

# A Survey Paper on Dynamic Server Load Balancing with Power Metering for Virtual Machine Live Migration in Cloud

Sowmya Naik P.T  
Research Scholar, Dept. of. CSE  
City Engineer College  
Bangalore, India

Rekha B  
M.Tech 2<sup>nd</sup> Sem, Dept.of. CSE  
City Engineer College  
Bangalore, India

**Abstract**—Virtualization technology has the capacity of run multiple and different operating system on single physical server. Cloud computing which enables the efficient use of computing resources and reduces the operating cost. Each virtual machine run independently and is any one of the servers goes down, all its vm's are automatically restarted or continued on another server, with zero downtime and through live migration, virtual servers can be migrated to another physical server for tasks such as performing maintenance on the physical servers without shutting them down. VM power metering is mainly reducing power consumption of data centers. Deals the issues of VM power metering and server models, sampling with VM power metering methods and accuracy methods. Analyzing their efficiencies, evaluate their performance. To achieve load balancing with virtual machine live migration and monitoring of system resources with system loads across the cluster can be distributed over physical machines An automatic decision making mechanism is to consolidate virtual servers and shut down the idle physical machines during the off-peak hours, while activating more machines at peak times.

**Keywords**— *Virtual machine, cloud computing, power metering, load balancing.*

## I. INTRODUCTION

Cloud computing has obtain computing resources, information services and entertainment services. Data centers are getting more powerful, energy consumption of data centers. Hardware and software solutions such as power capping and power saving scheduling. Virtualized environment consolidating virtual machines and switching off idle servers is a common technique for energy saving. Server virtualization is one physical server into multiple virtual servers. Each of these virtual servers can run its own operating system and applications, and perform as if it is an individual server. Virtualization is the process of creating virtual copies of resources that can be deployed on a physical server.

Power metering of VMs is a difficult task to measure energy consumption accurately for each VM. First, server power metering cannot be directly used for VMs and no device that can measure the power consumption of each VM, power models of servers cannot be directly applied to measure VM power. Second, power consumption of each VM is composed of the amount of each hardware resource power consumed by the VM. The power consumption of hardware resources is changing dynamically with applications and difficulties is to measure the power consumption of hardware resources. Third, VM power can be affected by other VMs on the same host competing resources together. The cloud monitoring systems like HPILO [6] and Green Cloud[5] can only measure the power consumption in the granularity of server and resource. VM power metering has two classes of methods: white box method and black box method.

## II. RELATED WORKS

A. *Libvirt*: This mechanism can be used to report detailed implementation states of each virtual machine, such as CPU time, memory, and network traffic and provide features of virtual machines migration and database information. Libvirt is an open source API for virtual platform management. It can support various hypervisors such as KVM, Xen, OpenVZ, User Mode Linux, QEMU and Virtual Box. C, C++ C#, Perl, Java, OCaml, PHP, Python, and Ruby are all supported by Libvirt, allowing programmers to manage various hypervisors and develop different cloud management systems efficiently.

B. *Live Migration*: Process of transferring a running virtual machine or running application between two different physical machines without disrupting/disconnecting the client machine and running application. Storage, CPU, Network Connectivity and Memory of the virtual machine are transferred from the original host machine to the destination host machine. There are two ways of live migration: precopy migration and postcopy migration.

Precopy is a virtual machine migration technique, all memory pages are migrated in preparation stage; once the writable work sequence (WWS) is small enough or the predefined

iteration threshold has been reached, the target virtual machine will be brought down and moved. All CPU states and memory pages are transferred to the destination machine. Precopy copies memory to the target virtual machine until dirty memory pages fall below a threshold. Then, the CPU states, device states, and remaining memory pages are finally moved and the VM starts to operate. In pre-copy memory migration, hypervisor typically copies all the memory pages from source node to destination node while the VM is still running on the source. If some memory pages change (become 'dirty') during this process, they will be recopied until the rate of recopied pages is not less than page dirtying rate. VM will be stopped on the original host, the remaining dirty pages will be copied to the destination, and the VM will be resumed on the destination host.[3] The time between stopping the VM on the original host and resuming it on destination is called "down-time", and ranges from a few milliseconds to seconds according to the size of memory and applications running on the VM. There are some techniques to reduce live migration down time, such as using probability density function of memory change.

Postcopy firstly copies CPU states and device states and resumes its operation, followed by copying memory. At the beginning, there will be no memory pages in the destination machine and the VM starts to function only when all memory images are completely copied. Postcopy VM migration is initiated by suspending the VM at the source. With the VM suspended, a minimal subset of the execution state of the VM (CPU state, registers and, optionally, non-pageable memory) is transferred to the target. Post-copy sends each page exactly once over the network. In contrast, pre-copy can transfer the same page multiple times if the page is dirtied repeatedly at the source during migration. On the other hand, pre-copy retains an up-to-date state of the VM at the source during migration, whereas with post-copy, the VM's state is distributed over both source and destination. If the destination fails during migration, pre-copy can recover the VM, whereas post-copy cannot.

**C. ZigBee Digital Power Meter.** ZigBee digital power meter to monitor the energy consumption of the hosts. Wireless data transmission, convenience for setting, low energy consumption, longer transmission distance, and data centralization. ZigBee is a developed wireless technology to provide a short distance and low bandwidth solution to data transmission of automatic control. Medium access control layer and physical layer by IEEE 802.15.4 and is capable of handling a wireless transmission network of up to 65,000 devices. Distance between each device ranges from standard 100m to several 1000meters/ kilo meters. ZigBee network contain different types of node in that mainly three types of nodes is there. Coordinator is mainly responsible for network path establishment and network IP address assignment. Then Router is device that responsible for building network and forward data packet within network and assigns IP address to child devices. Router can connect multiple networks with recovering the path of data packets from network. Last, End devices can join an existing network and transform or receive data packets in the network.

### III. ENERGY CONSUMPTION MODEL FOR VM POWER METERING

Total power consumption of a physical server consists of two components  $PM_{Static}$  and  $PM_{VM}$ .  $PM_{Static}$  is the fixed power of a server regardless of running VMs or not,  $PM_{VM}$  is the dynamic power that is consumed by VMs running on it. Suppose there are  $n$  VMs and each virtual machine of them is denoted by  $VM_j$ ,  $1 \leq j \leq n$ ,

$PM_{VM_i}$  denote the energy consumed by  $VM_j$ .

$$PM_{Total} = PM_{Static} + \sum_{j=1}^n PM_{VM_j}$$

$$= PM_{Static} + \sum_{j=1}^n PM_{VM_j}$$

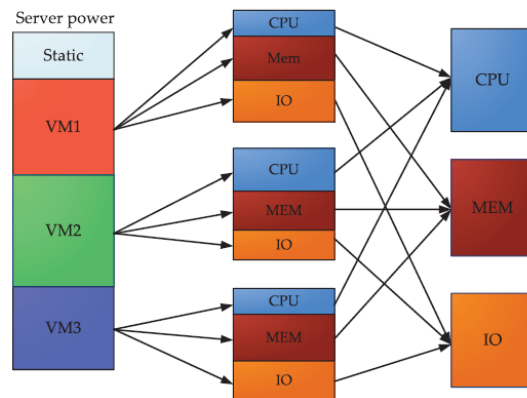


Figure 1: Power Consumption of VMs in the Server.

$PM_{VM_j}$  can be decomposed into the power consumption of components such as CPU, memory and IO, denoted by  $PM^{CPU}_{VM_j}$ ,  $PM^{Mem}_{VM_j}$ ,  $PM^{IO}_{VM_j}$  respectively.  $PM^{IO}_{VM_j}$  includes general energy cost of all devices that involve IO operations such as disk and network data transfer.

Power consumption of  $VM_i$  is:

$$PM_{VM_i} = PM^{CPU}_{VM_j} + PM^{Mem}_{VM_j} + PM^{IO}_{VM}$$

### III. MODELING INFORMATION COLLECTIONS

Virtual machine power metering usually has three steps: information collection, modeling and estimation. Since VM power is closely related to the hardware resources, PMCs and the power consumption of the server. Power measuring for servers, approaches for modeling information collection and sampling rate in the following:

#### A. POWER MEASURING FOR SERVERS

The measuring of physical server power is the basis for VM power metering. There are two types of methods to measure physical server power: External attached power meter and

Internal meter like power sensors and motherboard. The external power meter like PDU (Power Distributed Unit) has the merit of flexibility. PDU can be easily attached or detached from the machine without affecting normal operations of the system. Watts UP PRO logs power information inside the flash of PDU and the loggings can be downloaded into the computer when needed externally attached power meter can save a lot of investment in updating current infrastructure for data centers. But it is almost infeasible to apply external PDU on a large scale. Internal power meter such as power sensors and motherboard inside the server has the merit of management. Power information can be accessed through distributed programming interfaces or command lines even GUI. Dell Power Series server is just the type of server with power sensors inside. It can provide comprehensive power information about the CPU, memory, network, motherboard, and fans through Dell Open Manage suite [8].

## B. MODELING INFORMATION COLLECTION

Since there is no such device that can directly measure VM power, information such as resources or events of the system can be converted into power consumption by modeling methods. There is a need to track that information in the granularity of each VM and collecting modeling information for VM power metering.

### 1) COLLECTING HARDWARE RESOURCE POWER

VM power is usually composed of the portions of hardware resource power that is consumed by the VM. Power consumption of each hardware resource, some works in [9]-[12] use wires to connect self-developed power sensors or registers directly to the motherboards. But those methods are infeasible for large scale data centers due to the complexity and unavailability of the knowledge about architectures of modern servers [14]. Dell Power Series servers provide power information in the granularity of the hardware resource. For servers without power sensors inside, the modeling method could be used to estimate the power consumption of each hardware resource in the server.

## 2 COLLECTING RESOURCES INFORMATION

### a)CPU INFORMATION COLLECTION

Power consumption of CPU is affected including cache usage, specific instructions and frequencies. To estimate the power consumption of CPU of the server and VMs on it, the modeling method can be used by correlating CPU related information to power. The information can be categorized into three types: CPU utilization, PMCs and time slices of processors.

Methods for CPU information collection are as follows:

- Calculate CPU utilization using active time divided by the total time of the processor for a certain period. To account CPU utilization by each VM, the usages of virtual processors is calculated first

by tracking CPU performance counters in Hyper-V, and then transforming the usage of the virtual processor into the utilization of the physical processor. It should be noted that Xen trace can be used to track CPU utilization for each VM on Xen hypervisor.

- Collects PMCs of CPU by instrumenting a program in the hypervisor. The program collects PMCs in hypervisor and writes the information into a sharing buffer, which is then mapped into the address space of a monitor program at user-level. Thus, the information of PMCs can be obtained through the monitor program. To account the PMCs for each VM, it can be calculated through deviation of PMCs between the two consequent scheduling switches of the processor.
- Time slices of processors to account the portion of CPU usage by each VM. When vCPU scheduling happens, the time slices is allocated to a VM. Thus, the time slices can be captured from a data structure that can map physical CPU ID to vCPU ID (VM). The amount of CPU resource used by each VM can be inferred from time slices of the processor used by the VM.

### b)MEMORY INFORMATION COLLECTION

The reading and writing throughput of memory has significant effect on memory power and developed an external instrument that can accurately capture the throughput of memory. Kansal [15] and Krishnan [19] proposed a light-weighted method using LLC (Last Level Cache) misses to indirectly estimate the power consumption of memory. Krishnan believes LLC misses can reflect the utilization of memory at different levels. Many processors expose the information as performance counters so that LLC misses can be tracked using tools like Oprofile. LLC misses of each VM can also be obtained in the same way and estimates the power consumption of memory by modeling the number of memory accesses of the server and each VM on it.

### c) IO INFORMATION COLLECTION

The power consumption of IO (Input and Output) is usually consumed by devices such as disk and network. Most work in the literature take only disk as the power consuming component for IO and the network power is so low that it can be ignored. Disk information collection methods are as follows:

- Estimates the power consumption of disk by throughput of disk that can be obtained from hypervisor. Each VM it can be obtained through performance counters in the Hyper-V virtual storage device and the Hyper-V virtual IDE controller. The power contribution from varying spinning speeds of the disks is usually not considered, because it is rarely used in data centers.
- Time to finish an IO request is closely related to disk power. It can be calculated through the size of a

request divided by the transfer rate. Note though that the transfer rate is dynamically calculated using the byte size of 50 requests divided by the transfer interval.

- Method to get disk and network throughput in the granularity of VM on the Xen hypervisor. The IO requests of disk are mostly initiated by VM, so the disk throughput information for each VM can be easily obtained from the requests captured at the hypervisor level same way for writing throughput of network. Reading throughput of network cannot be captured from hypervisor are initiated by senders. We can obtain through put of both disk and network for each VM on Xen hypervisor.

### 3) TOOLS FOR INFORMATION COLLECTION

Modeling information obtained through tools provided by Linux Oses or developed by other organizations could be used. System events and register values can be collected through PMC tools like Oprofile [21], Pfmmon [22] and Perf-suite [23]. All of those tools can profile the system in the granularity of process. Cluster monitoring tool for large data centers, to collect modeling information. Monitoring tools provided by Linux such as iostat and sar in toolkit sysstat can be used for modeling information collection. Some other event information can be obtained by reading Linux files like /proc and /sys. In addition, specialized tools are developed to collect modeling information on Xen [26] KVM [27] and VMware. For example, Xentop [28], XenMon [29] and XenoProf [30] are designed to collect information of host and that of each VM in Xen platform, an enhanced perf [31], [32] is for KVM, and ReTrace [33] for VMware. With those tools, collecting modeling information for VM power metering is no longer a challenge for virtualized platform.

### C. SAMPLING RATE

Sampling is used for modeling information collection and it will affect the normal running of the server and the accuracy will be lowered otherwise. Goal of sampling rate setting is to keep high accuracy with low overhead. Most researchers empirically set sampling rate to 1 or 2 seconds. McCullough in [34] holds that 1 second is the right choice and the accuracy cannot be enhanced obviously by adjusting the interval. While in [9], 2 seconds sampling is proved to be suitable through experiments on power overheads of different sampling intervals. In addition, Chen Q n [24] proposes that sampling interval can be adjusted to be 5 seconds for stable applications, and 1 second for applications with dynamic workload and power. In summary, it is reasonable to set and adjust sampling rate flexibly according to real applications.

## IV. METHODS FOR VM POWER METERING

Based on the sampling information, we will discuss detailed methods for VM power metering including basic procedures, white box and black box methods, and finally mathematic methods in modeling. Virtual machine power metering usually includes three steps as follows:

- **Collection of information:** Collecting necessary information for modeling such as the power consumption of server, resources utilization and PMCs that can be obtained by approaches mentioned above for the server and each VM.
- **Modeling:** build a proper mathematic model using the most power-related resources or PMCs as variables, and then train the model using samples collected to generate a set of parameters.
- **Estimation:** calculate the power consumption of each VM by taking the information of each VM into the model with latest parameters.

VM power metering can be classified into two categories: white-box method and black-box method. So the following will discuss the two methods in detail:

### A. WHITE BOX METHOD

White box method is also called pitching in / proxy method in VM power metering. A running proxy program is inserted into each VM to collect resources utilization and CPU usage or PMC events of the VM for power modeling.

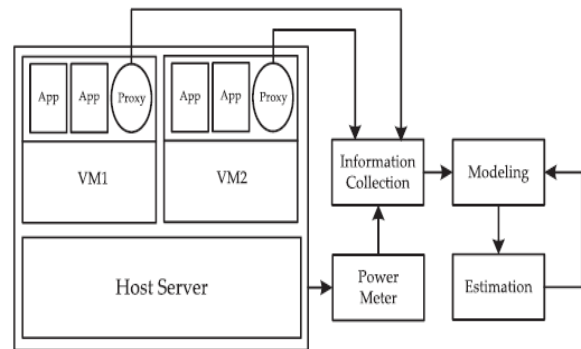


Figure 2: Architecture of white box method.

In the architecture, there are several VMs running on the host machine with each VM executing several existing applications inside. The proxy program in each running virtual machine is responsible for collecting resource utilization and CPU usage of the virtual machine. Collection module gathers information related to virtual machine from the proxy of each VM and power meter. Modeling module will collect data/ information and produces a set of model parameters then estimate the power consumption of each virtual machine. Estimation module will have some feedback to the modeling module when errors exceed a certain threshold, then modeling module retrain the samples to update parameters of the model and white box method is to measure the power consumption of each virtual machine. Modeling information is collected by a running proxy inside each virtual machine and the server power is collected through an external attached PDU. Resource usage information collected inside each virtual machine cannot objectively reflect the usage of hardware resources by the virtual machine. Black box method that collects events for each VM at host level

B. BLACK BOX METHOD

The black box method collects information of each VM at host level. The architecture of black box method is referring same method to that of white box method, as is shown in Figure 3. Difference lies in that VM proling information such as PMCs are collected outside virtual machine at hypervisor level. A typical example of this architecture is Xen virtualization platform and we can use Xenoprofile as tool to collect events of each VM on it. The black box method for VM power metering is based on the power model of the physical server. The power consumption of each VM is calculated by taking resources used or PMCs of each VM into the server power model. The key is to distinguish and account for the resources or PMCs for each VM. The accuracy of the power model for physical server will directly affect the result of VM power metering. PMCs are the counters that record the accumulation values of the registers or events of the system. PMCs are used to power consumption of system, virtual machine and applications.

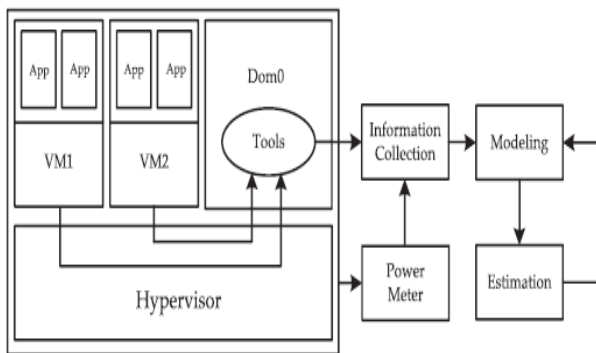


Figure 3. Architecture of Black Box Method

Two categories of VM power metering methods using PMCs: component-based models with PMCs representing each component and pure PMC based models.

1) COMPONENT BASED VM POWER ESTIMATION

The power consumption of a physical server consists of static power and dynamic power as is mentioned in section 3. The dynamic power is composed of the power consumption by CPU, memory, IO and so on, denoted as PCPU, PMem, PIO, etc. respectively. Thus, the total power can be expressed:

$$PM_{Total} = PM_{Static} + PM_{CPU} + PM_{Mem} + P_{IO} + \dots$$

2) PURE PMC BASED VM POWER ESTIMATION

Pure PMC based models use only PMCs for modeling, avoiding the inaccuracy incurred by empirical PMCs selection for the components. The PCA is excellent in reducing the dimension of raw data and selecting the most influenced independent components

V. CHALLENGES FOR VM POWER METERING

Many techniques have been developed in measuring the power consumption of server such as PDU (power distribution unit), an external power meter for older servers

and inner power sensors for the motherboard of newer servers. However, there is no such devices that can directly measure the power consumption of VMs. Therefore, software methods using the mathematical models have been proposed: the resource based models and the PMC based models. There are some challenges in the implementation of the two types of models:

- Distinguishing the activities of each VM in resource utilization or PMC changes, so that we can quantify the contribution of each VM to the power of each resource.
- Determining what resources or PMCs should be considered for the measuring of VM power.
- Deciding the type of mathematic models to be used for PMC based model.

Two challenges like to get more accurate power consumption for the VMs. To the best of our knowledge, machine learning methods like regression tree could be used to replace the linear models, because it will automatically divide the values of the resource in different dimensions into segments, and then train each generated segment leaf using linear model. Therefore, VM power metering using machine learning models is the future research direction. It could be used to automatically select key features (here refer to the resource utilization or changing values of the events). Features with similar characters could be clustered, and a separate model could be trained using data from the cluster.

VI. POWER SAVING THROUGH VM MIGRATION

A. Load Balancing across physical machines. Initially, four virtual machines (VMs) are deployed in each physical machine and each VM is equipped with 2CPU and 1GB of memory. When the system is idle, all virtual machines on the physical machine, node 100, will be migrated to node 200. We observed that the migration and power-off procedure took around 86 seconds.

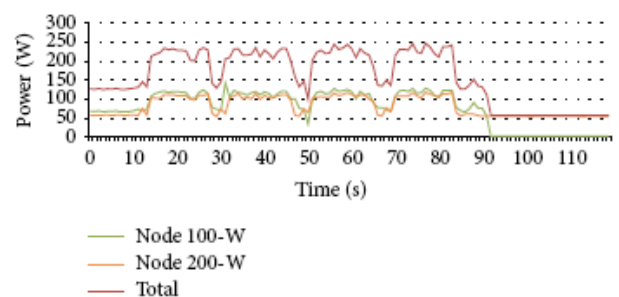


Figure 4: Energy consumption chart during migration

Figure 4: Shows that distribution of power consumption during the migration period, where the “total” stands for the total energy consumption (in watt) of two physical machines. At the beginning, total is 127, occasionally reaches up to 228W while the migration is processing. Once all VMs were successfully migrated to node 200, node 100 was turned off. The experimental results show that only

58W of energy is required to run eight active virtual machines on node 200, which is 69W less as compared with running over both physical machines. Network traffic used during the migration process is shown in Figure 5 and the perspective CPU loading of each VM is depicted in Figure 6.

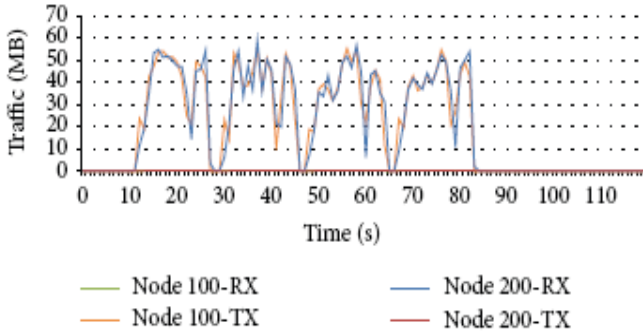


Figure 5: Internet traffic chart throughout network during migration.

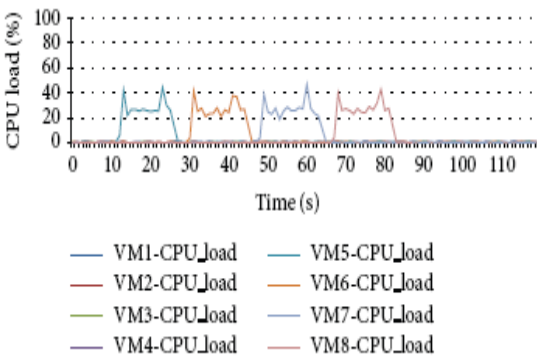


Figure 6: CPU Load in virtual machines across the network during migration.

B. Decision Results of Load Balance in Physical Machines. Effectiveness of load balancing to generate the CPU loading of each virtual machine. Among 8 VMs, which are evenly distributed over 2 physical servers, CPU loadings are particularly assigned as 80%, 40%, and 6 VMs with 20% each. Both VMs with 80% and 40% CPU utilization reside at the same physical machine, node 100, to experience the unbalanced loading. Each VM with 40% CPU utilization was chosen as a target for the purpose of migration. Through the operations of the migration module, this targeted VM is migrated to another physical machine, node 200. Effectiveness of load balancing is shown in both Figures 7 and 8. Figure 7 indicates that the migration process takes around 11 seconds, from 33 seconds to 44 seconds. When the migration is in progress, there are about 10% of increases in total power usage. Once the migration is accomplished, both CPU utilization and power consumption of two physical machines tend to be equivalent.

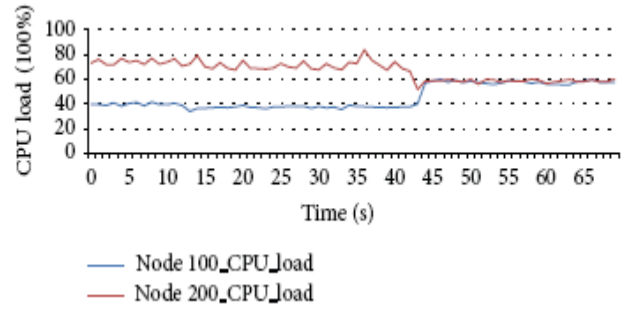


Figure 7: Energy consumption chart within network.

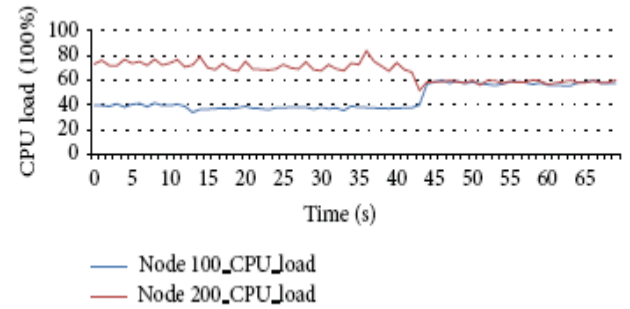


Figure 8: CPU Load in physical machines in same network.

## VII. CONCLUSION

Achieve load balancing through migration in clouds and resources can be efficiently utilized to meet the quality of service request. In the experiment, when the CPU load of one physical machine is approaching 100%, When the system is not busy, the consolidation process of the such mechanisms accomplished by activating the virtual machine live migration automatically. This results in powering off the idle physical machine and consequently, reducing the energy consumption. Experiment of load balancing, CPU loads of two physical servers are 80% and 40%, respectively. The proposed mechanism enables the assignment of new coming request, so that the CPU utilizations of both the machines are promptly balanced with around 60% each. Virtualization frameworks are flexible and can be easily extended. Huge cloud involves many numbers of server and virtual machines, an efficient VM migration with multiple machines. Estimating VM power at the software level tools for information collection, modeling methods, and estimation.

Black box method using PMC information for modeling for system power without violating the integrity of each VM. Learning methods could be used to enhance the accuracy of current method in VM power metering. VM power metering has power budgeting, fair billing and power-aware scheduling for the future green cloud data centers. Cluster dynamic load balancing method in virtual environment to balance the load across network virtualization in data center with flexible and cost effective

## REFERENCES

- [1] P. Kurp, "Green computing," *Commun. ACM*, vol. 51, no. 10, pp. 11–13, 2008.
- [2] P. Ranganathan, "Recipe for efficiency: Principles of power-aware computing," *Commun. ACM*, vol. 53, no. 4, pp. 60–67, 2010.
- [3] L. Liu et al., "GreenCloud: A new architecture for green data center," in *Proc. 6th Int. Conf. Ind. Session Auton. Comput. Commun. Ind. Session*, 2009, pp. 29–38.
- [4] HPiLO[Online]. Available: [http://h17007.www1.hp.com/us/en/enterprise/servers/management/ilo/#.U4Wk4uM\\_jKc](http://h17007.www1.hp.com/us/en/enterprise/servers/management/ilo/#.U4Wk4uM_jKc), accessed May 29, 2014.
- [5] WattsUp Meter. [Online]. Available: <https://www.wattsupmeters.com/secure/index.php>, accessed May 29, 2014.
- [6] Public APIs of Schleifenbauer PDU. [Online]. Available: <http://sdc.sourceforge.net/index.htm>, accessed May 29, 2014.
- [7] H. Yang, Q. Zhao, Z. Luan, and D. Qian, "iMeter: An integrated VM power model based on performance profiling," *Future Generat. Comput. Syst.*, vol. 36, pp. 267–286, Jul. 2014.
- [8] J. Jenne, V. Nijhawan, R. Hormuth. (2009). Dell Energy Smart Architecture (Desa) for 11g Rack and Tower Servers.
- [9] W. Dargie and A. Schill, "Analysis of the power and hardware resource consumption of servers under different load balancing policies," in *Proc. IEEE 5th Int. Conf. Cloud Comput. (CLOUD)*, Jun. 2012, pp. 772–778.
- [10] E. M. Elnozahy, M. Kistler, and R. Rajamony, "Energy-efficient server clusters," in *Proc. 2nd Int. Conf. Power-Aware Comput. Syst.*, [Online]. Available: <http://www.dell.com2003>, pp. 179–197.
- [11] D. Economou, S. Rivoire, C. Kozyrakis, and P. Ranganathan, "Full-system power analysis and modeling for server environments," in *Proc. IEEE Int. Symp. Comput. Archit.*, 2006, pp. 1–4.
- [12] C. Mobius, W. Dargie, and A. Schill, "Power consumption estimation models for processors, virtual machines, and servers," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 6, Jun. 2014.
- [13] A. Kansal, F. Zhao, J. Liu, N. Kothari, and A. A. Bhattacharya, "Virtual machine power metering and provisioning," in *Proc. 1st ACM Symp. Cloud Comput.*, 2010, pp. 39–50.
- [14] "CloudComputing—Wikipedia," [http://en.wikipedia.org/wiki/Cloud\\_computing#Private\\_cloud](http://en.wikipedia.org/wiki/Cloud_computing#Private_cloud).
- [15] F.H.MbubaandW.Y.C.Wang, "Softwareasa service adoption: impact on IT workers and functions of IT department," *Journal of Internet Technology*, vol. 15, no. 1, pp. 103–114, 2014.
- [16] M.-Y.Luo, "Designandimplementation of aneducation cloud," *Journal of Internet Technology*, vol. 15, no. 2, pp. 229–240, 2014.
- [17] "Virtual machine—Wikipedia," [http://en.wikipedia.org/wiki/Virtual\\_machine](http://en.wikipedia.org/wiki/Virtual_machine).
- [18] M. Mishra, A. Das, P. Kulkarni, and A. Sahoo, "Dynamic resource management using virtual machine migrations," *IEEE Communications Magazine*, vol. 50, no. 9, pp. 34–40, 2012.
- [19] L. Liu, R. Chu, Y. Zhu, P. Zhang, and L. Wang, "DMSS: a dynamic memory scheduling system in server consolidation environments," in *Proceedings of the 15th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops (ISORCW '12)*, pp. 70–75, April 2012.
- [20] A. Beloglazov and R. Buyya, "Energy efficient resource management in virtualized cloud data centers," in *Proceedings of the 10thIEEE/ACMInternational SymposiumonCluster,Cloud,and Grid Computing (CCGrid '10)*, pp. 826–831, May 2010.