



OPTIMIZING CLOUD RESOURCE MANAGEMENT THROUGH ENERGY-EFFICIENT VM PLACEMENT TECHNIQUES

Dipak Dabhi^{*1}, Ankit Kharwar², Diya Vadhwani³, Pritesh Prajapati⁴,
Vishal Barot⁵, Anamika Mittal⁶, and Dhvani Modi⁷

^{1, 5, 6, 7}School of Engineering and Technology, Gujarat Technological University, Ahmedabad, Gujarat, India.

²Sarvajanik College of Engineering & Technology, Sarvajanik University, Surat, Gujarat, India.

³Adani institute of Infrastructure Engineering, Ahmedabad, Gujarat, India.

⁴Charotar University of Science and Technology, Changa, Gujarat, India.

¹<https://orcid.org/0000-0002-2190-8056>, ²<https://orcid.org/0000-0002-1933-4866>, ³<https://orcid.org/0000-0003-1495-3403>

⁴<https://orcid.org/0000-0002-6544-8514>, ⁵<https://orcid.org/0000-0002-9139-9541>, ⁶<https://orcid.org/0009-0003-8726-1114>

⁷<https://orcid.org/0009-0002-0563-8858>

Email: *ap_dipak_dabhi@gtu.edu.in, ankit.kharwar@scet.ac.in, diya.vadhwani@adaniuni.ac.in,
priteshpnp.007@gmail.com, vishal.barot@gtu.edu.in, ap_anamika@gtu.edu.in, ap_dhwani@gtu.edu.in

ARTICLE INFO

Article History

Received: March 13, 2026

Reviewed: April 16, 2026

Accepted: July 16, 2026

Published: August 31, 2026

Keywords:

Cloud Computing,
VM Consolidation,
Energy Efficiency,
SLA Violation,
VM Placement.

ABSTRACT

Cloud computing has seen rapid growth in recent years, leading to a surge in demand for data center services. To meet this demand, data centers deploy a large number of servers, resulting in substantial energy consumption. Virtual Machine Consolidation (VMC) is an effective strategy to reduce energy usage by shutting down underutilized servers while ensuring that Service Level Agreements (SLAs) are maintained. The VMC process comprises four key steps: detecting overloaded hosts, identifying underloaded hosts, selecting virtual machines (VMs), and determining their placement. This research presents the Energy-Efficient Virtual Machine Placement (EEVMP) approach, which aims to optimize resource utilization by selecting suitable destination hosts for migrating VMs based on utilization and resource skewness. The proposed method is evaluated using the CloudSim simulator. Experimental results show that EEVMP consistently outperforms existing placement strategies such as PABFD, IQRMC, PEBFD, ESVMP, and HVMAP in terms of energy efficiency and overall performance.



Copyright ©2026 by authors and Galileo Institute of Technology and Education of the Amazon (ITEGAM). This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

I. INTRODUCTION

Cloud computing has experienced significant growth in recent years, resulting in increased demand for data center services. To accommodate this demand, data centers operate a large number of servers, which in turn leads to high energy consumption. Virtual Machine Consolidation (VMC) is a widely adopted technique to minimize energy usage by turning off underutilized servers while still adhering to Service Level Agreements (SLAs) [1]. The VMC process involves four main stages: detecting overloaded hosts, identifying underloaded hosts, selecting virtual machines (VMs), and determining their optimal placement [2]. This study introduces the Energy-Efficient Virtual Machine Placement (EEVMP) strategy, designed to enhance resource utilization by selecting destination hosts for migrating VMs based on their utilization levels and resource imbalance. The approach is validated through simulations conducted using the CloudSim toolkit. Results indicate that EEVMP consistently outperforms conventional VM placement methods—namely PABFD, IQRMC, PEBFD, ESVMP, and HVMAP—by offering superior energy efficiency and system performance [3]. Figures 1 and 2 illustrate the number of active hosts before and after the application of the virtual machine consolidation method. Figure 1 shows that five hosts are needed to handle user requests; however, some hosts are significantly underutilized, while others are moderately utilized. Despite varying levels of utilization, all hosts consume considerable energy. To address this, some underutilized hosts can be put into sleep mode, and their VMs can be migrated to other hosts. Cloud data center resources are heterogeneous, meaning each can have a distinct physical configuration. Service providers may face challenges migrating VMs between hosts.

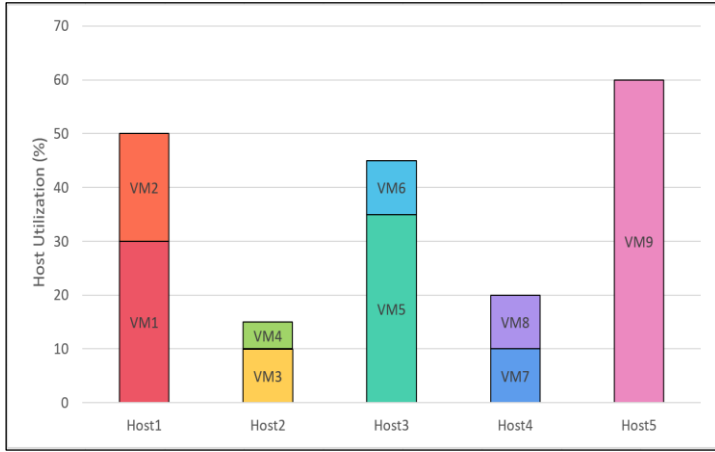


Figure 1: PM Utilization before VMC.
Source: Authors, (2026).

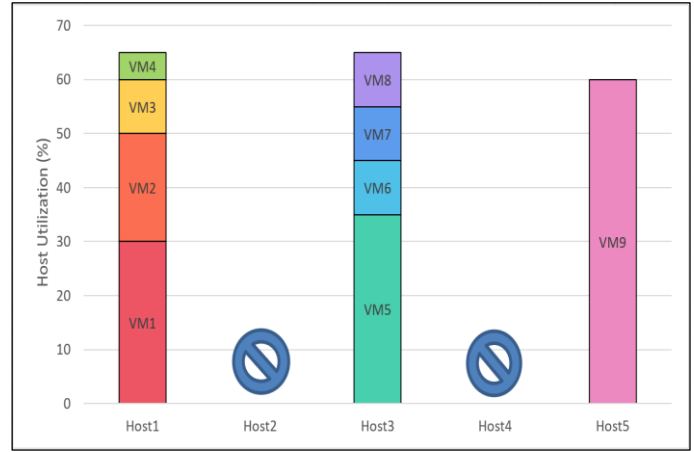


Figure 2: PM Utilization after VMC.
Source: Authors, (2026).

VMC resource allocation policies use various techniques to adaptively allocate the necessary VMs to minimize the number of active Physical Machines (PMs) while conserving energy. The primary objectives of VM consolidation are to avoid host overload and promote green computing. To prevent host overload, the VM resource request must be lower than the host's available resources. Otherwise, the Service Level Agreement (SLA) between the user and the service provider would be breached [4]. SLAs are crucial for ensuring Quality of Service (QoS) and service availability for end users. In terms of green computing, cloud service providers aim to fulfill user requests with the minimum number of active hosts possible without violating SLAs [5]. Figure 3 illustrates the classification of VM Consolidation Policies [6].

Host Overload Detection Algorithm	Static Threshold (Th)	Median Absolute Deviation (MAD)	Interquartile Range (IQR)	Local Regression (LR)	Robust Local Regression (LRR)
Host Underload Detection Algorithm	Host with Minimum Utilization				
VM Selection Algorithm	Minimum Migration Time (MMT)	Random Choice (RC)	Maximum Correlation (MC)	Minimum Utilization (MU)	
VM Placement Algorithm	First Fit (FF)	First Fit Decreasing (FFD)	Best Fit Decreasing (BFD)	Worst Fit Decreasing (WFD)	Power Aware Best Fit Decreasing (PABFD)

Figure 3: VMC Process Policy.
Source: Authors, (2026).

In a cloud data centre, multiple VMs may run on a single physical machine with host utilisation varying dynamically based on user demands; when consumption reaches 100%, performance degrades and SLAs are breached, making overload detection the first critical step in VM consolidation, followed by identification of the specific VMs responsible for the overload. Simultaneously, underutilised hosts must also be detected so their workloads can be transferred to other hosts, forcing them into standby mode to reduce overall energy consumption [7]. Once an overloaded host is identified, the offending VMs are selected, added to a migration list, and moved to alternative hosts to minimise performance deterioration, while all VMs from an underloaded host are migrated entirely [8]. The final step, VM placement, determines the optimal destination host for each migrating VM using a bin-packing-like algorithm; in this study, we developed the Energy-Efficient VM Placement for Optimized Resource Utilization in Cloud Data Centers (EEVMP) and benchmarked it against existing techniques including PABFD, IQRMC, PEBFD, ESVM, and HVMAP.

II. RELATED WORK

The VM workload consolidation approach is a promising research area for increasing cloud computing energy efficiency. Many strategies are advised to speed up the virtual machine consolidation process, including (a) detecting overloaded hosts, (b) detecting underloaded hosts, (c) selecting VMs, and (d) placing them. The VMC method focuses primarily on three criteria. The first is based on the Quality-of-Service approach, which emphasizes effective use of computer resources. The second option is to minimize data center power use by turning off underutilized hosts. Another critical challenge is maintaining a balance between energy use and SLA violations throughout the VMC process. By [1] offered an alternate technique for identifying overloaded/underloaded hosts, VM selection, and VM placement. The author created heuristic-based strategies for identifying overload hosts, including IQR, MAD, LR, LRR, and static thresholds. For Underload host identification, the least utilized host is deemed underutilized.

The author proposed the MC, MU, MMT, and RS algorithms for selecting VMs from overutilized hosts, as well as the PABFD policy for VM placement. Although it provides better power consumption outcomes than other standard approaches, average SLA violation and VM migration will be greater. In turn [9] improved their host overload detection system and migration control. After identifying an overloaded host, it calculates the VM prediction value using the host's mean and standard deviation, then checks to see whether the prediction is larger than one and if so, the host is labelled as overloaded. The results show that the proposed method beats the present methodology in terms of performance. According to [10] proposed many power-efficient VM placement approaches, including Medium Fit Power Efficient Decreasing (MFPED), PEBFD, and Power Efficient First Fit Decreasing (PEFFD). The proposed approach was evaluated on the OpenStack Neat cloud service platform. By [11] suggested a glowworm swarm optimization VM placement policy for reducing energy and SLA violations. By [12] discussed implementing a VM consolidation approach that is Quality-of-Service aware, employing the K-medoid interquartile range mid-range (KMIMR) technique. By [13] proposed two modified placement policies—ESVMP and HVMAP—for VM consolidation in cloud data centers.

III. PROPOSED WORK

The EEVMP (Energy-Efficient VM Placement for Optimized Resource Utilization in Cloud Data Centers) technique is introduced here, focusing on selecting the most suitable destination host based on utilization levels and host skewness. Skewness refers to the uneven usage of multiple resources on a server. A data center server comprises various resources such as CPU MIPS, Memory, and Bandwidth. In our proposed EEVMP algorithm, we evaluate three key resources (CPU, Memory, and Bandwidth) during the decision-making process. Let n be the number of resources considered, and r_i represent the usage of each resource. The resource skewness of a server is defined. Another parameter used is the Host Consumption, which averages the utilization of all server resources. When the power usage of the host surpasses a certain threshold, it is deemed overloaded. The usage of an overloaded host is indicated by P , which is the squared sum of its cloud resource consumption beyond the threshold. Let RE denote the total count of overloads on the server, and r_i indicate the overload threshold. Equations for CurrentSkewness and CurrentUtilization before VM migration are provided in Eq. (1) and Eq. (2), respectively.

$$CurrentSkewness = \sqrt{(\sum(r_i - \bar{r})^2 / n)} \quad (1)$$

$$CurrentUtilization = \sum(r - r_i)^2, \quad r \in R \quad (2)$$

It is crucial to ensure that the destination server does not become overutilized following the VM migration. Therefore, we must calculate the NewSkewness and NewUtilization of the server post-migration. The equations for determining NewSkewness and NewUtilization are provided in Eq. (3) and Eq. (4). The Skewness Difference (SkeDif) and Utilization Difference (UtilDif) must be lower than their respective difference values, and the server's NewUtilization must be below 65% utilization.

$$NewSkewness = \sqrt{(\sum(r_i + v_m - \bar{r})^2 / n)} \quad (3)$$

$$NewUtilization = \sum(r + v_m - r_i)^2, \quad r \in R \quad (4)$$

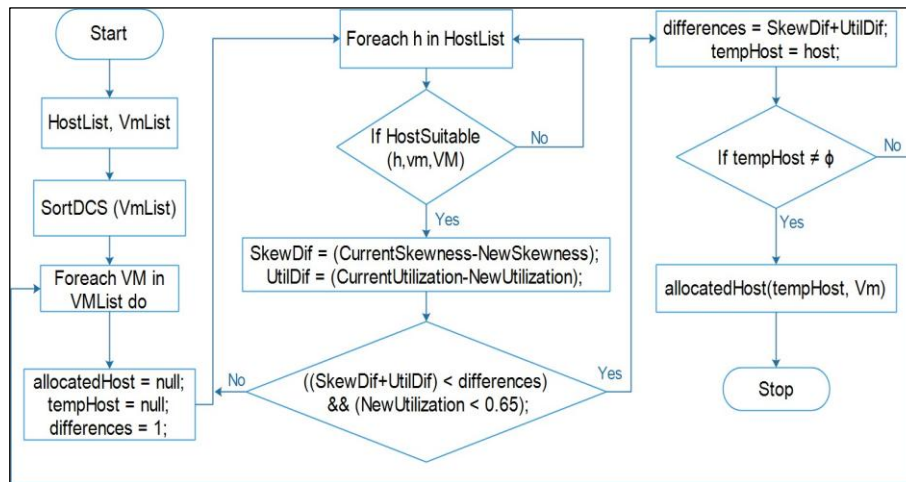


Figure 4: EEVMP Flowchart.

Source: Authors, (2026).

The proposed EEVMP flowchart is presented, highlighting the technique's use of host skewness and utilization before and after VM migration to efficiently identify destination hosts. The EEVMP placement policy first sorts all VMs in descending order based on CPU utilization for migration (Step 1). It then checks each VM in the VmList to find the destination host (Steps 2–5), and evaluates all hosts in the HostList as potential destination hosts (Step 6). If the chosen host has sufficient resources to accommodate the VM without causing overload, it calculates the CurrentSkewness and Utilization before VM Migration and the New Skewness and Utilization after VM Migration.

Algorithm 1 Energy-Efficient VM Placement (EEVMP)

Input: vmList, hostList
Output: allocatedHost

```

1) sort vmList in the order of decreasing CPU utilization;
2) foreach vm in vmList do
3)   allocatedHost = null;
4)   differences = 1;
5)   tempHost = null;
6)   foreach host in hostList do
7)     if host has enough resource for vm then
8)       CurrentUtilization = host.CurrentUtilization();
9)       CurrentSkewness = host.CurrentSkewness();
10)      NewUtilization = host.NewUtilization();
11)      NewSkewness = host.NewSkewness();
12)      UtilDif = Math.abs(CurrentUtilization-NewUtilization);
13)      SkewDif = Math.abs(CurrentSkewness-NewSkewness);
14)      if ((UtilDif + SkewDif) < differences) && (NewUtilization <= 0.65) then
15)        differences = UtilDif + SkewDif;
16)        tempHost = host;
17)      end if
18)    end if
19)  end
21) allocatedHost = tempHost;
22) return allocatedHost;
23) end

```

Figure 5: EEVMP Algorithm.

Source: Authors, (2026).

IV. EXPERIMENTAL SETUP

The simulations were conducted on an Acer computer equipped with a Core i5 CPU, 8 GB of RAM, Windows 10 64-bit operating system, a 125 GB SSD, and a 1 TB hard drive. The CloudSim application was utilized to simulate the specified activities. Our experiment was carried out in a data center comprising 800 physical heterogeneous hosts—400 HP ProLiant ML110 G4 and 400 HP ProLiant ML110 G5 [14]. Table 1 provides detailed specifications of the hosts [8]. Table 2 presents the VM configuration based on various speeds and RAM. Table 3 details the server power consumption, and Table 4 outlines the default implementation scenario.

Table 1: PMs Configuration.

HP ProLiant ML110 G4	1860	1	4	1000
HP ProLiant ML110 G5	2660	2	4	1000

Source: Authors, (2026).

Table 2: VMs Configuration.

High CPU VM	2500	1	870
Extra Large VM	2000	1	1740
Small VM	500	1	613
Micro VM	100	1	613

Source: Authors, (2026).

Table 3: Power Usage by PM at Different Load Levels (Watts).

HP ProLiant ML110 G4	86	89.4	92.6	102	114.2	117
HP ProLiant ML110 G5	93.7	97	101	110.8	123.7	135

Source: Authors, (2026).

Table 4: Implementation Scenario.

Hosts	800 (400 HP ProLiant ML110 G4 + 400 HP ProLiant ML110 G5)
VMs	1600 (mixed types)
Simulation Tool	CloudSim 3.0
Overload Detection	IQR-based
VM Selection	MMT (Minimum Migration Time)
VM Placement	EEVMP (proposed)
Dataset	PlanetLab (10 traces)
Evaluation Metrics	EC, VMM, SLAV, ESLAV, ESM, NHS

Source: Authors, (2026).

IV.1 DATASETS

The datasets were crucial in comparing performance. We utilized PlanetLab workload traces, which are publicly available real-world datasets, to evaluate our method's performance. PlanetLab serves as a functional cloud for the global research network. These traces include data on CPU consumption of numerous VMs from servers located in 500 different locations across the globe. The dataset traces were randomly collected between March 2010 and April 2020. CPU utilization values for VMs are recorded every 5 minutes in the daily dataset. Table 5 shows the specific properties of the dataset.

Table 5: PlanetLab Datasets.

20110303	42.3	15.2
20110306	38.1	12.8
20110309	40.5	13.9
20110322	44.7	16.1
20110325	41.0	14.5
20110403	52.3	18.7
20110409	45.8	16.5
20110411	45.2	16.0
20110412	39.8	13.6
20110420	38.6	12.4

Source: Authors, (2026).

IV.2 EVALUATION PARAMETERS

In cloud computing, measuring the quality of cloud services is crucial. In our tests, we assessed the algorithm's performance using six parameters : In cloud computing, measuring service quality is essential, and our algorithm's performance was assessed using six key parameters: Energy Consumption (EC), which calculates the total energy consumed by the data center over a given period based on varying host utilization levels; VM Migrations (VMM), which impacts application performance by causing an average CPU usage reduction of around 10% during downtime and must be minimized; SLA Violation (SLAV), which measures Quality of Service using two sub-metrics — Overload Time Fraction (OTF) and Performance Degradation due to Migrations (PDM) — expressed as $SLAV = SLATAH \times PDM$; Energy and SLA Violation (ESLAV), which balances EC and SLAV and is calculated as $ESLAV = EC \times SLAV$; Energy, SLA Violation and Migration (ESM), which jointly assesses total energy consumption, SLA violations, and migration impact as $ESM = EC \times SLAV \times VMM$; and finally, Number of Host Shutdowns (NHS), which counts the physical hosts powered down to conserve energy while maintaining acceptable performance levels within the data center [15].

$$SLATAH = (1/N) \sum (Ts_i / Ta_i) \quad (5)$$

$$PDM = (1/M) \sum (Cd_j / Cr_j) \quad (6)$$

$$SLAV = SLATAH \times PDM \quad (7)$$

$$ESLAV = EC \times SLAV \quad (8)$$

$$ESM = EC \times SLAV \times VMM \quad (9)$$

V. RESULTS AND DISCUSSIONS

The experiment was conducted based on six primary evaluation parameters: Total Energy Consumption (EC), VM Migrations (VMM), SLA Violation (SLAV), Energy and SLA Violation (ESLAV), Energy-SLA Violation and Migration (ESM) and the Number of Host Shutdowns (NHS), evaluated using PlanetLab datasets containing 10 days of data. Our proposed EEVMP strategy was compared against PABFD, IQRMC, PEBFD, ESVMP and HVMAP placement policies.

Table 6: Energy Consumption (EC) for PlanetLab Dataset.

20110303	177.10	119.10	114.97	109.41	112.92	84.56
20110306	135.49	90.73	86.15	78.10	84.72	61.43
20110309	156.64	103.89	99.87	91.91	97.95	68.70
20110322	191.16	123.00	118.43	107.58	116.32	82.82
20110325	165.80	107.00	102.75	88.36	101.38	69.88
20110403	245.78	161.86	157.70	144.44	154.77	108.42
20110409	196.13	127.67	123.53	117.53	121.61	87.62
20110411	193.26	125.25	120.20	115.31	118.34	83.78
20110412	168.31	110.66	106.13	95.61	104.27	70.94
20110420	147.37	92.41	88.11	81.04	86.17	58.51
Average	177.70	116.16	111.78	102.93	109.85	77.67

Source: Authors, (2026).

Table 6 highlights that EEVMP achieves the lowest energy consumption, with an average of 77.67, significantly reducing energy usage compared to PABFD (177.70, a 56.3% reduction), IQRMC (116.16, 33.1% reduction), and other policies, with the best energy efficiency observed on dataset 20110420 (58.51 vs. 147.37 for PABFD).

Table 7: Virtual Machine Migration (VMM) for PlanetLab Dataset.

20110303	23035	19405	15899	9208	17460	1663
20110306	18144	14196	11980	4948	12824	953
20110309	20762	17914	15006	8535	16906	1411
20110322	25566	24637	21531	12041	22924	1756
20110325	21760	20246	16992	5016	18271	1281
20110403	31659	26389	22811	11645	24952	1494
20110409	25526	22829	19272	12075	20966	1459
20110411	25039	21757	19006	12138	20909	1379
20110412	21956	19623	16681	6409	17874	1120
20110420	20491	19307	16540	12225	18159	1136
Average	23394	20630	17572	9424	19125	1365

Source: Authors, (2026).

Table 7 showcases a dramatic reduction in VM migrations, where EEVMP maintains an average of just 1,365 migrations, reducing them by over 94% compared to PABFD (23,394 migrations), with the most notable improvement in dataset 20110306, recording only 953 migrations versus 18,144 for PABFD.

Table 8: SLA Violation (SLAV) for PlanetLab Dataset.

20110303	0.00701	0.00579	0.00455	0.00398	0.00325	0.00437
20110306	0.00744	0.00548	0.00425	0.00470	0.00298	0.00220
20110309	0.00756	0.00660	0.00537	0.00490	0.00401	0.00311
20110322	0.00609	0.00570	0.00516	0.00587	0.00378	0.00280
20110325	0.00675	0.00596	0.00500	0.00576	0.00353	0.00264
20110403	0.00632	0.00577	0.00479	0.00617	0.00345	0.00195
20110409	0.00640	0.00566	0.00476	0.00518	0.00350	0.00240
20110411	0.00638	0.00551	0.00497	0.00461	0.00363	0.00239
20110412	0.00680	0.00568	0.00497	0.00600	0.00336	0.00211
20110420	0.00730	0.00757	0.00678	0.00447	0.00498	0.00269
Average	0.0068	0.0060	0.0051	0.0052	0.0036	0.0027

Source: Authors, (2026).

Table 8 shows EEVMP significantly minimizes SLA violations, achieving an average SLAV of 0.0027, marking a 60.3% reduction from PABFD (0.0068) and 25.0% lower than HVMAP (0.0036), with the lowest SLAV recorded in dataset 20110403 (0.00195).

Table 9: Energy and SLA Violation (ESLAV) for PlanetLab Dataset.

20110303	1.24124	0.68961	0.52304	0.43581	0.36697	0.36933
20110306	1.00766	0.49746	0.36611	0.36724	0.25217	0.13515
20110309	1.18491	0.68606	0.53651	0.45040	0.39316	0.21344
20110322	1.16470	0.70116	0.61144	0.63152	0.43919	0.23198
20110325	1.11981	0.63721	0.51381	0.50885	0.35806	0.18428
20110403	1.55417	0.93410	0.75551	0.89064	0.53466	0.21175
20110409	1.25560	0.72316	0.58758	0.60923	0.42601	0.21039
20110411	1.23285	0.69066	0.59794	0.53144	0.42999	0.20037
20110412	1.14410	0.62814	0.52771	0.57336	0.35015	0.14941
20110420	1.07579	0.69951	0.59707	0.36189	0.42878	0.15725
Average	1.19808	0.68871	0.56167	0.53604	0.39791	0.20634

Source: Authors, (2026).

Table 10: Energy, SLA Violation and Migration (ESM) for PlanetLab Dataset.

20110303	28591.9	13381.8	8315.8	4013.0	6407.3	614.2
20110306	18283.1	7061.9	4386.0	1817.1	3233.8	128.8
20110309	24601.1	12290.0	8050.9	3844.1	6646.8	301.2
20110322	29776.7	17274.4	13164.9	7604.1	10067.9	407.4
20110325	24367.1	12901.0	8730.6	2552.4	6542.2	236.1
20110403	49203.4	24649.9	17234.0	10371.5	13340.9	316.4
20110409	32050.4	16509.0	11323.8	7356.5	8931.8	307.0
20110411	30869.4	15026.6	11364.4	6450.7	8990.6	276.3
20110412	25119.9	12326.0	8802.7	3674.7	6258.6	167.3
20110420	22044.0	13505.4	9875.5	4424.1	7786.2	178.6
Average	28491	14493	10125	5211	7821	293

Source: Authors, (2026).

Table 11: Number of Host Shutdowns (NHS) for PlanetLab Dataset.

20110303	5439	976	1017	770	968	929
20110306	4324	919	958	777	926	783
20110309	4793	993	1039	784	985	788
20110322	5555	977	1036	772	963	778
20110325	5058	1014	1050	773	990	786
20110403	7050	958	1025	757	967	768
20110409	5776	979	1026	767	967	775
20110411	5715	995	1025	767	966	782
20110412	5132	991	1037	774	975	781
20110420	4454	971	1017	777	952	784
Average	5330	977	1023	772	966	795

Source: Authors, (2026).

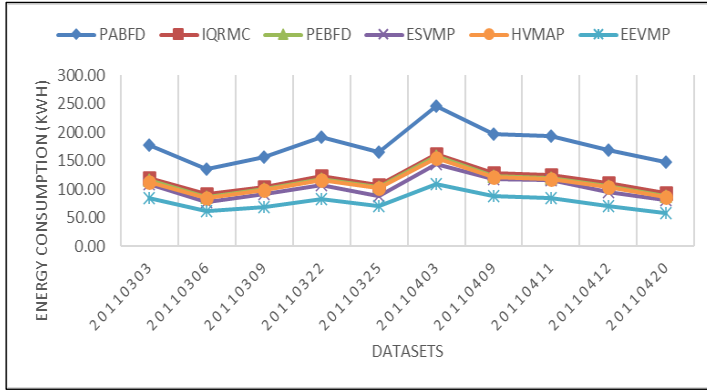
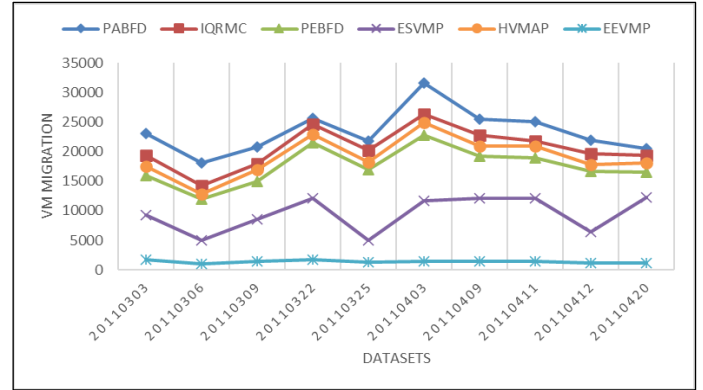
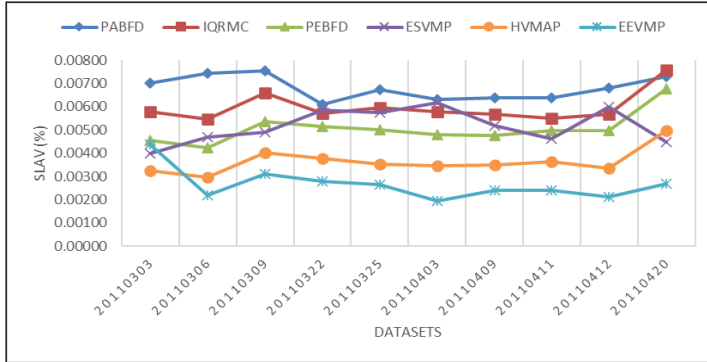
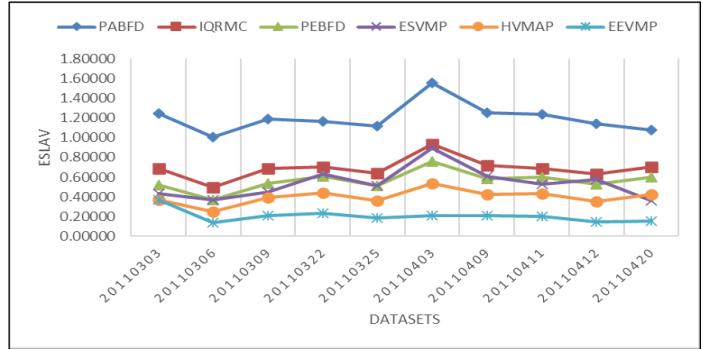
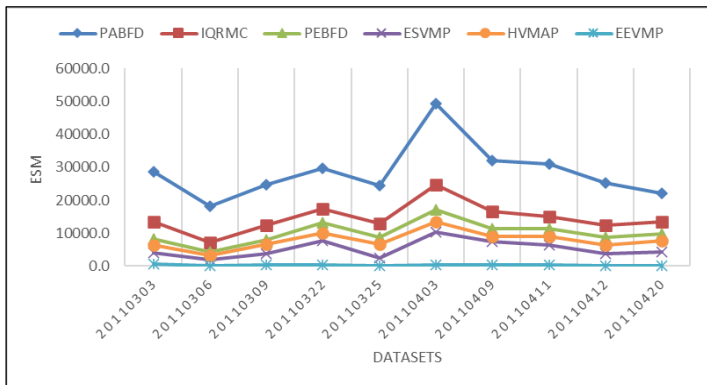
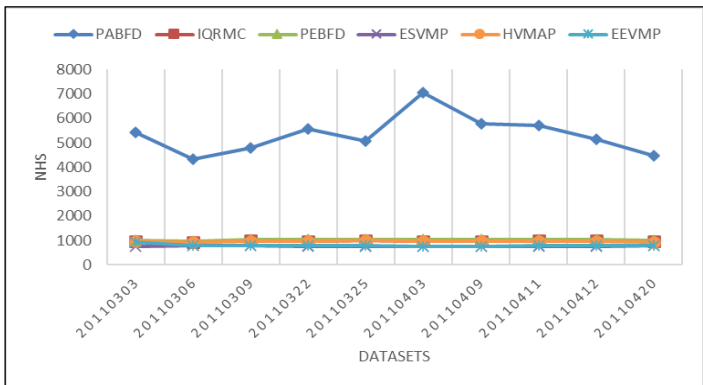
Figure 6: EC for PlanetLab Dataset.
Source: Authors, (2026).Figure 7: VMM for PlanetLab Dataset.
Source: Authors, (2026).Figure 8: SLAV for PlanetLab Dataset.
Source: Authors, (2026).Figure 9: ESLAV for PlanetLab Dataset.
Source: Authors, (2026).Figure 10: ESM for PlanetLab Dataset.
Source: Authors, (2026).Figure 11: NHS for PlanetLab Dataset.
Source: Authors, (2026).

Table 12: EEVMP Performance Comparison for ESLAV with Other Algorithms.

20110303	-108.274	-60.4907	-34.45	-16.5151	0.640808
20110306	-152.695	-114.543	-92.15	-92.3924	-60.4236
20110309	-138.945	-105.084	-86.16	-71.3888	-59.2541
20110322	-133.561	-100.558	-89.98	-92.5382	-61.7432
20110325	-143.476	-110.271	-94.41	-93.6526	-64.0854
20110403	-152.036	-126.081	-112.4	-123.167	-86.5232
20110409	-142.595	-109.855	-94.54	-97.3245	-67.7637
20110411	-144.079	-110.051	-99.6	-90.4818	-72.8548
20110412	-153.798	-123.14	-111.7	-117.314	-80.3685
20110420	-148.988	-126.584	-116.6	-78.8381	-92.6679
Average	-141.845	-108.666	-93.21	-87.3612	-64.5044

Source: Authors, (2026).

Further analysis of Table 12, which compares EEVMP's SLA Violation performance with other algorithms, underscores its superiority in handling SLA constraints. The negative values indicate the percentage reduction in ESLAV achieved by EEVMP. The average SLAV improvement across datasets demonstrates that EEVMP significantly reduces SLA violations. The average SLAV reduction achieved by EEVMP is 141.845 for PABFD, 108.666 for IQRMC, 93.21 for PEBFD, 87.3612 for ESVM, and 64.5044 for HVMAP, indicating that EEVMP surpasses all other algorithms in maintaining SLA integrity. These findings validate that EEVMP consistently achieves a lower SLA violation rate compared to traditional VM placement policies, thereby ensuring higher Quality of Service (QoS) and improved cloud resource management.

VI. CONCLUSIONS

The primary challenge in cloud computing is to reduce energy consumption and SLA violations while simultaneously meeting user demands. The evaluation of the proposed Energy-Efficient Virtual Machine Placement (EEVMP) strategy demonstrates its effectiveness in enhancing resource management and maintaining service quality in cloud data centers. The results consistently show that EEVMP outperforms existing VM placement policies, including PABFD, IQRMC, PEBFD, ESVM, and HVMAP, across multiple performance metrics. It significantly reduces energy consumption while ensuring efficient workload distribution, leading to lower operational costs and improved sustainability. Additionally, by minimizing virtual machine migrations, EEVMP reduces system overhead, thereby enhancing overall stability and performance. EEVMP demonstrates an improvement of 141.85% over PABFD, while improvements over IQRMC, PEBFD, ESVM, and HVMAP are 108.67%, 93.21%, 87.36%, and 64.50%, respectively. Future research will focus on further enhancing VM placement strategies by integrating AI-driven predictive analytics to improve energy efficiency and minimize SLA violations.

VII. AUTHOR'S CONTRIBUTION

Conceptualization: Dipak Dabhi, Ankit Kharwar, Diya Vadhwani, Pritesh Prajapati, Vishal Barot, Anamika Mittal, and Dhvani Modi.

Methodology: Dipak Dabhi, Ankit Kharwar, Diya Vadhwani, Pritesh Prajapati, Vishal Barot, Anamika Mittal, and Dhvani Modi.

Investigation: Dipak Dabhi, Ankit Kharwar, Diya Vadhwani, Pritesh Prajapati, Vishal Barot, Anamika Mittal, and Dhvani Modi.

Discussion of results: Dipak Dabhi, Ankit Kharwar, Diya Vadhwani, Pritesh Prajapati, Vishal Barot, Anamika Mittal, and Dhvani Modi.

Writing – Original Draft: Dipak Dabhi, Ankit Kharwar, Diya Vadhwani, Pritesh Prajapati, Vishal Barot, Anamika Mittal, and Dhvani Modi.

Writing – Review and Editing: Dipak Dabhi, Ankit Kharwar, Diya Vadhwani, Pritesh Prajapati, Vishal Barot, Anamika Mittal, and Dhvani Modi.

Resources: Dipak Dabhi, Ankit Kharwar, Diya Vadhwani, Pritesh Prajapati, Vishal Barot, Anamika Mittal, and Dhvani Modi.

Supervision: Dipak Dabhi, Ankit Kharwar, Diya Vadhwani, Pritesh Prajapati, Vishal Barot, Anamika Mittal, and Dhvani Modi.

Approval of the final text: Dipak Dabhi, Ankit Kharwar, Diya Vadhwani, Pritesh Prajapati, Vishal Barot, Anamika Mittal, and Dhvani Modi.

VIII. REFERENCES

- [1] A. Beloglazov, "Energy-efficient management of virtual machines in data centers for cloud computing," Ph.D. thesis, University of Melbourne, Australia, 2013.
- [8] A. Khosravi, L. L. H. Andrew, and R. Buyya, "Dynamic VM placement method for minimizing energy and carbon cost in geographically distributed cloud data centers," IEEE Trans. Sustain. Comput., vol. 2, no. 2, pp. 183–196, 2017.
- [9] M. Xu and R. Buyya, "Managing renewable energy and carbon footprint in multi-cloud computing environments," J. Parallel Distributed Comput., vol. 135, pp. 191–202, 2020.
- [10] Z. Á. Mann and M. Szabó, "Which is the best algorithm for virtual machine placement optimization?" Concurr. Comput. Pract. Exp., vol. 29, no. 10, 2017.
- [11] C. Zhang, Y. Wang, Y. Lv, H. Wu, and H. Guo, "An energy and SLA-aware resource management strategy in cloud data centers," Sci. Program., 2019, pp. 1–16.
- [14] H. Li, T. Li, and Z. Shuhua, "Energy-performance optimisation for the dynamic consolidation of virtual machines in cloud computing," Intl. J. Services Operations and Informatics, vol. 9, no. 1, pp. 62–82, 2018.
- [12] Y. Saadi and S. E. Kafhali, "Energy-efficient strategy for virtual machine consolidation in cloud environment," Soft Comput., vol. 24, no. 19, pp. 14845–14859, 2020.

- [13] R. Zolfaghari and A. M. Rahmani, "Virtual machine consolidation in cloud computing systems: Challenges and future trends," *Wirel. Pers. Commun.*, vol. 115, no. 3, pp. 2289–2326, 2020.
- [2] M. A. H. Monil and R. M. Rahman, "Implementation of modified overload detection technique with VM selection strategies based on heuristics and migration control," in *14th IEEE/ACIS Intl. Conf. on Computer and Information Science*, 2015, pp. 223–227.
- [3] F. F. Moges and S. L. Abebe, "Energy-aware VM placement algorithms for the openstack neat consolidation framework," *J. Cloud Comput.*, vol. 8, p. 2, 2019.
- [4] D. A. Alboaneen, H. Tianfield, and Y. Zhang, "Glowworm swarm optimisation algorithm for virtual machine placement in cloud computing," in *2016 Intl IEEE Conf. UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld*, 2016, pp. 808–814.
- [5] N. Khattar, J. Singh, and J. Sidhu, "An energy efficient and adaptive threshold VM consolidation framework for cloud environment," *Wirel. Pers. Commun.*, vol. 113, no. 1, pp. 349–367, 2020.
- [6] D. Dabhi and D. Thakor, "Energy and SLA-Aware VM Placement Policy for VM Consolidation Process in Cloud Data Centers," in *Sustainable Technology and Advanced Computing in Electrical Engineering*, 2022, pp. 351–365.
- [7] Q. Zhou, M. Xu, S. S. Gill, C. Gao, W. Tian, C. Xu, and R. Buyya, "Energy efficient algorithms based on VM consolidation for cloud computing: Comparisons and evaluations," in *20th IEEE/ACM Intl. Symposium on Cluster, Cloud and Internet Computing (CCGRID)*, 2020, pp. 489–498.
- [15] B. Shrimali and H. Patel, "Multi-objective optimization oriented policy for performance and energy efficient resource allocation in cloud environment," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 32, no. 7, pp. 860–869, 2020.