JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Binary search tree approach based on cloud-VM resource allocation

[1]Madeva N B, [2]Nandini B N, [3]Shree Lakshmi D P, [4]Chiranjeevi, [5]Dr. Hareesh K

[1]UG Student, [2]UG Student, [3]UG Student, [4]UG Student, [5]Professor & Head
[1]Department of Computer Science and Engineering
[1]Government Engineering College, KR Pet, Mandya, Karnataka, India

Abstract - *Cloud computing is becoming more and more well-liked every day because of its adaptability and prompt availability of necessary resources. Requests are granted by offering the requested virtual machine in accordance with the conditions. If the vm placement takes longer than expected or the intended vms is not accessible, the guarantee will have been broken. It is quite challenging to forecast the quantity of customers and their needs in a continually changing market. The main resource is the virtualization technology. As a result, the virtual machine will be placed such that resources are used effectively and the agreement is not broken.*

To improve resource use, reduce wealth distribution time, and reduce service level agreement violations, the author of the current work presents a binary search tree-based virtual machine way to find the best. The binary search tree-based virtual machine deployment strategy's outcomes have been examined and contrasted with those of other techniques reported in the literature using the Cloudsim simulator.

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

1. INTRODUCTION

The process of permanently storing information or data on computers and momentarily caching it on client netbooks, laptops, sensors, etc. is known as cloud computing.

Figure 1. Cloud Computing functioning as utility

Figure 1 demonstrates how cloud computing functions as a utility that consumers may access of any location in the globe. Rapidly, software is being created for millions of users to use as a service rather than running on their own machines thanks to cloud technology [1, 2].

2. *RESOURCE ALLOCATION*

The allocation of resources is the main draw of multithreading computing. The main goal of resource allocation is to make sure that the proper resource is assigned to the job at the appropriate moment. Additionally, it takes into account how to best use available resources. Additionally, it makes sure that no resource is misused or underused. further difficulties with the 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. The allocation of resources is a part of their integration with the host computer. At any given time interval (0,t), if there are x number of virtual machines and y number of host machines, we must choose the one that is the most efficient. More combinations mean more time spent deciding which one is the best. There are a number of conditions that must be met for a resource allocation method to be effective [3, 4].

2.1 *VM Placement Algorithm Classifications*

Deploying a virtual machine is the process of installing a specified virtual machine on an appropriate real computer. Virtual machine and server mapping aims to maximize resource efficiency while minimizing VM migration, energy use, and service level agreement violations. The three kinds of placement algorithms are based on the VM placement's objectives:

- **Quality of service-based approach:** Once the objectives of VM migration strategy is to upgrade the effectiveness of services, such as to reduce service time, improve productivity, and reduce SLA violations, that VM placement strategy is referred to as excellence of service-based strategy. The use of materials is efficient as a consequence of this strategy.
- **Power based approach**
  A VM placement technique is referred to as being power-based when its objective is to decrease power usage. This is necessary for the reasons listed below:
    - Higher cooling costs will be the consequence of higher electricity use.
    - The cost of electricity is rising daily.

Computation based approach: We may use a static or dynamic VM placement strategy, depending on the kind of processing precondition. Static VM placement strategy refers to the process of placing virtual machines in accordance with some predetermined rule. While it is known as the flexible VM placement strategy when it is dependent on the current circumstances [3].

2.2 *VM Placement Approaches*

- **Rank Based Approach:** Open Nebula, an open-source cloud platform, employs this strategy. The host computers are given a predetermined priority. The calendar takes the host computer's priority into account when mapping VMs to hosts. This algorithm's input is a predetermined task requirement and ranking, and its output is a translation of the virtual machine to the host [5].
- **Greedy Algorithm:** Nimbus and Eucalyptus, two open-source cloud platforms, use this strategy. This strategy is really straightforward and simple to put into practice. The host computer that can initially execute the user-configured virtual machine is chosen to map with both the virtual machine. This algorithm's result is the host device id, and its input is the vm requirement [6].
- **Round Robin Algorithm:** Nimbus and Eucalyptus, two open-source cloud platforms, use this strategy. The task need is not taken into account in this method. The host ID that has been linked with a VM is stored by the VM management. When the next job is ready, the VM is mapped to the next address in the list [7].

3. *CLOUDSIM SIMULATOR*

Software called CloudSim is available for free and without a license at http://www.cloudbus.org/CloudSim/. It is a Java-based programming library. The JDK may be integrated to build this library so that it can be utilized directly Then run the programme. Cloudsim is used with Programming language IDEs (Integrated Development Environments), such as Canvas or Beans, to build and test the programs more rapidly. The Cloud computing library may be accessible and the cloud method can be used by using the Eclipse or NetBeans IDE. The following operations are simulated using the Cloudsim library:

Large scale cloud computing at data centers

- Support for modelling and simulation of massive cloud computing data centers. Virtualized server hosts with customisable rules.
- Assistance with the design and control of virtualized server hosts, using programmable host resource provisioning strategies.
- Support for energy-aware computer clusters modelling and simulation.
- Offer the ability to simulate and model statement applications and data centre network architectures.
- Support for federated cloud modelling and simulation.
- Support for system dynamics element placement as well as simulation pause and resume.
- Support for user-defined host allocation and resource allocation rules for virtual machines.
- Host allocation rules that are user-defined for virtual machines [11].

4. *RELATED WORK*

Xu Gaochao et al. in [8] By incorporating the artificial bee colony (ABC) idea also with irregular initializing idea, binary search idea, and Stochastic selection policy, we have developed a novel causal live VM migration policy for cloud environments that has better global discovery and local extortion capabilities. The Bayes theorem has also been applied to enhance the ABC-based approach in order to get the final optimum solution more quickly. A long-term efficient optimisation for power saving is achieved as a consequence of the complete method. Results show that PS-ABC decreases incremental power usage and preserves VM functionality more than prior studies while operating and migrating. It improves the quality and usefulness of the results of a live VM migration.

A. Mevada et al. in [9] they've proposed an improved policy for cloud load balancing virtual machine placement that's more efficient. For improved load distribution, lower energy usage, and improved VM placement, authors have developed a modified version of the power-based VM placement method. A real-time or simulations implementation of the suggested technique is still pending.

S. Bose et al. in The dynamic deployment of virtualized resources, together with energy-aware cloud load balancing, has been shown by Their technique reduces the overall number of servers in use while also increasing the efficiency with which those resources are used. In addition, they have devised a way to cope with overcrowding while minimizing the amount of energy required. Experiments have shown that the suggested method is capable of achieving the significant results .reduction in energy consumption and improves the overall resource utilization.

5. *PROPOSED WORK*

A virtual machine host mapping approach based on a binary search tree has been developed to maximize resource consumption and minimize service level agreement violations. The suggested scheme's framework may be seen in Figure 2.

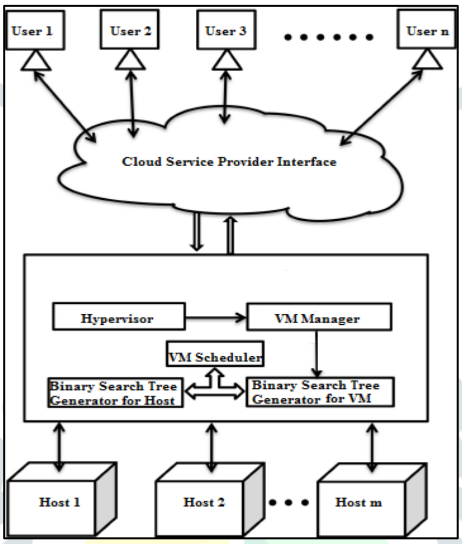

Fig 2. Framework for binary search tree based VM host mapping approach
User forwards their task through interface of cloud service

will require at most log (N) time to search the node. Through the cloud service's interface, the user sends their job N stands for the binary search tree's total number of nodes. Based on the availability of resources, a binary search tree for accessible host computers is also produced. In log (M) time, where M is the number of host machines, this will guarantee the optimal matching of the VM with the required host. Once a new host machine has been identified, its ID is supplied to the VM manager, who subsequently associates the VM with the host. After each new task arrival or leave, the binary search tree is updated. Host supply (HR) and VM resource need (VMR) are both given as numerical values. The following formula is used to compute the matching factor M:

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

As a result, whenever a host machine is searched for using a binary search tree, the host machine that produces the matching factor value 1 or 1 is always sought for. The suggested work is represented algorithmically as follows:

Algorithm BST_VMH_Mapping ()
{

- Initialize the set H using the set of host computers and resources that are currently accessible;
- Create a binary search tree that corresponds to the host computers;
- While (task is there)
  {
  Determine the task's needs;

```
            Create a virtual machine with the required resources;
            Create the binary search tree that corresponds to the virtual machines;
            Search both binary search trees such that the matching factor (M) for the VM host mapping will be 1 or 1;
            Complete the mapping;
            the binary search tree has been updated;
    }
}
```

internet service provider to CSP. The Virtualisation module generates a virtual machine with the required specifications based on the job requirements it receives. The active virtual machines are all monitored by the VM management. A binary search tree with a list of all outstanding VMs is created by the VM management. The binary search tree is created using the resource requirements for a virtual machine. This binary tree is sent to the virtual machine scheduler. The virtual machine with the most resources is chosen by the VM scheduler. The list of accessible VMs has been maintained using a binary search tree, thus it

The construction and search of a binary search for hosting resource distribution will take a logarithmic amount of time if there are N virtual machines and M hosts (M). As a result, the total time needed to map N virtual machine instances to a local computer that meets the requirements is N(log(M)), which is linear in nature. As a result, the suggested approach's time complexity is both linear in nature and also maps the VM to the optimal host machine.

6. PERFORMANCE & ANALYSIS

Three distinct scenarios have been created in order to evaluate and contrast the suggested strategy with the current one. Comparative studies have been done between the proposed technique and rank-based, greedy-first-fit, and round-robin approaches.

In scenario 1, ten distinct virtual machines (VMs) with resource needs ranging from 256 to 2048 have been created. Additionally, 10 host computers are created, each with 512 to 2048 MB of resources.
Figure The efficient use of resources in various ways is seen in Figure 3. It has been discovered that the suggested binary search-based strategy does not waste resources in the form of chunks, while chunk waste occurs in previous systems when resources are spread over several hosts. Additionally, figure 4 shows that the rank-based strategy requires more time than other vm host mapping approaches.

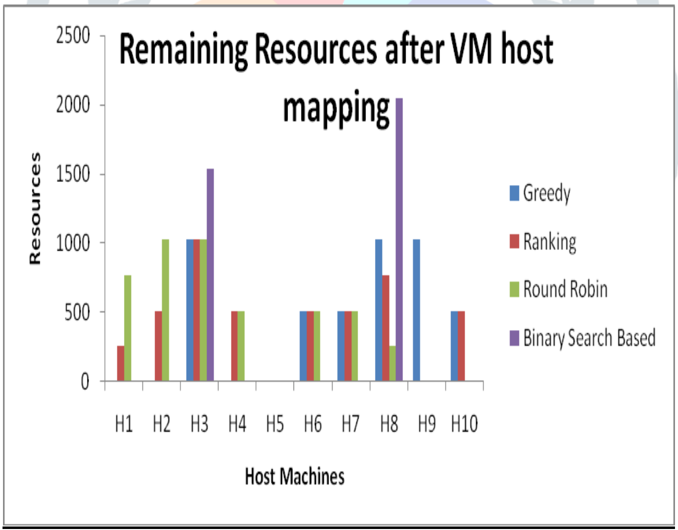

Fig -3: Usage of resources for different VM host mapping approach

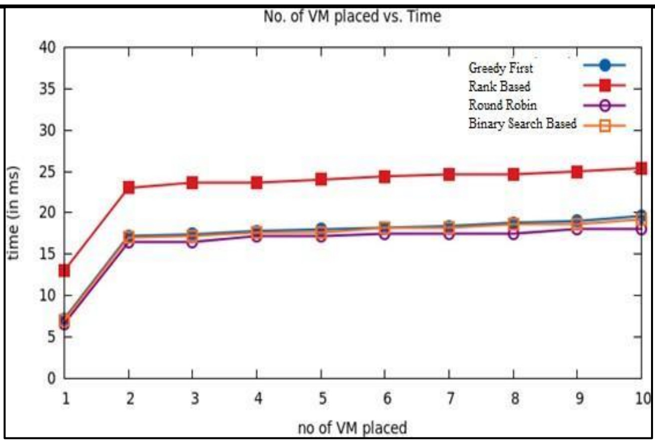

Fig -4: Comparison of different VM host mapping approach with respect to time

Twenty separate virtual machines (VMs) with resource needs ranging from 256 to 4096 have been created in scenario 2. Additionally, 10 host computers with resources ranging from 1024 to 4096 are created.

According to figure 5, the binary search-based approach results in the most VM host mappings when compared to other VM host mapping policies. This shows how successful the suggested policy is.

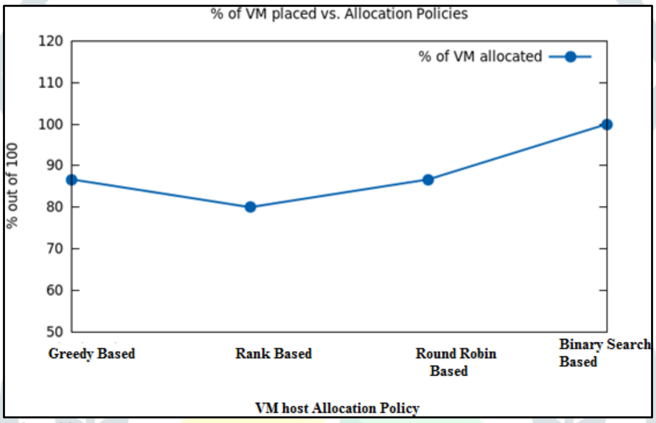

Fig -5: Comparison of different VM host mapping approach with respect to number of VM allocated

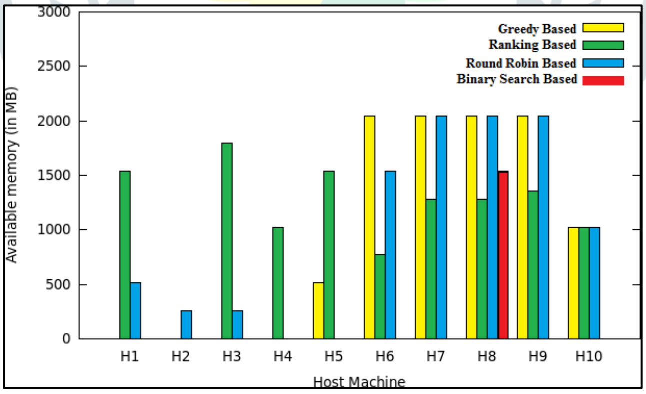

Fig -6: Usage of resources for different VM host mapping approach

Additionally, it can be shown from figure 6 that the suggested binary search-based strategy does not waste resources in the form of chunks, in contrast to previous systems that do.

50 distinct virtual machines (VMs) with resource needs ranging from 512 to 2048 have been created in scenario 3. Additionally, 30 host computers with resources ranging from 1024 to 4096 are created.

The SLA infringement is calculated by the formula (100-(Service delivered/Service requested)100).
Figure 7's graph illustrates that although there was no SLA infringement in the case of the binary hunt VM host mappings, there was in the case of all the other mapping

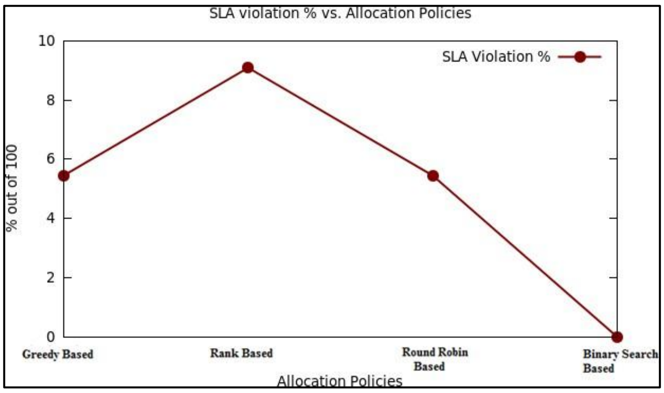

Fig -7: Comparison of different VM host mapping approach with respect to SLA violation

7. CONCLUSION

One of the main problems with cloud computing is the implementation of virtual machines. This is due to the fact that many users are requesting various resources at various times from various places. They get the virtual machines they requested within the time frame they requested. Therefore, a community cloud requires an effective technique for placing virtual machines. It must be designed in such a way that it not only maximizes the use of resources, but also cuts down on resource provisioning time and service level agreement violations.

The various virtual machine deployment strategies employed by various cloud service providers have been examined in the current article. Following that, using the cloudsim simulator, the new policy was explained and contrasted with the current one. It has been determined that the suggested technique performs better than the current one on all relevant metrics.

By integrating the live vm migration feature with the suggested virtual machine placement strategy, the performance of the proposed method may be further improved.

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”

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, 20