

A Survey of Energy Aware Cloud's Resource Allocation Techniques for Virtual Machine Consolidation

Asif Farooq

Department of Computer Science,
University of Sargodha, Lahore Campus,
Lahore, Pakistan

Tahir Iqbal, Muhammad Usman Ali,
Zunnurain Hussain

Department of Computer Science, Bahria University,
Lahore Campus, Lahore, Pakistan

Abstract—As the demand for cloud computing environment is increasing, new techniques for making cloud computing more environment-friendly are being proposed with an aim to convert traditional cloud computing into green cloud computing. A standout amongst the most imperative complications in cloud computing is streamlining of energy utilization because its importance is increasing rapidly. There are numerous strategies and algorithms used to limit the energy utilization in the cloud. Methods incorporate DVFS, UP-VMC, Utility based MFF, HCT, AVVMC, ACO, and ESWCT. In this survey, a review of energy-aware techniques is presented for making virtual machines more energy efficient in a cloud computing. Working on each technique is briefly explained. A comparative analysis is also given for comparing multiple efficient techniques with respect to performance metrics.

Keywords—Cloud computing; energy aware; green cloud computing

I. INTRODUCTION

Cloud computing is defined as a large group of interconnected servers and computers with a primary objective of providing reliable services and custom-made computing environments to its users. As cloud computing grows the demand for resources required to support such a large network is also increasing. Hundreds of data servers are added daily to data centers around the world to support the growing needs of cloud computing which lead to power consumptions issues. The complete framework for cloud computing depends upon datacenters for data storage and processing. These data centers mainly consist of hundreds of thousands of servers which are unfriendly to the environment in terms of carbon emissions. According to [1] one data center can annually produce up to 170 million tons of carbon, which is estimated to increase up to 670 million tons in 2020 worldwide [2]. With a huge consumption of electricity, the operational costs are also high. The total amount of consumed energy is estimated up to 250 billion kilowatt-hours by a single datacenter [2]. Hence these factors make a datacenter unfriendly to the environment. To overcome such problems cloud's resources must be managed in an optimal method. Allocation of resources plays an important role in maximizing the efficiency as well as reducing the power consumption of the entire system.

In Resource Allocation (RA), Available resources are assigned to needed cloud applications in such a way that it provides a level of satisfaction to its customers while ensuring the overall efficiency of the system. The RA should be beneficial to end users and economical to the service provider [13]. If resources are not managed precisely then it may starve cloud services. Thus energy efficient cloud resource allocation consists in identifying and assigning resources to each incoming user request in such a way that:

- The user requirements are completely fulfilled.
- Least possible number of resources are used.
- Data center energy efficiency is optimized.

Virtualization technology is adopted by data centers to overcome energy and resource efficiency. In virtualization process, each server's resources is logically divided into a certain number of independent partitions. Each of these logical servers or partitions now becomes a Virtual Machine (VM), capable of running independent operating system and applications. Thousands of physical servers can now run millions of VMs, through live VM migration techniques VMs can be transferred from server to server hence balancing the amount of VMs. This helps to switch a physical server in an idle (suspended) state if its resources are not being currently used up to a certain limit [3].

Enhancement of portability, manageability, and security are the main advantages of using VMs. Isolation can also be achieved, ensuring that VMs in a single physical server should not affect with each other. The virtualization process also blocks guest operating system of any VM from directly accessing the hardware resources of the physical server. Due to multiple overheads associated with VMs, regular monitoring is required to avoid performance degradation [4]. As multiple VMs depends on one single server, so the failure of one single server could crash hundreds of VMs. VM energy aware consolidation techniques are adapted for efficiently increasing cloud's resource utilization. For effective utilization of data center resources two step are carried out:

- 1) Effective Placement of VMs in PMs.
- 2) Optimizing allocated resources using live migration techniques.

The primary goal for the placement of VMs is to increase the throughput of cloud's resources and to lower the power consumptions [3]. Two main types of placement algorithms are categorized:

- Power Based Placement Algorithms.
- Application QoS Based Placement Algorithms.

During live migration of a VM, the memory state is uninterruptedly transferred from one PM to another. Live VM migration is the key for improving energy efficiency and resource optimization. Live VM migration causes following problems for dynamic placement of VMs.

- To determine if a PM is over-loaded (which requires migration of VM(s) from one host to other.
- To determine when a host is under-loaded (switching a PM into suspended state by migrating all of its VMs to other PM(s).
- To determine which VM(s) must be transferred from an over-loaded PM.
- To determine new places for migrating VM(s) either from an over-loaded or under-loaded PM(s).

II. LITERATURE REVIEW

In [3] a dynamic VM consolidation utilization prediction approach UP-VMC is presented, both current and future resource utilization is handled for VMs consolidation. A regression-based prediction model is used for future resource utilization prediction. Authors also proposed a VM allocation algorithm to enhance the QoS while minimizing the number of migrations using prediction models. UP-VMC shows significant improvements in the reduction of energy and VM migrations. Another approach [4] based on deploying a self-managing VM placement solution that assigns VMs to PMs dynamically with respect to the utilization of resources. This proposed solution is based on utility functions with a main goal of maximizing the IaaS provider profit as well as reducing energy consumption cost. Provided results shows improvements as compared to existing heuristics based solution. In [5] authors proposed an Ant Colony Optimization (ACO) ACO metaheuristic-based server consolidation mechanism to overcome the issue of power consumptions as well as maximizing the resource utilization in large virtualized data centers. Authors compared proposed solution with existing methods and show the effectiveness of the proposed algorithm. In [6], authors proposed two algorithms (Honeybee algorithm and HCT) for VM placement. Both of these proposed algorithms are tested again many overload detection VM selection algorithms. Results are concluded based on energy consumption along with VM migration and SLA. Results conclude that the proposed HCT algorithm shows improvement in energy consumption, migration and SLA. Another approach [7] for energy-aware VM placement is proposed using a technique called Minimum Correlation Coefficient (MCC). Authors defined the VM placement problem with flexibility among bin sizes and prices and proposed a power-aware PABFD method. To adjust the trade-off between SLAV and the amount of energy consumed Fuzzy

AHP method is adopted. This method delivers a satisfactory trade-off between energy consumptions and SLA violations. Two energy aware algorithms [8] are proposed (ESWCT) and (ELMWCT) for scheduling VMs using a workload-aware consolidation technique with a primary goal of reducing energy consumption in virtualized data centers. Multiples factors including imbalance utilization and resource utilization are deployed to compare against multiple scheduling algorithms. Results from both algorithms show good power savings and balanced resource utilization. In [9] an another algorithm for dynamic placement is proposed in which authors applied ant colony optimization (ACO) algorithm on multi-dimension bin packing (MDBP) algorithm. Results are compared with a traditional greedy algorithm (FDD). Both of these algorithm are implemented and validation in simulations. On average the new approach consume 4.1% less power as compared to FDD algorithm but comes a tradeoff with computational time, as ACO based algorithm took longer time for computation than FDD. In [10], author proposed another energy aware EEVS scheduling algorithm, which works by finding an optimal frequency to process VMs on a PM. Proposed approach also support DVFS and works by allocating VM to highest performance power ration PM. Results are simulated and show that EEVS algorithm can reduce up to 20% power consumption and increase 8 % of processing capacity in a best-case scenario. In [11] another nature-inspired ACO based algorithm of VM placement is proposed with a primary goal to get a non-dominated solution set that concurrently reduces the energy consumptions. The proposed ACO based VMPACS algorithm is compare with two single objective algorithm including FFD and SACO, obtained results shows that VMPACS can search solution space more efficiently to obtain a solution that use minimum number of servers and maximum resource utilization. This solution improved the overall efficiency hence less power is consumed. Two new approaches [12] are proposed related to VM placement issues, both of the proposed approaches are originated by modifying the traditional firefly algorithm with hierarchical cluster and meta-heuristics. Results related to power consumptions are deduced and compared with tradition HCT and honeybee algorithm, claiming

2% less energy consumption than original firefly algorithm and 12% less energy consumption than honeybee algorithm. Below table shows the techniques and algorithms that are selected to conduct this review paper (Table I). Each paper is selected on the basis of this paper's primary goal of energy efficiency.

TABLE I. AN OVERVIEW OF ENERGY EFFECTIVE VM CONSOLIDATION TECHNIQUES

Techniques / Algorithms	Year of Publish	Ref.
Utilization Prediction aware VM Consolidation (UP-VMC)	2016	[3]
Self-managing utility functions	2016	[4]

HCMFF and MFF algorithm	2016	[12]
EEVS algorithm	2015	[10]
ACO metaheuristic-based server consolidation mechanism	2014	[5]
Honeybee Cluster Technique (HCT)	2013	[6]
Minimum Correlation Coefficient Approach	2013	[7]
ESWCT – ELMWCT algorithm	2013	[8]
VMPACS algorithm	2013	[11]
ACO-Based Approach	2011	[9]

III. PERFORMANCE METRICS

The main goal of all of these approaches is to minimize the power consumption as much as possible either by reducing the amount of active PMs or VMs migrations in a manner that guarantees that SLAs are not violated. Hence the performances of listed techniques are measured on the following metrics:

A. SLA Violations

Service Level Agreement (SLA) is defined as a contract in-between cloud service provider and its customers. This agreement states the performance standards that the provider will obligated to meet. A number of performance and quality factors are established some of which include:

- The availability and uptime of services.
- Response time for applications.
- Notification Schedule for network changes.

For different types of customer infrastructures SLA may specify other parameters such as uninterruptable power supplies. SLA Violations (SLAV) is a performance metric independent form workload applied on any VM in Virtualized data centers. Total performance of VM is measured by obtaining SLAV in over-utilization (SLAVO) and in migration (SLAVM). Both of these evaluation metrics are independent of each other and carry equal importance in SLAV measurements.

$$SLAV = SLAVO \times SLAVM$$

B. Energy Consumption

Overall energy that is consumed by a physical resource during workloads is defined as energy consumption and is usually measured in kilo watts per hour (KwH). Consumption of PM usually depends upon the usages of CPU and memory during application workload. Although studies [9] have pointed that CPU often consumed more energy as compared to

other PM resources. Therefore PM resources utilization is mostly represented by CPU utilization.

IV. TECHNICAL REVIEW

In this Section, a brief overview for the proposed methodologies of each paper is presented. The working of algorithms is explained briefly to develop an insight for each paper.

A. UP-VMC

In [3] the proposed algorithm (UP-VMC) follows two steps procedure for VM Consolidation:

Step 1: Algorithm first start to target the VMs from overloaded and predicted overloaded PMs. Host should be consider overloaded or predicted overload as soon as any of the resources (CPU or Mem.) of PM is exceeded from a set threshold limit. Two regression based prediction models are used for resource utilization, Linear Regression and K-Nearest Neighbor Regression (K-NNR). These models predict the overloading of a PM upon the CPU and memory usage. To determine the relationship for PMs and VMs for current and future resource utilization following equations are used:

$$PU_{pde} = \alpha + \beta U_{pde} \quad (1)$$

Where PU_{pde} and U_{pde} are capacity vectors for the predicted and current used PM respectively. α and β are regression coefficients.

$$PU_{v} = \alpha + \beta U_{v} \quad (2)$$

Similarly (2) predicts the resource utilization of VM.

Three selection policies were selected including Minimum Migration Time (MMT), Maximum Load (MaxL) and Minimum Load (MinL).

Step 2: In second step this algorithm target to eliminate the least-loaded PMs as much as possible. It transfers all VMs from under-loaded PM to most-loaded PMs While ensuring that the destination PM should not be over-loaded by the migrated VMs.

B. Self-Managing Utility Function:

In [4], author proposed self-managing policy, with a goal to maximize the profits by reducing energy consumption and SLA violations during VM placement. Equation (3) defines of the utility function ($Utility(a, t)$) for self-managing policy as:

$$income(a, t) - (EECost(a, t) + EVCost(a, t) + PDMCost(a, t)) \quad (3)$$

Where a represents a map for assigning selected VMs to PMs, (t) represents time period of each assignment, $income(a, t)$ represents the total amount of hosting income from cloud's customers, $EECost(a, t)$ is the estimated amount of energy to be consumed and $EVCost(a, t)$ is the cost of SLAVs

For the development of utility based model five algorithms were presented. Following is an overview for each of the algorithms.

Algorithm-1 estimates the CPU utilization of PM under the workload of an assignment and records the utilization of each CPU of every PM that is linked to the given assignment.

Algorithm-2 calculates the expected amount of energy that is to be consumed in a given assignment in a time interval t .

Algorithm-3 calculates the violation cost of over-loaded PMs.

Algorithm-4 calculates the violation cost due to VM migration.

Algorithm-5 provides a pseudo code of an adopted algorithm that is modified with genetic algorithm for finding such VMs to PMs Assignments that maximizes the utility function.

C. AVVMC

In [5] AVVMC, a null solution along with a set of PMs and randomly selected VMs are assigned to each ant. After initialization a random ant is selected to choose a single VM applying probabilistic decision rule on all feasible VMs. The selected VM is then assigned next to its current PM.

Once every ant has a solution, best solution is picked based upon the value of objective function. For assigning a VM desirably to PM pheromone levels are associated with all VMs. Heuristics values are calculated dynamically for every assignment to balance out resource utilization. Equation (4) is used to initially assign pheromone levels and (5) is used to update pheromone levels.

$$\tau_0 = PE_{FFDL1Norm} \quad (4)$$

where $PE_{FFDL1Norm}$ is the solution of Packing Efficiency of the FFD heuristic.

$$\tau_{v:p} = (1-\delta) \tau_{v:p} + \delta \Delta \tau_{v:p} \quad (5)$$

Where (δ) indicates a decay parameter for pheromone and $\Delta \tau_{v:p}$ indicate pheromone reinforcement applied on each $v - p$ pairs.

D. Honey Bee Cluster Technique (HCT)

In [6] a Honey Bee Cluster Technique (HCT) is proposed, the algorithm works in following steps:

1) Resources (CPU, Memory) are clustered using a technique called hierarchical clustering in which each cluster behaves as a single resource.

2) Assignment is categorized and honeybee parameters ($n, m, s, Iteration, \alpha$) are initialized. Where $(n),(m),(s)$ represents the number of employed bees, on-looker bees and scout bees respectively, $(iteration)$ represents the max iteration number and (α) represents initial penalty parameter value.

3) A solution is constructed for employee Bee initialization. After that every employee bee locates a suitable VM for each task.

4) Evaluation of fitness function for each employed bee is calculated using (6).

$$Fitness = \frac{CT_{max}}{CT_i} + \frac{1}{PT_i} \quad (6)$$

where CT_{max} is total completion time of a task, CT_i is completion time of specific task and PT_i is the processing time for specific task.

E. PABFD-MCC

In [7], author presented an energy efficient architecture for IaaS layer along with a placement approach for VMs using a method called PABFD-MCC. The key idea behind this approach is to provide a suitable tradeoff between SLA violations and energy consumption, as the resource usage in a VM gets higher, the probability of host overloading is also increased along with a risk of SLA violations.

1) The correlation of CPU utilization among VMs is estimated, along with the CPU correlation between migrating VMs.

2) A best suitable host is selected for VM placement based on score assigned to each host, score is calculated on the basis of following factors:

- Power consumption of PM after VM allocation
- Correlation coefficient between migrating VMs and VMs running on PM.

F. ESWCT and ELMWCT

In [8] a two part algorithm is proposed, First part of algorithm is ESWCT which locates a placement for VM to get an effective and balanced host's resource utilization. Imbalance Utilization Value (IUV_i) is calculated among multi dimension recourses of physical server. Equation (7) is used to define IUV_i as:

$$\frac{(CPU_i^u - IR_i^u)^2 + (MEM_i^u - IR_i^u)^2 + (NET_i^u - IR_i^u)^2}{3} \quad (7)$$

Where IR_i^u is integrated resource utilization of a server (i) which is defined in Eq.8, CPU_i^u MEM_i^u NET_i^u is the average usage of CPU, MEM and NET of server (i) respectively.

$$\frac{CPU_i^u + MEM_i^u + NET_i^u}{3} \quad (8)$$

ESWCT works on following three steps:

1) Compute the capability of every component of each PM.

2) Get the component capability of VM.

3) Assign VM to PM with smallest IUV_i value.

ELMWCT is the second part of the algorithm, first it chooses the VM which are needed for migration, for this purpose node utilization threshold vector is introduced. Afterward VMs are allocated to PM using ESWCT algorithm.

G. Ant Colony Optimization (ACO) based Workload Placement

Another [9] ACO optimized VM placement algorithm is proposed for VM consolidation with a goal to reduce the

number of active PMs and maximizes resource utilization. A multi-dimension bin packing workload placement problem is modeled for validating proposed solution. The algorithm works in following steps:

- Input parameters including a set of VMs and active PMs along with vectors including resource demand and resource capacity are given to the algorithm.
- Parameters are initialed and a pheromone trail is set to τ_{max} . After this, algorithm starts its iterations up to a defined number (*nbCycles*) in every iteration each ant opens up a bin and starts building its solution.
- VMs are then assigned to PMs using probabilistic decision rule until there is no VM left or bin capacity is full.
- Once each ant finished building its solution the best solution is saved.
- Values of τ_{min} and τ_{max} are computed and pheromone levels are updated using pheromone update rule.

Following equations are used to for probabilistic decision rule and pheromone update, respectively.

$$P_v^i = \frac{[\tau_{i,v}]^\alpha \cdot [\eta_{i,v}]^\beta}{\sum u \in N_v [\tau_{u,v}]^\alpha \cdot [\eta_{u,v}]^\beta} \quad (9)$$

where $(\tau_{i,v})$ represents pheromone based desirability and $(\eta_{i,v})$ describes heuristics information.

$$\tau_{i,v} = (1 - \rho) \cdot \tau_{i,v} + \Delta\tau_{i,v}^{Best} \quad (10)$$

where $(\Delta\tau_{i,v}^{Best})$ represents iteration's best item bin pheromone amount.

H. EEVS Algorithm

In [10], the proposed algorithm performs scheduling in three phases, VMs allocation, Updating VMs and cloud reconfiguration. To set an optimal frequency for each PM, DVFS technology is adopted. Set of VMs and PMs are given to algorithm as input and schedule of VM, energy consumption and processing time is the output from algorithm. Algorithm works in following steps:

- Sorts PMs in a decreasing order of optimal performance power ratio.
- Starts VM allocation for a specific time period.
- Optimal frequency is computed along with practical optimal frequency and assigned to each primary machine.
- Information for active PMs and running VMs are updated.
- After this, phase three for cloud reconfiguration begins.

The proposed methodology although consumes less amount of energy and processes more VMs than compared algorithm but there is some downsides to this approach. Two assumptions were made, VM migration and performance-

power penalties of processor transition were ignored during the validation of this approach. Without these values this approach may not work well in practical cloud environments.

I. VMPACS

A nature inspired ACO approach [11] VMPACS is proposed. In this algorithm input parameters including a set of VMs, PMs with associated resource demand and the specific limit for resource utilization is given to the algorithm and a Pareto set P is output by the algorithm. The primary working VMPACS is almost identical to other nature inspired algorithm with a slight change for the use for pseudo-random proportional rule for VM placement. MGGA algorithm is selected for the comparison of VMPACS, as MGGA claims to be effective in multi objective VM placement problem. Following Pseudo-Random proportional rule is used by an artificial ant (*k*) to select a VM (*i*).

$$i = \underset{s}{\operatorname{argmax}}_{u \in \Omega_k(j)} \{ \alpha \tau_{u,j} + (1-\alpha) \eta_{u,j} \} \quad (11)$$

where α represents a control parameter for pheromone trail and q represents a uniformly distributed random number between 0 and 1, $\Omega_k(j)$ represents a set of VMs which qualifies for placement in PM (*j*).

VMPACS follows these steps to perform its operation:

- Parameters were initialized and value to all the pheromones trails are set to τ_0 .
- Each ant is set to receive all VMs request, and starts assigning VMs using pseudo-random proportional rule to active PMs. Once an ant finished building its path pheromone values are updated along with a global update of each Pareto set.

J. HCMFF and MFF

In [12] two energy effective approaches were proposed, Modified firefly algorithm (MFF) is implemented by modifying traditional firefly algorithms and hierarchical cluster based MFF (HCMFF) is implemented by modifying firefly algorithm with hierarchical clusters. Firefly approach uses male and female species of fireflies to populate its individuals. Both of these species can be assigned to different task, hence in proposed approach male fireflies represent PMs and females represent VMs. Both of these proposed algorithms were simulated and results were compared with traditional algorithms showing an effective reduction of energy consumption while maintaining SLA violations.

Following objective function is used to determine the attractiveness using the brightness of firefly.

$$x_j = PEnum_j \times PEmips_j \times VMbw_j \quad (12)$$

where $PEnum_j$ is the number of processors allocated to VM_j , $PEmips_j$ is total number of instructions of all processors in millions per second for each VM_j ,

MFF works in following steps:

- Once initialization of parameters an introductory population of fireflies was created.

- For all male and female fireflies light intensity is initialized, after this a firefly (*i*) is moved toward (*j*), with varying attractiveness of light due to distance, light intensity values were updates.
- Best solution is found by ranking fireflies.

HCMFF works with the same principal of MFF but with a slightly different approach.

By using a technique called hierarchical clustering, resources are clustered in terms of resource, bandwidth and memory.

- Every cluster behaves as a single resource in which VMs are categorized regarding requirements, after those firefly parameters were initialized.

- Firefly population is initialized and attractiveness of each individual is computed based on the objective function.
- Current solution is then found by ranking fireflies.

V. COMPARATIVE ANALYSIS

To compare selected energy aware VM consolidation approaches, the above mentioned techniques are compared side by side on the basis of multiple factors. Number of VMs and PMs used to simulate results are different in each proposed methodologies. A detailed comparative analysis is shown in Table II.

TABLE II. COMPARATIVE ANALYSIS OF ENERGY AWARE VM CONSOLIDATION TECHNIQUES

Technique	Compared With	PMs	VMs	Resource	Assessment Model	Selection Policy	SLAV	Energy(KwH)	VM Migrations
UP-VMC [3]	PUP-VMC VUP-VMC ACS-VMC SERCON MBFD MFFD	Load dependent	265	CPU Mem.	K-NNR	MMT	6.5x10 ⁻⁶	68	1700
						MinL	9.8x10 ⁻⁶	102	2800
						MaxL	1.5x10 ⁻⁵	70	2400
Utility Based [4]	Heuristic Based	100	150	CPU	Genetic algorithm	Self-Managing	2.5x10 ⁻⁵	90	1550
MFF [12]	Honey Bee	800	1052	CPU	Firefly algorithm	IQR-MC	9x10 ⁻⁶	34.77	888
						LR-MMT	1x10 ⁻⁴	35.33	867
						MAD-MMT	1.1x10 ⁻⁴	33.93	855
HCMFF [12]	HCT	800	1052	CPU	Firefly algorithm	IQR-MC	8x10 ⁻⁵	34.17	889
						LR-MMT	6x10 ⁻⁵	35.09	815
						MAD-MMT	1.6x10 ⁻⁴	32.91	873
EEVS [10]	MBFD	100	700	CPU	optimal frequency	random	-	10 (K-Watts)	-
AVVMC [5]	MMVMC VECTORGREEDY FFDL1Norm FFDVOLUME	100	1000	CPU MEM IO	Ant Colony System	Pseudo-random Proportional Rule	-	21 (K-Watts)	-
HCT [6]	Honey Bee	800	1000	CPU Mem.	Honey bee	IQR/RS	1x10 ⁻⁴	34.29	852
						LR/MU	8x10 ⁻⁵	36.85	869
MMC [7]	LR-MMT PAFDB	80	100	CPU	Minimum Correlation Coefficient	LR/MMT	7.81x10 ⁻⁵	11.69	-
ESWCT&ELMWCT [8]	Random iV-Value iDAIRS iVectorDot	100	500	CPU	-	ELMWCT	-	23 (K-Watts)	-
VMPACS [11]	MGGA	Load dependent	200	CPU Mem.	ACO	Random proportional rule	-	11.75 (K-Watts)	-
ACO [9]	FDD	30	100	CPU	ACO	Probabilistic decision rule	-	131.41	-

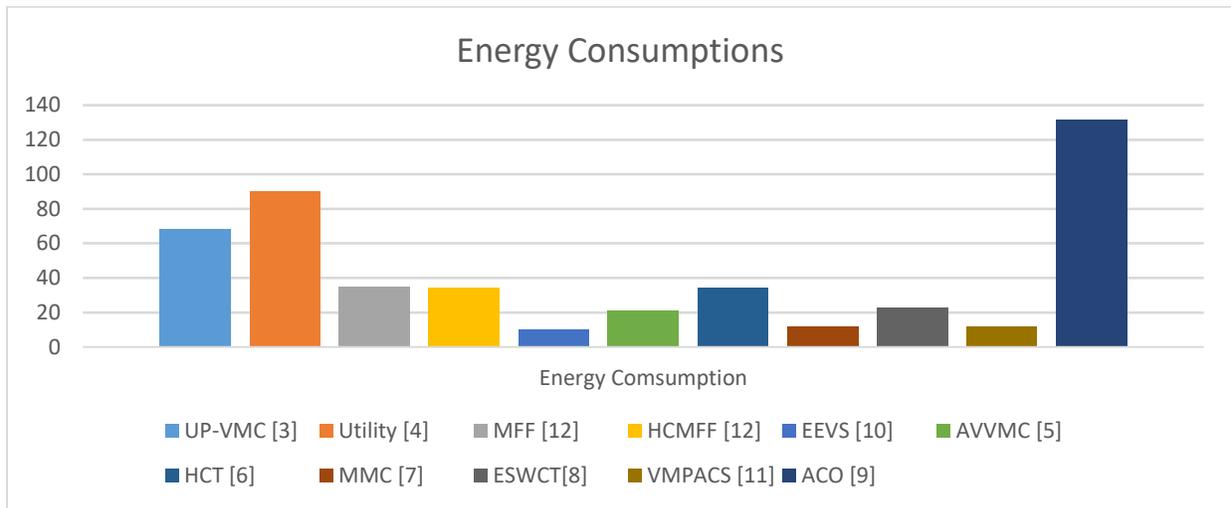


Fig. 1. Energy consumptions of multiple techniques in kWh.

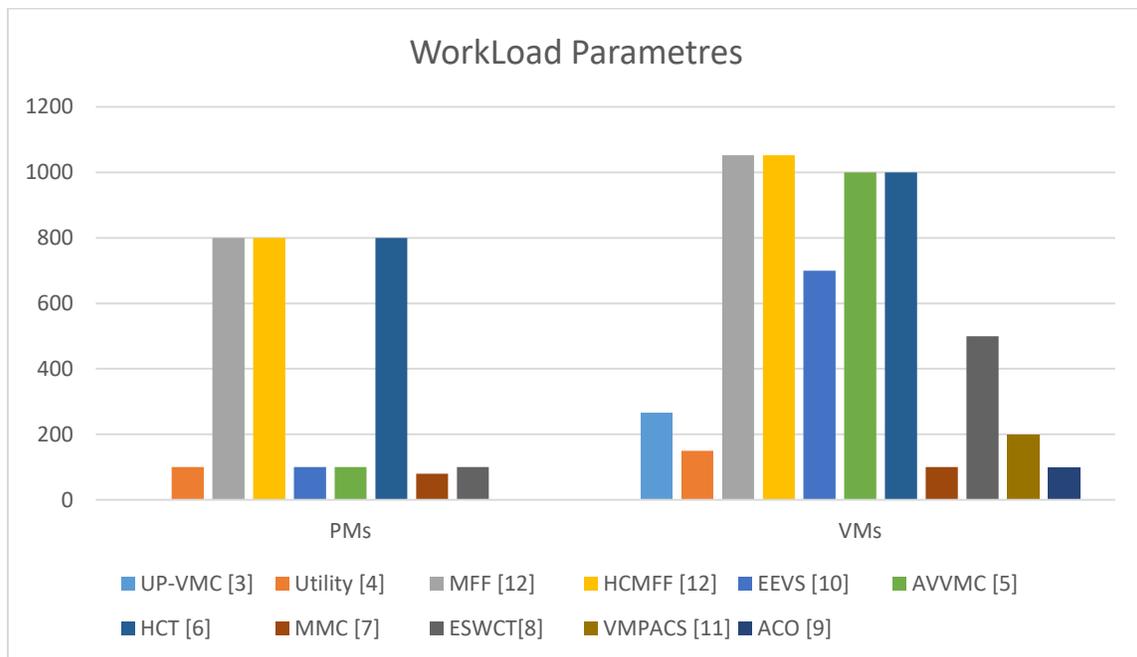


Fig. 2. Number of active PMs and VMs during workload testing.

Fig. 1 and 2 shows the amount of consumed energy by multiple energy aware techniques and the number of VMs and PMs used for the validation of results, respectively.

VI. CONCLUSION

As cloud computation applications are increasing daily, the number of servers used in cloud data centers are also increasing abruptly. Hundreds of thousands servers running in thousands of cloud datacenters around the world, consuming a large amount of energy. These vast power consumptions increase the volume greenhouse gases around the world. Hence a solution is required to reduce these energy consumptions to make cloud datacenters environmental friendly.

In this survey paper, multiples techniques for energy aware virtual machine consolidations are reviewed. All techniques have a primary goal of energy reduction with the minimum loss of cloud's applications efficiency. A technical analysis is given describing the proposed methodologies briefly, followed by a detailed comparative analysis.

Hence this survey paper will optimistically motivate future researchers to come up with smarter and secured optimal resource allocation algorithms and framework to strengthen the cloud computing paradigm.

REFERENCES

- [1] Shaden M.Allsmail,Review of Energy Reduction Techniques for Green Cloud Computing,2016.
- [2] Ankita Choudhary,A Critical Analysis of Energy Efficient Virtual Machine Placement Techniques and its Optimization in a Cloud Computing Environment,2015.

- [3] Fahimeh Energy aware VM Consolidation in Cloud Data Centers Using Utilization Prediction Model, IEEE transaction on cloud computing, 2016.
- [4] Abdelkhalik Mosa, Optimizing virtual machine placement for energy and SLA in clouds using utility functions, Journal of Cloud Computing: Advances, Systems and Applications, 2016.
- [5] Md Hasanul Ferdous et.al, Virtual Machine Consolidation in Cloud Data Centers using ACO Metaheuristic, 2014.
- [6] Ajith Singh, cluster based bee algorithm for virtual machine placement in cloud data centre, Journal of Theoretical and Applied Information Technology,2013
- [7] Negin Kord, An Energy Efficient Approach for Virtual Machine Placement in Cloud Based Data Centers,2013
- [8] LI Hongyou, Energy Aware Scheduling Scheme Using Workload Aware Consolidation Technique in Cloud Data Centres,2013.
- [9] Eugen Feller, Energy Aware Ant Colony Based Workload Placement in Clouds, 2011.
- [10] Youwei Ding, Energy efficient scheduling of virtual machines in cloud with deadline constraint, 2015.
- [11] Yongqiang Gao, A multi objective ant colony system algorithm for virtual machine placement in cloud computing,2013
- [12] Esha Barlaskar, Energy efficient virtual machine placement using enhanced firefly algorithm, 2016.
- [13] Zoha Usmani, A Survey of Virtual Machine Placement Techniques in a Cloud Data Center,2015