# Energy-aware Virtual Machine Consolidation for Cloud Data Centers

Dabiah Ahmed Alboaneen[1 2], Bernardi Pranggono[1] and Huaglory Tianfield[1]

[1]*School of Engineering and Built Environment, Glasgow Caledonian University, Glasgow G4 0BA, United Kingdom*

[2]*College of Education, University of Dammam, Jubail, Kingdom of Saudi Arabia*

*{Dabiah.Alboaneen, b.pranggono, h.tianfield}@gcu.ac.uk*

***Abstract*—One of the issues in virtual machine consolidation (VMC) in cloud data centers is categorizing different workloads to classify the state of physical servers. In this paper, we propose a new scheme of host's load categorization in energy-performance VMC framework to reduce energy consumption while meeting the quality of service (QoS) requirement. Specifically the underloaded hosts are classified into three further states, i.e., underloaded, normal and critical by applying the underload detection algorithm. We also design overload detection and virtual machine (VM) selection policies. The simulation results show that the proposed policies outperform the existing policies in CloudSim in terms of both energy and service level agreements violation (SLAV) reduction.**

***Keywords*— virtual machine consolidation (VMC); energy-aware; energy-efficient; cloud data center**

## I. INTRODUCTION

Data centers require huge amounts of energy to operate, resulting in high operating costs and carbon dioxide ($CO_2$) emissions. According to statistics, data centers consume up to 3% of all global electricity production while producing 200 million metric tons of $CO_2$ in 2012. This percentage is expected to increase significantly in the next years [1].

In recent years, significant research has focused on making data centers more sustainable and environment-friendly, particularly in reducing their energy consumption. Virtualization technology plays an important role in reducing power consumption in data centers by creating multiple virtual machines (VMs) on a single physical server (referred to as host) and implementing the process of virtual machine consolidation (VMC).

Due to the dynamic workloads, the number of VMs located in a host may vary, causing either performance degradation in case of CPU overutilization or increase in energy consumption in the other case. Hence, VMC in a cloud data center needs to perform live migration of VMs, i.e., to move a VM between hosts to meet the varied workloads and minimize the number of active hosts.

The remainder of this paper is organized as follows. In section II we discuss the research problem and questions. In section III we discuss previous relevant work. In section IV we propose a new scheme of host's load categorization in VMC framework. In section V we design the overload detection and VMs selection policies. In section VI we present the initial results. Finally, we conclude the paper in section VII.

## II. RESEARCH PROBLEM

### *A. Research Motivation*

In general, the VMC process can be divided into four tasks [2]:

1. Apply the overload detection policies to evaluate if a host is overloaded. If yes, then some VMs should be migrated from it to other active hosts or to be activated, to avoid performance degradation.
2. Apply the overload detection policies to evaluate if a host is underloaded. If yes, then all VMs should be migrated from it so that the host can be switched to a low-power mode.
3. Apply VMs selection policies to select VMs to migrate from the overloaded host.
4. Apply VMs placement policies to allocate the selected VMs on other active hosts or to be activated.

Typically, hosts in cloud data centers are categorized into three states: overloaded, underloaded and idle. A host is overloaded when the total CPU demand of VMs in the host exceeds the capacity of the CPU host and causes service level agreements violation (SLAV). Underloaded host means the host is currently in use with no SLAV. Idle host means it is available but not currently in use. The common VMC process is first migrating VMs from overloaded hosts to avoid performance degradation until they become underloaded. Next, the VMs from underloaded hosts need to be migrated to other hosts in order to switch off all the underloaded hosts to save energy. The VMC process in cloud data centers can be illustrated in Figure 1.

The above-mentioned VMC framework has a good performance in terms of energy consumption and SLAV reduction. However, it can be seen from the VMC process illustrated in Figure 1 that there is no state for the host when its load is between overloaded and underloaded states. Specifically, the hosts that are not considered as overloaded are not necessarily underload where all VMs need to be migrated. The hosts may be in a normal load that no action needs to be taken, or below the normal load but not really underloaded. That is critical as they have the capacity to receive the VMs to be migrated until they reach the normal load. Therefore, it is crucial to redesign the host's load categorization in the VMC framework to make it more efficient. The main expected benefit from classifying underloaded hosts into three further states is to speed up the VMC process by determining from the beginning which hosts are suitable for placing the VMs selected for migration and which hosts are not.

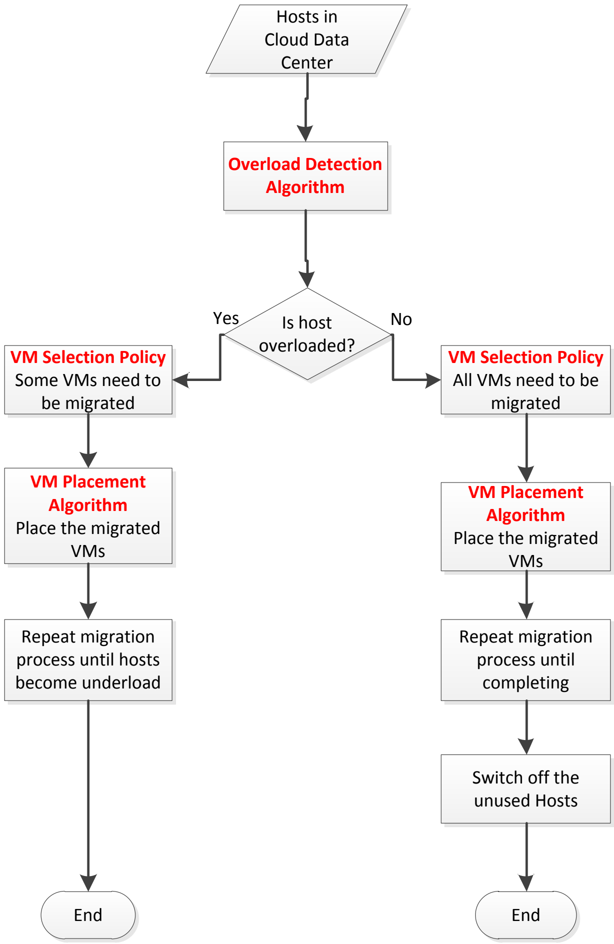


**Figure 1. VMC process in cloud data center**

### *B. Research Questions*

The key questions that have to be answered are:

- How to decide when, which VMs, and where to migrate to provide more energy-performance tradeoff?
- How to redesign the categorization of underloaded hosts?
- How to conduct the VM placement to reduce the energy consumption, number of migration VMs and SLAV in cloud data centers?
- How to keep a balance between minimizing energy consumption and providing QoS performance in cloud data centers?

### *C. Research Contributions:*

- To undertake a literature review on VMC algorithms in cloud data centers. The review will help identify the advantages and the limitations of each algorithm.
- To develop a new scheme of host's load categorization in VMC framework.
- To develop the overload detection, VMs selection and VMs placement algorithms to save energy consumption in cloud datacenters.
- To evaluate the proposed algorithms and policies using CloudSim [3] and on OpenStack Neat platform [4].

## III. RELATED WORK

Pinheiro et al. [5] studied energy management at the data center level. The authors have developed an algorithm that can switch nodes on and off dynamically according to the expected performance and power implications of the decision.

Nathuji and Schwan [6] applied dynamic VMC for minimizing energy consumption of a data center. The authors have shown the energy advantages obtained by VMC. They concluded that the energy consumption can be significantly minimized when VMs are consolidated and the improvement of power consumption yields up to 34%.

In [7], the authors have classified the VMC problem into four sub-problems: detect overloaded hosts, detect underloaded hosts, select VMs to migrate and place the selected VMs. To detect the overloaded hosts, a fixed utilization threshold policy (THR) was proposed. If the CPU utilization of a host drops below the lower threshold, all VMs should be migrated from this host. Then the host has to be switched to the low power mode in order to save energy. On the other hand, if the CPU utilization exceeds the upper threshold, some VMs should be migrated from this host in order to prevent performance degradation. The authors have proposed four polices for selecting VMs to migrate: single threshold (ST), Minimization of Migrations (MM), highest potential growth (HPG) and random choice (RC). The simulation results showed the flexibility of the proposed algorithms.

However, setting fixed value for the threshold is inappropriate for an environment that has dynamic and changing workloads. Therefore, as a continuous work to the previous study in [7], the authors in [8] have studied the problem of allocating fixed utilization thresholds and suggested that the system has to adjust its behavior automatically based on the workload patterns presented by the applications. The authors have proposed a novel technique for the dynamic VMC with auto-adjustment of the threshold values based on a statistical analysis of the historical data collected during the lifetime of VMs, which guarantees a high level of meeting the service level agreements (SLA) [8].

The idea of the dynamic threshold (DT) is based on the random variable that represents the sum of the CPU utilization by all VMs located to the host. The simulation findings showed that the DT is better than other migration-aware policies in terms of the level of SLA violation (SLAV) (<1%) and the number of VM migrations. However, the level of energy consumption was the same.

The authors in [2] have proposed four adaptive threshold utilization algorithms to estimate the CPU utilization and to detect the overloaded hosts based on the statistical analysis of historical data collected during the lifetime of VMs. The algorithms are: median absolute deviation (MAD), inter quartile range (IQR), local regression (LR) and robust local regression (RLL) with three different VM selection policies: the minimum migration time (MMT), random selection (RS) and maximum correlation (MC). The findings of implementing and comparing the proposed algorithms indicate that the dynamic VMC algorithms significantly outperform static allocation algorithms. The MMT policy produces better results compared with the MC and RS policies, dynamic VMC algorithms based on LR outperform the static threshold and adaptive-threshold-based algorithms.

In [9], the authors have proposed a novel dynamic VM allocation and VM selection policies for reducing energy consumption and SLAV in cloud data centers. The mean and standard deviation of CPU utilization for VM were used to decide which hosts were considered overloaded. In addition, the positive maximum correlation coefficient was used to select VMs from overloaded hosts for migration. The study indicates that the proposed overload detection and selection policies outperform the implemented polices in CloudSim in terms of reduction in SLAV. However, the previous policies in [2] perform slightly better than the proposed policies in terms of energy consumption.

The study in [10] focused on the VM selection task during the VMC process in cloud data centers and proposed a novel approach to illustrate how to use dynamic criteria to select VM to be migrated instead of the fixed ones used before. They modeled VM selection as a dynamic decision-making (DDM) task by using fuzzy Q-learning (FQL) to integrate multiple criteria in VM selection policies, specifically the maximum and minimum CPU utilization of VM. The simulation showed that the FQL approach outperforms the state-of-the-art VM selection algorithms using fixed criteria in terms of the energy-performance trade-off in cloud data centers.

In [11], the authors proposed a VMC framework for cloud data centers to achieve a better trade-off between energy consumption and SLA performance. They have implemented a SLAV decision algorithm (SLAVDA) to decide if a host is overloaded with SLAV. The authors have changed the first step about choosing the overloaded hosts of the existing VMC in CloudSim in [2]. They have classified the host states overload into two categories: overload with SLAV and overload with no SLAV. In the next step, the authors select VMs from the overload with SLAV hosts until they become overload with no SLAV. The simulation findings indicate that the proposed algorithm (SLAVDA) is better than the existing VMC framework with a reduction of 11.8%~27% in energy consumption, 57.9%~78.4% in SLAV and 63.2%~84.1% in energy-performance metrics.

Compared to the previous works, the hosts in cloud data centers are classified into either overloaded or underloaded hosts while ignoring other states of the host. If a host is not overloaded, it is not necessarily in an underloaded state; it may be in a normal load or close to a normal load. In VMC, we need to fine-grain the level of underload in order to improve the VMC process in terms of minimizing the number of VMs, energy consumption and SLAV. In this paper, we propose a new scheme of host's load categorization in energy-efficient VMC framework to minimize searching in all underloaded hosts to place the VMs to be migrated. Therefore, this will lead to a better performance than the existing VMC framework both in terms of minimizing energy consumption and meeting the QoS requirement.

## IV. Proposed Host's Load Categorization for Virtual Machine Consolidation

### A. *Host's Load Categorization*

The new host's load categorization classifies the hosts into four states, instead of three, i.e., overloaded, normal, critical and underloaded. Some changes will be made in step 1 and step 2; but step 3 will remain the same. The proposed VMC steps are as follows:

i. Classify the hosts in data centers into four states:
   1. Evaluate whether the host is considered as an overloaded host. Some VMs should be migrated to the critical host until it becomes normal.
   2. Evaluate whether the host is considered as an underloaded host. All VMs should be migrated to the critical host and then turn off the underloaded host.
   3. Evaluate whether the host is considered as a normal loaded host (between overloaded and critical). No action is needed.
   4. Evaluate whether the host is considered as a critical host (between underloaded and normal). No VMs migration is needed. The host is available to receive VMs until it reaches normal load.

ii. Apply VMs selection policies to select VMs from overloaded hosts to be migrated.

iii. Apply VMs placement algorithm to place the VMs selected for migration on critical hosts only.

iv. Switch off all underloaded hosts to become idle to save energy. Some idle hosts may be turned on and become underloaded hosts.

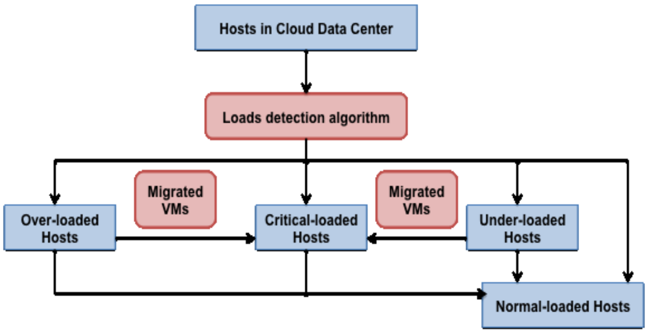


Figure 2. VMC flowchart based on the new host's load categorization

Figure 2 shows the flowchart of VMC based on the new host's load categorization. First, we use the overload detection algorithm to classify the hosts into two host states: overloaded and underloaded; and then we further classify the underloaded hosts into normal, underloaded and critical by using the underload detection algorithm. Next, the VM selection policies will be implemented to select VMs to continuously migrate from overloaded and underloaded hosts to the critical hosts until the load becomes normal. Finally, we apply the VM placement algorithm to reallocate the VMs selected for migration on the critical hosts based on the most efficient host that saves more energy.

### B. *Overload Detection Policies*

The overload detection policy is used to decide whether a host is considered as overloaded or not by predicting the host CPU utilization. Four policies have already been implemented in CloudSim [2]: MAD, IQR, LR and LRR policies. It is suggested that the algorithm that uses LR policy to predict CPU utilization outperforms the other algorithms [2].

### C. *Overload Detection Policy - Mean (Mn)*

This policy aims to use the mean only to set an adaptive upper CPU utilization threshold based on VMs' average CPU utilization. The mean is equal to the sum of CPU utilization divided by the total number of VMs.

Assume we have a set of VM CPU utilizations as $\{U_1$, U2, …. $U_n\}$, we sort these values in increasing order. Then we calculate the sum of CPU utilization of VMs (denoted as $X_t$) on host $H_i$ in time t as:

$$Xt = \sum_{j=1}^{Ni} Uij,t \quad (1)$$

where N is the number of VMs.

Then we can calculate the mean (denoted as M) as:

$$M = \frac{1}{N}\sum_{t=1}^{N} Xt \quad (2)$$

According to [2], the prediction CPU utilization threshold (denoted as T) is defined as:

$$\mathrm{T} = 1\text{-s} \cdot \mathrm{M} \quad (3)$$

where s is the safety parameter and it is a constant value. When T is higher than the current CPU utilization of $H_i$, then $H_i$ is considered as an overloaded host.

### D. *VM Selection Policies*

The VM selection policy is used to select VMs from overloaded hosts to be migrated to underloaded hosts to prevent performance degradation. There are four policies in CloudSim for selecting VMs to migrate from overloaded hosts: MMT, MU, RS and MC policies. It is suggested that MMT policy outperforms other policies because it selects the VM to be migrated that requires the minimum migration time [2]. It is calculated by the amount of RAM utilized by VM divided by the network bandwidth available for the host. Due to the fact that all network links have the same amount of bandwidth (1 Gbps), the migration time only considers the utilized RAM on VM.

### E. *VM Selection Policy - Maximum Requested Bandwidth (MBW)*

The maximum requested bandwidth (MBW) policy aims to select VM $V_i$ from overloaded host $H_i$ that has the maximum requested bandwidth to migrate it. The step is repeated until the host $H_i$ is considered not overloaded.

## V. Methodology and Performance Metrics

### A. *Simulation Setup*

As our goal is to implement infrastructure as a service (IaaS), we need to evaluate the proposed policies on a large-scale virtualized data center infrastructure. As implementing repeatable large-scale experiments on a real infrastructure is not practical, CloudSim toolkit 3.0.3 simulator [3] was selected to evaluate the proposed policies. CloudSim is widely used to simulate cloud system components such as data centers and VMs. It supports policies for VMs allocation and selection, power models for data center resources and provides different types of workloads. A data center with 800 nodes was simulated with heterogeneous servers and VMs. Half of the servers are HP ProLiant ML110 G4, and the other half are HP ProLiant ML110 G5. We use the same power models provided in the website [12] for both servers, as shown in Table 1. The characteristics of hosts and VMs used in the experiments are listed in Table 2 and 3, respectively.

Table 1. Power consumption by two types of hosts at different load level in Watts [6]

| Host | 0% | 10% | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 100% |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **HP ProLiant G4** | 86 | 89.4 | 92.6 | 96 | 99.5 | 102 | 106 | 108 | 112 | 114 | 117 |
| **HP ProLiant G5** | 93.7 | 97 | 101 | 105 | 110 | 116 | 121 | 125 | 129 | 133 | 135 |

Table 2. Characteristics of the two types of hosts used in experiments

| # of hosts | Type | # of cores | MIPS | RAM | Storage | BW |
|---|---|---|---|---|---|---|
| 400 | HP ProLiant ML110 G4 | 2 | 1860 | 4096 | 1 GB | 1 GB |
| 400 | HP ProLiant ML110 G5 | 2 | 2660 | 4096 | 1 GB | 1 GB |

Table 3. Characteristics of the four types of VMs used in experiments

| VM | # of cores | MIPS | RAM | Storage |
|---|---|---|---|---|
| Type 1 | 1 | 2500 | 870 | 2.5 GB |
| Type 2 | 1 | 2000 | 1740 | 2.5 GB |
| Type 3 | 1 | 1000 | 1740 | 2.5 GB |
| Type 4 | 1 | 500 | 613 | 2.5 GB |

### *B. Real Workload*

The real workload is provided as a part of the CoMon project, a monitoring infrastructure for PlanetLab [13]. We selected five days from the workload traces collected by Beloglazov and Buyya in April 2011 [2]. During the simulations, each VM is randomly assigned a workload trace from one of the VMs from the corresponding day. The number of VMs in each day is shown in Table 4.

Table 4. The number of VMs in the real workload

| Date | 3 April | 9 April | 11 April | 12 April | 20 April |
|---|---|---|---|---|---|
| **No. of VMs** | 1463 | 1358 | 1233 | 1054 | 1033 |

### *C. Performance metrics*

The tradeoff between minimizing the energy consumption and meeting QoS requirements is very important in cloud based data centers. Meeting QoS requirements is usually formalized in the form of SLAs. There are also some characteristics such as minimum throughput or maximum response time delivered by the deployed system which can determine the level of QoS. These characteristics vary from one application to another, therefore it is necessary to define a workload independent metric that can be used to evaluate the SLA delivered to any VM deployed in an IaaS.

According to the study in [2], the SLAs are delivered when 100% of the CPU utilization requested by applications inside a VM is satisfied. Therefore, we used SLA-related metrics defined in [2] to evaluate the proposed policies. These are: energy consumption, SLATAH (SLAV time per active host), PDM (performance degradation due to migrations), SLAV, ESV (Energy and SLAV) and number of VMs migration.

*1) Energy consumption*

This metric represents the total energy consumed by the physical data center resources.

*2) SLAV*

This combined metric can be calculated from both SLATAH and PDM metrics as equation (4):

$$SLAV = SLATAH \cdot PDM \tag{4}$$

where

$$\text{SLATAH} = \frac{1}{N}\sum_{i=1}^{N}\frac{Tsi}{Tai} \tag{5}$$

$$\text{PDM} = \frac{1}{M}\sum_{j=1}^{M}\frac{Cdj}{Crj} \tag{6}$$

where $T_{si}$ is the SLAV time on host $H_i$, $T_{ai}$ is the active time of host $H_i$, M is the total number of VMs, $C_{dj}$ is the performance degradation of the VM j caused by migrations and $C_{rj}$ is the overall CPU capacity requested by VM j.

*3) ESV*

This combined metric can be calculated from both energy consumption and SLAV as [2]:

$$ESV = Energy \cdot SLAV \tag{7}$$

*4) Number of VMs migration*

This metric represents the total number of migrated VMs. The least number of VMs migrations is the best for decreasing the performance degradation. Because the same amount of CPU capacity is allocated to a VM on the destination host, each VM migration triggers SLAV. Hence, it is important to minimize the number of VMs migration.

## VI. SIMULATION RESULTS

We have evaluated three implemented overload detection policies (MAD, IQR and LR) and MMT as VM selection policy, the proposed overload detection policy (Mn) and the proposed VM selection policy (MBW) through CloudSim in terms of SLAV, energy consumption and number of VMs migration. We evaluated each algorithm on real workloads. The results show that the combination of mean and maximum requested bandwidth (Mn_MBW) was able to reduce the total energy consumption more effectively than the other three algorithms. In addition, the Mn_MBW was also able to minimize the number of VM migration more efficiently than the other three algorithms. The reason is that the VM selection policy based on choosing the maximum requested bandwidth to migrate has resulted in migrating few VMs from overloading hosts. The total energy consumptions consumed by all algorithms, the ESV and the number of VM migrations are shown in Table 4 and Figures 3, 4 and 5.

Table 4. The energy consumption, SLAV, ESV and number of VM migrations of the four algorithms (mean values)

| Algorithm | Energy (KWh) | SLATAH % | PDM % | SLAV $10^{-5}$ | ESV $10^{-3}$ | No. of VM migration |
|---|---|---|---|---|---|---|
| MAD_MMT | 195.528 | 4.992 | 0.062 | 324.2 | 633.9 | 27851.4 |
| IQR_MMT | 200.176 | 4.97 | 0.064 | 320.2 | 640.9 | 28061.8 |
| LR_MMT | 172.18 | 6.298 | 0.082 | 505.8 | 870.8 | 30209.2 |
| Mn_MBW | 171.984 | 1.19 | 0.002 | 5.6 | 9.63 | 2150 |

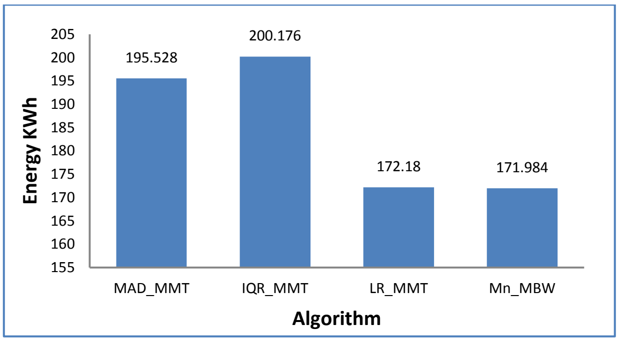


Figure 3. Energy consumption of the four algorithms

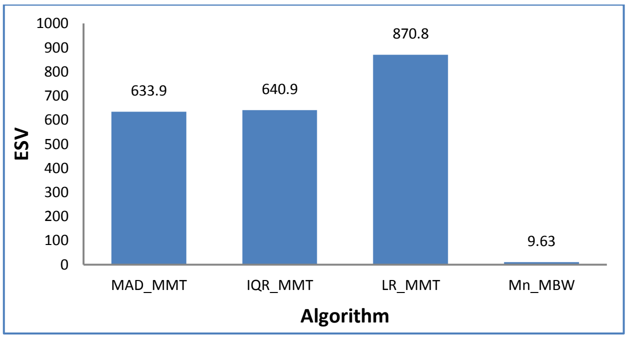


Figure 4. ESV of the four algorithms

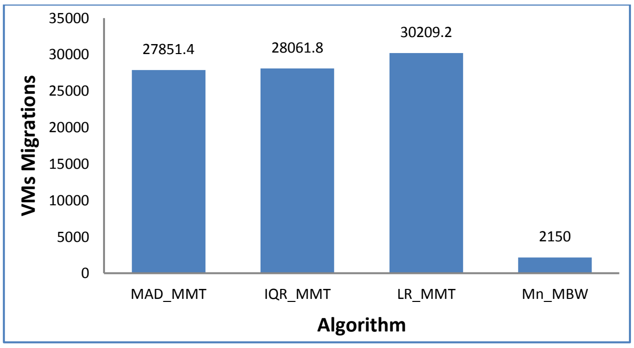


Figure 5. Number of VMs migrations of the four algorithms

## VII. CONCLUSION

We proposed a new scheme of host's load categorization in VMC framework in cloud based data centers to reduce energy consumption while meeting QoS requirements. The main idea is to classify the underloaded hosts into three further states, i.e., underloaded, normal and critical by applying underload detection algorithms. We also designed an overload detection policy called Mn which uses the *mean* to predict the upper threshold and VM selection policy, called MBW, based on the maximum requested bandwidth. The simulation results show that the proposed policies outperform the existing policies in CloudSim with regards to both energy and SLAV reduction.

## REFERENCES

[1]. DatacenterDynamics 2012, "Datacenter Dynamics 2012 global census," *Datacenter Dynamics 2012 Global Census*.

[2]. A. Beloglazov and 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 (CCPE), Wiley Press*, New York, USA, vol. 24, no. 13, pp. 1397-1420, Sep. 2012.

[3]. R. N. Calheiros, R. Buyya, and A. Beloglazov, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and Experience, Wiley Press*, pp.23-50, Jan. 2011.

[4]. A. Beloglazov and R. Buyya, "OpenStack Neat: A Framework for Dynamic and Energy-Efficient Consolidation of Virtual Machines in OpenStack Clouds," *John Wiley & Sons, Ltd*, USA, 2014.

[5]. E.Pinheiro, R. Bianchini, E.V. Carrera, and T. Heath, "Load balancing and unbalancing for power and performance in cluster-based systems." *Workshop on compilers and operating systems for low power*, Barcelona, pp. 182, Spain, 2001.

[6]. R. Nathuji and K. Schwan, "Virtualpower: Coordinated power management in virtualized enterprise systems," *Proc. of the ACM SIGOPS Symp. on Op. Sys. Principles, ACM press*, pp.265-278, Dec. 2007.

[7]. A. Beloglazov, J. Abawajy, and R. Buyya, "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing," *Future Generation Computer Systems*, pp.755-768, May. 2012.

[8]. A. Beloglazov and R.Buyya. "Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers," *Proceedings of the 8th international workshop on middleware for grids, clouds and e-science ACM*, pp. 4, 2010.

[9]. Z.Cao and S. Dong, "Dynamic VM Consolidation for Energy-Aware and SLA Violation Reduction in Cloud Computing," *Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2012 IEEE 13th International Conference*, pp. 363, 2012.

[10]. S.S.Masoumzadeh and H. Hlavacs, "Integrating VM selection criteria in distributed dynamic VM consolidation using Fuzzy Q-Learning," *CNSM*, pp. 332, 2013.

[11]. Z.Cao and S. Dong, "Energy-Aware Framework for Virtual Machine Consolidation in Cloud Computing", *High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing (HPCC_EUC), 2013 IEEE 10th International Conference*, pp. 1890, 2013.

[12]. All Published SPECpower_ssj2008 Results, http://www.spec.org/power_ssj2008/results/power_ssj2008.html

[13]. [13] K.S. Park and V.S. Pai, "CoMon: a mostly-scalable monitoring system for PlanetLab,"*ACM SIGOPS Operating Systems Review*, pp.65-47, 2006.