I Q R based Approach for Energy Efficient Dynamic VM Consolidation for Green Cloud Data Centers

Praveen Shukla
M.Tech Scholar
Department of Computer Science and Engineering
Maulana Azad National Institute of Technology
Bhopal, 462051, India

R.K. Pateriya
Associate Professor
Department of Computer Science and Engineering
Maulana Azad National Institute of Technology
Bhopal, 462051, India

ABSTRACT
With the advent of cloud computing in the arena of IT field energy consumption and service level agreement (SLA) violation emerge as a major problem, which reduces the profit of cloud service providers (CSP) and affect the cloud customers by fencing the reusability and scalability of the cloud data center services. This problem needs to be eradicate for the efficient resource provisioning in cloud data center. To satisfy the customer need virtual machine (VM) migration technique is required to balance the load of entire data center. Therefore we need to transfer the virtual machine of the overloaded host to the light weighted host using virtual machine migration technique. Due to frequent load balancing of cloud data center enormous amount of energy consumption takes place. This enhances the overall energy cost and degrades the performance of cloud data center. This paper proposes an Energy Efficient Dynamic VM Consolidation algorithm for reducing energy consumption.

Keywords
Cloud computing, Cloud service provider (CSP), Energy consumption, SLA violation, Load balancing, VM migration.

1. INTRODUCTION
Cloud computing has emerged as a new computing model for providing application, platform, or computing resources (processing power, RAM, band width, storage etc.) to cloud customer on pay as per-use basis. It is a proven service delivery model. It offers three types of services like Infrastructure as a Service (IaaS), Platform as a service (PaaS), and Software as a Service (SaaS) [1]. The main component of the cloud computing is virtualization technology that helps in the emergence of cloud computing in global IT environment [2]. Virtualization is a methodology of partitioning the resources among multiple execution environments (virtual machine) by applying technology such as hardware partitioning, software partitioning and OS partitioning. Virtualization technology provides the cost saving, hardware reducing, elasticity, scalability, and on demand resource provisioning [1].Virtualization improves the performance in terms of resource utilization and application flexibility by multiplexing VMs on the same set of hardware hosts. That's why most of the organization attracted towards the cloud computing paradigm and moving away from traditional computing model. The main component of virtualization in cloud is virtual machines.

The virtual machine is an abstraction layer between hardware and the end user, which have an ability to run multiple operating systems with its application inside them [2].Cloud is a multi-tenant environment where multiple clients generate

multiple requests would unbalance the server in cloud data center. This may arise the need of appropriate load balancing strategies to balance the load among multiple hosts without disrupting the current user services. Load balancing is actually implemented through virtual machine migration technique. There exist various algorithms considering VM migration. Each of the algorithms uses different models for achieving VM migration.

VM migration is a technique that migrate the entire VM from one host to another host. In live migration VM can be migrated without disrupting the application running on it [3].

VM consolidation is used to manage work load of cloud data center using various techniques such as

a) Load balancing :- The load balancing is triggered when host is over loaded then some virtual machine need to be migrated from over loaded host to another host to balance the load among the host. This is shown in fig. 1.
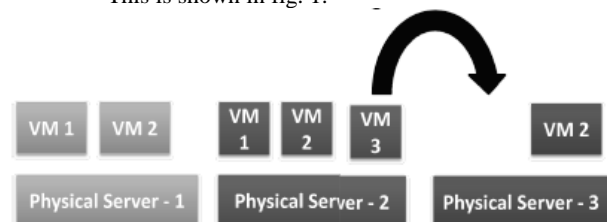

Fig 1(a): Cloud Load Balancing [Before migration]

Fig 1(b): Cloud Load Balancing [After migration]

b) Server consolidation:-The Server consolidation is a technique to reduce active host by consolidating VMs onto the minimum number of host and turn idle host into power saving mode. It implements the green computing by reducing the required power to operate cloud system [4].This is shown in fig. 2.

Fig 2(a): Cloud Server Consolidation [Before migration]

Fig 2(b): Cloud Server Consolidation [After migration]

Service level agreement (SLA) is a legal contract between cloud service provider (CSP) and customer to define Quality of Services (QoS). SLA is violated if any party doesn't fulfill the customer requirement at any given point of time as specified in SLA. In case of SLA Violation, penalty has to be paid by the cloud service provider which reduces the overall profit of cloud service provider [5].

Now a day's cloud computing technology becomes more popular among the different organization. Large number of user accessing the cloud computing environment which induces huge amount of data to be stored on cloud data center. The large amount of data consumes an enormous amount of energy. The total energy cost includes data computation and transmission costs along with data center cooling energy cost [D]. According to statistics, the investment of energy cost over the infrastructure of data centers are three times of IT operations [6].

In this paper, our focus is to design energy efficient dynamic VM consolidation technique that works on reducing energy consumption, VM migration, and SLA violation. It optimizes the resource provisioning in order to provide efficient resource utilization. The subsequent sections are organized as follows: section 2 presents related work on energy efficient dynamic VM consolidation in data center. Section 3 presents the methodology and metrics for proposed work. Section 4 represents the experimental setup and result analysis. In section 5 represents conclusion and future work.

2. RELATED WORK

This section gives brief literature review on research works in terms of VM migration, energy aware resource allocation, Interquartile range method.

Horvath et al. [7] proposed an energy saving approach by dynamically adjusting the server voltages to minimize the total power consumption while still meeting delay constraints in a multi-tier web application environment. Nathu ji et al. [8] have proposed virtual power approach to manage online power in virtual data centers, which has been implemented by integrating power management mechanism and policies in these virtual environments.

Kusic et al. [9] have implemented and validated a Limited Look-ahead Control (LLC) structure for dynamic resource provisioning and power management in virtual environments. LLC is used to maximize the profit of the system by minimizing energy consumption as well as SLA violations. It decides (i) The number of virtual machine to provision to each application, (ii) The number of host on which virtual machine is allocated, (iii) The CPU share to each virtual machine, (iv) The number of host machine to power on.

In literature [10], Srikantaiah have applied multi-dimensional bin packing problem of allocating and migrating workloads to achieve optimal operation. Virtual machine consolidation based on two characteristics CPU and disk usage. But experiments show the model can well balance energy consumption and performance in virtual environment. Stoess et al. [11] have proposed a novel framework for energy management in modular, multi-layered operating system

structure. The framework provides a unified model to partition and distribute energy, and mechanisms for energy-aware resource accounting and allocation.

Beloglazov et al. [12] have implemented an energy-aware resource allocation for VMs consolidation in CloudSim. Firstly they set a fixed upper threshold for hosts in data center.

Secondly probes each host utilization in constant period, if it exceeds the threshold, and then marks it as an overloaded host. Finally, choose VMs from those hosts selected in the second step to migrate. However, fixed threshold is not suitable for virtual environment with mixed workloads. In the recent work [13], they replace the fixed threshold with dynamic threshold. Furthermore four different VM allocation algorithms are presented to judge overloaded host and three VM selection algorithms are proposed to select VM for migration. Experiment show that allocation and selection algorithm can extensively save energy consumption.

Zhibo Cao et al. [14] propose a novel allocation and selection policies for dynamic virtual machine consolidation in virtualized data center to reduce energy consumption and SLA violation. They uses the mean and standard deviation of CPU utilization for VM to determine the hosts overloaded or not, then use the positive maximum correlation coefficient to select VMs from those overloading host for migration.

3. METHODOLOGY AND METRICS FOR PROPOSED WORK

3.1 Interquartile Range (IQR) method for finding dynamic threshold

The interquartile range (IQR) is an estimate of variability, based on dividing a data set into quartiles. It is the difference between the upper and lower quartile in a data set.
Steps for finding Interquartile Range:

1. Sort the data set in increasing order.
2. Find the median for the ordered set (Q_2).
3. Divide the data set into two halves.
4. Find the median for the first half of the ordered data set (Lower Quartile Q_1).
5. Find the median for the second half of the ordered data set (Upper Quartile Q_3).
6. IQR = Upper Quartile – Lower Quartile.

Here data set defines set of the host utilization. We propose a method based on two threshold values, lower threshold and upper threshold. The median for the first half of the ordered data set (host utilization) is used to calculate the lower threshold value, while median of second half of the ordered data set (host utilization) is used to calculate the upper threshold value. This is shown in example as follows:

Let us assume the utilization of each host (in terms of percentage).

List of host utilization [23,65,10,75,50,84,15,30,90,12]

1. After sorting [10, 12, 15, 23, 30, 50, 65, 75, 84, 90]
2. Median = (30+50) / 2 =40
3. First half [10, 12, 15, 23, 30], Second half [50, 65, 75, 84, 90]
4. Median of the first half =15, which is used as lower threshold value.
5. Median of Second half = 75, which is used as upper threshold value.

3.2 Overloaded host detection and load balancing

For detecting overloaded host we need to find the host utilization. Assumes that there are N host in the cloud data center, where i^{th} host is denoted as H_i and number of VMs on H_i as K_i, current request of jth cloudlet on host H_i is denoted as $C_{i,j}$ and host utilization is denoted as H_u.

$$H_{u_i} = \frac{\sum C_{i,j}}{M_i} \qquad \text{... (1)}$$

Where M_i denotes the total MIPS of ith host.

In this section, we adopt a dynamic threshold value based on the interquartile range of host utilization and it is evaluated in section 3.1. If the host utilization is greater than the upper threshold value then this host is considered as overloaded host. Over utilization of host is caused by unpredictable as well as destructive VM provisioning. It may result in performance degradation and high energy consumption. When over utilization occurs load balancing is much needed for balancing over utilized host. In our algorithm after finding the overloaded host we select VM from VMmigrationList1 for migrating some VM form this host to another host.

VMmigrationList1 is determined by using various policies:

1. The minimum utilization policy: It is a simple method to select VMs which have minimum utilization among other VMs.
2. The random choice policy: It randomly selects any VM and checks whether host is still overloaded then repeat this method again.
3. The minimum migration time policy: It selects a VM which has minimum migration time among other VMs.
4. The maximum utilization policy.

After selection of VM from VMmigrationList1 then checks best suitable form MostLikelyOverloadedHostList and if its utilizationAfterAllocation (utlzAfterAllocation) is greater than previous utilization on this host (prevUtlzOnThisHost) along with powerAfterAllocation (powerAfterAllocation) is less than min Power then select this host for VM migration.

Algorithm 1: VM placement algo for Overloaded Host
1. For each VM in VMmigrationList1 {
2.   For each Host in MostLikelyOverloadedHostList {
3.   If (this host is suitable for VM) {
4.   Calculate utlzAfterAllocation;
5.   Calculate powerAfterAllocation;
6.   If ((utlzAfterAllocation> prevUtlzOnThisHost)
            && (powerAfterAllocation < minpower))
7.   targetHost = thisHost
8.   }
9.   }
10.   Add (VM, Host) pair to MigrationMap
11. }

Algorithm 2: For finding MostLikelyOverloadedHostList
1. For each Host in descending order of capacity {
2.   If (lowerThreshold<hostUtilization <upperThreshold)
3.   Add (host) to MostLikelyOverloadedHostList;
4.   }

3.3 Underloaded host detection and server consolidation

In case of under utilization energy wastage is more because servers typically need up to 70 % of their peak energy even at their low utilization level. So there is a need of VM migration technique which consolidates VM onto the minimum number of server for reducing energy consumption. Lower threshold is used to detect whether host is under loaded or not. Host utilization is calculated form eq. (1).

If host utilization is less than lower threshold then it is considered as under loaded. Then all the VM of this host are selected to migrate to other host by applying server consolidation technique and this host is switched to idle mode [12].

In this algorithm VMmigrationList (VMmigrationList2) contains all VMs from under loaded host then checks suitable least loaded host form MostLikelyUnderloadedHostList. Whereas utlzAfterAllocation means utilizationAfterAllocation, prevUtlzOnThisHost means previous utilization on this host.

Algorithm 3: VM placemen algo for Underloaded Host
1. For each VM in VMmigrationList2 {
2.   For each Host in MostLikelyUnderloadedHostList {
3.   If (this host is suitable for VM) {
4.   Calculate utlzAfterAllocation;
5.   Calculate powerAfterAllocation;
6.   If ((utlzAfterAllocation > prevUtlzOnThisHost)
            && (powerAfterAllocation < minpower)) {
7.   targetHost = thisHost
8.   }
9.   }
10.   }
11.   Add (VM, Host) pair to MigrationMap
12. }
13. Apply the server consolidation

In algorithm 4 we replace host list by Most Likely Underloaded Host List which is defined as follows

---

Algorithm 4: For finding MostLikelyUnderloadedHostList
1. For each Host in data center {
2.   If (hostUtilization < lowerThreshold)
3.   Add (host) to MostLikelyUnderloadedHostList;
4.   }

3.4 Energy Efficient Dynamic VM Consolidation: Proposed Algorithm

Input: VMmigrationList1, VMmigrationList2, MostLikelyOverloadedHostList, and MostLikelyUnderloadedHostList.

Output: VM Allocations to avoid SLA violation and reduce Energy consumption.

For each Host in data center {

1. Finding overloaded host {
2.   If host is overloaded then
3.   Select VM from VMmigrationList1 until host is overloaded
4.   Placement of VM from over loaded host
       (VM placement algo for Overloaded Host);
5.   }
6. Finding under loaded host {
7.   If host is under loaded then
8.   Select all VMs from this host
9.   Add all VMs to VMmigrationList2
10.   Placement of VM from underloaded host
       (VM placement algo for Underloaded Host);
11.   }
12. Break
13. }

The proposed algorithm finds over loaded host in first step then selects the virtual machine for migration from overloaded hosts and place this VM using VM placement algo for Overloaded Host. After load balancing, finds an under loaded host and select all the virtual machines from that host. Place these VMs over least loaded host by using VM placement algo for Underloaded Host.

4. IMPLEMENTATION

In this section, we are going to discuss about the implementation part and it is having two subsection experiment setup and experimental results.

4.1 Experiment setup

For performing experiments physical machine has a 3.40GHz Intel(R) core(TM) i7 processor, 2GB RAM, 640 GB hard disk, and 64-bit Operating System Windows 7 Professional. In our experiment latest version of cloudsim toolkit, cloudsim 3.0.3was used for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithm [15]. It is a java based simulator to provide rich library and user friendly GUI and is better than other simulators like gridsim or gangsim.

Cloudsim consist of various entities such as data centers, hosts, brokers, virtual machines and cloudlets. Data center contains multiple hosts and each host configured as 10 GB RAM, bandwidth up to 100 Gbps, and has 3000 the capacity (MIPS). Each VM is configured as image size of 2500 MB, 128 MB for RAM, 1 processing units, and has 1000 the processing capacity (MIPS).For a single data center, brokers are configured to communicate between users and data center [16]. Broker is an entity which is responsible for assigning a task to a data center. It receives requests from user and sends it to the best suitable data center, after the successful completion of tasks data center returns an acknowledgement to Broker.

4.2 Experimental result

We have analyzed different scenarios by taking 10 hosts, 20 virtual machines and various numbers of tasks (cloudlets) to evaluate the performance of proposed algorithm.
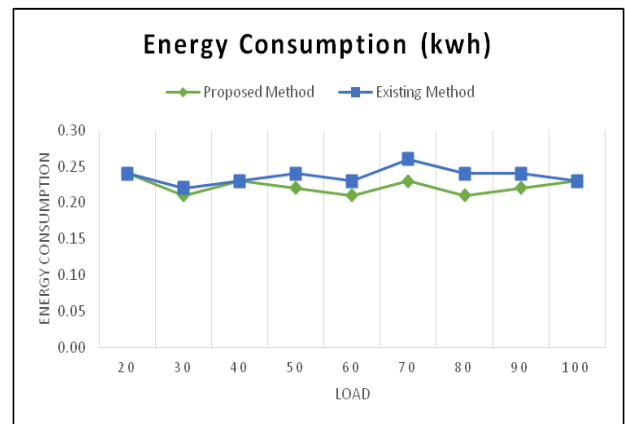

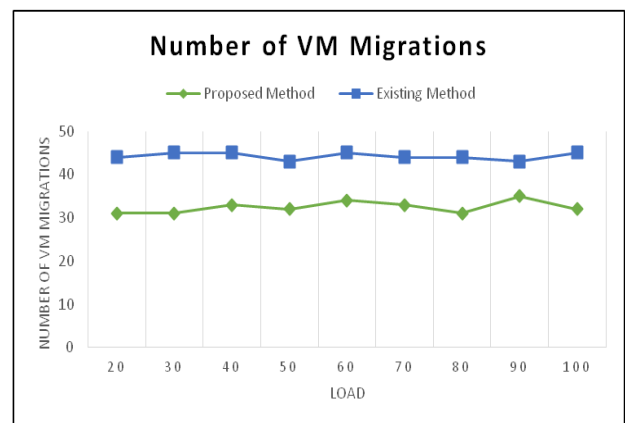**Fig 3. Energy Consumption comparison between Proposed and Existing method**

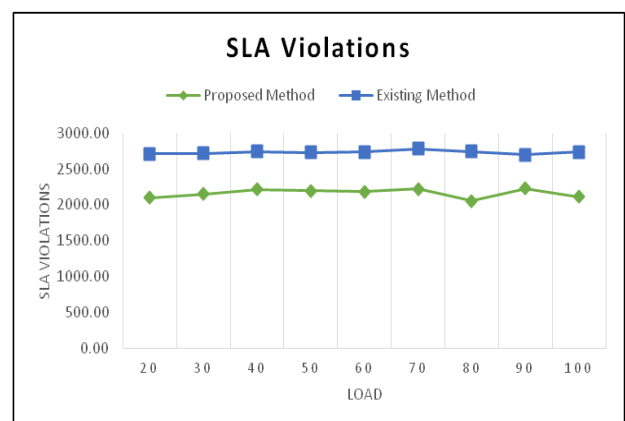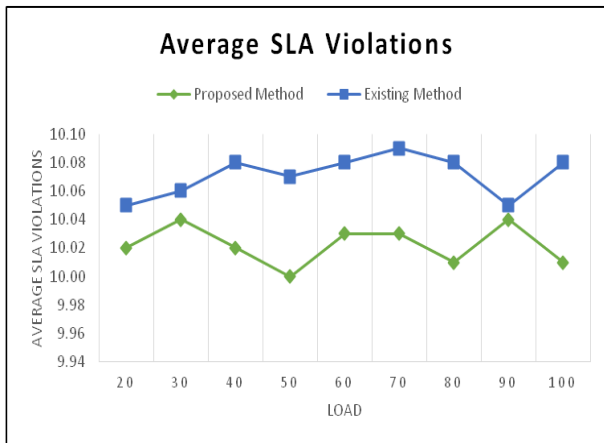**Fig 4. Number of VM Migration comparison between Proposed and Existing method**

Fig 5. SLA Violation comparison between Proposed and Existing method

Fig 6. Average SLA Violation comparison between Proposed and Existing method

The result shows that proposed method Energy Efficient Dynamic VM Consolidation algorithm performs much better than existing method in all scenarios. For metric Number of VM migrations, it gives less number of VM migrations as compared to existing method in all situations. Percentage of improvement in result for metric Number of VM migrations is

$$percentage\ of\ Improvement$$
$$= \left(\frac{(AvgMig_{old} - AvgMig_{new}) \times 100}{AvgMig_{old}}\right)\% \qquad \dots (2)$$

Where $AvgMig_{old}$ denotes average number of VM migrations for existing method, $AvgMig_{new}$ denotes average number of VM migration for proposed method. In case of number of VM migration the proposed method gives 30 percent enhancements over existing one. For every metric such as SLA violations, Energy consumption and Average SLA violation proposed method showed improvement compared to old method.

5. CONCLUSION AND FUTURE WORK

In this paper, the proposed Energy Efficient Dynamic VM Consolidation algorithm not only minimizes number of VM migrations but also reduces both energy consumption and SLA violations. The result shows better performance for every metrics and efficient resource utilization of data centers. It is a critical task to make tradeoff between energy consumption and SLA violation. As far as number of VM migration and SLA violation is concerned we have achieved a lot of improvement. Proposed method consolidates VM onto the minimum number of server for reducing energy consumption. But still in case of energy consumption more work is needed for further improvement.

Energy consumption for each host totally depends upon the energy model on the host. Therefore in future we will design another strategy to reduce energy consumption while keeping low level of SLA violations and less number of VM migrations.

6. REFERENCES

[1] Jing Huang, Kai Wu, and Melody Moh, "Dynamic Virtual Machine Migration Algorithms Using Enhanced Energy Consumption Model for Green Cloud Data Centers," © IEEE 2014.

[2] Kamyab khajehei, "Role of Virtualization in Cloud Computing,"International Journal of Advanced Research in Computer Science and Management Studies (ijarcsms),Volume 2, Issue 4, April 2014.

[3] P. Getzi Jeba Leelipushpam, and Dr. J.Sharmila, "Live VM Migration Techniques in Cloud Environment- A Survey,"IEEE Conference on Information and Communication Technologies (ICT) © 2013.

[4] Yatendra Sahu, R.K. Pateriya, and Rajeev Kumar Gupta, "Cloud Server Optimization with Load Balancing and Green Computing Techniques Using Dynamic Compare and Balance Algorithm,"5th International Conference on Computational Intelligence and Communication Networks© IEEE 2013.

[5] Linlin Wu, Saurabh Kumar Garg, Steve Versteeg, and Rajkumar Buyya, "SLA-Based Resource Provisioning for Hosted Software-as-a-Service Applications in Cloud Computing Environments," IEEE Transactions on services computing, VOL. 7, NO. 3 JULY-SEPTEMBER 2014.

[6] C.L.Belady, "In the Data center, power and cooling costs more than the equipment it supports," Nov. 2013.

[7] G. Chen, W. He, J. Liu, and S. Nath, "Energy-aware server provisioning and load dispatching for connection-intensive internet services," in proceedings of the 5th USENIX symposium on networked system design and implementation, 2008.

[8] R. Nathuji, K. Schwan, "Virual power Coordinated power management in virtualized enterprise system," Proc. of the ACM press, Dec. 2007.

[9] D. Kusic, J.O.Kephart, J.E.Hanson, et al. "Power and Performance Management of Virtualized Computing Environments via Look ahead Control," Cluster Computing, Springer press, 2009.

[10] S. Srikantaiah, A. Kansal, F. Zhao, "Energy aware Consolidation for Cloud Computing," Proc. USENIX Workshop on Power Aware Computing and Systems in Conjunction with OSDI, 2008.

[11] J. Stoess, C. Lang, F. Bellosa, "Energy Management for Hypervisor based Virtual Machines," USENIX Annual Technical Conference on Proceedings, July 2007.

[12] A. Beloglazov, J. Abawajy, R. Buyya, "Energy-aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing," Future Generation Computer Systems May 2012.

[13] A. Beloglazov, R. Buyya, "Optimal Online Deterministic Algorithms and Adaptive Heuristics for Energy and Performance Efficient Dynamic Consolidation of Virtual Machine in Cloud Data Center," Concurrency and Computation Practice and Experience (CCPE), Wiley press Sep 2012.

[14] Zhibo Cao, Shoubin Dong, "Dynamic VM Consolidation for Energy-aware and SLA Violation Reduction in Cloud Computing," 13th International Conference on Parallel and Distributed Computing, Applications and Technologies© IEEE 2012.

[15] R.N. calheiros, R.Ranjan, A.Beloglazov, and R.Buyya, "CloudSim a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," Software-practice and Experience, 2011.