

RESEARCH ARTICLE

E2SVM: Electricity-Efficient SLA-aware Virtual Machine Consolidation approach in cloud data centers

Vaneet Kumar¹, Aleem Ali¹, Payal Mittal², Ibrahim Aqeel^{3*}, Mohammed Shuaib³, Shadab Alam³, Mohammed Y. Aalsalem³

1 Department of Computer Science and Engineering, Chandigarh University, Gharuan, India, **2** University Institute of Pharma Sciences, Chandigarh University, Gharuan, India, **3** College of Engineering and Computer Science, Jazan University, Jazan, Saudi Arabia

* iahmed@jazanu.edu.sa



OPEN ACCESS

Citation: Kumar V, Ali A, Mittal P, Aqeel I, Shuaib M, Alam S, et al. (2024) E2SVM: Electricity-Efficient SLA-aware Virtual Machine Consolidation approach in cloud data centers. PLoS ONE 19(6): e0303313. <https://doi.org/10.1371/journal.pone.0303313>

Editor: Jacopo Soldani, University of Pisa, ITALY

Received: January 15, 2024

Accepted: April 23, 2024

Published: June 10, 2024

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: <https://doi.org/10.1371/journal.pone.0303313>

Copyright: © 2024 Kumar et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All relevant data are within the manuscript.

Funding: The authors extend their appreciation to the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia, for funding this research work through the project number ISP-

Abstract

Cloud data centers present a challenge to environmental sustainability because of their significant energy consumption. Additionally, performance degradation resulting from energy management solutions, such as virtual machine (VM) consolidation, impacts service level agreements (SLAs) between cloud service providers and users. Thus, to achieve a balance between efficient energy consumption and avoiding SLA violations, we propose a novel VM consolidation algorithm. Conventional algorithms result in unnecessary migrations when improperly selecting VMs. Therefore, our proposed E2SVM algorithm addresses this issue by selecting VMs with high load fluctuations and minimal resource usage from overloaded servers. These selected VMs are then placed on normally loaded servers, considering their stability index. Moreover, our approach prevents server underutilization by either applying all or no VM migrations. Simulation results demonstrate a 12.9% decrease in maximum energy consumption compared with the minimum migration time VM selection policy. In addition, a 47% reduction in SLA violations was observed when using the medium absolute deviation as the overload detection policy. Therefore, this approach holds promise for real-world data centers because it minimizes energy waste and maintains low SLA violations.

Introduction

Cloud computing is an architecture that delivers a cost-effective, flexible, and shared pool of resources to its clients [1]. These resources can be hardware or software (e.g., memory, CPUs, bandwidth, services, and applications), and serve in a pay-as-you-go model. With this client's favorable features, cloud computing has become a popular computing platform for users [2]. Although edge computing reduces the computational load of cloud data centers (CDCs) even than emerging technologies powered by the cloud, such as blockchain, and the Internet of things (IoT), they generate heterogeneous data traffic for processing on CDCs [3, 4].

According to a report by Globe Newswire, cloud data centers are projected to experience a compound annual growth rate of 17.5%, i.e. from USD 0.37 trillion in 2020 to USD 0.83

2024. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

trillion by 2025 [5]. A statistical survey has shown that data centers consume 7% of the world-wide generated electricity, with an anticipated growth of 13% by 2030 [6]. Poor server utilization provokes energy consumption as an idle state server consumes approximately 70% of its peak power [7]. Another study indicates that servers typically operate at less than 50% of their maximum capacity [8]. Beloglezove et al. proposed an architectural framework for optimizing data center resources called VM consolidation.

This process is divided into four layers (1) underutilized server detection to put them in low power mode; (2) determining over-utilized hosts for smooth provisioning of requested resources; (3) selection of VMs for migration; and (4) assigning selected VMs to active or reactivated hosts [9]. Although VM consolidation maximizes resource utilization by shifting the workload from little-loaded servers and turning them to power-saving mode, it introduces a new problem of performance degradation with aggressive VM migrations that may raise SLA violations [10]. Existing studies work on a joint solution considering adaptive threshold (underload/overload detection) for maintaining low SLA violations and VM selection as well as VM placement for energy efficiency [11–13]. Nevertheless, none of the existing VM selection approaches consider a reduction in SLA violations while minimizing energy consumption.

A schematic diagram of energy consumption and SLA violation tradeoff is shown in Fig 1. The hashed arrow represents reduced energy consumption by minimizing active servers and reduced SLA violations by minimizing unnecessary migrations. To stabilize the tradeoff between SLA violations and energy consumption, third and fourth-layer policies for VM consolidation are developed.

The key contributions of our research can be summarized below:

- A novel Energy-Efficient SLA-aware Virtual Machine (E₂SVM) consolidation algorithm is designed to deal with energy-SLA tradeoff. This algorithm significantly reduces unnecessary migrations and thereby controls the reactivation of hosts.

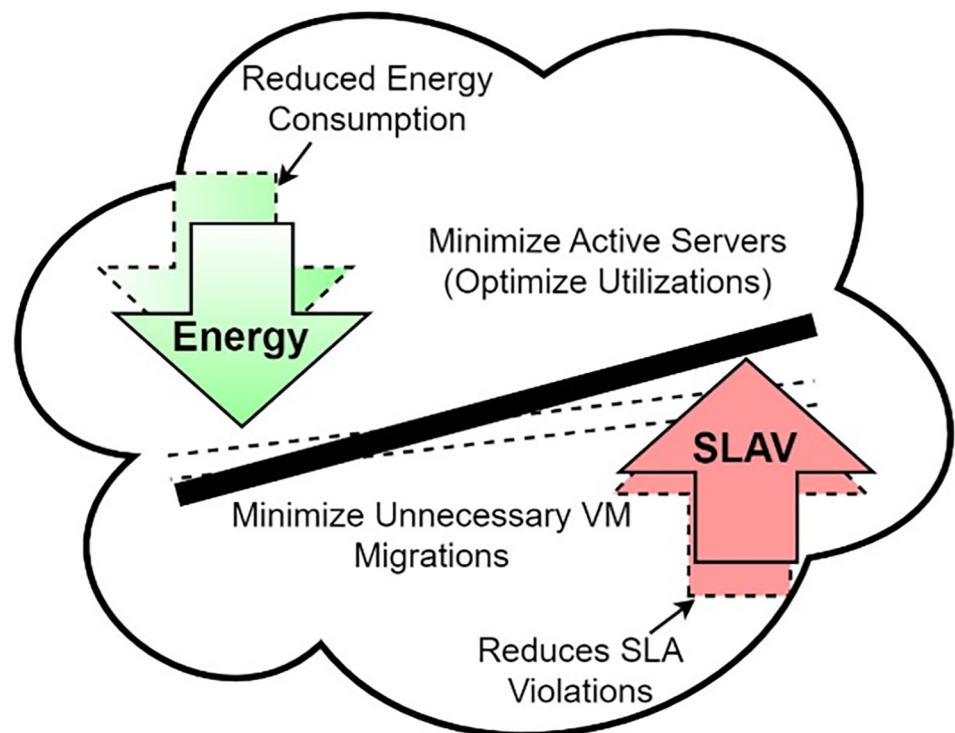


Fig 1. The trade-off between energy consumption and SLAV in cloud data center.

<https://doi.org/10.1371/journal.pone.0303313.g001>

- A dynamic Minimum utilization Maximum deviation (MuMd) algorithm is proposed to select VMS from an over-utilized server. The basic idea of this algorithm is to migrate the VMs to improve the stability of the consolidation process, thereby maintaining SLA.
- We introduced the MuMd aware Best Fit Decrease (MuMdBFD) VM placement algorithm. This algorithm maximizes resource utilization with optimal use of fluctuations in resource provisioning.

The efficiency of the proposed algorithm has been studied using computer-based simulations implemented in MatLab and exhibits its superiority over existing VM selection policies. The proposed algorithm is assessed on dynamic workload to check its validity. Compared with other energy-aware algorithms the proposed algorithm can handle unnecessary VM migrations with heavy load fluctuations and shows a maximum 27.3% reduction in the number of migrations.

The paper is structured as follows: related work section provides a comprehensive review of the literature, focusing on the research challenges concerning energy-efficient SLA-aware VM consolidation. Material and methods section details the application of the E2SVM algorithm. In Experiment Setup and Efficiency Matrices section, the cloud data center and diverse metrics used to evaluate the performance of the proposed method are discussed. Results and Discussion section presents the results obtained from the proposed study. Finally, Conclusion and Future Work section outlines the conclusions of this study and future research works.

Related work

The energy consumption is linearly proportional to CPU utilization, while the quality of Services (QoS) in terms of SLA is controlled by migration thrashing [14]. Although abundant studies have sought to address these issues, even then minimization of SLA violations, maximization of resource utilization, and energy efficiency are still challenging issues [15]. Authors [16] analyzed EC₂ instances and found that 20% of performance degradation is due to poor communication systems. Authors in [17] proposed adaptive-based algorithms focusing on past CPU utilization of allocated virtual machines to detect or predict overloaded hosts. Authors in [18] proposed load aware three gear threshold (LATHR) overloaded host detection algorithm to encircle the power of adaptive threshold and static threshold policies. Before these approaches, Beloglazov and Buyya proposed various adaptive and statistical methods to fix the upper threshold values required for the migration process [19]. In [20], the authors have designed multi-resource selection (MRS) VM consolidation using the concept of hottest resource to determine overloaded hosts. They introduced the concept of resource temperature and resourced correlation to determine potential VM for migration.

After detecting an overloaded host, VM selection has always been a part of VM consolidation to make an optimal decision. Beloglazov and Buyya proposed minimum migration time (MMT), maximum correlation (MC), and random selection (RS) VM selection policies. The RC policy selects VMs randomly to reduce the load from the host. The MMT approach selects the VM with least memory utilization among all the VMs from the overloaded host, taking minimum migration overhead. The MC approach selects VM with maximum correlation and proves more power saving but lacks QoS. In contrast, MMT provides better performance incurring more power.

Additionally, Zhou et al. [21] offered five VM selection rules and a three-threshold energy-saving algorithm (TESA) to enhance SLA. Furthermore, Zhou et al.'s [22] innovative flexible three-threshold energy aware (ATEA) VM placement method has decreased energy use and SLA violations. Minimum memory size and lowest CPU utilization constraints are taken for

designing VM selection policies. Monil and Rahman have introduced fuzzy logic-based energy-saving VM consolidation algorithms [23]. The standard deviation, memory size, and correlation have been used for defuzzification to select VMs for migration.

Further, Yadav et al. [24] have come up with an idea of 'M estimation regression' (MeReg) for managing the energy-SLA tradeoff. They used VM's CPU utilization history to detect the hotspot server to address this issue. In continuation, authors in [25] have proposed two adaptive heuristic models, gradient recent-based regression (Gdr) and bandwidth responsive algorithm for VM selection whereas maximum correlation percentage (MCP) to fix dynamic upper threshold. Noteworthy progress has been recounted regarding energy efficiency and QoS in the cloud data center (DC). Yadav et al. [26] have extended the research for over-utilized host detection using the least medial square regression (LmsReg) heuristics algorithm and introduced a utilization-based prediction method for VM selection. These heuristic algorithms significantly reduced energy consumption with improved SLA violations. This approach is high energy saving but lacks in maintaining low SLA violations. In the latest work by Mandal et al. [27], they introduced an approach aimed at relocating VMs with a dual objective: minimizing size while consuming high energy. While this approach demonstrates significant energy savings, it falls short in maintaining low SLA violations.

All the above approaches improved energy efficiency, focusing on unnecessary VM migrations by simultaneously overlooking fluctuations and current CPU utilization. Migrations of VMs with constant load can fall the DC in migration thrashing (MT) due to redundant migrations resulting in performance degradation. Similarly, heavily loaded VMs should not be migrated to achieve minimum live migration cost. As studied in the literature that requested workload is uncertain in the real world. Thus, utilization aware robust statistical method needs to be designed for handling dynamic consolidation.

Materials and methods

In this section, we will discuss the resource provisioning architecture of VM consolidation in a dynamic environment, which involves using virtual machines as shown in Fig 2 [28]. The main challenges of this architecture are achieving maximum resource utilization and delivering high-quality services. The quality of service is measured using SLA metrics. In order to optimize resource utilization, VMs with high remaining utilization and maximum deviation are migrated to hosts with high remaining utilization and minimum deviation [29].

The fundamental process of consolidating VMs to enhance resource utilization and guarantee QoS maintenance is shown in Algorithm 1.

ALGORITHM 1: ENERGY-EFFICIENT SLA AWARE VIRTUAL MACHINE (E2SVM) CONSOLIDATION ALGORITHM

Input: h_{list}, vm_{list}

Output: h_{chosen} : Detected destination host

1. for each host h in data center
2. for each VM on current host
3. Step-1: $Uthr_h = \text{MeanAbsoluteDeviation}(h_{list}) || \text{DoubleThreshold}$
4. Step-2: $MigVM_{list} = \text{MuMd}(Host_{over}, Uthr_h)$
5. Step-3: $h_{chosen} = \text{MuMdBFD}(H_{list}, MigVM_{list}, uth)$
6. end

The process begins with selecting the list of allocated VMs as input. Then, the workload on each VM is determined at regular time intervals. During each interval, overutilized and underutilized hosts are identified using MAD and THR policies. In the subsequent steps, we delve into the discussion of a dynamic workload-aware VM selection policy called Minimum Utilization Maximum deviation (MuMd) under subsection A. By migrating VMs with minimal utilization, energy efficiency can be enhanced. Additionally, by considering maximum deviation, we can control fluctuations in host status, thereby reducing the number of migrations. This

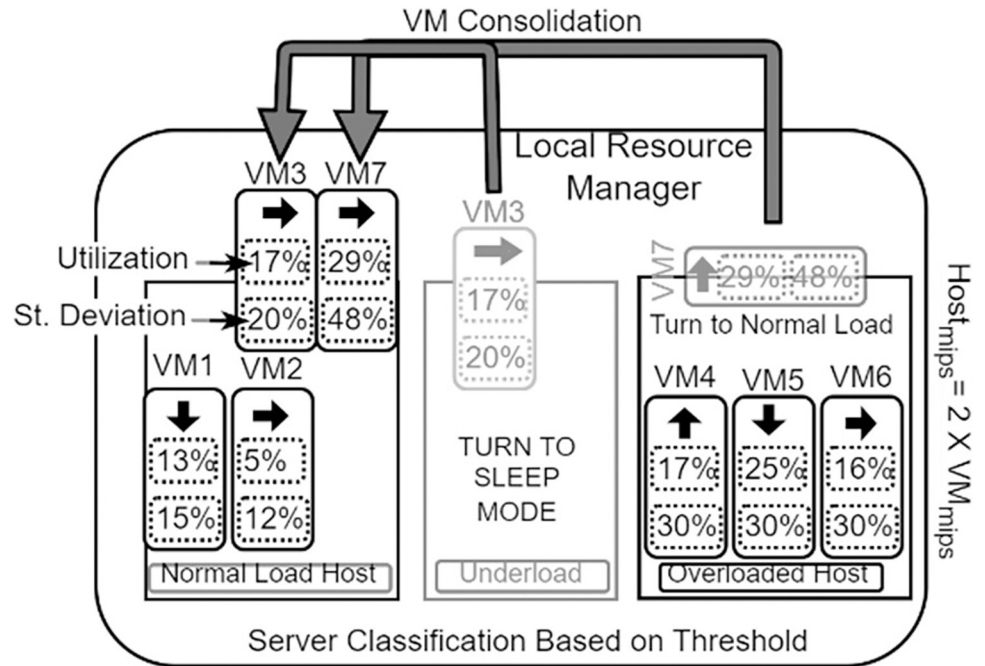


Fig 2. Multi-objective VM consolidation architecture.

<https://doi.org/10.1371/journal.pone.0303313.g002>

approach is important because VMs with high standard deviation and low current utilization on a host are more likely to cause host overload once again. Following this, the selected VMs from the previous step are mapped to suitable hosts utilizing the suggested VM placement algorithm discussed in subsection B. The combined behavior of these proposed algorithms promotes higher resource utilization, controlled migrations, and ultimately leads to improved SLA compliance and energy efficiency. Table 1 provides a list of notations used throughout this paper.

Proposed MuMd based VM selection algorithm

It has been proved that prudent migrations can optimize VM consolidation and save unwanted traffic load. Therefore, SLA-obedient energy-efficient VM selection is as important as overutilized and underutilized host detection. The suggested algorithm calculates the CPU fluctuations ratio with the summation of current CPU utilization and fluctuations individually for all the virtual machines using the following equation [30]. Further, one or more VMs with the maximum ratio is being selected for migration, explained using flowchart shown in the

Table 1. Notations.

V_h	Number of virtual machines
δ_v	Deviation in utilization of VM for last n iterations
L_v	Load ratio of current VM
$curr_uti_v$	Current utilization of VM v
$mips$	Million instructions per second
$CPU_{mips}(v_\tau)$	CPU utilization in mips at time τ
$aveCPU_{mips}(v)$	Average CPU utilization in mips of VM v

<https://doi.org/10.1371/journal.pone.0303313.t001>

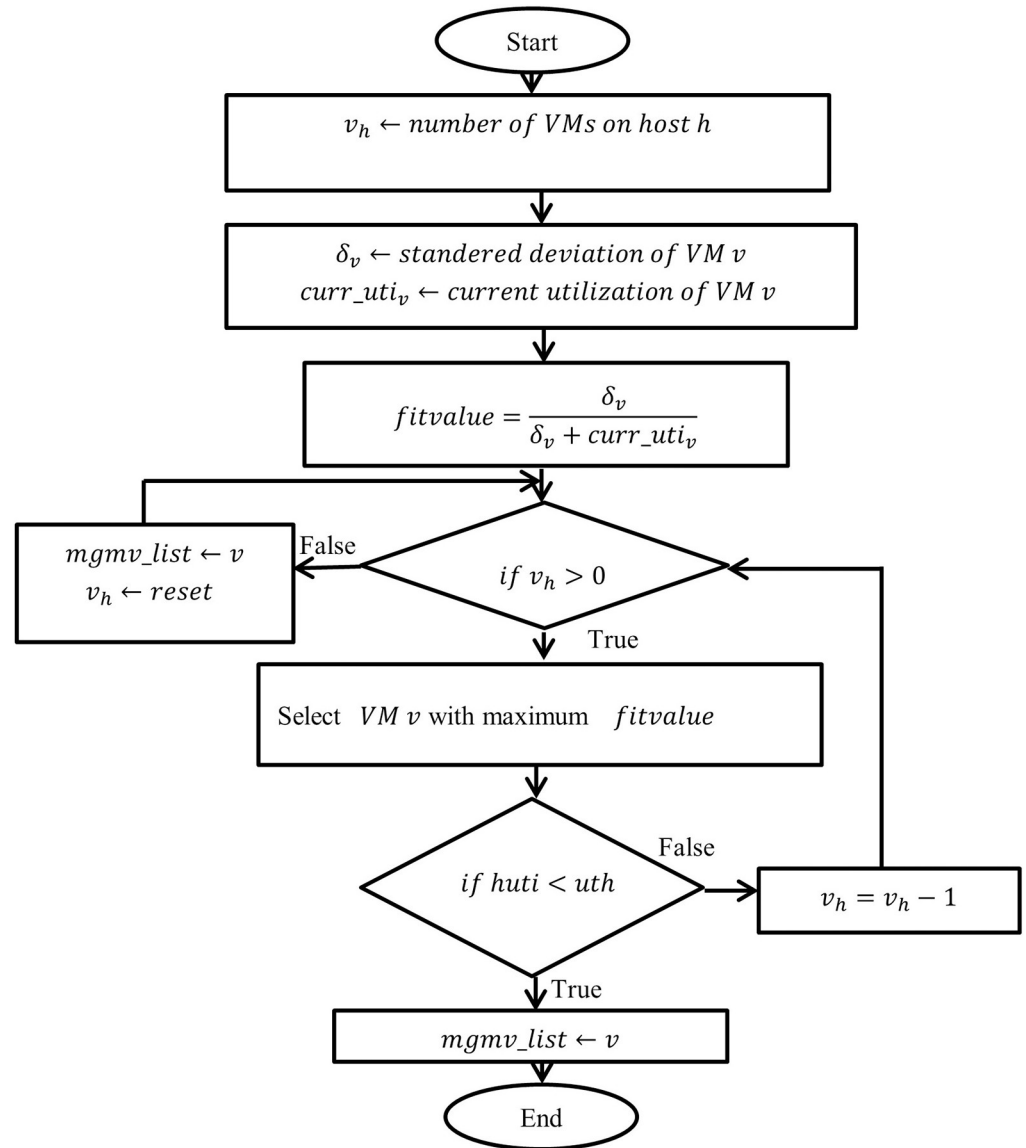


Fig 3. Flow chart of proposed technique (MuMd).

<https://doi.org/10.1371/journal.pone.0303313.g003>

Fig 3.

$$V_h = \{v | v \text{ is a VM allocated on the host } h\}$$

$$ratio = \frac{\delta_v}{\delta_v + curr_uti_v} \times L_v \tag{1}$$

$$L_v = \frac{curr_uti_v}{availability} \tag{2}$$

Where the set of allocated VMs on host h are represented by V_h . The current CPU utilization $curr_uti_v$, of virtual machine v is unitized according to standard deviation (δ_v) for last n iterations as referred in (3).

Let us consider $CPU_{mips}(v_\tau)$ representing the MIPS allocated to virtual machine v at time τ . The CPU utilizations of previous itr number of different time slots are denoted as $CPU_{mips}(v_{\tau-1}), CPU_{mips}(v_{\tau-2}) \dots CPU_{mips}(v_{\tau-itr})$ and $aveCPU_{mips}(v)$ is the average CPU utilization of virtual machine v .

$$\delta_v = \sqrt{\frac{1}{itr} \sum_{i=1}^{itr} (CPU_{mips}(v) - aveCPU_{mips}(v))^2} \quad (3)$$

Algorithm 1 shows the implementation of the proposed work in two phases. In the first phase, Line 5- Line 8 determine the CPU utilization ($huti$) of host in MIPS. As indicated by Line 10 onwards, in the second phase, one or more VMs are selected until the victim host changes its state to normal load, narrated in Fig 2. Line 13 is used to find the ratio of CPU utilization of individual VMs as represented in (1). The best virtual machine will be selected as indicated by Line 14–16 after finding the ratio of all allocated VMs. The proposed algorithm (MuMd) is represented in algorithm 1, and Fig 3 represented the flow of algorithm pictorially.

ALGORITHM 2 PROPOSED ALGORITHM (MUMD VMS SELECTION)

1. **Input:** VM_list from overloaded host h
2. **Output:** Selected VMs
3. $MigVM_list \leftarrow NULL$
4. $Huti \leftarrow 0$
5. $MaxRatio \leftarrow MIN$
6. $V_h \leftarrow$ Each and every virtual Machines on h
7. **for** each v in V_h **do**
8. $huti \leftarrow huti + v.utilization$
9. **end for**
10. **while** ($huti \geq Uthr$)
11. **for** each VM v in V_h **do**
12. $\delta_v \leftarrow Stdev(v)$
13. $ratio = \frac{\delta_v}{\delta_v + curr_uti_v}$
14. **if** $ratio \geq MaxRatio$ **then**
15. $MaxRatio \leftarrow ratio$
16. $v_{chosen} \leftarrow v$
17. **end if**
18. **end for**
19. $huti \leftarrow huti - CPU_{mips}(v)$
20. $MigVM_list \leftarrow v_{chosed}$
21. **end while**
22. **end**

The entanglement of the suggested algorithm depends upon the total number of hosts (M) and virtual machines (N), or $O(M+N)$.

Proposed MuMd based VM placement algorithm

VM placement is considered to be a bin packing optimization problem [31]. As bin packing is an NP-hard problem thus, placement for increasing idle hosts while considering intra VM dependency is a big challenge [32]. Power-aware best-fit decrease (PABFD) is the single prime objective VM placement algorithm. The flow of the proposed MuMd based Best Fit Decrease (MuMdBFD) VM Placement policy is discussed as pseudo-code in algorithm 2. The algorithm reinforces the full flow of VM allocations to the suitable servers and depicts that the system restricts SLA violations with controlled migration thrashing, hence reducing energy consumption.

ALGORITHM 3: MUMD BASED BEST FIT DECREASE ALGORITHM (MUMDBFD)

Input: H_{list} : all hosts, vm : a new virtual machine, uth : upper threshold

Output: h_{chosen} : the selected destination host

1. $h_{chosen} \leftarrow null$
2. $hratio \leftarrow 0$
3. $MinRatio \leftarrow MAX$
4. Choose cluster c
5. **for** each host h in HC_c **do**
6. $huti \leftarrow 0$
7. $V_h \leftarrow$ All virtual Machines on h
8. **for** each v in V_h **do**
9. $huti \leftarrow huti + v.utilization$
10. $hratio = hratio + \frac{\delta_v}{\delta_v + curr_uti_v}$
11. **end for**
12. **if** $huti > 0$ **then** // check active/inactive state of host
13. $utilization \leftarrow huti + v.utilization$
14. **if** $utilization > 1$ **then**
15. *continue*
16. **else if** $utilization \leq uth$ **then**
17. $ratio = hratio + \frac{\delta_{vm}}{\delta_{vm} + curr_uti_{vm}}$
18. **if** $ratio \leq MinRatio$ **then**
19. $MinRatio \leftarrow ratio$
20. $h_{chosen} \leftarrow h$
21. **endif**
22. **endfor**
23. Return: h_{chosen}

In Algorithm 2, Line 8, CPU utilization ($huti$) of the host is determined. Line 9 determines the CPU utilization to standard deviation ratio ($hratio$) for each virtual machine (VM) allocated to host h . The purpose of obtaining $hratio$ is to identify a suitable host where the unsettled CPU utilization is maximized for VM placement. Overall deviation is minimum among all the hosts as described by line 16-line 18. After all iterations algorithm returns the best host with high remaining utilization and comparatively minimum deviation.

Theorem 1: The time complexity of the algorithm $O(M \times N)$ and depends upon the total number of hosts (M) and virtual machines (N).

Proof: E₂SVM algorithm comprises of three steps in sequence. Initially, adaptive threshold is determined for overload server detection in $O(M \times k)$ time, where k is the number of virtual machines on a server. In the second step VM is selected in $O(M + N)$ time. Finally, VMs in migration list are allocated or placed in suitable host in $O(M \times N)$ time. Therefore, the overall time complexity of the proposed algorithm is $O(M \times N)$.

Experiment setup and efficiency matrices

In this section, we will explain the simulation setup of a cloud data center for comparison and evaluation of existing algorithms with proposed VM selection approach. Thereafter, basic definitions (includes energy consumption models, energy efficiency metric, and SLA violation metrics). In the end, results and discussion are presented in the next section. Table 2 below provides list of abbreviation and notations used to define the different efficiency matrices used in this study.

Modeling of physical machines and VM requests

The experimental setup to run application on IaaS environment, involves configuring N virtual machines $\{V = (v_1, v_2, \dots, v_N)\}$ and M servers $\{S = (s_1, s_2, \dots, s_M)\}$. Each host h is delineated by its processing capacity ($CPU_{mips}(h)$) in MIPs and PC_h as its power usage is measured in watts per second. Likewise, the processing capacity of each virtual machine, represented by $v \in (v_1, v_2, \dots, v_N)$, is characterized by $CPU_{mips}(v)$. we simulate a data center with 290 virtual machines

Table 2. Abbreviations and notations.

H_h	Set of M hosts
$CPU_{mips}(h)$	CPU utilization of host h in $mips$
$CPU_{mips}(v)$	CPU utilization of VM v in $mips$
PC_h	Power consumption of host h
PC^{max}	Power consumption at full utilization
$CPU_{mips}(h_\tau)$	CPU utilization of host in $mips$ at time τ
EC_h	Energy consumption by host h
TEC	Total energy consumed by data center
a_{vh}	Allocation of v on host h .

<https://doi.org/10.1371/journal.pone.0303313.t002>

and 100 physical hosts of heterogeneous configuration as mentioned in Tables 4 and 5.

Modeling of energy consumption

The CPU utilization, measured in MIPS, is recognized as a crucial factor in governing energy consumption because the power consumption (PC_h) of a server is directly proportional to its CPU usage, as indicated in Eq 4 [33]. To accurately describe the energy consumption, we use real data on energy consumption, all of which derive from the SPECpower benchmark [34]. For the experimental study of the proposed work, the HP G4 server is utilized and details of energy consumption is illustrated in Table 3. The maximum power (PC^{max}) consumed by the HP G4 server amounts to 117 watts [35]. It is observed that, when a server remains idle, it still consumes approximately 70% of its total power. Consequently, underutilized servers consume more power per instruction (about 0.7 times) compared to fully utilized servers.

$$C. PC_h = \begin{cases} 0.7 \times PC^{max} \times CPU_{mips}(h) + 0.3 \times PC^{max} & , CPU_{mips}(h) > 0 \\ 0 & , Otherwise \end{cases} \quad (4)$$

Due to the dynamic nature of workloads, the CPU workload of a host (h) differs over time (τ), which can be represented as $CPU_{mips}(h_\tau)$. As a result, the energy usage (EC_h) over a given time period can be considered an inherent factor of power consumption.

$$D. EC_h = \int_{\tau_0}^{\tau_1} PC_h(CPU_{mips}(h_\tau)) dt \quad (5)$$

SLA Metric

Beloglazov and Buyya provided an explanation for SLA violations within the context of the Infrastructure as a Service (IaaS) model, as demonstrated in Eq (6). They introduced the concept of SLATH, which represents the time duration during which SLA violations occur on an active host operating at maximum CPU utilization. Additionally, they introduced PDM as an abbreviation for Performance Degradation by VM resulting from live migrations.

$$SLAV = SLATH \times PDM \quad (6)$$

Table 3. Power consumption by the two servers at different load levels in watts.

Server Utilization	0%	20%	40%	60%	80%	100%
Power Consumption	86	92.6	99.5	106	112	117

<https://doi.org/10.1371/journal.pone.0303313.t003>

IER Metric

Instruction Energy Ratio is another metric that consider both the performance and energy consumption in Million instructions per joule as referred in (7).

$$IER(\mathcal{M}) = \frac{\sum_{h=1}^M \sum_{v=i}^N (a_{vh} \cdot CPUmips(v))}{TEC} \quad (7)$$

This study's primary objectives are to increase the number of instructions carried out per unit of energy usage and reduce SLA breaches.

Results and discussion

The effectiveness of the proposed algorithm is evaluated through computer-based simulation conducted in Matlab version R2017a. The specifications of the hosts and virtual machines, obtained from prior studies, are presented in Tables 4 and 5. To ensure result accuracy, the data center undergoes ten iterations during the simulation. The validation of the findings is done by examining SLA violations and energy efficiency metrics. Additionally, the previous section on Experiment Setup and Efficiency Matrices discusses the utilization of IER to demonstrate energy savings per instruction

The effectiveness of the proposed algorithm is evaluated against existing VM selection policies, namely minimum migration time (Mmt), maximum correlation (Mc) [36], and maximum utilization minimum size (MuMs) [37]. In the Mmt policy, the VM with the least remaining time to complete migration is chosen. Conversely, in the Mc policy, the VM with the highest correlation among others on the oversubscribed server is selected for migration to alleviate the workload below the threshold level. In our simulations, we utilize the medium absolute deviation (MAD) and interquartile range (Iqr) policies to establish threshold levels.

The proposed approach reduces energy consumption by an average of 9.9% and 9% compared to the Mad and Iqr algorithms, respectively, as shown in Figs 4 and 5. Fig 6 depicts the energy consumption in joules per million instructions using IER. According to the interpretation, the proposed approach demonstrates a significant 13.3% improvement in the instructions performed per unit of energy.

SLA violations are a critical concern for cloud service providers because financial penalties may be imposed by users for the duration of the performance degradation. SLA is considered to be a standard parameter to determine the quality of service. The proposed approach in the present study reduces an average 10% SLA violations as shown in Fig 7.

After selecting the victim VMs, it needs to be migrated to other suitable host machines. Thus, migration of virtual machines also degrades the desired performance level and imparts processing overhead to the entire system. Fig 8, reveals that the suggested algorithm has a significantly lower number of migrations than MM and RC algorithms. In order to place

Table 4. Specifications of data center.

Number of Hosts	Number of Virtual Machines	Data Center
290	100	1

<https://doi.org/10.1371/journal.pone.0303313.t004>

Table 5. Specifications of machines.

Type	RAM (MB)	CPU	MIPS
Host	8192	1	1000–3000
Virtual Machine	128	1	250–1000

<https://doi.org/10.1371/journal.pone.0303313.t005>

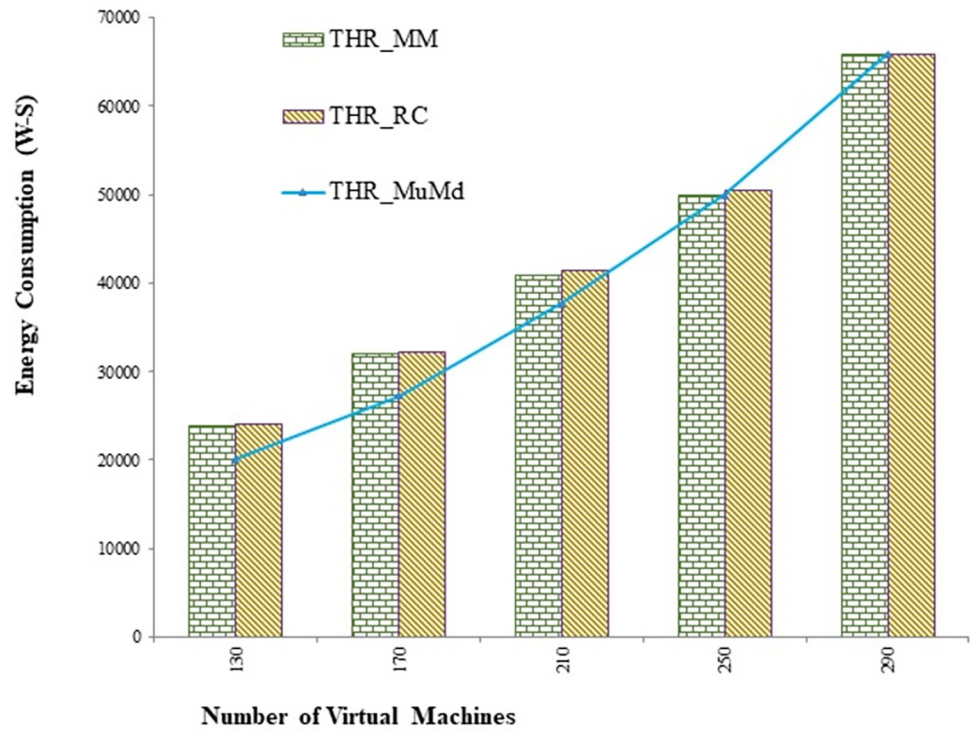


Fig 4. Energy usage with varying VMs for double threshold (THR) as overload detection policy.

<https://doi.org/10.1371/journal.pone.0303313.g004>

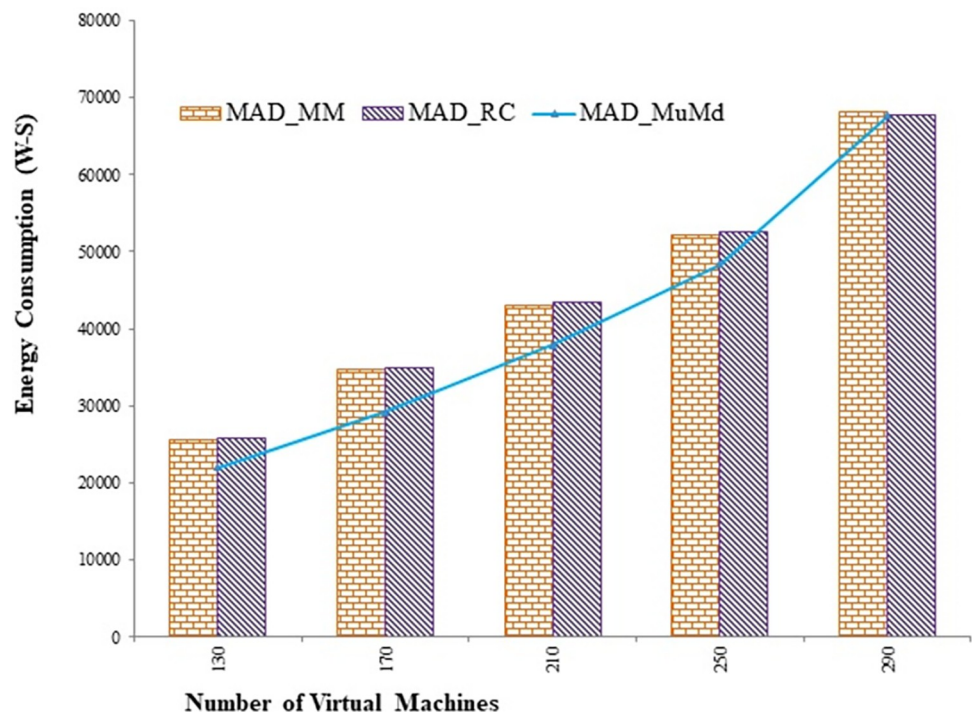


Fig 5. Energy usage with varying VMs for Median Absolute Deviation (MAD) as overload detection policy.

<https://doi.org/10.1371/journal.pone.0303313.g005>

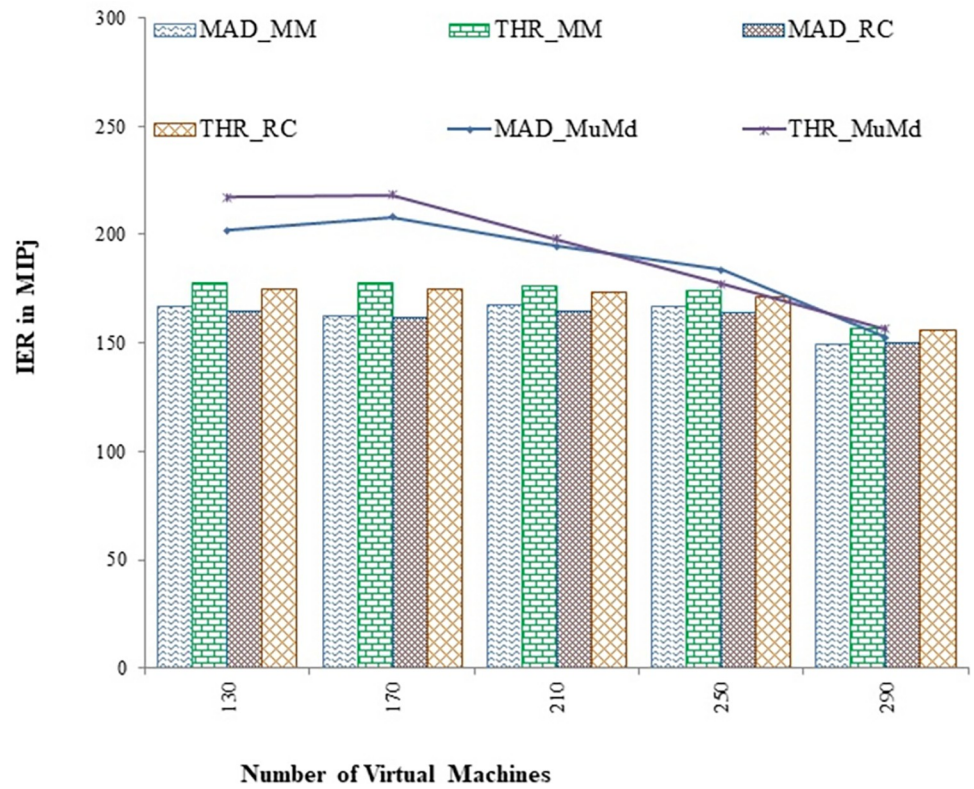


Fig 6. Change in IER for MAD and THR as overload detection policies v/s increasing virtual machines.

<https://doi.org/10.1371/journal.pone.0303313.g006>

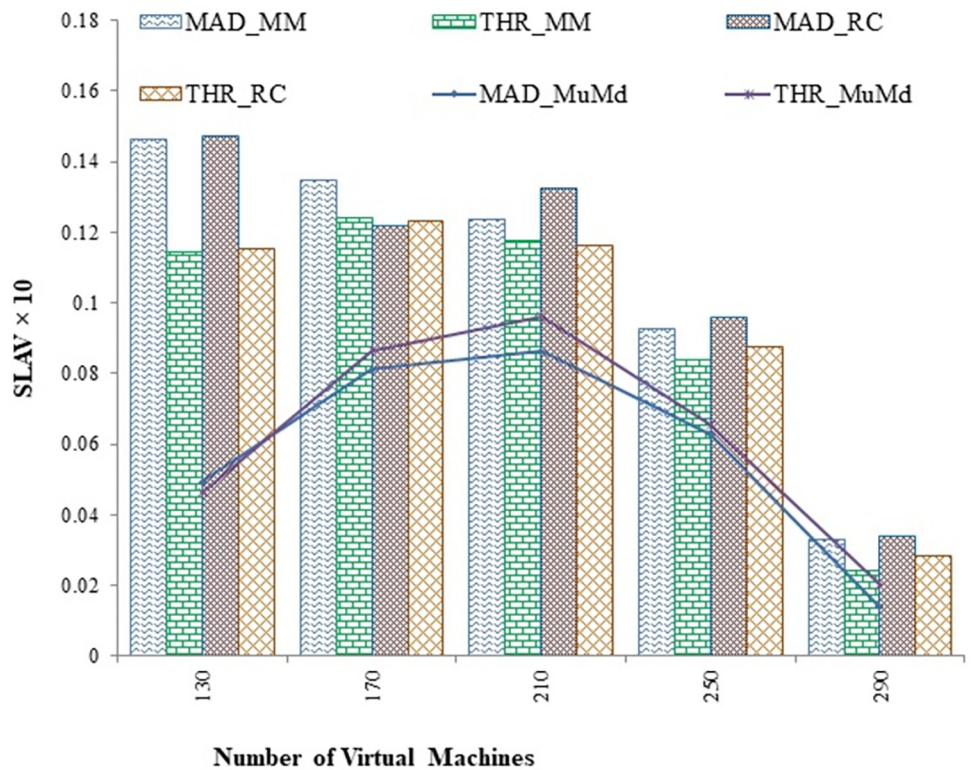


Fig 7. SLA Violations for MAD and THR as overload detection policy v/s virtual machines.

<https://doi.org/10.1371/journal.pone.0303313.g007>

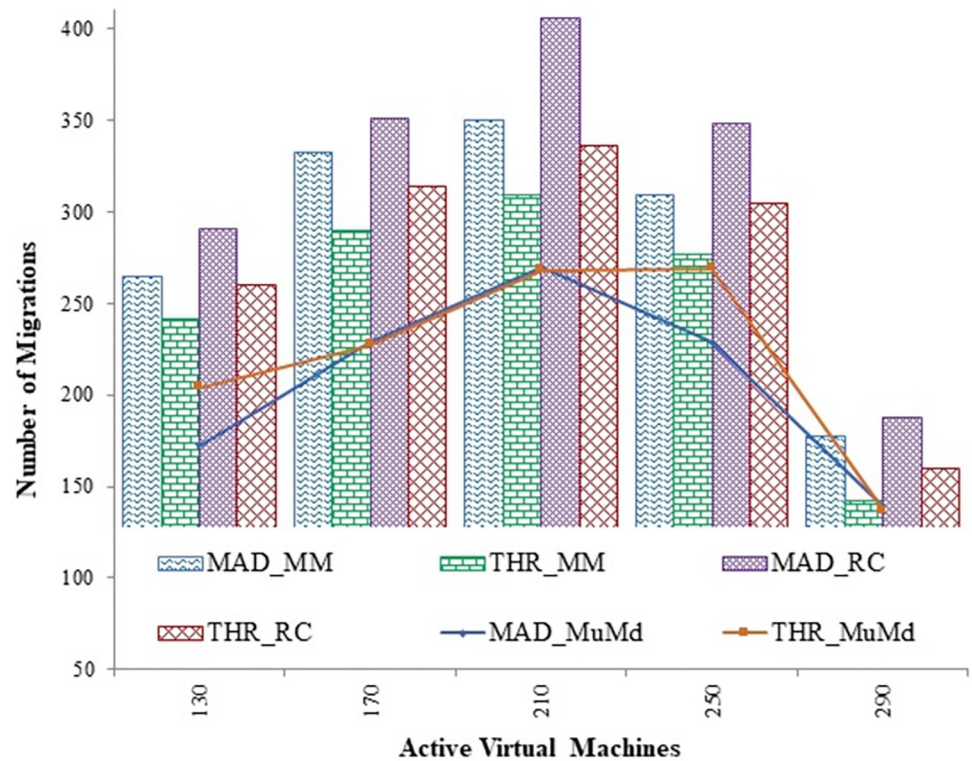


Fig 8. Change in migrations count for MAD and THR as overload detection policies against increasing virtual machines.

<https://doi.org/10.1371/journal.pone.0303313.g008>

underloaded hosts in a low-power mode, all virtual machines (VMs) residing on those hosts are migrated. After the installation of migrable VMs, Fig 9 compares the number of idle hosts that are still present. The results demonstrate that the proposed technique exhibits superior performance compared to the MAD and THR algorithms, with reductions of 10.4 percent and 6.2 percent in idle hosts, respectively.

Results in Fig 10 reveals the improvement of overall percentage of suggested algorithm (MuMd) with Minimum Migration (MM) and Random Choice (RC) VM selection algorithms using Median Absolute Deviation (MAD) and Double Threshold (THR) as host overload detection policies. It can be clearly noted that the proposed scheme offers improvement for energy consumption, idle servers, migrations count, instruction energy ratio and service level agreement.

Conclusion and future work

In this paper, we proposed an Energy Efficient SLA-aware Virtual Machine Consolidation (E2SVM) algorithm. Initially, we employed existing Median Absolute Deviation (MAD) and Interquartile Range (IQR) policies for detecting overloaded servers. Subsequently, victim virtual machines were selected based on fluctuations using the last 10 historical utilizations and current resource requirements. The proposed Minimum Utilization Maximum Deviation (MuMd) approach considered both past CPU utilization and current resource requirements simultaneously.

This study addressed two critical challenges in the IaaS cloud platform: reducing energy consumption and preventing server overutilization to minimize SLA violations. We

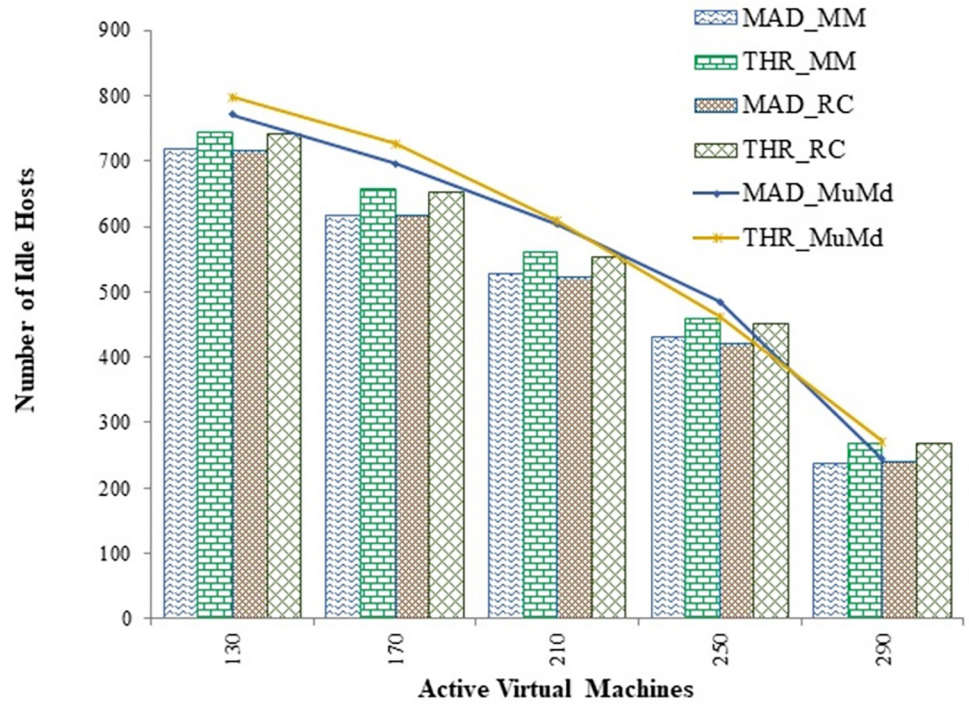
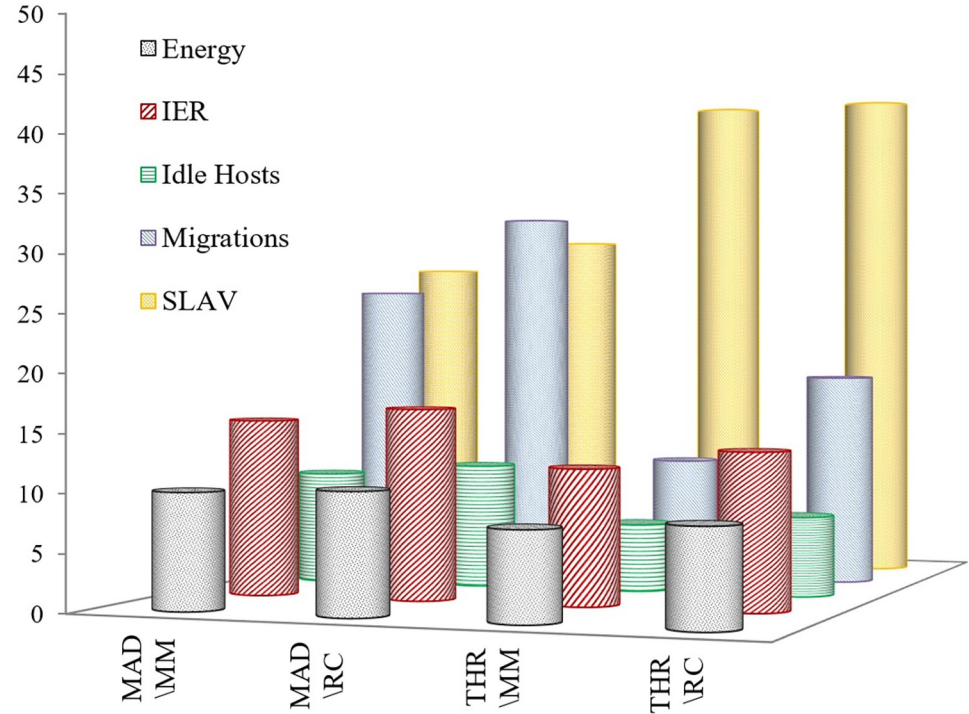


Fig 9. Controlling idle servers for MAD and THR as overload detection policies against increasing virtual machines.

<https://doi.org/10.1371/journal.pone.0303313.g009>



MuMd vs Others

Fig 10. Percentage improvement in various metrics using proposed algorithm compared with existing policies.

<https://doi.org/10.1371/journal.pone.0303313.g010>

implemented the proposed approach using computer-based simulations and evaluated it in Matlab version 2017a. The simulation results demonstrated significant improvements compared to existing benchmark algorithms, including an average reduction of 18.9% in the number of migrations, 12.7% in idle hosts, and a 9.5% decrease in electrical energy usage.

In the future, we aim to incorporate machine learning algorithms for forecasting server workload and apply our approach to real-world data centers.

Author Contributions

Conceptualization: Vaneet Kumar, Shadab Alam.

Data curation: Mohammed Shuaib.

Formal analysis: Aleem Ali, Payal Mittal, Ibrahim Aqeel, Shadab Alam, Mohammed Y. Aalsalem.

Funding acquisition: Ibrahim Aqeel, Mohammed Y. Aalsalem.

Investigation: Payal Mittal, Mohammed Shuaib.

Methodology: Vaneet Kumar, Aleem Ali, Payal Mittal.

Project administration: Shadab Alam, Mohammed Y. Aalsalem.

Supervision: Mohammed Y. Aalsalem.

Validation: Mohammed Shuaib.

Visualization: Ibrahim Aqeel.

Writing – original draft: Vaneet Kumar, Aleem Ali, Shadab Alam.

Writing – review & editing: Payal Mittal, Ibrahim Aqeel, Mohammed Shuaib, Mohammed Y. Aalsalem.

References

1. Flynn MJ. Very high-speed computing systems. *Proc IEEE Int Conf Cloud Comput Intell Syst CCIS*. 1966; 54: 1901–1909. <https://doi.org/10.1109/PROC.1966.5273>
2. Zhou Z, Shojafar M, Alazab M, Abawajy J, Li F. AFED-EF: An Energy-Efficient VM Allocation Algorithm for IoT Applications in a Cloud Data Center. *IEEE Trans Green Commun Netw*. 2021; 5: 658–669. <https://doi.org/10.1109/TGCN.2021.3067309>
3. Buvana M, Loheswaran K, Madhavi K, Ponnusamy S, Behura A, Jayavadeivel R. Improved resource management and utilization based on a fog-cloud computing system with IOT. *Microprocess Microsyst*. 2021; 2: 103815. <https://doi.org/10.1016/j.micpro.2020.103815>
4. Chinnasamy P, Albakri A, Khan M, Raja AA, Kiran A, Babu JC. Smart Contract-Enabled Secure Sharing of Health Data for a Mobile Cloud-Based E-Health System. *Appl Sci*. 2023; 13: 3970.
5. Temjanovski R, Bezovski Z, Jovanov T. Cloud computing in logistic and Supply Chain Management environment. *J Int Board Contents*. 2021; 6: 23–33. <https://doi.org/10.46763/JOE216.10023t>
6. Shaw R, Howley E, Barrett E. Applying Reinforcement Learning towards automating energy efficient virtual machine consolidation in cloud data centers. *Inf Syst*. 2021; 107: 101–122. <https://doi.org/10.1016/j.is.2021.101722>
7. Uddin M, Shah A, Memon J. Measuring Efficiency of Tier Level Data Centers to Implement Green Energy Efficient Data Centers Measuring Efficiency of Tier Level Data Centers to Implement Green Energy Efficient Data Centers. 2013. <https://doi.org/10.5829/idosi.mejsr.2013.15.2.2356>
8. Bermejo B, Juiz C, Guerrero C. Virtualization and consolidation: a systematic review of the past 10 years of research on energy and performance. *J Supercomput*. 2019; 75: 808–836. <https://doi.org/10.1007/s11227-018-2613-1>
9. Goyal P, Mittal P. Pharmacognostical and Pharmacological Review on Crinum Latifolium. *Plant Cell Biotechnol Mol Biol*. 2020; 21: 207–211.

10. Aqeel I, Khormi IM, Khan SB, Shuaib M, Almusharraf A, Alam S, et al. Load Balancing Using Artificial Intelligence for Cloud-Enabled Internet of Everything in Healthcare Domain. *Sensors*. 2023; 23: 5349.
11. Mikram H, El Kafhali S, Saadi Y. Server Consolidation Algorithms for Cloud Computing: Taxonomies and Systematic Analysis of Literature. *Int J Cloud Appl Comput*. 2022; 12: 1–24. <https://doi.org/10.4018/JCAC.311034>
12. Kumar V, Jindal B. Load Aware Hotspot Selection for SLA Improvement in Cloud Computing and Protect Environment by Reduction In CO 2 Emissions. *IOP Conf Ser Earth Environ Sci*. 2021; 889: 012–028. <https://doi.org/10.1088/1755-1315/889/1/012028>
13. Garg V, Jindal B. Resource optimization using predictive virtual machine consolidation approach in cloud environment. *Intell Decis Technol*. 2023; Preprint: 1–14. <https://doi.org/10.3233/IDT-220222>
14. Alharbi F, Tian YC, Tang M, Zhang WZ, Peng C, Fei M. An Ant Colony System for energy-efficient dynamic Virtual Machine Placement in data centers. *Expert Syst Appl*. 2019; 120: 228–238. <https://doi.org/10.1016/j.eswa.2018.11.029>
15. Hsieh SY, Liu CS, Buyya R, Zomaya AY. Utilization-prediction-aware virtual machine consolidation approach for energy-efficient cloud data centers. *J Parallel Distrib Comput*. 2020; 139: 99–109. <https://doi.org/10.1016/j.jpdc.2019.12.014>
16. Schad J, Dittrich J, Quiané-Ruiz J-AA. Runtime measurements in the cloud: Observing, analyzing, and reducing variance. *Proc VLDB Endow*. 2010; 3: 460–471. <https://doi.org/10.14778/1920841.1920902>
17. Ranjbari M, Akbari Torkestani J. A learning automata-based algorithm for energy and SLA efficient consolidation of virtual machines in cloud data centers. *J Parallel Distrib Comput*. 2018; 113: 55–62. <https://doi.org/10.1016/j.jpdc.2017.10.009>
18. Vaneet K, Balkrishan J. Energy efficient virtual machine migration approach with SLA conservation in cloud computing. *J Cent South Univ*. 2021; 28: 760–770.
19. Buyya R, Beloglazov A, Abawajy J. Energy-efficient management of data center resources for cloud computing: A vision, architectural elements, and open challenges. *International Conference on Parallel and Distributed Processing Techniques and Applications*. 2010. pp. 1–12. <https://doi.org/10.1002/cpe.1867>
20. Hieu NT, Di Francesco M, Yla-Jaaski A. A multi-resource selection scheme for virtual machine consolidation in cloud data centers. *6th IEEE Int Conf Cloud Comput Technol Sci*. 2015; 2015(2): 234–239. <https://doi.org/10.1109/CloudCom.2014.130>
21. Zhou Z, Hu ZG, Song T, Yu JY. A novel virtual machine deployment algorithm with energy efficiency in cloud computing. *J Cent South Univ*. 2015; 22: 974–983. <https://doi.org/10.1007/s11771-015-2608-5>
22. Zhou Z, Hu Z, Li K. Virtual Machine Placement Algorithm for Both Energy-Awareness and SLA Violation Reduction in Cloud Data Centers. *Sci Program*. 2016; 2016: 1–11. <https://doi.org/10.1155/2016/5612039>
23. Monil MAH, Rahman RM. VM consolidation approach based on heuristics fuzzy logic, and migration control. *J Cloud Comput*. 2016; 5: 1–18. <https://doi.org/10.1186/s13677-016-0059-7>
24. Yadav R, Zhang W. MeReg: Managing energy-SLA tradeoff for green mobile cloud computing. *Wirel Commun Mob Comput*. 2017; 2017: 1–11. <https://doi.org/10.1155/2017/6741972>
25. Yadav R, Zhang W, Kaiwartya O, Singh PR, Elgendy IA, Tian YC. Adaptive Energy-Aware Algorithms for Minimizing Energy Consumption and SLA Violation in Cloud Computing. *IEEE Access*. 2018; 6: 55923–55936. <https://doi.org/10.1109/ACCESS.2018.2872750>
26. Yadav R, Zhang W, Li K, Liu C, Shafiq M, Karn NK. An adaptive heuristic for managing energy consumption and overloaded hosts in a cloud data center. *Wirel Networks*. 2020; 26: 1905–1919. <https://doi.org/10.1007/s11276-018-1874-1>
27. Mandal R, Mondal MK, Banerjee S, Srivastava G, Alnumay W, Ghosh U, et al. MECpVmS: an SLA aware energy-efficient virtual machine selection policy for green cloud computing. *Cluster Comput*. 2023; 26: 651–665. <https://doi.org/10.1007/s10586-022-03684-2>
28. Thakur N, Goswami M, Mittal P, Goyal P, Sood A. Research Article Solid Dispersion: A Novel Approach for Bioavailability Enhancement. 2016; 39: 265–272.
29. Rana N, Abd Latiff MS, Abdulhamid SM, Misra S. A hybrid whale optimization algorithm with differential evolution optimization for multi-objective virtual machine scheduling in cloud computing. *Eng Optim*. 2022; 54: 1999–2016.
30. Qamar S, Amaan M, Rahman MI, Aqeel I, Shuaib M, Khormi IM, et al. Cloud data transmission based on security and improved routing through hybrid machine learning techniques. *Soft Comput*. 2023; 1–8.
31. Rana N, Latiff MSA, Abdulhamid SM, Chiroma H. Whale optimization algorithm: a systematic review of contemporary applications, modifications and developments. *Neural Comput Appl*. 2020; 32: 16245–16277.

32. Ahmad RW, Gani A, Siti SH, Shiraz M, Xia F, Madani SA. Virtual machine migration in cloud data centers: a review, taxonomy, and open research issues. *J Supercomput.* 2015; 71: 2473–2515. <https://doi.org/10.1007/s11227-015-1400-5>
33. Saadi Y, El Kafhali S. Energy-efficient strategy for virtual machine consolidation in cloud environment. *Soft Comput.* 2020; 2: 14845–14859. <https://doi.org/10.1007/s00500-020-04839-2>
34. Zhou Z, Abawajy J, Chowdhury M, Hu Z, Li K, Cheng H, et al. Minimizing SLA violation and power consumption in Cloud data centers using adaptive energy-aware algorithms. *Futur Gener Comput Syst.* 2017. <https://doi.org/10.1016/j.future.2017.07.048>
35. Zhang X, Wu T, Chen M, Wei T, Zhou J, Hu S, et al. Energy-aware virtual machine allocation for cloud with resource reservation. *J Syst Softw.* 2019; 147: 147–161. <https://doi.org/10.1016/j.jss.2018.09.084>
36. Beloglazov A, Buyya R. Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in Cloud data centers. *Concurr Comput Pract Exp.* 2012; 24: 1397–1420. <https://doi.org/10.1002/cpe.1867>
37. Yadav R, Zhang W, Chen H, Guo T. MuMs: Energy-aware VM selection scheme for cloud data center. 28th International Workshop on Database and Expert Systems Applications, DEXA. Lyon, France; 2017. pp. 132–136. <https://doi.org/10.1109/DEXA.2017.43>