International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 1 Issue 10, December- 2012

Energy Efficient Threshold Based Approach for Migration at Cloud Data Center

Dimple Maheshwari
Alpha College Of Engineering and Technology,
Ahmedabad,India.

Purnima Gandhi
Alpha College Of Engineering and Technology,
Ahmedabad,India.

Richa Sinha
Kalol Institute Of Technology
Ahmedabad,India

Abstract

Cloud Computing is one of the fast spreading technologies for providing utility-based IT services to its user. large scale data-centers consume Large amount of electrical 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. Virtualization is a best approach to reduce this power consumption by consolidating multiple virtual servers onto a smaller number of computing resources. Power consumption by data-centers can be reduced by leveraging live migration of VMs and switch off idle nodes. So, we proposed a Single,Double and Dynamic threshold based approach for CPU utilization for host at data center.

Keyword:*Cloud Computing, Energy Efficient,Virtual Machines (VM), Green IT, Live Migration.*

1. Introduction

Cloud computing is modeled to provide service [1] rather than a product. In recent years, IT infrastructures continue to grow rapidly driven by the demand for computational power created by modern compute-intensive business and scientific applications. However, a large-scale computing infrastructure consumes enormous amounts of electrical power leading to operational costs that exceed the cost of the infrastructure in few years. For example, in 2006 the cost of electricity consumed by IT infrastructures in US was estimated as 4.5 billion dollars and tends to double by 2011 [3]. Except for overwhelming operational costs, high power consumption results in reduced system reliability and devices lifetime due to overheating. Another problem is significant CO2 emissions that contribute to the greenhouse effect.

One of the ways to reduce power consumption by a data center is to apply virtualization technology. This technology allows one to consolidate several servers to one physical node as Virtual Machines (VMs) reducing the amount of the hardware in use. Recently emerged Cloud computing paradigm leverages virtualization and provides on-demand resource provisioning over the Internet on a pay-as-yougo basis [2].

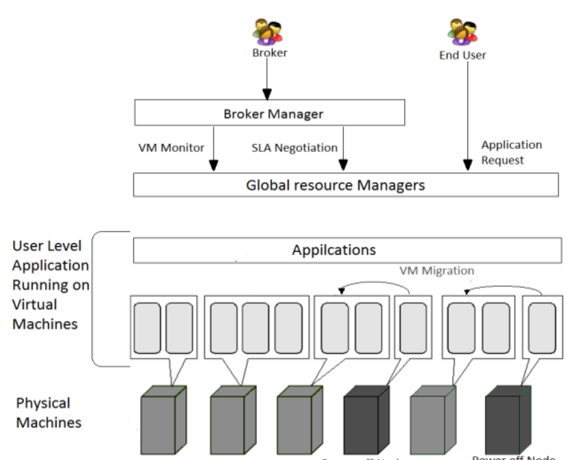


Figure1: System View of Cloud Environment

Figure 1, shows the actual system view cloud computing environment. There are mainly two types of actors on cloud: end-user and brokers. The end-user 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 monitor 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.

1.1 Literature Survey

1.1.1 Energy Utilization Issues on Data Centers

The Large 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 warning 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.

1.1.2 Utilization Concept of CPU

In general terms, CPU usage is the amount of time for which the CPU is used to process the instruction of a program. Similarly, when an application request for resource on cloud, VMs are mapped with pools of physical server [14]. These VMs are so placed, to fulfill the CPU utilization of its host so that multiple tasks can be done at once.

1.1.3 Theory of Live Migration

Live Migration for load balancing (Figure 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.

2. Power Model

Power consumption by computing nodes in data centers is mostly determined by the CPU, memory, disk storage and network interfaces. In comparison to other system resources, the CPU consumes the main part of energy, and hence in this work we focus on managing its power consumption and efficient usage. Moreover, the CPU utilization is typically proportional to the overall system load.

Recent studies [6, 11] show that the power consumption by servers can be accurately described by a linear relation- ship between the power consumption and CPU utilization, even when Dynamic Voltage and Frequency Scaling (DVFS) is applied. The reason lies in the limited number of states that can be set to the frequency and voltage of CPU and the fact that voltage and performance scaling are not applied to other system components, such as memory and network in- terface. Moreover, these studies show that on average an idle server consumes approximately 70% of the power consumed when it is fully utilized. Therefore, for our experimental studies we de_ne the power consumption as a function of the CPU utilization (P(u)) as shown in (1).

$$P(u) = k \cdot P_{max} + (1 - k) \cdot P_{max} \cdot u, \quad (1)$$

Pmax is the maximum power consumed when the server is fully utilized; k is the fraction of power consumed by the idle server (i.e. 70%); and u is the CPU utilization. For our experiments Pmax is set to 250 W, which is a usual value for modern servers. For example, according to the SPECpower benchmark, for the fourth quarter of 2010, the average power consumption at 100% utilization for servers consuming less than 1000Wwas approximately 259 W.The utilization of the CPU may change over time due to the workload variability. Thus, the CPU utilization is a function of time and is represented as u(t). Therefore, the total energy consumption by a physical node (E) can be defined as an integral of the power consumption function over a period of time as shown in (2).

$$E = \int P(u(t))dt \quad 2$$

2.1 SLA Violation Calculation

Meeting QoS requirements is extremely important for Cloud computing environments. QoS requirements are commonly formalized in the form of SLA, which can be determined in terms of such characteristics as minimum throughput or maximum response time delivered by the deployed system. As these characteristics can vary for di_erent applications, it is necessary to de_ne a generic metric that can be used in our simulation experiments to estimate the level to which the SLA are delivered by the infrastructure.

$$SLAvio = \frac{Userreq - Userallocate}{Userreq}$$

Where *Userreq* is the MIPS users request and *Userallocate* represents the MIPS actual distribution to users

3. ALLOCATION POLICIES

The system operation can be divided in two parts: (1) selection of VMs that have to be migrated to optimize the allocation; and (2) placement of the VMs selected for migration and new VMs requested by the users on physical nodes. We discuss these parts in the following sections.

3.1 VM Selection

3.1.1 Single Threshold (ST)

In our previous work we have proposed four heuristics for choosing VMs to migrate [12]. The first heuristic, Single Threshold (ST), is based on the idea of setting an upper utilization threshold for hosts and placing VMs while keeping the total utilization of the CPU below this threshold. The aim is to preserve free resources to prevent SLA violation due to consolidation in cases when the resource demand by VMs increases. At each time frame all the VMs are reallocated using the Modifed Best Fit Decreasing (MBFD) algorithm (Section 4.2) with an additional condition of keeping the upper utilization threshold not violated. New placement is achieved by live migration of VMs.

3.1.2 Double Threshold (DoT)

The other three heuristics are based on the idea of setting upper and lower utilization thresholds for hosts and keeping the total utilization of the CPU by all the VMs between these thresholds. If the CPU utilization of a host falls below the lower threshold, all VMs have to be migrated from this host and the host has to be switched off in order to eliminate the idle power consumption. If the utilization exceeds the upper threshold, some VMs have to be migrated from the host to reduce the utilization in order to prevent potential SLA violation. We have proposed three policies for choosing VMs that have to be migrated from an over-utilized host.

1. Minimization of Migrations (MM)- migrate the least number of VMs to minimize migration overhead

2. Highest Potential Growth (HPG) - migrate VMs that have the lowest usage of CPU relatively to re-quested in order to minimize total potential increase of the utilization and SLA violation

3.Random Choice (RC) - choose the necessary number of VMs randomly

3.1.3 Dynamic Thresholds

As mentioned earlier, fixed values for the thresholds are unsuitable for an environment with dynamic and unpredictable workloads, in which different types of applications can share a physical resource. The system should be able to automatically adjust its behavior depending on the workload patterns exhibited by the applications. Therefore, we propose a novel technique for auto-adjustment of the utilization thresholds based on a statistical analysis of the historical data collected during the lifetime of VMs. 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 :

Uvm=totalRequestedMips / totalMips for that VM

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

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. UpperT for each host separately by following equations .

$$S = \Sigma U_{vm} \qquad Sq = \sqrt{\Sigma U_{vm}2}$$
$$UT = 1-(((Pu * Sq) + s)-((Pl * Sq)+s))$$

where, for each host we preserve amount of CPU capacity by upper (Pu) and lower (Pl) probability limits.

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 [12], we considered that if the CPU utilization is above 30%, lower threshold (LT) is always 0.3. So, we define equations for calculating lower threshold for each node as follows .

$$S = \Sigma U_{vm} / n \qquad Sq = \sqrt{(\Sigma U_{vm} - Sum)2}$$
$$LT = s - (Pl * sq), \quad \text{if CPU utilization is} < 30\%$$
$$= 0.3, \quad \text{if CPU utilization is} >= 30\%$$

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

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. The complexity of the algorithm is proportional to the sum of the number of non over-utilized host plus the product of the number of over-utilized hosts and the number of VMs allocated to these over-utilized hosts.

Algorithm 1: Dynamic Thresholds (DT)

```
Input: hostList, vmList  Output: migrationList
vmList.sortDecreasingUtilization( )
foreach h in hostList do
hUtil ← h.util( )
bestFitUtil← MAX
  while hUtil > h.upThresh() do
     foreach vm in vmList do
        if vm.util() > hUtil-  h.upThresh() then
           t  ← vm.util() - hUtil + h.upThresh()
           if t < bestFitUtil then
                bestFitUtil ← t
                bestFitVm   vm
      else
           if bestFitUtil = MAX then
               bestFitVm  ← vm
               break
     hUtil← hUtil - bestFitVm.util()
     migrationList.add(bestFitVm)
     vmList.remove(vm)
  if hUtil < lowThresh() then
      migrationList.add(h.getVmList())
      vmList.remove(h.getVmList())
return migrationList
```


3.2 VM Placement

The VM placement can be seen as a bin packing problem with variable bin sizes and prices, where bins represent the physical nodes; items are the VMs that have to be allocated; bin sizes are the available CPU capacities of the nodes; and prices correspond to the power consumption by the nodes. As the bin packing problem is NP-hard, to solve it we apply a modi_cation of the Best Fit Decreasing (BFD) algorithm that is shown to use no more than 11/9 .OPT + 1 bins (where OPT is the number of bins provided by the optimal solution) [13]. In our modi_cation (MBFD) we sort all the VMs in the decreasing order of current CPU utilizations and allocate each VM to a host that provides the least increase of the power consumption caused by the allocation. This allows the leveraging the nodes heterogeneity by choosing the most Power efficient ones first. The pseudo-code for the algorithm is presented in Algorithm. 2. The complexity of the algorithm is n.m, where n is the number of nodes and m is the number of VMs that have to be allocated.

Algorithm 1: Modified Best Fit Decreasing (MBFD)

```
Input: hostList, vmList  Output: allocation of VMs
vmList.sortDecreasingUtilization()
foreach vm in vmList do
    minPower ← MAX
    allocatedHost ← NULL
    foreach host in hostList do
        if host has enough resource for vm then
            power ← estimatePower(host, vm)
            if power < minPower then
                allocatedHost ← host
                minPower ← power
    if allocatedHost ≠ NULL then
        allocate vm to allocatedHost
return allocation
```

4. RESULTS

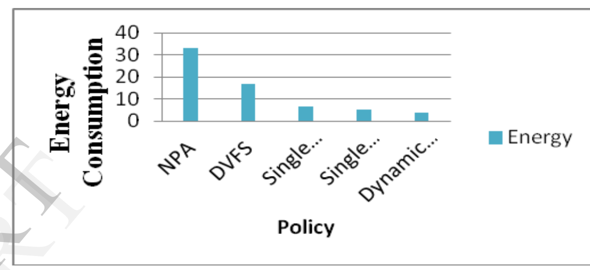
We tested our work on Cloudsim Toolkit [15]. In our experiment, we have worked with just one datacenter. We took up with 10 and 20 VM on those host. Each node comprises of one CPU core with 10 GB ram/network bandwidth and storage space of 1TB. The host comprises of 1000, 2000 and 3000 MIPS accordingly. For each virtual machine on host ram size is 128MB and bandwidth size is 2500 MB with 250, 500, 750 and 1000 MIPS accordingly.

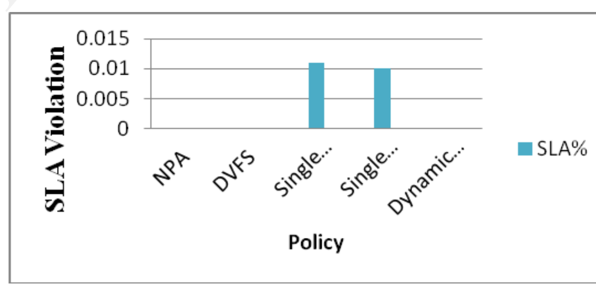
Firstly, we tried to work on Cloudsim Toolkit. Then we went on studying the power examples already implemented i.e. DVFS [14] and NPA [13]. These examples are not following the migration policy. Then we tried implementing single threshold on it. In which, a static assignment of upper limit threshold value is done. Finally, we moved on implementing our concept of dynamic threshold using the threshold theories stated in previous section. We compared the DVFS and NPA algorithms with the Single threshold(ST) algorithm and Dynamic Threshold Algorithm.The results are as shown in below Table 1.

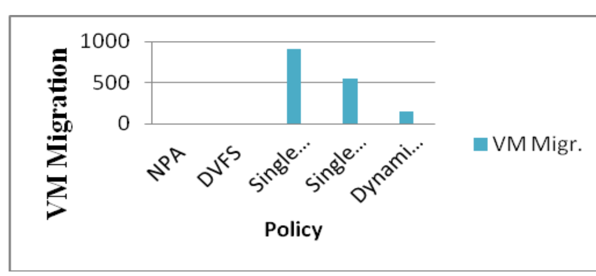
Table 1: The comparison of 4 types of power aware algorithms

Policy	Energy	SLA %	VM Migr.	Avg. SLA, %
NPA	33.01	0	0	0
DVFS	16.74	0	0	0
Single Thr 80%	6.75	0.01	906	9.98
Single Thr 90%	5.1	0.01	556	9.66
Dynamic Thr	4.03	0	155	9.62

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 it entire dependency is limited to voltage and frequency. The single threshold is violating the maximum number of SLA with nominal energy consumption.
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 Following are the results as shown in graphs.







5. 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. For our future work, 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.

6. Reference

[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, C. S. Yeo, and S. Venugopal, “Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities,” in Proceedings of HPCC’08. IEEE CS Press, Los Alamitos, CA, USA, 2008.

[3] R. Brown et al., “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] R. Buyya, A. Beloglazov, J. Abawajy, Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges, in: Proceedings of the 2010 International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA 2010, Las Vegas, USA, 2010.

[11] D. Kusic et al. Power and performance management of virtualized computing environments via lookahead control. Cluster Computing, 12(1):1{15, 2009.

[12] A. Beloglazov and R. Buyya. Energy e_cient allocation of virtual machines in cloud data centers. In Proc. of the 10th IEEE/ACM Intl. Symp. on Cluster, Cloud and Grid Computing (CCGrid 2010), 2010.

[13] M. Yue. A simple proof of the inequality FFD (L)< 11/9 OPT (L)+ 1,for all l for the FFD bin-packing algorithm. Acta Mathematicae Applicatae Sinica, 7(4):321{331, 1991.

[14] Jason Sonnek and Abhishek Chandra Virtual Putty: “Reshaping the Physical Footprint of Virtual Machines” HotCloud ,2009

[15]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