

# Energy Efficient Dynamic Integration of Thresholds for Migration at Cloud Data Centers

Richa Sinha  
Information Technology  
L.D. College of Engineering  
Ahmedabad, India

Nidhi Purohit  
Information Technology  
L.D. College of Engineering  
Ahmedabad, India

Hiteishi Diwanji  
Computer Department  
L.D. College of Engineering  
Ahmedabad, India

## ABSTRACT

Cloud Computing is one of the fast spreading technologies for providing utility-based IT services to its user. Large-scale virtualized data-centers are established to meet this requirement. Data centers consume large amount of computation power for providing efficient and reliable services to its user. Such large consumption of electrical energy has increased operating cost for the service providers as well as for the service users. Moreover, a large amount of carbon dioxide is emitted, results into increased global warming in near future. From our studies we concluded that, power consumption can be reduced by live migration of the virtual machines (VM) as required and by switching off idle machines. So, we proposed a dynamic threshold based approach for CPU utilization for host at data center. This consolidation will work on dynamic and unpredictable workload avoiding unnecessary power consumption. We will not only meet energy efficiency requirement but would also ensure quality of service to the user by minimizing the Service Level Agreement violation. We would also validate the proposed technique results with higher efficiency.

## General Terms

Algorithms, Table

## Keywords

Energy Efficient, Green IT, Cloud computing, Live Migration, CPU Utilization, VM Selection, VM Placement

## 1. OVERVIEW

Cloud computing is modeled to provide service [1] rather than a product. Services like computation, software, data access and storage are provided to its user without its knowledge about physical location and configuration of the server which is providing the services. Cloud works on the principle of virtualization of resources with on-demand and pay-as-you-go model policy [2]. End – user does self- service to access an available pool of computing resources to do its job in just few minutes instead of taking months.

The main advantage of using cloud as a service is that, it reduced the end-user cost of buying resource like software and other applications.

It also increases the storage area of the private computers without any additional storage. End - user no distance wireless communication technology is examined and the range of wireless data transmission is 100 m.

longer need to bother about up gradation of the resources. Cloud is more flexible and mobile. It also allows IT to shift its focus on innovation rather than worrying about constant server updates and other computing issues.

In order to provide services, large-scale data centers are established. These data center contain thousands of running

computational nodes providing virtualization by placing many virtual machines (VMs) on each node.

## 1.1 Energy Utilization Issues on Data Centers

The enormous amount of electrical energy is needed to run a data center which is either obtained by the organization outsourcing it to cloud in pay back as service that they used from cloud or by directly from the power sources. This causes emission of large amount of carbon dioxide which will lead to many environmental issues in near future. First and foremost is global warming and greenhouse effect.

The power consumption by IT infrastructure has doubled from 2000 to 2006 and will double again till 2011. US uses about 61 billion kWh energy which leads to the total cost of 4.5 billion dollar of electricity bill which incurred by the companies. Such data centers in US are alone using 1.5 % entire electricity of US [3]. Facebook's data centers are using 10.52% of total power used for entire IT data centers which highest of all. Second on list is Google with 7.74% of total power consumption and next is YouTube with 3.27% and so on [4]. According to a survey, the data taken from 5000 servers showed that only 10-15% of their total capacity is used [5]. The inadequate usage results into underutilization of the resources causing large scale unnecessary power consumption. According to another survey, an idle machine unnecessarily uses 70% power of data centers [6], again resulting into consumption of large amount of energy. If just a corner amount of this energy can be saved by any means, a new direction can be given to support green revolution. Moreover, this extra power can be utilized at some other areas for betterment in term of social aspects.

So, we concluded from our studies that most of the power is wasted because of underutilization and ideality of resources at data centers. In our approach, we have considered these factors to save energy.

## 1.2 CPU Utilization

Live migration of VMs is done for three reasons: resource requirement, power consumption and affinity of VM. By migrating a VM across physical nodes at data centers our problem can be solved. Main advantage is that it separate hardware and software and also brief about fault management, load balancing and low-level system maintenance [7].

## 1.3 Live Migration

Live migration of VMs is done for three reasons: resource requirement, power consumption and affinity of VM. By migrating a VM across physical nodes at data centers our problem can be solved. Main advantage is that it separate hardware and software and also brief about fault management, load balancing and low-level system maintenance [7].

Live Migration for load balancing (Figureure 1) is done for two types of VMs: underloaded VM and overloaded VM. An underloaded VM are those VM which are underutilizing its CPU capacity. All the VM of such node are migrated to those nodes whose residual capacity is big enough to hold them. So the latter node is switched off to save power. An overloaded VM is one which has already crossed its utilization capacity. In this case, migration is done to underloaded VM [7, 8, 9]. Live migration if taken place continuously can lead to the performance degradation of the node. So continuous monitoring scheme can applied to minimize the VM migration and ensuring Quality of service by minimizing the SLA violation.

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. We make some conclusion and other possible direction for future research in Section 6.

## 2. PREVIOUS THEORIES

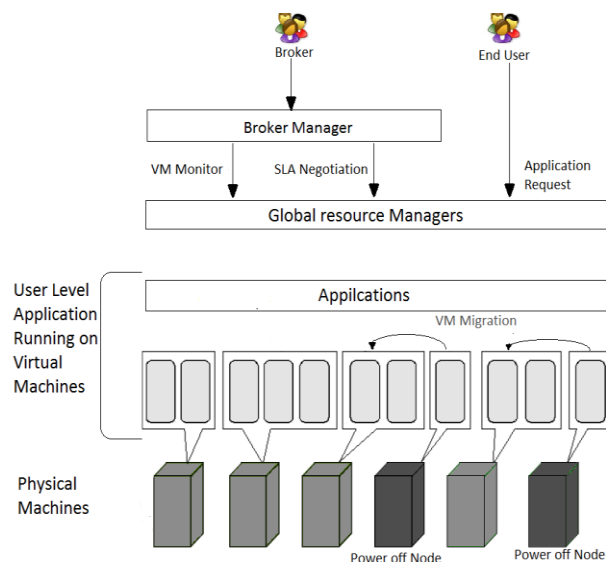
Many related study and work has been proposed for energy management scheme on data centers for cloud. In [10], resource allocation at data centers is done according to the priority, but it doesn't apply VM migration for optimization. In [11], an energy management scheme is proposed by comparing multiple strategies for finding minimum-power network subsets across a range of traffic pattern, but finding such networks on geographically separated big network may decrease the performance of the network. In [12], a threshold based reactive approach to dynamic workload handling but this approach is not much suited in IaaS environment. In [13], again a threshold based approach is proposed using single threshold value as upper limit for utilization but the node has to remain active even if the load is much less than threshold value. In [14], a DVFS (Dynamic Voltage Frequency Scaling) based scheme is proposed.

The systems adjust itself dynamically for lower power consumption using frequency and voltage. Similar to this, in [13] again a Non Power aware approach for lower power consumption is made. These approaches may degrade with quality of service for the end-user. VMware Distributed Power Management [15] operates based on thresholds with the lower and upper utilization thresholds set to 45% and 81% respectively. It was not justifiable to decide the upper and lower limit of threshold in unpredictable workload.

So, here we propose a scheme based on dynamic threshold which determines the CPU utilization dynamically for unpredictable workloads.

## 3. SYSTEM MODEL

We consider the cloud computing system model as shown in the Figureure 1, the target systems are of IaaS Environment.



**Figure 1: Block Diagram of Proposed Work**

In a cloud computing environment, we considered mainly two types of actors on cloud: end-user and brokers. The end-user service requests for the application on cloud and brokers process these request. As per our system, we have considered two major roles for brokers: SLA Negotiation and VM Monitor. The SLA Manager takes care that no Service Level Agreement (SLA) is violated and VM Monitor monitors the current stated of virtual machines periodically at specific amount of time. All these request are taken by a global resource manager which decides what type of application is been requested and accordingly the VM machine is generated at physical nodes.

We have considered two types of Manager: 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.

Our system model considers three main theories.

### 3.1 Power verses Utilization

Many studies [17,18] shows the power consumption by servers can be described by a linear relation between the power consumption and CPU utilization. These studies as say that an average power consumed by an idle server is 70% of power consumed by fully utilized server. So, we considered the power consumption as CPU utilization  $P(u)$  by as shown in (1):

$$P(u) = P_{max} (0.7 + 0.3 u)$$

where Pmax is 250 W for modern computing server and u is the CPU utilization[20]. But, CPU utilization change with respect to time i.e. u(t). So the total energy consumed (E) as shown in (2):

$$E = \int P(u(t)) dt$$

So the total energy consumption can be measured from CPU utilization from this model.

### 3.2 Cost

We consider our result on two types of cost.

#### 3.2.1 Cost at data center

With decrease in power consumption using live migration which results in decreasing operating cost for the data center. We consider here cost as shown in (3):

$$C_{dc} = c * E$$

(3)

where c is the cost of 1 kW power.

#### 3.2.2 Cost at user side

Again due to decrease power consumption user side cost is decreased. User side cost can be calculated as shown in (4):

$$\begin{aligned} C_{user} = & C_{second} * \text{no. of seconds} \\ & + C_{bandwidth} * \text{total allocated bandwidth} \\ & + C_{memory} * \text{total allocated Ram} \\ & + C_{Storage} * \text{total allocated Storage} \end{aligned} \quad (4)$$

where cost per second, cost per bandwidth, cost per memory and cost per storage is decided by the datacenter.

### 3.3 SLA Violation

QoS needed to be met for Cloud computing environments. QoS is determined in the form of SLA (Service Level Agreement), which is determined either by minimum throughput or maximizes response time. This can differ from system to system. For our studies, we consider SLA violation as shown in (5):

$$SLA = \frac{\sum (\text{requested MIPS}) - \sum (\text{allocated MIPS})}{\sum (\text{requested MIPS})} \quad (5)$$

The percentage of this value will show CPU is not allocated even if it is demanded.

Therefore, in order to increase the QoS for the end-users, our prior goal is to minimize this SLA from getting violated.

## 4. PROPOSED STRUCTURE

Here, we proposed dynamic threshold based scheme. We divide the algorithm in two parts: (1) Selection of VM for migration and (2) Placing the VM on proper host.

### 4.1 Selecting VM for Migration

The selection of VM for migration is done to optimize the allocation. Here, we first calculated the CPU utilization of all VMs as shown below in (6):

$$U_{vm} = \frac{\text{totalRequestedMips}}{\text{totalMips for that VM}} \quad (6)$$

Beside utilization we also have considered allocated ram and bandwidth for both virtual machine and host as shown in (7):

$$Bw = \sum \text{current allocated bandwidth for VMs for host}$$

$$Ram = \sum \text{current allocated Ram for VMs for host}$$

And also  $Sum = \sum U_{vm}$

We, hence show in our scheme the two threshold values.

#### 4.1.1 Upper Threshold

The CPU will be considered overloaded when the utilization is above this value so we migrate some of the VMs. Here, so went on calculating this value i.e. Tupper for each host separately by following equations in (8):

$$temp = Sum + (Bw / Bw(host)) + (Ram / Ram(host))$$

$$Tupper = 1 - (((0.95 * temp) + Sum) - ((0.90 * temp) + Sum)) \quad (8)$$

where, for each host we preserve amount of CPU capacity by upper 0.95 and lower as 0.90 probability limits.

#### 4.1.2 Lower threshold

The node is considered to be underutilized when the CPU utilization is below this value so all VMs are migrated to other node. From our study in [13], we considered that if the CPU utilization is above 30%, lower threshold (Tlower) is always 0.3. So, we define equations for calculating lower threshold for each node as follows in (9):

$$sum1 = (Sum + Bw + Ram) / (n * Bw(host) + Ram(host));$$

$$sqrt = \sqrt{((Sum - sum1) * (Sum - sum1))};$$

$$Tlower = \begin{cases} sqrt - (0.3 * sum1), & \text{if CPU utilization is } < 30\% \\ 0.3, & \text{if CPU utilization is } \geq 30\% \end{cases} \quad (9)$$

where, we considered maximum probability limit for this threshold as 0.3 and n is number VMs on the host.

After defining the dynamicity of lower and upper threshold from the equation (8) and (9) respectively, we describe our theory for Dynamic Threshold based Live Migration as shown in the Algorithm 1. We have tried using a minimized migration policy rather simple migration policy for a better QoS.

**Algorithm 1:** Live Migration using dynamic threshold

**Input:** host list, VM list      **Output:** migration list

- Sort the VM list in the decreasing order of its VM utilization.
- 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 goto 3 else goto 7. Fix two best fit utilization : bfuupper and bfulower with max value assignment.
- Get the each VM for the current host. If VM utilization is greater than the difference of current host utilization and upper threshold value, then goto 4 else goto 5.
- If VM utilization – host utilization + upper threshold of host is greater than bfuupper then, bfuupper =

VMutilization –(host utilization – upper threshold) and best fit VM is current VM.

- If  $bfuupper = \max$  then, if (  $hostutilization - upperthreshold$ ) – VMUtilization is less than  $bfuupper$  then,  $bfuupper = (hostutilization - upper threshold) - VMutilization$ . And best fit VM is current VM.
- Adjust the value of host utilization as difference of current host utilization and best fit VM utilization and add the best fit VM to the migration list and remove the VM from the current host.
- If host utilization value is less than lower threshold value then add all the VM of the host to the migration list and remove all the VM from the host.
- Return the migration list.
- Return allocation

## 4.2 Placing of VM

We have considered placing of VM as a bin packing[22] 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 :** host list, VM list      **Output:** allocation of VMs

- Sort the VM list in the decreasing order of its VM utilization.
- For each VM in VM list, allocate minpower as maximum power and allocatedHost as null.
- For each host in host list, if host has enough resource for VM then estimate power of VM and host. If power is less than minpower then allocated host is current host and minpower is power difference of VM and host.
- If allocatedHost is not null then allocate VM to the allocatedHost.
- Return allocation

## 5. EXPERIMENTAL RESULTS WITH EVALUATION

We tested our work on Cloudsim Toolkit [21]. In our experiment, we have worked with just one datacenter. We took up with 10 host on this datacenter which in turn is running 20 virtual machines on those host. Each node comprises of one CPU core with 10 GB ram/network bandwidth and storage space of 1TB. The host comprises of 1000, 2000 and 3000 MIPS accordingly. For each virtual machine on host ram size is 128MB and bandwidth size is 2500 MB with 250, 500, 750 and 1000 MIPS accordingly. For our experiment we have just worked with one resource. Initially the VMs are considered to be utilized by 100% of time.

As shown in the previous section, we have used best fit decreasing order for placement of VMs on the host to save power. As per our theory, we have used two threshold values. The equations for calculating these threshold values are already discussed in the previous section. We have taken value of  $c$  as Rs 8 in order to calculate cost for billing at datacenters. We have also taken other cost value to calculate debits at user side as Cseconds as Rs 3 , Cmemory as Rs 0.05, Cstorage as Rs 0.001 and Cbandwidth as Rs 0.

Firstly, we tried to work on analysis of conceptual of live migration and its implementation on Cloudsim Toolkit. Then we went on studying the power examples already implemented i.e. DVFS [14] and NPA [13]. These examples are not following the migration policy. Then along with the understanding of live migration, we tried implementing single threshold on it. In this, a static assignment of upper limit threshold value is done and no concept of lower limit. While experimenting we considered this limit as 0.6. With this theory we implemented migration with two static values of the thresholds that is upper and the lower we considered 0.8 as upper threshold and 0.2 as lower threshold. Finally, we moved on implementing our concept of dynamic threshold using the threshold theories stated in previous section. We compared the non migrating algorithms like DVFS and NPA algorithms with the Single threshold(ST) algorithm. We came with some results as shown below in Table 1.

**Table 1: The comparison of 3 types of power aware algorithms**

	Energy kWh	Cost (DC) Rs	Cost (User) Rs	SLA violation	VM Migration
DVFS	0.25	2	53.4	----	----
NPA	0.86	6.88	160.2	----	----
ST	0.29	2.32	551.8	98.22%	44

As shown in the table 1, we concluded that by using power efficient policy for migration, energy usage can be minimized resulting into decreasing electricity bills for data centers. NPA is using maximum amount of power among all the theories taken into consideration resulting into more cost. DVFS may use less energy but for the real scenario it may change because its entire dependency is limited to voltage and frequency. The single threshold is violating the maximum number of SLA with nominal energy consumption.

**Table 2: The comparison of double(DoT) and dynamic threshold(DyT) algorithms**

	Energy kWh	Cost (DC) Rs	Cost (User) Rs	SLA violation	VM Migration
DoT	0.26	2.08	453.9	82.09%	30
DyT	0.25	2.03	1171.3	67.28%	22

As shown in the Table 2, in the double threshold the SLA violation has dropped by around 15% from the single threshold. For dynamic threshold, we found a more drop in SLA violation as compared to double threshold. This will help in improving QoS. As shown from the table 1 and 2 we can also see bit fluctuation in energy and cost is found.

After the above results, we continued to look into the behavior of our algorithm for all the theories mentioned in section 3 along with the VM Migration policies. From this analysis, we took into consideration 1 to 10 host with the VMs running on it. The host to VM ratio is 1:2. We have used some of the algorithms for this test bed to compare the performance of our algorithm. Following are the results as shown in graphs.

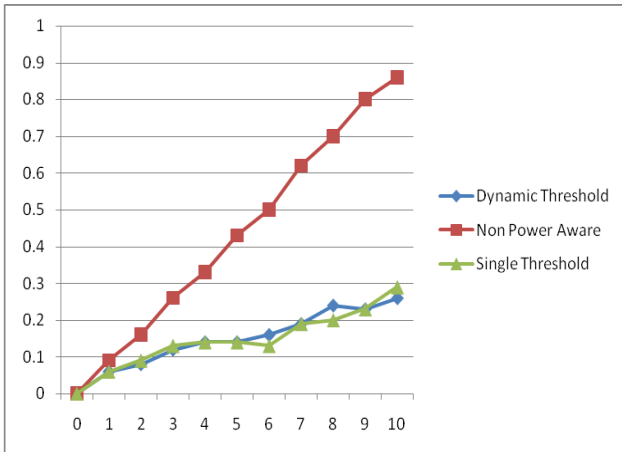


Fig 2: Energy Consumption v/s Host for different policies

In figure 2, as shown, NPA uses maximum energy with linear growth. ST and DyT uses almost similar amount of energy with ST using slightly more than DyT.

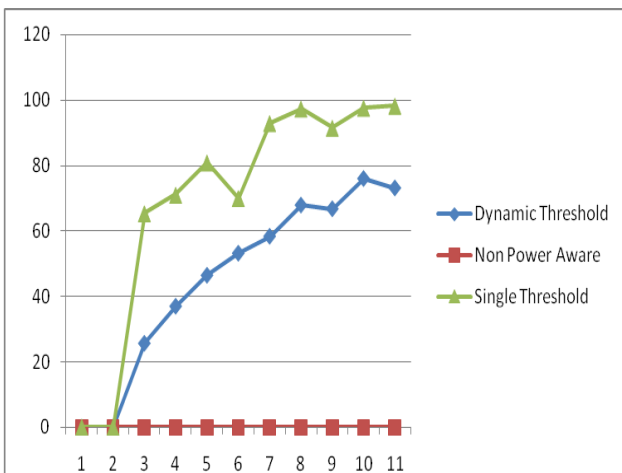


Fig3: SLA Violation v/s Host for different policies

In figure 3, NPA is not using any migration policy so the SLA violation is 0 for this, whereas by using ST we got SLA violation upto 98% which is too high. DyT has tried to minimize this by 20% to 25%.

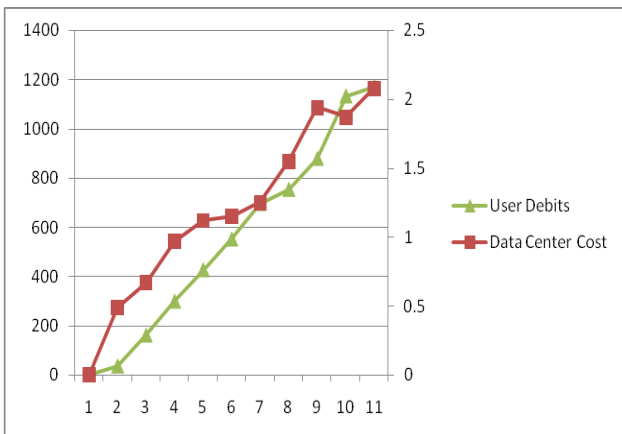


Fig 4: Cost v/s Host for Dynamic Thresholds

In figure 4 as shown, the left axis shows the user debts and the right axis shows data center cost. We have not tried to compare any algorithm because cost will increase as the energy consumption will increase. The graph above shows the increment in cost as number host for the resources increases

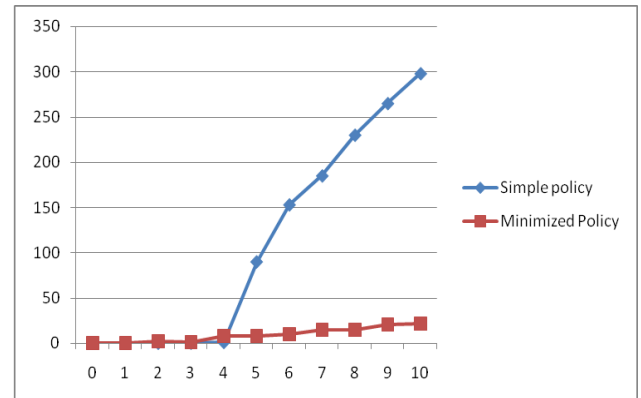


Fig 5: VM Migration policy v/s Host for Dynamic Thresholds

In figure 5 as shown, we have compared simple migration policy with minimized policy for live migration VM. We have huge amount of drop on VM Migration. This has resulted into better QoS by reducing the SLA violation.

## 6. CONCLUSION AND FUTURE WORK

From our study we conclude that dynamic consolidation of VM and switching off idle servers maximizes the energy efficiency of the resource. We proposed a dynamic threshold based CPU utilization for the dynamic and unpredictable workload for the cloud. The algorithm has tried to reduce the power consumption which can be a small step towards Green technology. Dynamicity has also reduced human labor of defining the threshold limit for the host and made the machine capable to calculate threshold value itself. Moreover, we have tried to implement minimized migration policy which gives better can quality of service to the users by minimizing the SLA violation for the resource. We also showed the cost structure at both user and data center side. By providing quality of service both broker and end – users will be benefited. This algorithm is been tested and simulated on with our results which clearly show that by increasing CPU utilization more work can be done.

For our future work, we would like have a test bed that can investigate the algorithm behavior with multiple numbers of resources. Secondly, we would also like to investigate this technique on real cloud setup and check what will be its exact reaction on real environment. This can be a small social step for significant decrease in emission of carbon dioxide along with reduction in infrastructure and operating cost.

## 7. REFERENCES

- [1] R. Buyya, CS Yeo, S. Venugopal, J. Broberg, I. Brandic, "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility, Future Generation Computer Systems, 2011
- [2] R. Buyya et al. Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities. In Proc. of the 10th IEEE Intl. Conf. on High Performance.
- [3] R. Brown. "Report to congress on server and data center energy efficiency: Public law 109-431". Lawrence Berkeley National Laboratory, 2008

- [4] Peer1 hosting site puts a survey on “Visualized: ring around the world of data center power usage”. From engadget.com ,2011
- [5] L. A. Barroso and U. Holzle. “The case for energy-proportional computing.” *Computer*, 2007
- [6] X. Fan, “Power provisioning for a warehouse-sized computer” In Proc. of the 34th Annual Intl. Symp. On Computer Architecture, 2007
- [7] C Clark, K Fraser, S Hand, J G Hanseny, E July,C Limpach, I Pratt, A Wareld, “Live Migration of Virtual Machines” NSDI’05 Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation ,2005
- [8] E Arzuaga, D R Kaeli, “Quantifying load imbalance on virtualized enterprise servers.” In WOSP/SIPEW ’10: Proceedings of the first joint WOSP/SIPEW international conference on Performance engineering, ACM, 2010.
- [9] H W Choi, H Kwak, A Sohn, K Chung, “Autonomous learning for efficient resource utilization of dynamic vm migration.” In ICS ’08: Proceedings of the 22nd annual international conference on Supercomputing, ACM, 2008.
- [10] Y. Song, “Multi-Tiered On-Demand resource scheduling for VM-Based data center” In Proc. of the 2009 9th IEEE/ACM Intl. Symp. on Cluster Computing,155, 2009.
- [11] B Heller,S Seetharaman, P Mahadevan,Y Yiakoumis, P Sharma,S Banerjee,N McKeown,” ElasticTree: Saving Energy in Data Center Networks”NSDI 2010
- [12] D. Gmach , “Resource pool management: Reactive versus proactive or let Ss be friends”. *Computer Networks*, 2009
- [13] A. Beloglazov, R. Buyya, “Energy efficient allocation of virtual machines in cloud data centers”. 10th IEEE/ACM Intl. Symp. on Cluster, Cloud and Grid Computing ,2010.
- [14] G Laszewskiy, L Wangz, A J. Youngez, X Hez,“Power-Aware Scheduling of Virtual Machines in DVFS-enabled Clusters, IEEE,2009
- [15] VMware Inc. “VMware distributed power management concepts and use”, 2010.
- [16] 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
- [17] X. Fan, “Power provisioning for a warehouse-sized computer” In Proc. of the 34th Annual Intl. Symp. On Computer Architecture, 2007
- [18] D. Kusic, “Power and performance management of virtualized computing environments via lookahead control”. *Cluster Computing*, 2009.
- [19] Jason Sonnek and Abhishek Chandra Virtual Putty: “Reshaping the Physical Footprint of Virtual Machines” HotCloud ,2009
- [20] R. yahyapour, C. Perez, E. Elmroth, I. M. Llorente, F. Guim and K. Oberle, “ Introduction” . Euro –Par 2011 Parallel Processing .Springer, 2011
- [21] R. Calheiros, R Ranjan, César A. F. De Rose, R. Buyya, “ CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services” , 2011