# Multi-Class Load Balancing Scheme for QoS and Energy Conservation in Cloud Computing

Olasupo O. AJAYI[1], Florence A. OLADEJI[2], and Charles O. UWADIA[3]
[1,2,3] University of Lagos, Nigeria
[1]olaajayi@unilag.edu.ng, [2]foladeji@unilag.edu.ng, [3]couwadia@unilag.edu.ng

**Abstract**

*he challenges of adhering to stringent Quality of Service requirements, efficiently utilize resources, and conserve energy consumption are constantly being faced by Cloud Service Providers. In a bid to proffer solutions to these challenges, numerous researchers have proposed varied solutions. However, there has yet to be an all-encompassing solution that tackles all these challenges at once, as these challenges are often times contrasting. Authors therefore usually focus on one then seek to manage the compromises on the other(s). In this work, we propose a new scheme for load balancing that uses multiple workload classes to guarantee end-to-end QoS while conserving energy with little compromise on either. Experiments were done using CloudSim toolkit and obtained results show that our scheme outperforms the other approaches both in terms of energy conservation and QoS adherence.*
Keywords: Load balancing, QoS cloud comutng, energy conservation

---

## Introduction

The relative decrease in cost of Internet access and the proliferation of smart devices has led to an increase in workloads at Cloud data centers. These increased workloads with varied requirements and a less than equal increase in resource levels have led to the need to efficiently utilize Cloud resources in order to effectively service these workloads and at the same time make money for the Cloud provider. There is also the dire need to conform to standards for green computing by reducing overall energy consumption and carbon emission levels.

One approach to energy conservation is server consolidation and multi-tenancy [1], [2]; which through the use of virtualization and virtual machines [3] seek to aggregates workloads on Physical Machines (PMs) together in a bid to reduce the total number of active PMs. Doing this however could have negative effects on user workloads as the illusion of dedicated PMs which Virtual Machines (VMs) provide to users is in not perfect and shared resources can sometimes be fiercely contested for by these workloads [2]. Cloud providers are then faced with the issue of contending with energy conservation versus guaranteeing QoS adherence.

In this work, an approach that uses multiple workload classes to guarantee an end-to-end QoS adherence while at the same time conserving energy is proposed.

The rest of this paper is organized as follows: section 2 discusses on related works, in section 3 the proposed Multi-Class load balancing approach is presented, experimental results are presented in section 4 and the paper is concluded in section 5.

## Related Works

### *Multi-Queue Workload Classification*

Classification of user workloads has been done by numerous authors some of which include: [4] where user workloads were split into two groups – Gold and Bronze based on user required response times. In the works of [5], [6] user workloads were grouped into three groups – Short, Medium and Long based on the user indicated burst time of each tasks. In works done by [7], [8] the authors used multiple user supplied criteria for classification of workloads. Though their works focused on workload preemption, they had to classify these workloads in order to determine priority of preemption. Reference [9] proposed a resource based classification of PMs using RAM, CPU and Bandwidth, in which user workloads were allocated to the PM that offered minimum completion time for such tasks. In the works of [10], [11] multiple SLA parameters (such as product type, account type, request type, response time etc.) were considered but

ultimately workloads were classified into three groups – Small, Medium and Max or Gold, Silver and Bronze respectively.

From literature it can be concluded that classification of user workloads is not a trivial tasks, as it is almost impossible to consider every requirement/criterion during these classifications, and because workload classification is outside the scope of this work, we simply adopted the state of the art approach used in [5], [6].

***Energy Aware Load Balancing of Workloads***

Reference [12] proposed an energy aware approach to tasks allocation and load balancing in Cloud Data Centers (DC). The focus of this work is on conservation of energy while minimizing SLA violations. Workloads on admission were allocated to PMs using a modified best fit descending algorithm called Power-Aware Best Fit Descending (PABFD), which performed a power-growth test prior to workload allocation and only allocates after confirming that such allocation would not make the power consumption of the PM greater than a preset threshold value. With respect to load balancing, the approach compares PMs' CPU utilization level against pre-set upper and lower threshold values to detect an over/under worked. If a PM's CPU utilization grows above the upper threshold, VMs are migrated off the PM similarly, if the CPU utilization is below the lower threshold, all VMs are migrated off and PM put to sleep to conserve energy.

Reference [14] improved on the work of [12], [13] by proposing a prediction based approach to resource management in Cloud computing called VMCUP. Rather than checking for CPU utilizations after allocation, this work predicts the short-term future state of the PM and determines if such a PM will be over/under utilized. This is a preventive approach which contrasts to the corrective approach used in [12].

Mosa and Paton [15] proposed a utility function based VM allocation approach to energy conservation, SLA adherence and profit maximization. The work identifies optimal allocation of VMs to PMs as a NP-hard problem and thus used a meta-heuristic genetic algorithm to achieve this goal in the most rewarding (profit) way. The authors employed a utility factor which was based on expected income less estimated energy, violation and performance degradation costs. The approach recorded improvements in terms of QoS adherence and energy conservation.

Notable shortcomings of some of these energy-aware approaches to load balancing are: heavy reliance on the use of agents to get status information of resources prior to and during the allocation and load balancing phases, which invariably leads to increased response time. With the exception of [15] scalability might be a challenge as the schedulers in these other approaches has to keep an active communication channel with all the PMs and VMs; this is impracticable especially in Cloud data centers with large number of PMs. The PM's CPU utilization level was the only metric used to measure QoS adherence; other factors such as class of payment, required response time and burst time could have been considered. The "power-growth" tests performed during allocation of VMs to PMs might seem effective but [16] shows that an idle PM consumes about 70% of its maximum usable power, hence there can only be an energy saving of 30% per PM at best when used. To this end, we propose an approach that uses workload classes for QoS and energy conservation in Cloud Computing.

**Proposed Approach**

We propose a hybrid scheme with feature sets from [12-14,17], called Multi-Class Load Balancer (MC-BAL). The proposed approach incorporates significant enhancements that address the short-comings of these approaches while leveraging on their individual strengths. It is a two-phased approach with phases described below.

In the first phase, user requests (VMs) are allocated to PMs using our Binary Search Best Fit Algorithm. The proposed algorithm is similar to [12] but uses the Binary Search Tree (BST) to speed up the search for a suitable PM. It has been proven that BST has an average, best and worst case running complexity of $O(\log_2 n)$ which for large entries, is much faster than the average and worst cases of the linear array search O(n) used in both [13],[14].

We introduce multiple workload classes to the allocation phase. There are three different classes of user workloads – Gold, Silver and Bronze and grouped based on their QoS requirements, with Gold being premium and bronze being best effort.

Like in the work of [14], the usage prediction model is used in the allocation of VMs to PMs however, the power growth check is removed. It is expected that the process of VM consolidation carried out in the load balancing phase would cater for energy efficiency as the higher the number of PMs actively running the higher the total energy consumption of the entire data center and vice versa. This is an analogy drawn from the works of [18],[19].We also introduce a Binary Search Best Fit allocation (BSBF), which is used in place of the PABFD. PABFD, searches linearly through all the PMs in the data center for the most suitable to host a VM. Our justification for this is, given a data center with N number of PMs, PABFD has to do N comparison at the worst case before allocating a PM to a user workload. If N is large, this process can slow down the allocation process and lead to an increase in delay time (SLA violation). This is where BSBF has an advantage. Being based on a binary search tree, it has an average and worse case search complexity of $\log_2 N$ thus able to find suitable PMs much faster than linear search based best fit descending used in [12] and [14].

In the load balancing phase the VMs allocation carried out in the allocation phase is improved on with a view of uniformly re-distributing allocated workloads amongst PMs. This would improve QoS adherence, as well as consolidate VMs onto fewer PMs to reduce overall energy consumption of the data center. This phase is split into two parts – utilization detection and VM-Migration. The utilization detection process is the same is in [14]. However, in choosing which VM to migrate, the class to which it belongs is considered.

This implies that all bronze class VMs if present in a PM would be selected for migration first before any silver class. Likewise all silver classed VMs would be selected before any gold classed VM is selected. This would ensure lower SLA violation for the gold class as a result indiscriminate VM migration. In the case of under-utilized PMs, all VMs are selected for migration irrespective of the class they belong to after which the underutilized PMs are put in sleep to conserve energy.

Algorithm 1: The Binary Search Best Fit Algorithm

1. Get total number of PMs in system
2. Arrange all PMs in ascending order of their available processing capacity (LcP)
3. Accept VMs to be allocated (VM_Set)
4. Foreach vm in VM_Set
    a. Get vm's requirement (wR)
    b. Build BinaryTree (BT_LcP) from LcP
    c. SuitablePM = MBS_Method(wR, BT_LcP)
5. MBS_Method(wR, BT_LcP) //recursive search

a. Search BT for a PM p, such that p.AvailableMIPS - wR → 0 //search for the most suitable PM
b. If found, return p
c. Else

i. Remove p from BT_LcP and update BT_LcP
ii. Return MBS_Method(wR, BT_LcP)

The MC-BAL builds on the works of [12] and [13] but with modifications leading to the following contributions:

1. QoS adherence and energy conservation through the use of workload classes. Though numerous works have used multiple workload classes, such as those of [4-11], most have focused on billing and/or QoS only. We do not know of any work where workload classes has been used for QoS and energy conservation.
2. The use of workload classes in VM migration, thus guaranteeing end-to-end QoS compliance
3. Binary Search Best Fit heuristic for the allocation of VMs to PMs, which speeds up allocations.

**Performance Evaluation**

***Experimental Setup***

To verify the efficiency of our proposed model, simulations were carried out using CloudSim toolkit [20] and the same experimental setup used in [12] and [14] was used for comparison purpose. The data center consisted of 800 heterogeneous PMs of two categories and with specifications and power consumption models based on benchmarked data from real servers [21]. These are depicted in table 1.

TABLE 1
SPECIFICATIONS OF THE PMS USED FOR SIMULATION

| Category | Make | CPU | Cores | Memory |
|---|---|---|---|---|
| 1 | HP ProLiant ML110 G4 | 1,860 MHz | Intel Xeon 3040, 2 cores | 4GB |
| 2 | HP ProLiant ML110 G5 | 2,600 MHz | Intel Xeon 3075, 2 cores | 4GB |

Data used for this experiment are from workload traces of over 5,000 PlanetLab VMs [22], measured at preset intervals of five minutes over a five day period and Google Test Cluster (GTC) [23] consisting of about 168 jobs recorded over a 7 day period.

***Evaluation Metrics***

The following metrics were used in order to maintain consistency and for comparison purpose with [14], they are: Energy consumption; Average number of power state changes per PM; Average SLA violation and Average job delay. In our experiments only the static threshold based overutilization host detection approach of CloudSim was considered. Also only the performance of workloads classified as Gold was of significant interest to us hence comparisons are based on this workload class only.

***Experimental Results***

In order to determine the utilization level of a CPU, static thresholds of 80% and 25% were set for both the upper and lower limits respectively. Above the upper threshold, the PM is classified as overworked and workloads are selected for migration from it, while below the lower threshold the PM is classified as underutilized and all workloads migrated from it.

The performance of MC-BAL was compared with the PABFD [12] and VMCUP [14] using the static threshold for 1,078 VMs logs

.

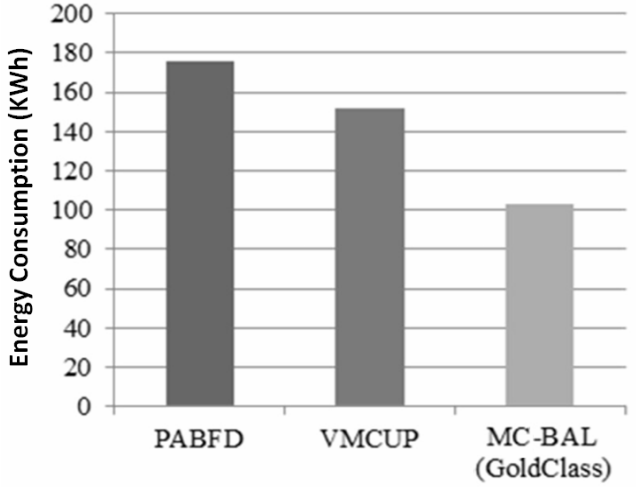


Fig. 1. Comparison of Total Energy Consumption - PlanetLab dataset

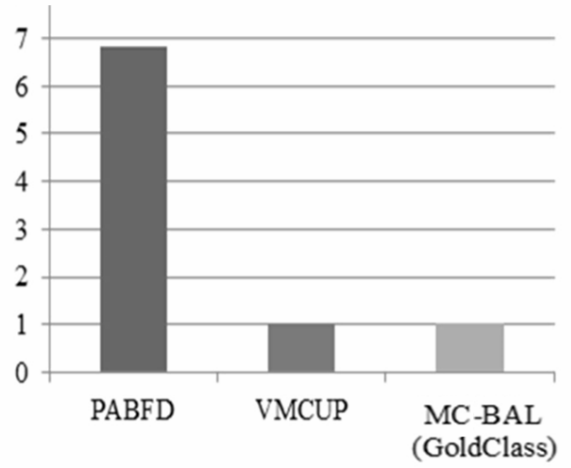


Fig. 2. Average number of power state changes per PM –PlanetLab dataset

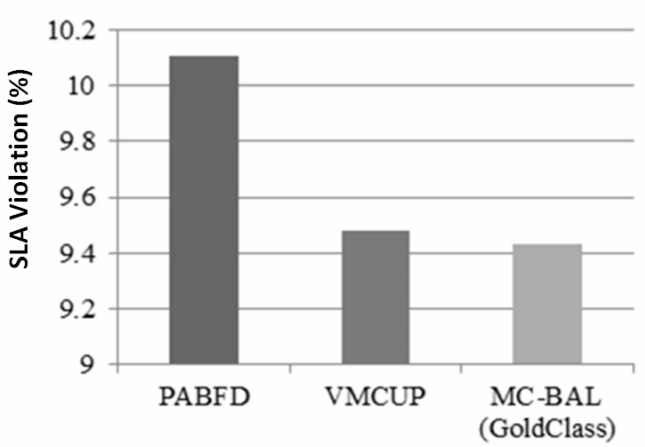


Fig. 3. Comparison of SLA Violation -PlanetLab dataset

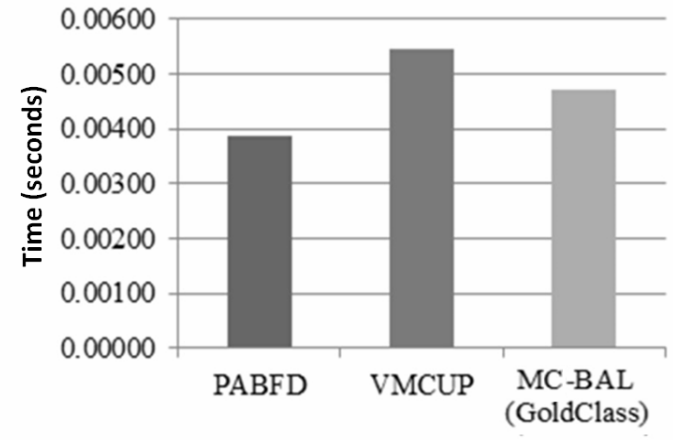


Fig. 4. Workload allocation delays – PlanetLab dataset

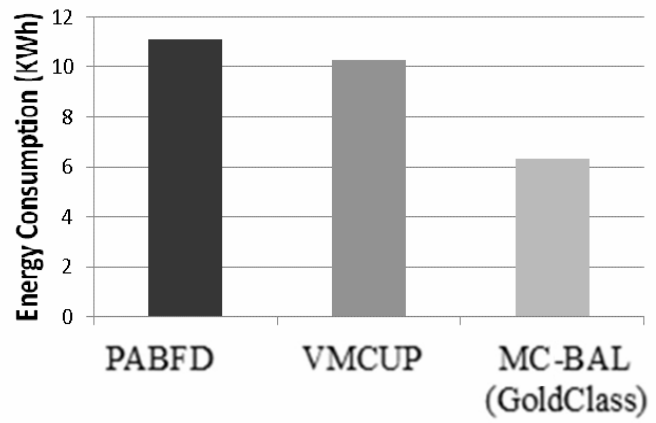


Fig. 5. Comparison of Total Energy Consumption – GTC dataset

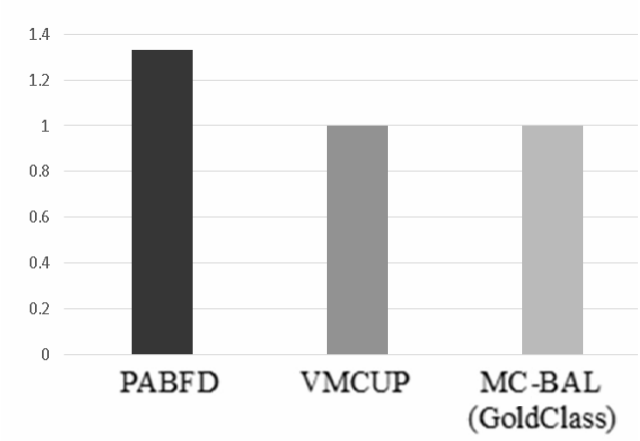


Fig. 6. Average number of power state changes per PM –GTC dataset

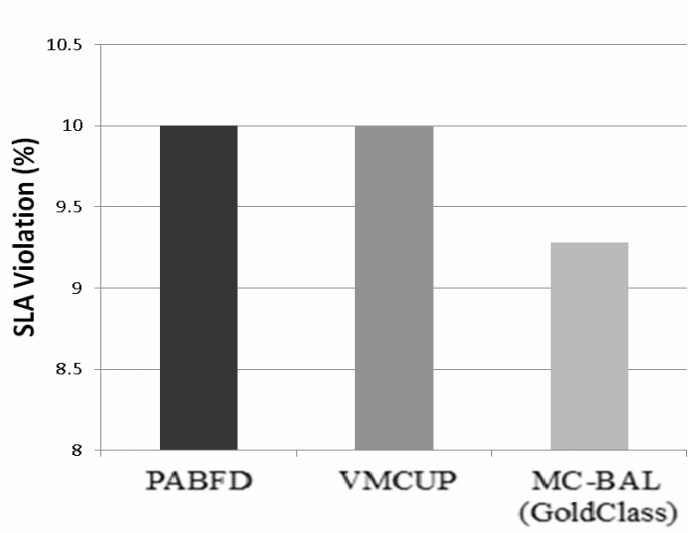


Fig. 7. Compariosn of SLA violations –GTC dataset

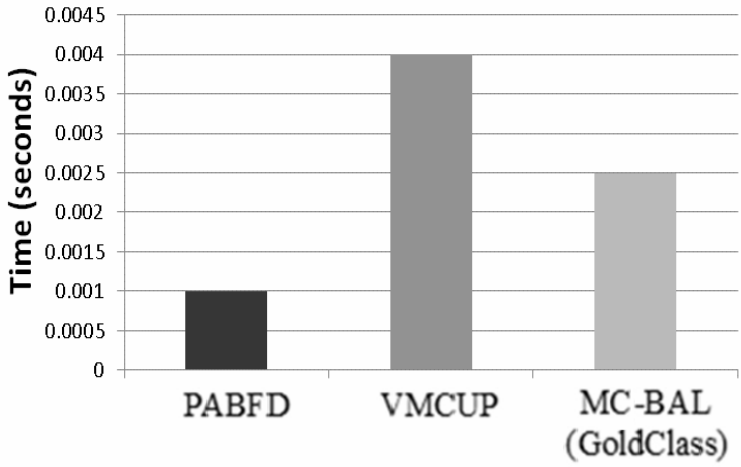


Fig. 8. Workload allocation delays – GTC dataset

Obtained results show a similar trend across both datasets. Fig. 1 shows that MC-BAL clearly outperforms both VMCUP and PABFD with a total energy value of 102.83KWh as against by 175.43KWh for PABFD and 151.42KWh for VMCUP. This implies that MQ-BAL is 70.6% and 47.3% more energy efficient than PABFD and VMCUP respectively

for PlanetLab dataset. The same trend is observed with the GTC in Fig. 5, where MC-BAL with a total energy consumption of 6.33KWh conserves energy better than PABFD (11.1KWh) and VMCUP (10.28KWh), representing a 42.9% and 38.2% improvement over PABFD and VMCUP respectively.

In terms of the average number of power state changes, Fig. 2 depicts that MC-BAL (1.02) slightly outperforms VMCUP (1.04) by 1.92% and PABFD (6.82) by about 85%. Consistent with PlanetLab results, MC_BAL also outperforms the other approaches using GTC dataset as depicted in Fig. 6. This implies that MC-BAL is able to better limit the frequency at which PMs are switched off and on.

Compliance to SLA requirements is depicted in Fig. 3 and Fig. 7 for both datasets. For PlanetLab dataset (Fig. 3), MC-BAL results in the least SLA violation with 9.43%. It edges out VMCUP (9.48%) by about 1.48% and clearly outperforms PABFD (10.11%). For GTC dataset (Fig. 7); MC-BAL also outperforms the other approaches with an average SLA violation of 9.38% as against 10% obtained for both PABFD and VMCUP; this represents an 8% improvement in SLA compliance. MC-BAL is thus able to guarantee end-to-end QoS adherence while providing services to user workloads.

Finally we introduced a last metric, which is job delay. This is the amount of time a VM spends waiting to be allocated to a PM. Fig. 4 shows that PABFD has the least delay at about 0.0039 second and VMCUP has the longest delay at 0.0055 seconds. Since both approaches apply the same linear search based Best Fit Descending (BFD) allocation algorithm, it implies that the utilization prediction algorithm used in VMCUP greatly slows it down. MC-BAL also uses the same utilization prediction algorithm used in VMCUP but the application of BSBF during the allocation phase accounts for the improvement in delay (0.0047 seconds) experienced by MC-BAL. A similar trend is also observed with the GTC dataset and depicted in Fig. 8.

**Conclusion**

Numerous research works have been done in resource management in Cloud computing, however most of them have focused on tackling a single challenge at a time or considering one as the primary challenge and others as secondary. In this work, an approach to load balancing is proposed that leveraging on the strengths of previous works while at the same time addressing most of their shortcomings is proposed. The proposed approach introduces a class-based workload migration coupled with a BSBF allocation technique. Implementation results show that our approach is better than other state of the art approaches in terms of overall energy conservation, SLA adherence and power state switching; and slightly below par in the area of workload delay

## REFERENCES


[1] Le-Quoc, M. Fiedler, C. Cabanilla, "The Top 5 AWS EC2 Performance Problems" Whitepaper. Datadog Inc, 2013.

[2] Y. Xu, Z. Musgrave, B. Nobel, M. Bailey, "Workload-Aware Provisioning in Public Cloud. Internet Computing", vol. 18, no. 4, IEEE Computer Society Press, 2014, pp.15-21.

[3] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the art of virtualization," Proc. of 19th ACM symposium on Operating systems principles, 2003, pp. 177.

[4] H. Goudazi, M. Pedram, "Multi-dimensional SLA-based Resource Allocation for Multi-tier Cloud Computing Systems" Intl Conf. on Cloud Computing (CLOUD), IEEE Computer Society Press, 2011, pp. 324-331.

[5] A. Karthick, E. Ramaraj, R. Subramanian, "An Efficient Multi Queue Job Scheduling for Cloud Computing", World Congress on Computing and Communication Technologies (WCCCT), IEEE Computer Society Press, 2014, pp. 164-166.

[6] V. Rajeshram and C. Shabarran, "Heuristics Based Multi Queue Job Scheduling for Cloud Computing Environment", International Journal of Research in Engineering and Technology, vol. 4 no. 5, 2015, pp. 163 – 166.

[7] K. Gouda, T. Radhika, M. Akshatha, "Priority Based Resource Allocation Model for Cloud Computing", International Journal of Science, Engineering and Technology Research, vol. 2, no. 1, pp. 215 -219, 2013.

[8] C. Pawar, R. Wagh, "Priority Based Dynamic Resource Allocation in Cloud Computing", Proc. of the Intl Symposium on Cloud and Services Computing, IEEE Cloud Computing, 2012, pp. 1-6.

[9] W. You, K. Qian, Y. Qian, "Hierarchical Queue Based Task Scheduling" Journal of Advances in Computer Networks, 2014, vol. 2, no. 2, pp. 138–141.

[10] L.Wu, S. Garg, R. Buyya, "SLA-based Resource Allocation for Software as a Service Provider (SaaS) in Cloud Computing Environments" 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2011, pp. 195-204.

[11] M. Macias, J. Guitart, "Client Classification Policies for SLA Enforcement in Shared Cloud Datacenters", Proc. of 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, 2012, pp. 156-163.

[12] A. Beloglazov, R. Buyya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in Cloud data centers", Concurrency and Computation: Practice and Experience, pp. 1397–1420.

[13] A. Beloglazov, J. Abawajy, R. Buyya, "Energy-Aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing", Future Generation Computing Systems, 2012, vol. 28 no. 5, pp. 755-768.

[14] N. Hieu, M. Francesco, A. Yla-Jaaski, "Virtual Machine Consolidation with Usage Prediction for Energy-Efficient Cloud Data Centers", Proc. of 8th IEEE International Conference on Cloud Computing, 2015, pp. 750-757.

[15] A. Mosa, N. Paton, "Optimizing Virtual Machine Placement for Energy and SLA in Clouds Using Utility Functions", Journal of Cloud Computing: Advances, Systems and Applications, 2016, 5:17

[16] Pennsylvania, "Computer Power Usage", University of Pennsylvania, 2013.

[17] Y. Lu, Q. Xie, G. Kliot, A. Geller, J. Larus, A. Greenberg, "Join-Idle-Queue: A Novel Load Balancing Algorithm for Dynamically Scalable Web Services", ACM Journal of Performance Evaluation, vol. 68, no. 11, 2011, pp. 1056-1071.

[18] M. Uddin, A. Rahman, "Server Consolidation: An Approach to Making Data Centers Energy Efficient and Green", Intl Journal of Scientific & Engineering Research, vol. 1, no. 1, 2010, pp. 1-7.

[19] R. Talaber, "Using Virtualization to Improve Data Center Efficiency", The Green Grid, White Paper 19, 2009.

[20] N. Rodrigo, Calheiros, R. Ranjan, A. Beloglazov, A. Cesar, F. De Rose, R. Buyya, "CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of

Resource Provisioning Algorithms", *Software: Practice and Experience (SPE)*, 2011, vol. 41, no. 1, pp. 23-50.

[21] SPECPower, "The SPECpower benchmark results for the fourth quarter of 2010", Online at https://www.spec.org/power_ssj2008/results/ res2011q1/power_ssj2008-20110124-00338.html

[22] K. Park and V. S. Pai, "Comon: A mostly-scalable monitoring system for planetlab," SIGOPS Oper. Syst. Rev., vol. 40, no. 1, pp. 65–74, 2006

[23] J. Wilkes, C. Reiss, "Google Cluster Usage Traces: format + schema of Google Workloads", 2011 http://code.google.com/p/googleclusterdata/