International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 06 | June -2017 www.irjet.net p-ISSN: 2395-0072

Load Balancing in Cloud Computing Through Virtual Machine Placement

Er. Anu Sharma [1], Er. Nancy Arora[2]

[1]*M.Tech. Student, Computer Science & Engineering, International Institute of Engineering & Technology, Samani, Kurukshetra, Haryana, India*
[2]*Assistant Professor, Computer Science & Engineering, International Institute of Engineering & Technology, Samani, Kurukshetra, Haryana, India*

Abstract - *Due to flexibility and timely availability of desired resources, cloud computing is getting more and more popular day by day. By providing the specified virtual machine as per requirement, requests are entertained. There will be violation of service level agreement if virtual machine placement will exceed the specified time or desired virtual machine is not available. In the dynamically changing environment, it is very difficult to predict the number of customer and their requirements. Virtual machine is the prime resource. So placement of virtual machine will be done in such a way that resources should be utilized efficiently and there should be no violation of service level agreement.*

In the present work, author has given a binary search tree based virtual machine placement approach to optimize the resource utilization, minimize the resource allocation time and minimize the service level agreement violation. Cloudsim simulator has been used to analyze and compare the results of binary search tree based virtual machine placement strategy with other strategies found in literature.

Key Words: **Cloud Computing, Load Balancing, Cloudsim, Virtual Machine Placement**

1. INTRODUCTION

Cloud computing may be defined as the way in which we can store data or information permanently on servers and temporarily cache them on the client side through laptops, computers, sensor, etc."

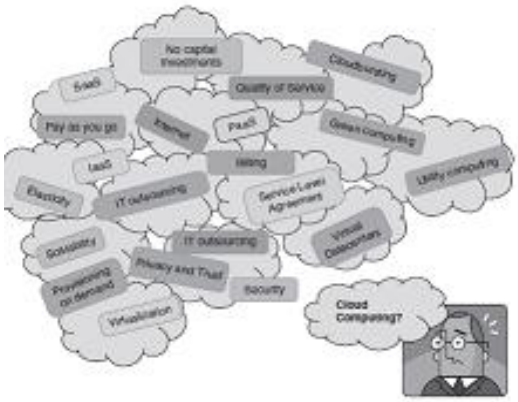

Fig -1: Cloud computing

Figure 1 shows that the cloud computing act as a utility where users are able to access services from anywhere in the world. Cloud computing is briskly turning the tables towards flourishing the software for millions of customers to deplete it like a service, instead of run on their individual computers [1, 2].

2. RESOURCE ALLOCATION

Concept of resource allocation is central point of attraction in parallel and distributed computing. The prime aim of resource allocation is to ensure the allotment of right resource to task at right time. While it also consider the optimize utilization of resources. It also ensures that no resource is over and under-utilized. Other issues associated with resource allocation are the minimization of cost and energy consumption. Resource allocation is implemented to achieve load balancing. Virtual machines in cloud environment are the prime resources. Their mapping with host machine involves the resource allocation. If there are x number of virtual machine at any time interval (0,t] and number of host machine are y, then among the total xy number of VM host combination, we have to chose that one which will be the best. Higher the number of combination, more time will be required to choose the best combination. So aim of resource allocation algorithm is to minimize the number of combination to be tested while also satisfying all the constraints [3, 4].

2.1 VM Placement Algorithm Classifications

Virtual machine deployment is the way of placing desired virtual machine on the suitable physical machine. The goal behind virtual machine and host mapping is to optimize the resources, minimize the VM movement, energy consumption and service level agreement violation. Based upon the goal of VM placement, placement algorithm can be divided into three categories:

- **Quality of service based approach:** When the goal of VM placement approach is to improve the quality of services i.e. minimize the service time, maximize the throughput and minimize the service level agreement violation then that VM placement approach is called quality of service based

approach. This type of approach results in efficient utilization of resources.

- **Power based approach:** When the goal of VM placement approach is to minimize the power consumption then that VM placement approach is called power based approach. This is needed due to following factors:
  - More power consumption will result in more cooling cost.
  - Price of power is increasing day by day.
- **Computation based approach:** Depending upon the nature of computation condition, we can have static or dynamic VM placement approach. When the VM placement is done according to some predefined rule, then it is called static VM placement approach. While, when it is based upon the present conditions then it is called the dynamic VM placement approach [3].

2.2 VM Placement Approaches

- **Rank Based Approach:** This approach is used by open source cloud platform Open Nebula. Host machines are assigned some pre defined priority. While mapping VM with host machine, scheduler considers the priority of host machine. Input to this algorithm is predefined rank and task requirement while output is mapping of VM with host [5].
- **Greedy Algorithm:** This approach is used by open source cloud platform Nimbus and Eucalyptus. This approach is very simple in nature and easy to implement. The host machine which can run the user defined configured virtual machine found first is selected to map with virtual machine. Input to this algorithm is virtual machine requirement and output is host machine id [6].
- **Round Robin Algorithm:** This approach is used by open source cloud platform Nimbus and Eucalyptus. In this approach task requirement is not considered. VM manager stores the id of host which has been mapped with VM. On arrival of next task, VM is mapped with the next host in the list [7].

3. CLOUDSIM SIMULATOR

Cloudsim is free and open source software available at http://www.cloudbus.org/CloudSim/. It is a code library based on Java. This library can be directly used by integrating with the JDK to compile and execute the code. To develop and test the applications quickly, Cloudsim is joined with Java-based IDEs (Integrated Development Environment) including Eclipse or NetBeans. Using Eclipse or NetBeans IDE, the Cloudsim library can be accessed and the cloud algorithm can be implemented. The Cloudsim library is used for the simulation of the following operations:

- Large scale cloud computing at data centers
- Virtualized server hosts with customizable policies.
- Support for modeling and simulation of large scale cloud computing data centers.
- Support for modeling and simulation of virtualized server hosts, with customizable policies for provisioning host resources to VMs.
- Support for modeling and simulation of energy-aware computational resources.
- Provide facility of simulation and modeling of data centre network topologies and message-passing applications.
- Support for modeling and simulation of federated clouds.
- Support for dynamic insertion of simulation elements, as well as stopping and resuming simulation.
- Support for user-defined policies to allot hosts to VMs, and policies for allotting host resources to VMs.
- User-defined policies for allocation of hosts to virtual machines [11].

4. RELATED WORK

Xu Gaochao et al. in [8] have presented a novel heuristic based live virtual machine migration policy for cloud environment by combining the artificial bee colony (ABC) idea with the uniform random initialization idea, the binary search idea, and Boltzmann selection policy to achieve an improved ABC-based approach with better global exploration's ability and local exploitation's ability. They have also used the Bayes theorem to further optimize the improved ABC-based process to faster get the final optimal solution. As a result, the whole approach achieves a longer-term efficient optimization for power saving. The experimental results demonstrate that PS-ABC evidently reduces the total incremental power consumption and better protects the performance of VM running and migrating compared with the existing research. It makes the result of live VM migration more high-effective and meaningful.

A. Mevada et al. in [9] have presented an Enhanced Energy Efficient Virtual Machine Placement Policy for Load Balancing in Cloud Environment. Authors have proposed a modified version of power based VM placement algorithm for reduce energy consumption, better load balancing and optimized VM placement. The proposed algorithm is yet to be implemented and tested under various real-time or simulation environment.

S. Bose et al. in [10] have presented an energy aware cloud load balancing technique using dynamic placement of virtualized resources. Their algorithm minimizes the total number of running servers and improves the resource utilization. Also they have introduced a method for dealing with the problem of overloading while reducing the energy used. Through experimental evaluations it is demonstrated that the proposed algorithm is able to achieve the substantial

reduction in energy consumption and improves the overall resource utilization.

5. PROPOSED WORK

A binary search tree based virtual machine host mapping scheme has been proposed to optimize the resource utilization along with minimizing the violation of service level agreement. Figure 2 shows the framework for proposed scheme.

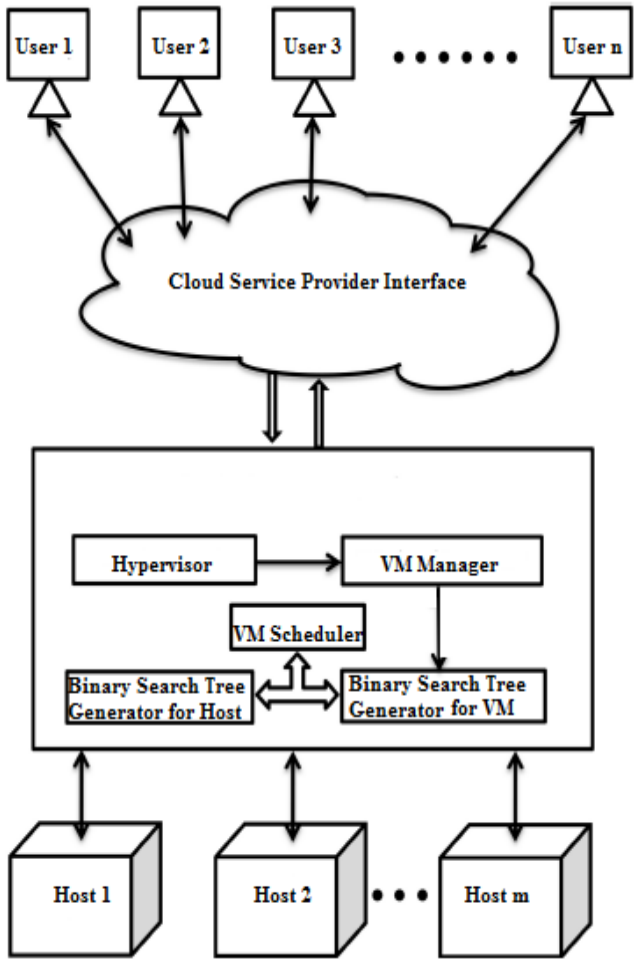

Fig -2: Framework for binary search tree based VM host mapping approach

User forwards their task through interface of cloud service provider over internet towards CSP. Depending upon the requirements of task received by Hypervisor module, virtual machine with desired specification is generated. VM manager keeps track of all live virtual machines. Through VM manager, binary search tree containing the list of all pending VM is generated. Resource Requirement to a VM is used to generate the binary search tree. This binary tree is forwarded towards VM scheduler. VM scheduler selects the highest resource required virtual machine. As binary search tree has been used to maintain the list of available VM, so it will require at most log (N) time to search the node. Here N is the number of nodes in binary search tree. Binary search tree for available host machines is also generated based upon the availability of resources. This will ensure the best mapping of VM with desired host in log (M) time where M is the number of host machine. After finding the suitable host machine, its id is returned to VM manager which then maps the VM with host. Both the binary search tree will be updated after every new arrival or departure of task. VM resource requirement (VMR) and host resource availability (HR) is expressed in the form of numeric value. Matching factor M is calculated using the following formula:

$$M = \frac{VMR}{HR} = \begin{cases} <1 & \text{suitable candidate} \\ 1 & \text{Best candidate} \\ >1 & \text{Not suitable} \end{cases}$$

So while searching the host machine in binary search tree for host, it always tries to find that host which makes the matching factor value1 or <1. Algorithmic representation of proposed work is as follows:

Algorithm BST_VMH_Mapping()
{
- Initialize the set H with available set of host machines along with available resources;
- Construct the binary search tree corresponding to host machines;
  While (task is there)
  {
    - Identify the requirements of task;
    - Generate the VM with the specified resource requirements;
    - Construct the binary search tree corresponding to virtual machines;
    - Now search the both binary search tree so that Matching factor (M) for VM host mapping will become 1 or <1;
    - Do the mapping;
    - Update both the binary search tree;
  }

}

Time Complexity Analysis: If there are N number of virtual machines and M number of hosts, then time complexity for constructing and searching of binary search tree for Host resource availability will be log(M). So total time required to map N virtual machines with host machine of required specification will be N(log(M)), which is linear in nature. So time complexity of proposed approach is not only linear in nature but it also maps the VM with best host machine.

6. RESULTS & ANALYSIS

To test and compare the proposed approach with the existing one, three different scenarios have been developed. Proposed approach has been compared with rank based, greedy first fit and round robin based approach.

In the **scenario 1**, 10 different VM whose resource requirements are varying from 256 to 2048 have been generated. Also 10 Host machines are generated which have resources varying from 512 to 2048.
Figure 3 shows the effective utilization of resources in different strategies. It has been found that in proposed binary search based approach resources are not wasted in the form of chunks while in other approaches, resources at different hosts are wasted in the form of chunks. Also from figure 4, it has been found that rank based approach takes more time in comparison to other vm host mapping approach.

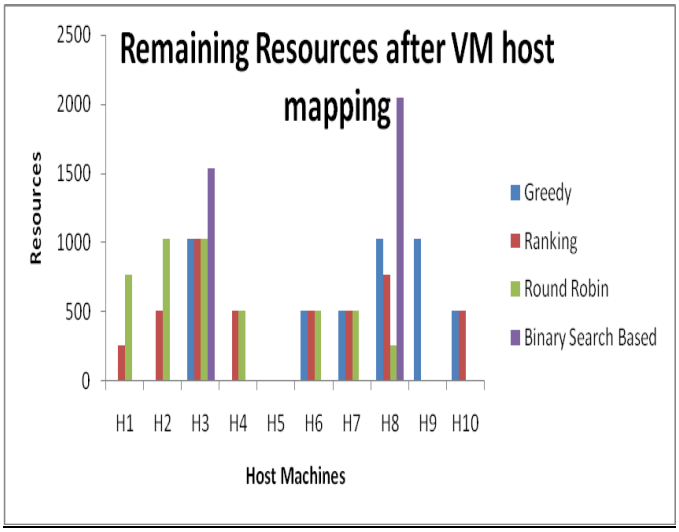

Fig -3: Usage of resources for different VM host mapping approach

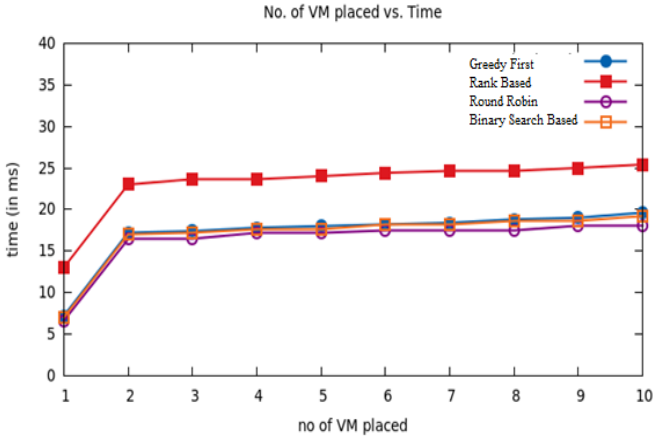

Fig -4: Comparison of different VM host mapping approach with respect to time

In the **scenario 2**, 20 different VM whose resource requirements are varying from 256 to 4096 have been generated. Also 10 Host machines are generated which have resources varying from 1024 to 4096.

From figure 5, it can be concluded that if we follow the binary search based, then maximum number of VM host mapping is done relative to other VM host mapping policies. This indicates the effectiveness of the proposed policy.

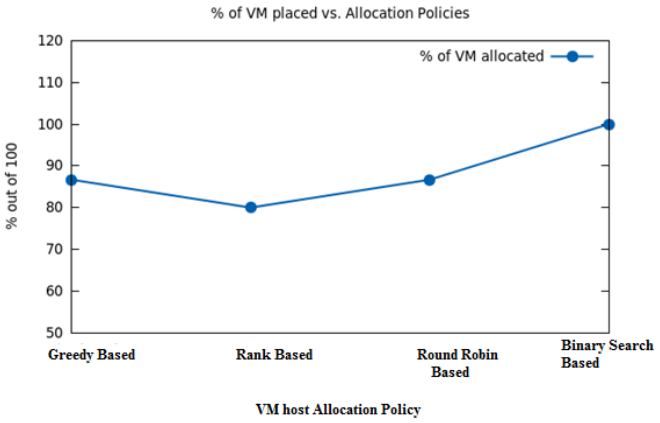

Fig -5: Comparison of different VM host mapping approach with respect to number of VM allocated

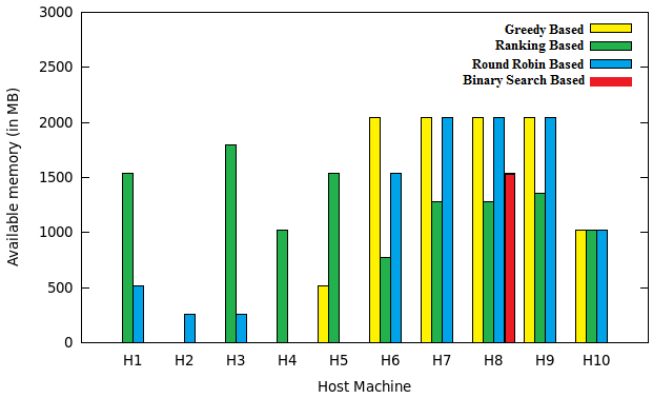

Fig -6: Usage of resources for different VM host mapping approach

Also from figure 6, it has been found that in proposed binary search based approach resources are not wasted in the form of chunks while in other approaches, resources at different hosts are wasted in the form of chunks.

In the **scenario 3**, 50 different VM whose resource requirements are varying from 512 to 2048 have been generated. Also 30 Host machines are generated which have resources varying from 1024 to 4096.

Following formula is used to calculate the SLA violation: (100-(Service delivered/Service Requested)100).

Graph of figure 7 shows that SLA violation is zero in case of binary search based VM host mapping, while it was non zero in other mapping.

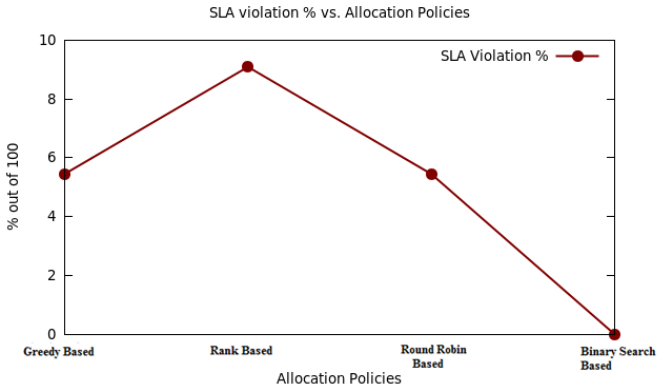

Fig -7: Comparison of different VM host mapping approach with respect to SLA violation

7. CONCLUSIONS

Among the different issues of cloud computing, deployment of virtual machine is major issue. It is so as different users are demanding different resources at different time from different locations. Their requests are entertained in terms of specified virtual machines within the specific time. So an efficient virtual machine placement approach is the need of cloud environment. It must be such that it not only optimize the resource utilization but also minimize the resource allocation time and minimize the service level agreement violation.

In the present paper, different virtual machine deployment approaches used by different cloud service provider has been analyzed. Then the new policy has been discussed and compared with existing using cloudsim simulator. It has been identified that proposed approach outperforms the existing one on all relevant parameters.

Functionality of proposed work can be further enhanced by combining the feature of virtual machine migration with proposed virtual machine placement approach.

REFERENCES

[1] P. Sasikala, "Cloud computing: Present Status and Future Implications," International Journal of Cloud Computing, vol. 1, no. 1, 2011, pp. 23-36.

[2] A. Jain and R. Kumar, "A Taxonomy of Cloud Computing," International Journal of Scientific and Research Publications. vol. 4(7), Jul. 2014, pp. 1-5.

[3] P. Patel and A. K. Singh, "A survey on resource allocation algorithms in cloud computing environment," In Golden Research Thoughts, vol. 2(4), 2012.

[4] A. Jain and R. Kumar, "A Comparative Analysis of Task Scheduling Approaches for Cloud Environment," International Conference On Computing for Sustainable Global Development, 2016, pp. 2602-2607.

[5] N. Bobro, A. Kochut and K Beaty, "Dynamic placement of virtual machines for managing sla violations," In 10th IFIP/IEEE International Symposium on Integrated Network Management, IM'07., 2007,pp. 119-128.

[6] B. Sotomayor, R. S. Montero, I. M. Llorente and I. Foster, "Virtual infrastructure management in private and hybrid clouds," Internet Computing, vol. 13(5), 2009, pp.14-22.

[7] H. Zhong, K. Tao and X. Zhang, "An approach to optimized resource scheduling algorithm for open-source cloud systems," In China Grid Conference (China Grid), 2010 Fifth Annual, 2010, pp. 124-129.

[8] G. Xu, Y. Ding, J. Zhao, L. Hu and X. Fu, "A novel artificial bee colony approach of live virtual machine migration policy using bayes theorem," The Scientific World Journal. 2013.

[9] A. Mevada, H. Patel and N. Patel, "Enhanced Energy Efficient Virtual Machine Placement Policy for Load Balancing in Cloud Environment," Int J Cur Res Rev, vol. 9(6), 2017.

[10] S. Bose and J. Kumar, "An Energy Aware Cloud Load Balancing Technique using Dynamic Placement of Virtualized Resources"

[11] R. Buyya, R. Ranjan & R.N. Calheiros, "Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities," High Performance Computing & Simulation, HPCS'09. International Conference, 2009, pp. 1-11.