International Journal of Cloud Computing and Services Science (IJ-CLOSER)
Vol.3, No.4, August 2011, pp. 211~217
ISSN: 2089-3337

A Proposed Energy Efficient with Balanced Resources Approach for the Virtual Machine Placement in Cloud Environment

Rajeev Kumar Gupta, R.K. Pateriya
Department of Computer Science & Engg , MANIT, Bhopal

Article Info

Article history:

Received Jun 15th, 2014
Revised Jul 20th, 2014
Accepted Aug 10th, 2014

Keyword:

Virtualization
Virtual Machine
Migration
Threshold
VM placement

ABSTRACT

Compared to traditional distributed computing paradigms, resource utilization with proper resource balancing is more challenging task in the cloud computing environment, because of the dynamic nature of the resources. Resources utilization can be increase by the virtualization. For the efficient utilization of the physical resources, VM should be placed on to the suitable host. VM placement is the process to select the appropriate host for the given VM. Resource utilization can be increased and power consume by the data centers can be reduced by the efficient VM placement algorithm. One of the major concerns for cloud providers is how to place virtual machine on physical machine to improve resource utilization and reduce energy consumption Numbers of virtual machine placements algorithms have been proposed that run under cloud computing environment. Main goal of these algorithms are either to save energy by shutting down some severs or maximizing the resources utilization, but none of them considered both. In this paper we proposed a method for the VM placement that increased the resources utilization with the resource balancing and reduced the power consumed by the data centers.

Corresponding Author:

First Author,
Department of Computer Science & Engg , MANIT, Bhopal
Email: rajeevmanit12276@gmail.com

1. INTRODUCTION

Cloud computing is a new emerging technology in the field of computer networks and Web provisioning [1]. Academic and industrial communities have paid significant attention to cloud, because of their attractive feature such as on-demand, easy to use, value added services etc. Several venders such as Amazon, HCL, Google and IBM are providing the cloud services and they are continuously working to optimize the usage of their own data centers. Three type of services are provided by the cloud such as Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) [2] and it can be deploy in three different way that is private, public and protected [3][4].

Virtualization [5, 6] is the core technology behind the cloud computing, it's allowed number of virtual machines (VMs) can be assign to the same physical host. So it's increased the resources utilization. In the cloud architecture each data center having a number of host and number of VM are running in each host. So it is very difficult to decide which VM allocated to which hosts. Andrzej et al in [7] argue that power consumed by the data center can be saved up to the 50% by suitable allocation of virtual machines. VM placement are required in two different situation either for placing new VM or to place a migrated VM. Proper placement of a VM play a vital role in the cloud environment, if initial placement of VM is not suitable, its' increase the number of servers as well as number of migration. For the energy effective solution number of active server and the number migration should be kept as low as possible. VM placement is a NP hard problem that means placing a VM in a proper PM is a challenging task [8-11].

2. RELATED WORK

T. Wood et al. [12], proposed a method called Sandpiper for detecting a hot spot to prevent SLA violations. They use the threshold to classify the PM as underloaded or overloaded. When the CPU utilization of a PM crosses the value of upper threshold, hotspots are detected. Sand volume which is defined by the equation-1 is used to place the VM to mitigate the hotspots.

$$\text{Sand\_volume} = \frac{1}{1-cpu} * \frac{1}{1-net} * \frac{1}{1-mem} \text{ ------- (1)}$$

$$\text{VSR} = \frac{\text{sand\_volume}}{\text{size}}$$

Where cpu, net and mem are the normalized utilization of the cpu, network and memory respectively and size is the memory footprint of the VM. According to this approach all PM are arranged into the descending order of their volume and all VM also arranged into the descending order of their VSR (volume to size ratio) value. When the hotspots is detected, algorithm select the VM which having the highest VSR value from the overloaded host and placed it to the least loaded PM among all the PM. If the least loaded physical machine not having an enough resources then the next least loaded PM are selected. Similarly if none of the PM has the enough resources to keep the selected VM, then VM with next highest VSR value are selected from the overload host. This method seems good, but there is a possibility to select wrong target physical machine, because they are not using the shape of the resource utilization and 3D (cpu, network, memory) resource information are converted into 1D that is sand_volume.

Khanna et al [13], proposed an algorithm for the server consolidation. Lower and upper CPU threshold are used to detect the hotspots and coldspots. Hotspots are detected when the CPU utilization cross the upper threshold limits and coldspots are detected when the CPU utilization below the lower threshold. All PM and VM are arrange into the increasing order of their CPU utilization. When the hotspots is detected VM with least CPU utilization are migrated to the host which has the lower residual capacity. If none of the PM can host the selected VM then next least usage VM are chosen from the sorted order. In the case of coldspots detection all VMs are migrated to the other host. Problem with this approach is they select the small VM, which increase the number of migration and they are also not focus to the resources balancing.

A. Beloglazov et al. [14], proposed a method for the server consolidation which reduce the power consume by the data centers. Upper and lower threshold are use to detect the overloaded and underloaded hosts. When the host is overloaded, they select the VM whose size is greater than or equal to the (host utilization-upper threshold) and place to the PM where it produced less increment in the power consumption among all the hosts. In this approach they reduced the power consumed by the data centers, but the resources balancing are not handled.

M. Mishra et al. [15], proposed an approach for placing a VM with the resources balancing. For the proper utilization of the resources it is necessary that the virtual machine which required more CPU and less memory should be placed on the physical machine which has low CPU and more memory utilization. Vector calculus is used to express all the resources related information. Dot product and the vector projection are used to select the PM for the VM placement. For calculating the load on the system they give the different weight to the each resource by using the weighting coefficient (W_i). Following equation are used to calculate the load on k^{th} host

$$\text{HL}_k = \sum_{i=1}^{n}(w_i) * \frac{\sum_{j=1}^{m}(RU_{ij})}{(Hcap_{ik})}$$

Where n is the type of resources i.e. cpu, memory, io etc, m is the number of VM in k^{th} host, RU_{ij} is the i^{th} type resources use by the j^{th} VM and $Hcap_{ik}$ is the i^{th} resource capacity of k^{th} host.

The main idea behind this approach is to select a target PM for placing a VM such that the resource utilization vector of the PM is complementary to the resource requirement vector of the VM. This method properly balanced the resources, but power consumption may increase compare to the previous approach.

3. PROPOSED WORK

Several methods have been proposed for the VM placement. All these methods are focus either on the power consumption or resource balancing, but none of them considered both. In this paper we proposed a method for the VM placement that increased the resources utilization and reduced the power consume by the data centers. Energy consumption can be reduced by minimizing the number the active host and system performance can be improved by the balanced resources utilization. To achieve this we are using an array to store the host. Load on the physical machine are used to decide the overloaded and underloaded situation. So firstly we have to calculate the load on the PM and VM.

3.1. Physical and Virtual Machine Load Calculation

In the cloud environment each hosts having a numbers of VM. So load on the PM is equal to the sum of all VM loads running on it. Most of the work done previously for calculating the load only considers the current load. Since load on the VM are change dynamically, so average load of VM gives more accuracy during the load calculation.

If P is the set of physical machine present in the system and m_i is number of VM in i^{th} host and T is the time span for monitoring the data. We divided this time span into n time slots T= [$(t_1 - t_0)$, $(t_2 - t_1)$, $(t_3 - t_2)$ ------------ $(t_k - t_{k-1})$]. Then load on the PM and VM can be calculated as

$$VL_i^T = \frac{1}{T} \left[\sum_{k=1}^{n} \sum_{r=1}^{m} (w_r)\, u_r^i \,(t_k - t_{k-1}) \right]$$

Where VL_i^T is the average load of the i^{th} VM in T time, m is the number of resources. In our approach we are using 3 resources (CPU, memory and bandwidth, so m = 3), W_r is the weights of r_{th} type resource and u_r^i is the fraction of r_{th} type resource, which is define as

$$u_r^i = \frac{r\ type\ resource\ used\ by\ the\ i\ VM}{Total\ r\ type\ resource\ capacity\ of\ PM\ inwhich\ i\ VM\ running}$$

And average load on the j^{th} PM

$$PL_j^T = \sum_{i=1}^{m_i} VL_i^T$$

Upper and lower threshold are use to detect the overloaded and underloaded hosts. When the host is underloaded all the VM in that host are move to the other host, which is known is server consolidation and if the host is overloaded, algorithm select the VM whose size is greater than or equal to the (host utilization-upper threshold). Now the selected VM are placed into the host, where the remaining resource capacity of the PM is similar to the resource requirement of the VM and produced less increment in the power consumption among all the hosts in the array. For this purpose six array are used which store the host. CPU, memory and bandwidth are the three main resources which are used in the cloud computing. CBM, CMB, MBC, MCB, BMC, BCM are the six array which are used to store the hosts. CBM array store the host where the available CPU > available Bandwidth > available Memory. Similarly if the available CPU > available Memory > available Bandwidth then the host are stored into the CMB array. First step for placing any VM, is to select the appropriate array where resource requirement of the VM belongs. In the selected array VM are placed to the PM, where its have a sufficient resource to fulfill the resource required of the VM produced less increment in the power consumption among all the hosts in the array.

3.2. Linear Programming Formulation for The VM Placement

Main of the placement algorithm is to reduce the number of active server and resource leakage. If n_p is the number of active server then

$$\text{Min} \sum_{i \varepsilon P} n_p$$

$$\text{Where } n_p = \begin{cases} 1 & \text{If PM is active} \\ 0 & \text{Otherwise} \end{cases}$$

In each dimension sum of resource required by all the VM placed on it, should be less than the total capacity of the PM.

$$\sum_{j=1}^{m_i} u^i_{cpu} < C^i_{cpu} \qquad \forall j \in m_i$$

$$\sum_{j=1}^{m_i} u^i_{mem} < C^i_{mem} \qquad \forall j \in m_i$$

$$\sum_{j=1}^{m_i} u^i_{bw} < C^i_{bw} \qquad \forall j \in m_i$$

Where C^i_{cpu}, C^i_{mem} and C^i_{bw} is the total cpu, memory and bandwidth capacity of the i^{th} host respectively. Each VM should be allocated to only one PM. If x_{ij} represent the i^{th} VM assign to the j^{th} PM then $\sum_{j \in P} x_{ij} = 1$

Thus, summarizing the formulation:

$$\text{Min} \sum_{i \varepsilon P} n_p$$

Subject to

$$\sum_{j=1}^{m_i} u^i_{cpu} < C^i_{cpu} \qquad \forall j \in m_i$$

$$\sum_{j=1}^{m_i} u^i_{mem} < C^i_{mem} \qquad \forall j \in m_i$$

$$\sum_{j=1}^{m_i} u^i_{bw} < C^i_{bw} \qquad \forall j \in m_i$$

$$\sum_{j \in P} x_{ij} = 1$$

4. ALGORITHM FOR VM PLACEMENT

```
1.  Input hostList, vmMigrationList, Output allocatedHost
2.  For each host in hostList
3.  If(remaining capacity of cpu > remaining capacity of mem> remaining capacity of bw)
4.  CMBList.add(host)
5.  endif
6.  elseif(remaining capacity of cpu > remaining capacity of bw> remaining capacity of mem)
7.  CBMList.add(host)
8.  endif
9.  elseif (remaining capacity of mem > remaining capacity of bw > remaining capacity of cpu)
10. MBCList.add(host)
11. endif
12. elseif (remaining capacity of bw> remaining capacity of mem > remaining capacity of cpu)
13. BMCList.add(host)
14. endif
15. elseif (remaining capacity of mem > remaining capacity of cpu > remaining capacity of cbw)
16. MCBList.add(host)
17. endif
18. elseif (remaining capacity of bw> remaining capacity of cpu > remaining capacity of mem)
19. BCMList.add(host)
20. endif
21. end for
22. For each VM in vmMigrationList
23. Find the appropriate array of host  in which VM belongs
24. For each host in the selected host array
25. hostUtil=host.getUtil()
26. if (vm.getUtil + hostUtil<=UT && vm.getUtil + hostUtil >=LT)
27. minPower ← MAX
28. allocatedHost ← NULL
29. If host has enough resource for VM then
30. power ← estimatePower(host, vm)
31.  if power < minPower then
32. allocatedHost ← host
33.  minPower ← power
34. End for
35.  if allocatedHost /= NULL then
36. allocate vm to allocatedHost
37. return allocation
38. else
39. Select the next array of host
40. Goto 30
```

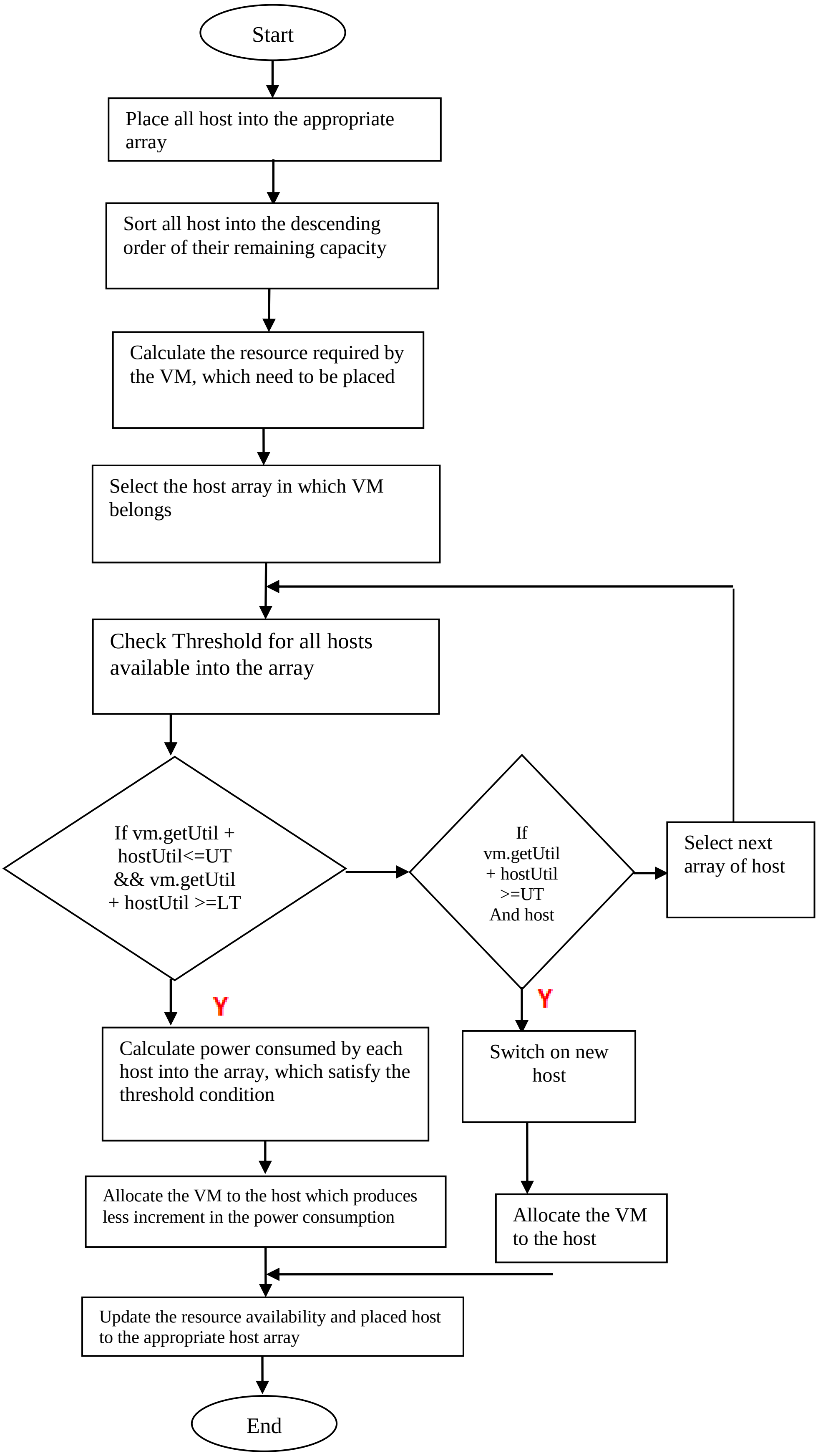

Figure 1. Flow diagram for VM placement

Algorithm for the server consolidation

1. If hostUtil < LT
2. For each VM in the underloaded host
3. Call VM placement algorithm
4. End for
5. End if

So proposed algorithm increase the resource utilization by selecting the suitable host and also reduce the power consume by the data centers.

5. EXPECTED OUTCOME

1. Reduce the resources leakage, by selecting the best host available in the available into the array
2. Since hosts are arranged into the decreasing order of their remaining capacity i.e. lightest host comes first. So it will reduce the searching time, because most of the time algorithms select the first host from the array.
3. Since we are calculating the load on the PM and VM using the history data in last T time span. So it will calculate the load accurately that will help us to reduce the number of migration.

6. CONCLUSION

Resource in the cloud computing are change dynamically and distributed geographically. So the proper placement of the VM is the challenging task. Efficient VM placement in the cloud are the necessary step, because it's affect the overall system performance. Number of migration and the number of active host can be reduced by the suitable VM placement approach. In this paper we proposed a method for the VM placement that increased the resources utilization with the resource balancing and at the same time reduced the power consumed by the data centers.

7. REFERENCES

[1]. R. Buyya et al., "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility", Future Generation Computer Systems, 2011.
[2]. Mell, P. et al."The NIST Definition of Cloud Computing", NIST Special Publication, 2011.
[3]. Sosinsky et al." Cloud Computing Bible", Wiley Publishing Inc 2012
[4]. Kim et al., "Experimental Study to Improve Resource Utilization and Performance of Cloud Systems based on Grid Middleware" proceeding in first International Conference on Internet 2009.
[5]. Borja Sotomayor, Kate Keahey, Ian Foster, and Tim Freeman, "Enabling cost-effective resource leases with virtual machines," ACM/IEEE International Symposium on High Performance Distributed Computing 2007 (HPDC 2007),2007.
[6]. L. Cherkasova, D. Gupta, and A. Vahdat, "When virtual is harder than real: Resource allocation challenges in virtual machine based it environments," Technical Report HPL 2007-25, February 2007.
[7]. Andrzej Kochut, "Power and Performance Modeling of Virtualized Desktop Systems", in Proceedings of MASCOT 2009.
[8]. L. Grit, D. Irwin, A. Yumerefendi. and J. Chase, "Virtual Machine Hosting for Networked Clusters: Building the Foundations for Autonomic Orchestration," in Proceedings of IEEE International Workshop on Virtualization Technology in DistributedComputing (VTDC), November 2006.
[9]. M. Cardosa, M. R. Korupolu, and A. Singh, "Shares and Utilities based Power Consolidation in Virtualized Server Environments," inProceedings of IFIP/IEEEIntegrated Network Management (IM), 2009
[10]. F. Hermenier, X. Lorca, and J.-M. Menaud, "Entropy: A Consolidation Manager for Clusters," in Proceedings of the 2009 ACMSIGPLAN/SIGOPSInternational Conference on VirtualExecutionEnvironments (VEE), 2009.
[11]. N. Bobroff, A. Kochut, and K. Beaty, "Dynamic Placement of Virtual Machines for Managing SLA Violations," inProceedings of IFIP/IEEE International Symposium on Integrated Network Management (IM), pp.119-128, May 2007.
[12]. Timothy Wood, Prashant Shenoy, and Arun. Black-box and gray-box strategies for virtual machine migration. 4th USENIX Symposium on Networked Systems Design and Implementation, 2007.
[13]. Gunjan Khanna, Kirk Beaty, Gautam Dhar, and Andrzej Kochut. Application performance management in virtualized server environments. Network Operations and Management Symposium NOMS 10th IEEE/IFIP, pages 373-381, 2006.
[14]. A. Beloglazov et al. "Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing", Elsevier journal of Future Generation Computer Systems, pp. 755–768, 2012.

[15]. Mayank Mishra and Anirudha Sahoo, “On Theory of VM Placement: Anomalies in Existing Methodologies and Their Mitigation Using a Novel Vector Based Approach”, IEEE 4th international conference on cloud computing in 2011.

BIOGRAPHY OF AUTHORS

Rajeev Kumar Gupta is a PhD candidate in the Department of Computer Science and Engineering at Maulana Azad National Institute of Technology, Bhopal, India. He has completed his Bachelor degrees in CSE at SATI, Vidisha and Master degree inCSE at MANIT, Bhopal. His research area is Cloud Computing, Distributed computing and Gried computing.

Dr. R. K. Pateriya is a Associate Professor in the Department of Computer Science and Engineering, at Maulana Azad National Institute of Technology, Bhopal, India. He is a member of the IEEE. His current research interests include Cloud and Distributed Computing, E- Commerce, Security etc. He has authored and published over 200 research papers .