ISSN:0975-9646

Virtual Machine Placement Based on Disk I/O Load in Cloud

Pathan Noumankhan Sayeedkhan
Information Technology Department
SGGS IE&T
Vishnupuri, Nanded

Mr. Shetty Balaji S.
Information Technology Department
SGGS IE&T
Vishnupuri, Nanded

Abstract— In cloud Virtualization allows to consolidate the services onto a lesser number of physical servers than originally required. The virtual machine is one of the most commonly used resource carriers in which business services are encapsulated. Virtual machine placement optimization, i.e., finding optimal placement schemes for virtual machines, and reconfigurations according to the changes of environments has become a challenging issue. In our work we have studied a disk I/O load based virtual machine placement algorithm i.e. FFDL and proposed a Static Disk Threshold Based Migration algorithm to optimized the performance of the virtual machine.

Index Terms— Cloud Computing, Virtualization, Virtual Machine, Virtual Machine Placement, VM Migration

INTRODUCTION

Cloud computing provides scalable computing and storage resources via the Internet. It delivers infrastructure, platform, and software (applications) as services, which are made available to consumers as subscription-based services under the pay-as-you-go model. In industry these services are referred to as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) respectively [1]. These infrastructure resources (computing and storage resources) based on a pay-per-use basis is delivered by several cloud providers. Each customer (user) can rent a VM from the cloud provider to execute his/her desired application.

One of the important problems in data centers that supports cloud computing model, is the placement of virtual machines (VMs) requiring varying resources over physical hosts of fixed capacity. In the IaaS delivery model, the key technology is how to efficiently virtualize the computing and storage resources of all physical machines (PMs) to provision a large number of virtual machines (VMs). Amazon EC2 provides 12 different instance types of VMs to meet the computing needs of users. Several virtualization technologies (e.g. Xen [2], KVM [3]) have been used in cloud computing systems. Based on the virtualization technology, more than one VM can be created in the same PM, each of which acquires its required resources by slicing a portion of resources from the PM.

There are a lot scheduling algorithms have been proposed to meet the different goals such as energy-saving, time-saving, CPU load balancing, Performance enhancements, but very few of them consider the Disk I/O load for performance enhancement. Many applications perform a huge amount of disk operations, such as data mining, signal processing, etc. which often cause a performance bottleneck [4].

From our study, we have observed that more than one virtual machine having disk intensive task when executed on the same physical host takes more time than the virtual machines having CPU intensive task. In this paper we propose a Static Disk Threshold based Migration scheme for disk intensive task executing in cloud computing environment.

The remainder of the paper is organized as follows. In Section 2, we discuss the related work. In Section 3, we present our system model. In section 4, we present our proposed approach, followed by evaluation and experimental results in Section 5. Lastly we make the conclusion in Section 6.

RELATED WORK

The problem of virtual machine placement has attracted considerable attention recently. In particular, [5] proposes a placement algorithm that determines the placement of virtual machines such that the energy consumption in computing and data centers is minimized. [6] employs a particle swarming algorithm to find a near-optimal placement for virtual machines by taking into account the task completion time of virtual machines, the cost of virtual machines, as well as the cost of communications between resources. [7] Propos a placement algorithm that places virtual machines, one at a time, such that the data transfer delay is minimized. The method of [7], however, assumes that only a single virtual machine can request a file transfer at any point of time, and therefore does not address the problem of optimal rate allocation. However, according to Amazon EC2, one of the most popular cloud computing providers, unlike other resources, data rates are not dedicated and have to be shared among virtual machines. [8] Proposes an instances placement algorithm FFDL that based on disk I/O for private cloud environment to deal with big data that would adopt the disk I/O load balancing strategy and reduce competition for the disk I/O load between instances. However the disk I/O performance of a virtual machine can only be observed when it is in running state. Therefore when more than one virtual machine enter into the cloud the FFDL algorithm places the VMs in the same manner as greedy algorithm.

SYSTEM ARCHITECTURE

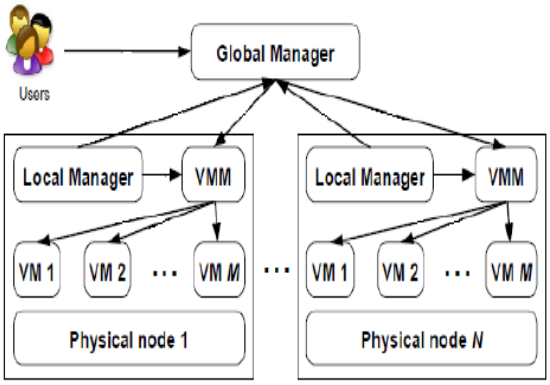

Figure 3.1: The System Model

We consider the system model same as proposed in [9], the target systems are of IaaS Environment. As shown in the above figure, the system models consist of global and local manager. The local managers, which are part of VM monitor, resists on each node and are responsible for keeping continuous check of when to migrate a VM and utilization of the node. The end-user sends its service request along with some CPU performance parameters like MIPS (Million Instruction per second), RAM, memory and network bandwidth to a global manager which in turns intimates the VM monitor for VM placement. The local manager reports the global manager about the utilization check of its node. And thus, global manager keeps the check of overall utilization of the resource.

We propose our scheme based on static disc threshold in two parts:

1. Selection of VM for migration.
2. Placing the VM on proper host.

PROPOSED SCHEME

The selection of VM for migration is done to optimize the allocation. Here, we first calculated the Disk utilization of all VMs as shown below:

$$\sum Uvm = \frac{TotalRequestedIOMips}{TotalIOMipsForThatVM}$$

Where IOMips stands for Input Output Million Instruction per Second. It simulates the beheviour of disk I/O operation in CloudSim toolkit.

- Upper Disk threshold value:
  In [10] it is considered that if the CPU utilization is above 80%, the upper threshold value will be 0.8. According to Amazon EC2, like CPU, network and disk of a physical host have to be shared among VMs, they are not the dedicated resources. Therefore we can consider 0.8 value as a disk threshold value. The Disk will be considered overloaded when the utilization is above this value so we migrate some of the VMs.
- Lower Disk Threshold value:
  If a host's disk is getting utilized under 30%, we will consider it as underutilized. This underutilized host will serve as candidate for virtual machine placement.

The following steps are followed for selecting a VM for migration.

Algorithm 1 Live Migration using static threshold

Input: host list, VM list
Output: migration list, target host list.

1. Sort the VM list in the decreasing order of its VM utilization.
2. For each host in host list compare the current host utilization value to the upper threshold value of that host. If the value is greater than threshold value go to 3 else go to 4.
3. Get the each VM for the current host. If VM utilization is greater than the or equal to the upper threshold value, add the VM into VM migration list.
4. If host utilization value is less than lower threshold value than, add the host into target host list.
5. Return the migration list and target host list.

Placing of VM

We have considered placing of VM as a bin packing type of problem. So, for placing the VM we have used BFD (Best Fit Decreasing) algorithm. We describe the algorithm for placing VM as below.

Algorithm 2 VM Placement with Best Fit Decreasing

Input: Target host list, VM list
Output: allocation of VMs

1. Sort the VM list in the decreasing order of its VM utilization.
2. For each host in target host list, if host has enough resource for VM then select the host as destination for VM migration, else do nothing.
3. Return allocation.

EXPERIMENTAL RESULTS

We tested our work on Cloudsim Toolkit [11]. In our experiment, we have worked with just one datacenter. We created 10 hosts on this datacenter which in turn is running 20 virtual machines on those hosts. Each node comprises of one CPU core with 5 GB ram/network bandwidth and storage space of 1TB. The host comprises of 5000 MIPS of execution speed and 1000 IOMIPS disk I/O execution speed. For each virtual machine on host ram size is 512MB and bandwidth size is 2500 MB with 1000 MIPS execution speed and 500 IOMIPS for disk I/O execution speed.

We created one cloudlet to run on each VM. Out of 20 cloudlets 8 cloudlets contains only CPU instruction to execute and 12 cloudlets contain both CPU and Disk I/O instruction.

As shown in the previous section, we have used best fit decreasing order for placement of VMs on the host to increase performance of virtual machine. As per our theory, we have used two threshold values, namely upper disk threshold and lower disk threshold value.

Firstly we run the simulation with virtual machines on datacenter using a greedy algorithm (First Fit round Robin). The results obtained are as follows:

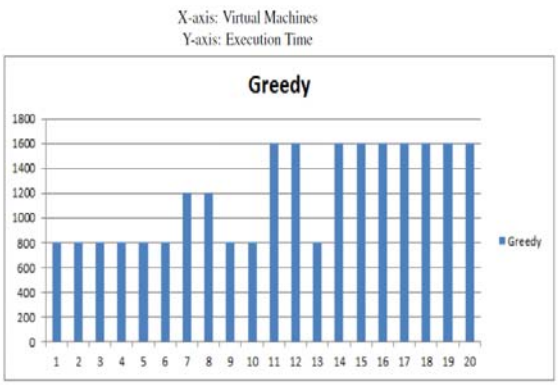

Figure 5.1: Performance: Greedy
The graph shows performance of Round Robin algorithm

Then we performed the simulation with the same virtual machine on the data center using FFDL algorithm. The results are compared with greedy algorithm.

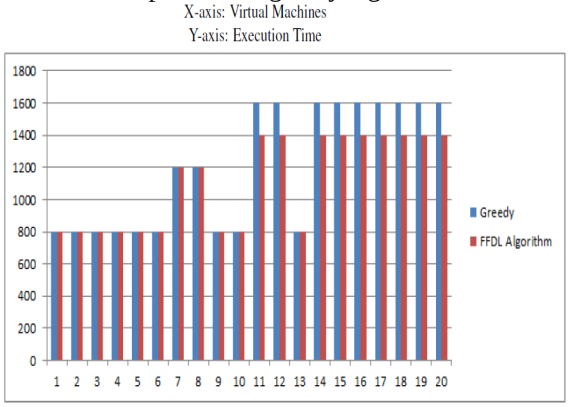

Figure 5.2: Performance: Greedy Vs FFDL
Graph shows the performance of Greedy and FFDL algorithm

Lastly we performed the simulation of virtual machine execution on datacenter using FFDL as initial placement algorithm and applied Static Threshold based Migration scheme for optimization. The results are compared with results obtained before.

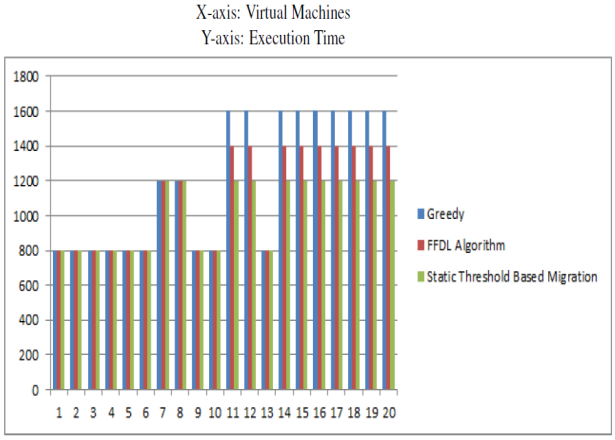

Figure 5.3: Performance: Greedy Vs FFDL Vs Static Threshold Based Migration
Graph shows the Performance Comparison of Greedy, FFDL and Static Threshold Based Migration Algorithms

CONCLUSION

The FFDL algorithm is more effective when virtual machines enter in cloud with regular interval. Since the disk I/O performance of a virtual machine can only be observed when it is in running state. Therefore when more than one virtual machine enter into the cloud the FFDL algorithm places the VMs in the same manner as greedy algorithm.

From comparison we conclude that the execution time obtained by applying Static Threshold Based scheme is less than execution time obtained by applying FFDL algorithm alone.

REFERENCES

[1] Rajkumar Buyya, Anton Beloglazov, Jemal Abawajy, "Energy-Efficient Management of Data Center Resources For Cloud Computing: A Vision, Architectural Elements, and Open Challeges", Proceedings of the 2010 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2010), Las Vegas, USA, July 12-15, 2010

[2] R. Wojtczuk, "Subverting the Xen Hypervisor,"Black Hat USA, 2008.

[3] I. Habib, "Virtualization with kVM," Linux Journal, article 8, no. 166, 2008.

[4] Nikolay Grozev and Rajkumar Buyya, "Performance Modeling and Simulation of Three-Tier Applications in Cloud and Multi-Cloud Environments". The Computer Journal, 2013.

[5] A. Beloglazov, R. Buyya, "Energy efficient allocation of virtual machines in cloud data centers", 10th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2010),

[6] Y.M. Chen, S.Y. Tsai, "Optimal Provisioning of Resource in a Cloud Service", IJCSI International Journal of Computer Science Issues, Vol. 5, Issue 6, 2010.

[7] J.T. Piao, J. Yan, "A Network-aware Virtual Machine Placement and Migration Approach in Cloud Computing", International Conference on Grid and Cloud Computing, 2010.

[8] Jian Guo, Zhao-Meng Zhu, Xiv-Min Zhou, Gong-Xuan Zhang, "An instances placement algorithm based on disk I/O load for big data in private cloud" IEEE International conference on 17-19 Dec. 2012, 287 – 290.

[9] A Beloglazov ,R Buyya, "Energy Efficient Resource Management in Virtualized Cloud Data Centers"10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, 2010