ < T_L) then
5) Migrate all VM form the selected host
6) end if
7) if ($host_{util}$ > T_U) then
8) choose smallest VM from the selected host
9) end if

ProperFit VM Selection for the migration

1) HostList ← {Available physical machine or host list}
2) for all host in HostList
3) $host_{util}$ ←host.getUtil()
4) if ($host_{util}$ < T_L) then
5) Migrate all VM
6) end if
7) if ($host_{util}$ > T_U) then
8) diff ← $host_{util}$ − T_U
9) vmList ← {All available virtual machine in selected host }
10) for all VM in the vmList (all VM in overloaded PM)
11) if VM_{util} > diff
12) Add VM to the VmMigList1
13) end if
14) end for
15) Sort VmMigList1 in to the increasing order of their value
16) Select first VM
17) End if
18) End for

Here VM_{util} and $host_{util}$ denotes utilization of VM and PM respectively, T_U and T_U denotes upper and lower limits of the host utilization.

III. Finding the PM for the Placement

After selecting the VM we need to select the suitable host for placing the VM. Proper selection of the PM is the challenging task. If the wrong PM is selected for the placement then it will increase the number of active server and number of VM migrations. In our approach we select the PM whose load is greater than 20 and less than 50. Reason for selecting this host is to minimize the number of active host and number of migrations.

VM Placement

1) vmList ← {List of all migrated VM available for the placement}
2) HostList ← {List of existing PM }
3) while (vmList != Null) do
4) for each host in the HostList do
5) if (T_L>host< T_U)
6) insert host to the HostList-1
7) end if
8) end for
9) if pmList-1 = Null
10) Actuate new PM
11) else
12) for all PM in the pmList-1
13) L_1 ← {Calculate host load before assigning the VM}
14) Assign VM to the PM
15) L_2 ← {Calculate host load after assigning the VM}
16) diff ← L_2 - L_1
17) Add diff into the diffList
18) end for
19) end if
20) Based on the diff value assemble all PM of PMList-1 in ascending order
21) Allocate VM to the first PM
22) Finally updates all resource of host like CPU, RAM and Bandwidth
23) end while

IV. SIMULATION RESULT

CloudSim simulation tool [12, 13] is used to evaluate the performance of the proposed approach. To check the efficiency of the proposed approach it is compare with the existing load balancing approach [11]. Efficiency of the proposed and competitive approach is measured in term of number of migrations, energy consumption and number of active server.

To create the cloud environment 10 number of PM is create with MIPS of 1000, 2000 and 3000 and size of RAM and bandwidth is 10000 MB and 100000 bit/sec respectively. Numbers of created VM during the experiment are 12, 15, 18 and 20.

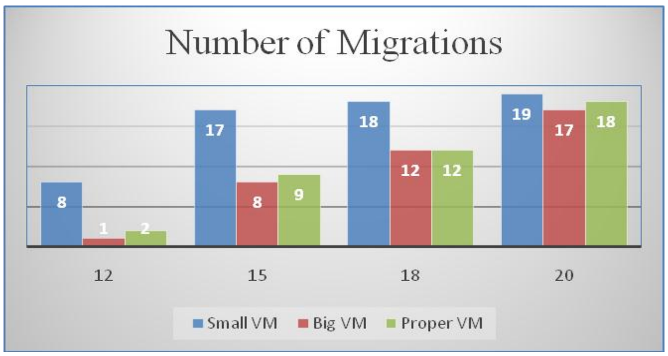

Figure 2: Number of Migrations for Small, Big and ProperFit SizeVM

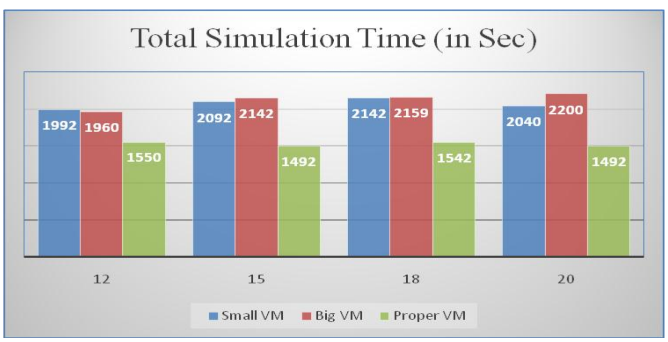

Figure 3: Total Simulation Time for Small, Big and ProperFit SizeVM

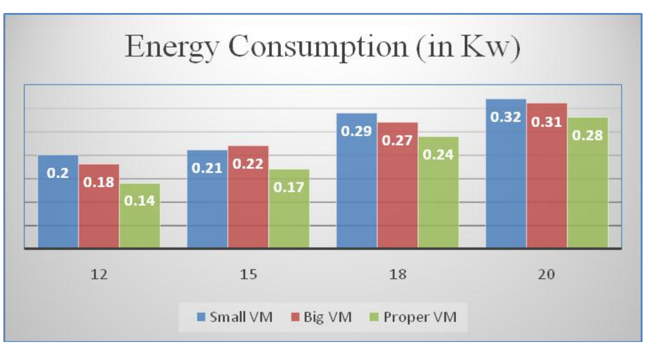

Figure 4: Electricity Consumption for Small, Big and ProperFit SizeVM

V. CONCLUSION

VM migration is the unique solution for mitigating the load balancing problem. The migration approach consist of three steps. These steps are selecting overloaded or underloaded PM, find the VM for the migration and the final step is to locate the appropriate PM to host this VM. Previous study says that choosing proper VM and allocate to the proper PM are the two difficult jobs in cloud and the performance of any load balancing approach is totally depends on these two steps. Larger VM selection may increase the TMT and DT whereas choosing a smaller VM raise the number of VM migration which lead in more SLA violation. Hence the proper VM selection is the prime requirement of any load balancing approach. In this paper we select the various type of VM and then compare them in term of migration, energy consumption and simulation time. Based on the result we can conclude that best fit VM gives the better result as compare to the small and large size VM.

REFERENCES

[1] R. Buyya et al., "Market Oriented cloud computing: vision, hype and reality for delivering it services as computing utilities," CoRR, 2008.

[2] P. Mell et al, "NIST definition of cloud computing," vol. 15, October 2009.

[3] Introduction to Cloud Computing, available at http://www.webopedia.com/TERM/C/cloud_computing.html

[4] Rajeev Kumar Gupta and R.K. Pateriya," Balance Resource Utilization (BRU) Approach for the Dynamic Load Balancing in Cloud Environment by using AR Prediction Model ", Journal of Organizational and End User Computing (JOEUC), Volume 29, Issue 4, Article 2, 2017.

[5] A. Rabiatul et al., "Virtual Machine Migration Implementation in Load Balancing for Cloud Computing", six IEEE international conference, 2014.

[6] A Jain et al., "A Threshold Band Based Model For Automatic Load Balancing in Cloud Environment", in proc. of IEEE International Conference on Cloud Computing in Emerging Markets, pp 1-7, 2013.

[7] A. Awad et al., " Enhanced Particle Swarm Optimization For Task Scheduling In Cloud Computing Environments", International Conference on Communication, Management and Information Technology (ICCMIT), ScienceDirect, pp. 920 – 929, 2015.

[8] G.Shobana et al., "Nature Inspired Preemptive Task Scheduling for Load Balancing in Cloud Datacenter", International Conference on Information Communication and Embedded Systems (ICICES), pp. 1-6, 2014.

[9] R. Addawiyah et al., "Virtual Machine Migration Implementation in Load Balancing for Cloud Computing", six IEEE international conference, 2014.

[10] Lei Xu et al., "An Integrated Dynamic Resource Scheduling Framework in On-Demand Clouds", journal of Information Science and Engineering, pp. 1-8, 2015.

[11] 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.63, pages 271-277, 2010.

[12] 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.

[13] B. Wickremasingh et al.. "A CloudSim-based Visual Modeller for Analyzing Cloud Computing Environments and Applications", in: Proceedings 24th International Conference on Advanced Information Networking and Applications (AI NA), 2010.

