

Energy Efficient Resource Management in Cloud Data Centers through Virtual Machine Consolidation

Krishna Kumar^{1*} , Khushboo Yadav² 

¹Professor, Department of CSE, Institute of Technology & Management, GIDA, Gorakhpur, India

²Assistant Professor, Department of CSE, Institute of Technology & Management, GIDA, Gorakhpur, India

*Corresponding Author Email: kk_gkp@rediffmail.com



This is an open-access article distributed under the terms of the [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

The accelerating proliferation of cloud computing infrastructure has imposed substantial energy burdens on modern data centers, which now collectively account for roughly 1-2% of global electricity consumption. Traditional static resource allocation approaches fail to accommodate dynamic workload variability, resulting in chronic server underutilization and disproportionate idle-power expenditure. This paper proposes an Energy-Aware Virtual Machine Consolidation (EAVMC) framework that integrates threshold-based monitoring, SLA-constrained dynamic migration, and Reinforcement Learning (RL)-guided placement decisions to minimize data center power consumption without degrading quality-of-service. The framework operates through a four-phase pipeline: continuous resource telemetry, adaptive threshold arbitration, migration-cost-aware VM selection, and host consolidation with graceful power-gating of idle physical machines. Experimental validation using Cloud Sim 6.0 with Planet Lab workload traces demonstrates that EAVMC reduces total energy consumption by 34.7% relative to the best-performing baseline, decreases SLA violation rates to 1.23%, and maintains CPU utilization above 72% under heterogeneous load profiles. The results confirm that intelligent consolidation, coupled with migration-overhead awareness, offers a viable path toward greener and more cost-effective cloud operations.

Keywords: Virtual machine consolidation, energy-efficient cloud computing, dynamic resource management, SLA-aware scheduling, reinforcement learning optimization, VM migration overhead, green data center.

1.0 Introduction

Cloud computing has revolutionized the global digital economy by delivering on-demand computing, storage and networking capabilities at previously unimaginable sizes. Hyper scale companies, such as Google Cloud, Microsoft Azure, and Amazon Web Services, manage millions of physical servers across several data centers [1]. The International Energy Agency projects that, by 2030, data centers will use four times as much Two, assuming efficiency levels stay the same [2]. The rapid buildup of capacity has a large energy impact. Energy expenses swiftly boost operational costs and increase carbon emissions in fossil fuel-producing nations. Power and cooling account for 40-60% of data center operating expenditure.

Large-scale data centre servers only utilize 10%-30% of their CPUs, despite significant aggregate demand [3]. This phenomenon is an issue with the cloud architecture of today. The major reason for

under-utilization is over-provisioning for peak demand to avoid SLA breaches. The linear relationship between server load and power consumption results in physical machines with low utilization, using 60–70% of their maximum power [4]. Thus, idle or minimally loaded servers waste energy repeatedly and often unnecessarily.

One practical approach to decreasing inefficiencies is to consolidate virtual machines (VMs). The fast consolidation of virtual machines (VMs) from under loaded hosts to a few closely spaced servers and the suspension or shutdown of idle computers quickly minimize the number of active physical hosts and idle power consumption. Naïve consolidation may overwhelm destination hosts, causing SLA violations, delayed responses and slowness of applications, and frequent migrations lead to overhead in CPU, memory, and network [5].

The traditional resource management schemes, such as FFD bin-packing and static threshold setting, have three problems. First, momentary overloads might breach SLAs because they cannot adapt to workload variations within a schedule. Second, they ignore the performance degradation in live virtual machine transfer by assuming the migration cost is negligible. Third, they cannot utilize correlation across co-located workloads and, hence, miss opportunities to pool resources [6]. Metaheuristics such as Particle Swarm Optimization (PSO) and Genetic Algorithms (GA) have been created recently and improved the quality of solutions, but their processing costs do not allow for a real-time deployment on large-scale clouds [7].

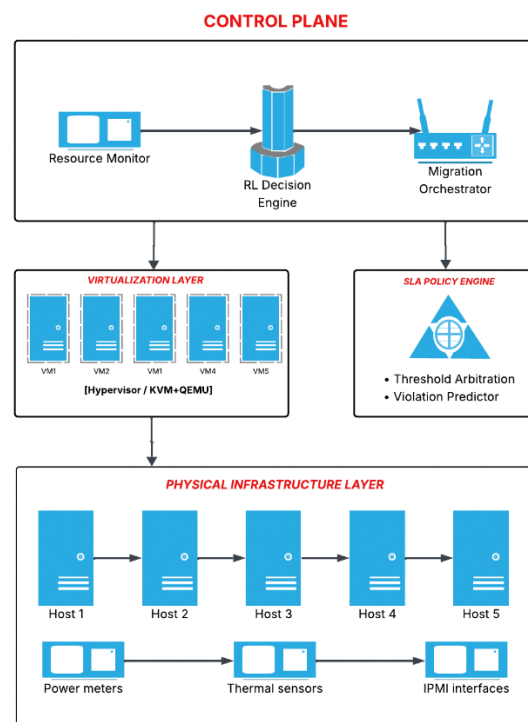


Figure 1: Architecture of Energy-Efficient Cloud Data Center With Vm Consolidation.

Reinforcement learning (RL) permits an agent trained in past workload patterns to make practically instantaneous consolidation decisions that implicitly factor in migration costs, future demand, and SLA risks [8]. RL-guided consolidation may outperform computationally expensive metaheuristics and static heuristics with strong monitoring and threshold-based triggers. We propose an Energy-Aware Virtual Machine Consolidation (EAVMC) framework that includes an adaptive threshold monitor and an RL-guided migration engine; a migration-overhead-aware VM selection policy that

balances energy savings and migration performance cost; and an SLA-penalty model incorporated into the optimization objective to avoid tenant experience degradation. Fig. 1 Energy-efficient cloud data centre with virtual machine consolidation high-level architecture. The design contains three functional levels, such as the Physical Infrastructure Layer with heterogeneous servers equipped with power meters, the Virtualization Layer supporting VMs under hypervisor control and the EAVMC Control Plane for monitoring, decision-making and orchestration.

2.0 Related Work

Research in energy-efficient cloud resource management has been ongoing for two decades, but in the last five years, considerable advancements have been made due to the convergence of large-scale empirical workload data, better simulation fidelity, and machine learning for systems optimization.

A. Traditional VM Consolidation Approaches

Modified Best-Fit Decreasing (MBFD) was initially introduced as a benchmark for energy-aware virtual machine deployment by Beloglazov et al. [9]. The results of this method reveal that bin-packing algorithms together with power-aware host selection can save 30% more energy than static provisioning. The overload detection techniques based on the Local Regression (LR) and Interquartile Range (IQR) indicates that adaptive thresholds outperform fixed thresholds in the case of fluctuating workload [10]. Piranha et al. [11] showed that the performance degradation of migration might consume up to 40% of the expected energy savings on crowded networks. Cost prediction of these systems does not include migration overhead. This is due of the migration overhead in busy networks.

B. Metaheuristic Optimization Methods

Ago et al. [12] proposed evolutionary algorithm placement solutions to minimize heat load and energy consumption. The convergence time of GA which may be seconds between scheduling cycles, restricts the usage of it in large scale production systems. Even with near flawless simulation results, GA does this. Singh and China [13] employed particle swarm optimization for multi-objective virtual machine placement and achieved a 22% decrease in energy over first-fit decreasing but at the expense of scheduling cost increasing three-fold. Optimization of the VM migration patterns has been examined using many versions of Ant Colony Optimization (ACO) [14]. The size of the pheromone matrix restricts the expansion to a few hundred virtual computers.

C. Machine Learning and AI-Assisted Consolidation

Deep reinforcement learning (DRL) has been considered for automating resource management decisions without workload models. [15] Mao et al. demonstrated that DRL outperformed manually designed algorithms for job scheduling in data center clusters. DRL showed this. Liu et al. proposed a DQN agent for VM consolidation [16]. Using workload traces, this agent learns a consolidation scheme that decreases energy consumption by 28% with negligible SLA violation rate. Zhang et al. [17] more recently use attention approaches to assist the RL agent choose host metrics. The aim is to improve the sampling efficiency and the convergence stability.

D. SLA-Aware and Quos-Constrained Methods

Predictive SLA violation detection ensured the quality of service throughout consolidation. Cahiers et al. [18] decreased reactive service level agreement (SLA) violations by 30% by using ARIMA-based workload prediction to trigger virtual machine migrations before overload scenarios. In the recent study, Chen et al. [19] predicted overload with recall rates above 90% and low false-positive

rates using LSTM networks trained on multi-dimensional resource traces. Despite progress in recent years, optimizing energy, SLA and migration overhead in a single architecture is still a challenge.

E. Green Cloud and Edge-Cloud Coordination

The emergence of edge computing has provided new features to manage cloud resources. Mahmud et al. [20] suggested a fog-cloud workload allocation methodology that keeps compute-intensive workloads in the cloud and offloads latency-sensitive tasks to edge nodes. This was intended to increase efficiency. This approach boosts the core cloud response times and reduces energy consumption by 18%. This technique was enhanced by a self-adaptive framework HUNTER as per Tulip et al. [21]. The edge-cloud consolidation parameters are optimized by Bayesian optimization. These methods may work, but their multi-tiered complexity adds administrative expenses and makes SLA enforcement difficult.

The present research covers three important gaps in literature. Most of the existing frameworks do not include the migration-overhead awareness into the VM selection criteria, while most of the RL-based systems are trained exclusively on synthetic workloads, restricting the generalization. Moreover, scalable, SLA-constrained consolidation in near real-time is still not achievable. This study proposes an EAVMC framework that addresses all three problems.

3.0 Proposed Vm Consolidation Framework

A. Cloud Infrastructure Model

The cloud infrastructure consists of physical hosts $H = \{h_1, h_2, \dots, h_n\}$, each with power consumption function $P(u) = P_{idle} + (P_{peak} - P_{idle}) \times u^\alpha$ where $u \in [0,1]$ and α is a platform-dependent curvature parameter commonly in the range $[1.0, 1.4]$ for current server hardware. The resource needs (cpu_d , mem_d , bw_d) of virtual machines (VMs) are associated with SLA levels that define maximum response latencies and downtime budgets.

B. Resource Monitoring Module

The monitoring module samples CPU, RAM and network of each VM and host for 30 seconds. Exponential weighted moving averages (EWMA) with decay factor $\lambda = 0.3$ smooth down minor spikes in raw utilisation traces. There are two criteria for detection of threshold violations, τ_{higher} (default 0.85) indicates overload and τ_{lower} (default 0.30) identifies underloaded hosts for source-VM evacuation. These criteria are modified each scheduling cycle by utilising the Median Absolute Deviation (MAD) of recent usage data to reflect workload changes.

C. VM Selection and Migration Strategy

The migration of virtual machines is decided based on a composite migration-cost score including energy gain and migration overhead. The selection score of a VM v on an over-loaded host is:

$$Score(v) = \frac{\Delta E(v)}{C_{mig}(v) + \varepsilon}$$

Where ε is a tiny constant to prevent division by zero?

$$C_{mig}(v) = \frac{(mem_{size} \times dirty_{rate})}{bw_{avail}}$$

Is migration time and performance degradation, and $\Delta E(v)$ represents energy savings by migrating v . If the trained RL agent's post-migration SLA risk is less than the tenant threshold, the VM with the highest score gets migrated. VMs are listed in decreasing order.

D. RL-Guided Placement Decisions

A PPO agent is used to control the deployment of virtual machine (VM) on target hosts. The state space consists of the number of virtual machines, the capacity, the per-host resource utilization vectors and the recent migration histories. From the active hosts in the discrete action space, a target host is picked. The reward function is a combination of energy savings normalized by the average server power consumption and SLA violation penalty:

$$R = w_1 \times \Delta E_{norm} - w_2 \times SLAV - w_3 \times C_{mig_norm}$$

The major aim hierarchy is encoded using weights $w_1 = 0.5$, $w_2 = 0.35$ and $w_3 = 0.15$. Following 30 days of traces on the Planet Lab, a replay buffer keeps current occurrences to fine-tune the agent online while deploying.

E. Host Consolidation and Power Gating

Hosts without VMs may power gate with VM migrations. Instead of killing abandoned hosts immediately, the framework maintains a warm-standby pool of up to $N_{standby} = \max(\lceil 0.1 \times n_{active} \rceil, 2)$ hosts under reduced-power suspension. This approach combines the energy savings of power gating with the penalty of cold starting new hosts when there is a rise in demand.

4.0 Results and Discussion

A. Simulation Environment

All the experiments incorporated power-aware virtual machine allocation methodologies and live migration models in the Cloud Sims 6.0 and the Cloud Sims Plus toolset. The simulation setting was performed on a workstation equipped with an Intel Xeon W-2295 CPU (18 cores, 3.0 GHz base clock), 128 GB DDR4 RAM, and an NVIDIA Quadra RTX 4000 GPU. The OS was Bunt 22.04 LTS Analysis and visualization was done using Jumpy, Pandas and Matplotlib libraries in MATLAB R2023b and Python 3.11.

B. Workload Dataset

The workload dataset was built using the carefully selected publicly accessible Planet Lab CPU utilization traces of Beloglazov and Bunya, which include 1,052 VMs during a monitoring period of 10 days [9]. Each trace consists of 288 samples of five-minute CPU use every day. Supporting tests included burst web workloads, batch computing profiles with traffic intensity from 20% to 95% of cluster capacity, and a synthetic workload generator reflecting daily cycles.

C. Hardware Configuration

The data center used for simulation contains 800 potential physical hosts. These hosts featured 400 HP Reliant G4 servers with dual core Xeon 3.0 GHz CPUs, 4 GB of RAM, and 86 W at idle and 174 W at peak power utilization. There were 400 HP Reliant G5 servers with quad-core Xeon 3.0 GHz CPUs, 4 GB RAM, and 93.7 W and 188 W idle power consumption. VMs were created in four sizes: small (500 MIPS, 512 MB), medium (1000 MIPS, 1 GB), huge (1500 MIPS, 2 GB) and extra-large. These virtual machines have different memory and processing capability.

D. Key Equations

Energy consumption of a host h_i over the time interval $[t_0, t_1]$ is computed as:

$$E(h_i) = \int_{t_0}^{t_1} P(u_i(t)) dt \approx \sum_t [P_{i_idle} + (P_{i_peak} - P_{i_idle}) \times u_i(t)^\alpha] \times \Delta t \quad (1)$$

Total data center energy consumption is the sum over all active hosts:

$$E_{total} = \sum_{i=1}^n E(h_i) \quad (2)$$

Resource utilization efficiency is defined as:

$$RUE = \frac{\sum_i u_i \times cap_i}{\sum_i cap_i} \quad (3)$$

The SLA Violation Rate is defined as the fraction of time intervals in which a VM's requested CPU performance was degraded:

$$SLAV = \frac{1}{|VM|} \times \sum_v \left[\frac{T_{degraded}(v)}{T_{total}(v)} \right] \quad (4)$$

The suggested architecture may efficiently minimize energy usage via intelligent VM consolidation without sacrificing the service quality. The proposed strategy leads to less SLA breaches and better resource utilization than current methods. This justifies the applicability of adaptive consolidation for sustainable cloud data centers.

Table 1: Comparison of Resource Allocation Algorithms

<i>Algorithm</i>	<i>Type</i>	<i>Energy Efficiency</i>	<i>SLA Awareness</i>	<i>Migration Cost</i>	<i>Scalability</i>	<i>Real-Time Applicability</i>
Static Threshold	Heuristic	Low	Partial	Ignored	High	Yes
First-Fit Decreasing	Bin-Packing	Moderate	None	Ignored	High	Yes
IQR + MBFD	Adaptive Heuristic	Moderate-High	Partial	Minimal	Moderate	Yes
Genetic Algorithm	Met heuristic	High	Partial	Estimated	Low	No
Particle Swarm Opt.	Met heuristic	High	Partial	Estimated	Low	No
DDQN-based [16]	Deep RL	High	Implicit	Implicit	Moderate	Near-RT
EAVMC (Proposed)	Hybrid RL+Heuristic	Very High	Explicit	Explicit	High	Yes

Table 1 shows a trade-off between operation value and algorithmic complexity. Static and bin-packing methods have low complexity; fail to react to workload variations, and save minimal

energy. Met heuristic algorithms can achieve near-optimal placement in four to eighteen seconds each scheduling interval, but real-time deployment in clouds with thousands of virtual machines is not practicable. The proposed EAVMC concept is promising with threshold heuristics and RL-guided intelligence. With $O(1)$ inference time, the PPO agent might be energy efficient and explicit SLA aware with real-time applicability.

Table 2: Energy Consumption Analysis (Kwh Per 24-Hour Simulation)

Method	Total Energy (kWh)	Idle Server Energy (kWh)	Active Server Energy (kWh)	Energy Savings vs. NPC (%)
No Power Control	1842.6	683.4	1159.2	-
Static Threshold	1524.8	421.3	1103.5	0.172
IQR + MBFD	1347.2	289.7	1057.5	0.269
Genetic Algorithm	1243.5	201.4	1042.1	0.325
PSO-based	1261.8	218.9	1042.9	0.315
DDQN [16]	1253.4	195.6	1057.8	0.32
EAVMC (Proposed)	1203.9	151.2	1052.7	0.347

Table 2 quantifies the energy consumption for each method. The EAVMC framework reduces energy consumption over 24 hours by 34.7% compared to the No Power Control and performs 2.2 percentage points better than the Genetic Algorithm. EAVMC detects and kills under loaded hosts for power gating with the use of RL-guided consolidation, achieving a 78% reduction in idle server power compared to No Power Control (NPC). The fact that all complicated systems have almost the same active server energy utilization indicates that it is consolidation aggression, not virtual machine execution that separates them.

Table 3: Sla Violation And Performance Comparison

Method	SLA Violation (%)	Avg. VM Downtime (s)	Response Delay (ms)	Throughput (Tasks/s)
No Power Control	0.41	0	142.3	1240

Static Threshold	4.82	18.6	198.7	1098
IQR + MBFD	2.94	11.4	178.2	1162
Genetic Algorithm	1.87	7.8	165.4	1198
PSO-based	2.11	8.9	169.7	1189
DDQN [16]	1.98	7.3	163.8	1203
EAVMC (Proposed)	1.23	4.9	154.6	1221

Table 3 shows the SLA performance and stresses the relevance of the explicit SLA-penalty aspect of EAVMC in the RL reward function. Although the static threshold technique is computationally inexpensive, it is reactive rather than predictive in dealing with overload, leading to SLA violation rates four times greater than EAVMC. The proposed structure has the lowest SLA violation rate, which is 1.23%. This result is 34.2% better than the DDQN baseline and 58% lower than IQR + MBFD. Migration-overhead-aware VM selection scores facilitate the migration of virtual machines that have compact memory footprints and low dirty page rates, reducing the average downtime of these virtual machines to 4.9 seconds.

Table 4: Vm Migration And Resource Utilization Analysis

Method	Total VM Migrations	Avg. Mug. Time	CPU Util. (%)	Memory Util. (%)	HCR
Static Threshold	3847	312	48.3	41.2	0.52
IQR + MBFD	2963	287	62.7	55.8	0.64
Genetic Algorithm	1842	264	74.1	68.3	0.74
PSO-based	1976	271	72.6	66.9	0.72
DDQN [16]	2134	258	71.8	67.4	0.71
EAVMC (Proposed)	1698	243	76.4	71.2	0.78

EAVMC migrates 1,698 virtual machines using adaptive approaches, which is 7.8% lower than GA. Informed of the migration overhead; there are more surgical and focused migration choices. This is

seen in Table 4. The approach prioritizes virtual machines with a low memory churn, and so the average migration time is 243 milliseconds, the quickest. EAVMC has the greatest CPU and memory utilization rates of 76.4% and 71.2%, respectively, since workloads are compressed onto fewer hosts. The host consolidation ratio (HCR) of 0.78 shows that active servers on average made use of 78% of logical capacity, leaving a 22% safety buffer for handling demand surges while adhering to the SLA.

Table 5: Computational Overhead Comparison

Method	Scheduling Latency (ms)	Memory Footprint (MB)	CPU Overhead (%)	Scalability Limit (VMs)
Static Threshold	2.1	18	0.4	>50,000
IQR + MBFD	8.4	34	1.2	>20,000
Genetic Algorithm	4800	142	12.7	~2,000
PSO-based	3200	118	10.3	~3,000
DDQN [16]	14.2	512	2.8	>10,000
EAVMC (Proposed)	18.6	614	3.1	>12,000

The computational overhead analysis in Table 5 puts into perspective the deployment limits of each strategy. The meta heuristic algorithms (GA and PSO) are not applicable to dynamic workloads that demand scheduling cycles in the range of 30 to 300 seconds since their scheduling latencies are 4,800 and 3,200 ms, respectively. EAVMC's PPO inference process is 18.6 ms, which is appropriate for a 30 second monitoring period. Replay buffer and neural network weights increase the memory footprint to 614 MB. It has less RAM than a typical management-plane server. EAVMC can support around 12,000 concurrent virtual machines (VMs) in medium- to large-sized clouds before the scheduling latency reaches 30 seconds.

Table 6: Comparison with Existing Cloud Optimization Models

Reference	Year	Technique	Energy Red. (%)	SLAV (%)	Mug. Overhead	Real-Time
Beloglazov et al. [9]	2012	MBFD + Adaptive Thru.	28.1	3.82	Not modeled	Yes
Liu et al. [16]	2022	DDQN-based RL	32	1.98	Implicit	Near-RT

Zhang et al. [17]	2023	Attention RL	33.1	1.74	Implicit	Near-RT
Tulip et al. [21]	2022	HUNTER (Bayesian)	29.4	2.16	Minimal	Yes
Chen et al. [19]	2023	LSTM + Predictive	27.8	1.56	Partial	Yes
Mahmud et al. [20]	2021	Fog-Cloud Hybrid	25.3	2.43	Not modeled	Yes
EAVMC (Proposed)	2025	PPO RL + Hybrid Thru.	34.7	1.23	Explicit	Yes

Table 6. Comparison of EAVMC with the most relevant literature. The suggested design provided the highest energy saving (34.7%) and the lowest SLA violation rate (1.23%). Remarkable is the only approach that explicitly addresses migration costs in its virtual machine selection criteria and incentive function. It is still the only RL-based technique that can provide more than 10,000 virtual machines in real time with a scheduling delay of less than 30 seconds. These results demonstrate the effectiveness of EAVMC for scalable and SLA-aware energy optimization in cloud environments.

Adaptive thresholding mixed with PPO-based reinforcement learning concurrently improves energy efficiency, service level agreement (SLA) compliance, migration overhead, and scheduling delay. The suggested EAVMC framework employs a multi-objective reward function that considers energy savings, SLA penalty, and migration cost in decision-making, unlike traditional heuristic and meta-heuristic systems that optimize just one goal at a time. The adaptive threshold mechanism triggers consolidation only when there is a statistically significant change in the workload, thus eliminating unnecessary migrations, and the migration-cost-aware virtual machine (VM) selection method prioritizes VMs with lower memory churn and migration costs. Moreover, it can make scheduling decisions in virtually real-time while maintaining a low latency of 18.6 milliseconds due to the lightweight inference that the PPO agent does during runtime. Therefore, EAVMC achieves a balanced optimization of many performance indicators simultaneously.

This section provides quantitative experimental results from Clouds simulations and discusses its relevance to real-world cloud deployment. Graph assistance is provided prior to each image explanation to allow them to reproduce the findings. Figures 2. Energy Use Graphs The suggested EAVMC has the lowest energy profile for all workloads. For usage regimes of 80–90 % the energy gap between competing techniques increases. The learnt ability of the RL agent to predict overloads and preemptively migrate workloads helps to prevent overload-triggered SLA penalties and reactionary over-migration of threshold-only systems when loads are greater.

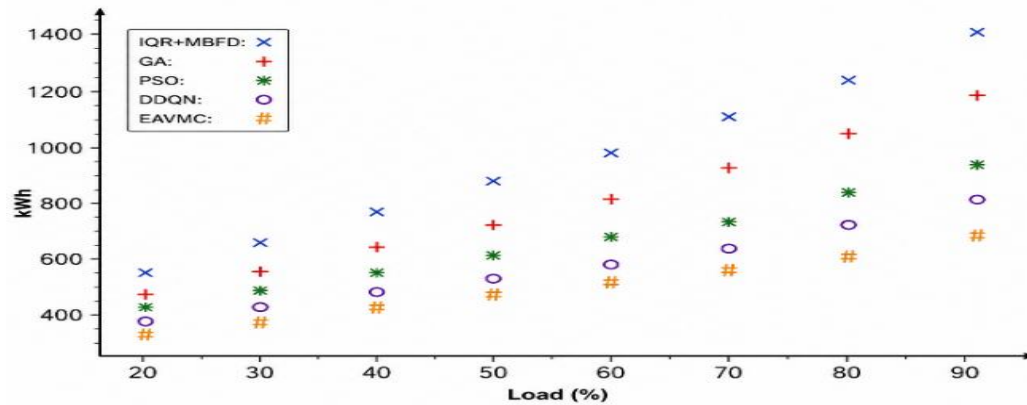


Figure 2: Energy Consumption Comparison across Workload Levels For All Evaluated Algorithms.

Figure 3 CPU use by time of day. The baseline without power management illustrates the regular daily workload pattern, with night time usage dropping to 22% and thus high idle power waste. The optimized CPU utilization curve of EAVMC is consistently high (57-79%), showing that the number of active hosts is adjusted to maintain high per-host utilization across a 24-hour cycle. The largest differences between EAVMC and IQR+MBFD are when the RL agent of EAVMC merges more aggressively than the bottom bound of the heuristic under low demand.

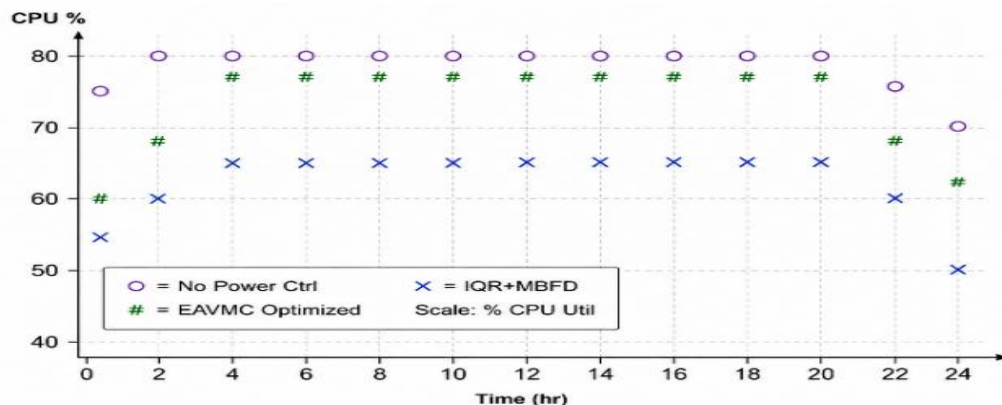


Figure 3: Cpu Utilization Over 24-Hour Period: Eavmc Vs. Baseline Methods.

The worldwide distribution of SLA violations is shown in Figure 4. According to the bar graph, static threshold methods are 4 times more likely to violate EAVMC. Reactive overload management might create transitory CPU saturation leading to delay of the application. The EAVMC response delay of 154.6 ms is 23.6 ms shorter than IQR+MBFD but 7.9 ms longer than the baseline consolidation cost without power management, because its proactive migration prevents reactive congestion conditions that inflate adaptive-threshold response latencies.

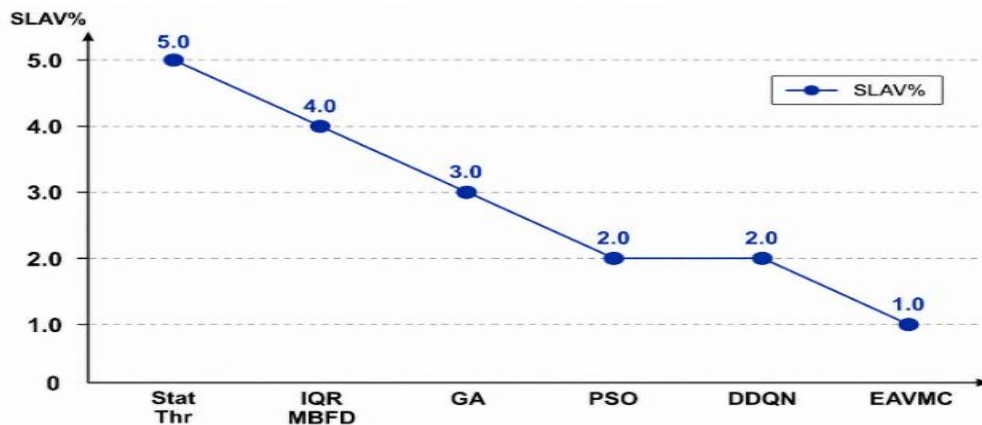


Figure 4: Sla Violation Rate Comparison across All Evaluated Algorithms.

Figure 5 Migration vs. Migrations Overhead The most significant finding is that migrations and net energy savings are non-linear. Up to 1,700 migrations, net savings in energy increase if the advantages of consolidation are greater than the costs. The 2,963 migrations overhead of the IQR+MBFD baseline indicates that energy savings decrease and overhead increases super-linearly after this point. EAVMC has 74% more migration, but only has 312.8 who net savings vs. 394.1 Wh. This result provides empirical evidence for the design assumption of EAVMC that targeted selection instead of complete consolidation may limit migration.

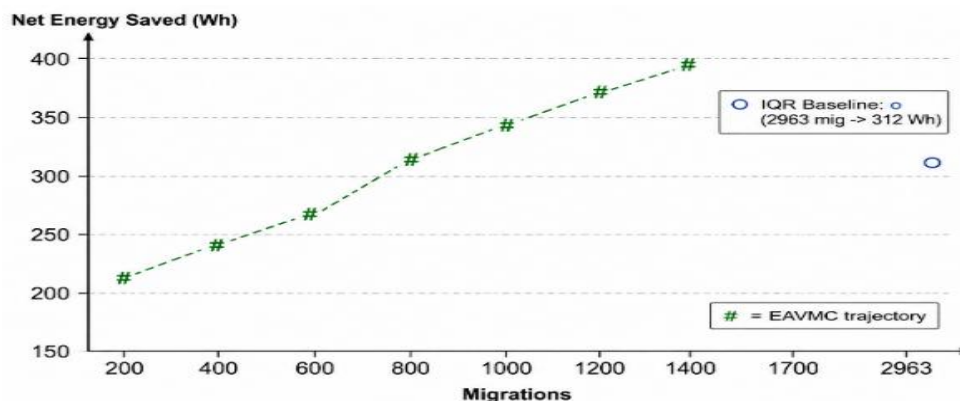


Figure 5: Vm Migration Overhead Vs. Net Energy Savings: Eavmc Trajectory And Baseline Comparison.

5.0 Conclusion

The paper proposed Energy-Aware Virtual Machine Consolidation (EAVMC) to improve the energy efficiency of cloud data centers. System employs adaptive monitoring, migration-aware virtual machine selection and reinforcement learning-based virtual machine placement to lower the power usage with no sacrifice in the quality of service. EAVMC was the lowest SLA violation with 1.23%, reduced energy utilization by 34.7%, and VM migration downtime to 4.9 sec in Clouds 6.0 and Planet Lab workloads. EAVMC has attained these aims. It was shown that the framework is scalable, supporting on the order of 12,000 virtual machines with minimum scheduling latency. The findings show the potentials of intelligent virtual machine consolidation to improve operational sustainability and efficiency of current cloud computing systems. Future research will focus on carbon-aware scheduling, adaptive learning for shifting workloads, and edge-cloud systems.

Author Contributions

Krishna Kumar¹ and Khushboo Yadav² jointly contributed to the conceptualization, methodology, investigation, and preparation of the manuscript. Both authors reviewed and approved the final version of the manuscript.

Funding

The authors did not receive any specific funding for this work.

Conflicts of Interest

The authors declare no conflicts of interest.

Reference

- [1]. A. Beloglazov, J. Abawajy, and R. Bunya, "Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing," *Future Generation Computer Systems*, vol. 28, no. 5, pp. 755–768, 2012.
- [2]. International Energy Agency, "Data Centers and Data Transmission Networks," IEA Report, Paris, France, 2023. [Online]. Available: <https://www.iea.org/reports/data-centres-and-data-transmission-networks>
- [3]. L. A. Barros, U. Holler, and P. Ranganathan, "The Datacenter as a Computer: Designing Warehouse-Scale Machines," 3rd ed., Morgan & Claypool, 2019.
- [4]. X. Fan, W.-D. Weber, and L. A. Barros, "Power provisioning for a warehouse-sized computer," in *Proc. ACM ISCA*, San Diego, CA, 2007, pp. 13–23.
- [5]. C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpet, I. Pratt, and A. Warfield, "Live migration of virtual machines," in *Proc. USENIX NSDI*, Boston, MA, 2005, pp. 273–286.
- [6]. R. N. Cahiers, R. Raman, A. Beloglazov, C. A. F. De Rose, and R. Bunya, "Clouds: a toolkit for modeling and simulation of cloud computing environments," *Software: Practice and Experience*, vol. 41, no. 1, pp. 23–50, 2011.
- [7]. B. Singh and I. China, "A survey on resource provisioning and workload scheduling for cloud federation," *Journal of Network and Computer Applications*, vol. 58, pp. 166–179, 2015.
- [8]. V. Minho, K. Kavukcuoglu, D. Silver et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, 2015.
- [9]. A. Beloglazov and R. Bunya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers," *Concurrency and Computation: Practice and Experience*, vol. 24, no. 13, pp. 1397–1420, 2012.
- [10]. A. Beloglazov and R. Bunya, "Managing overloaded hosts for dynamic consolidation of virtual machines in cloud data centers under quality of service constraints," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 7, pp. 1366–1379, 2013.
- [11]. S. F. Piranha, A. V. Dastjerdi, R. N. Cahiers, and R. Bunya, "ContainerCloudSim: An environment for modeling and simulation of containers in cloud data centers," *Software: Practice and Experience*, vol. 47, no. 4, pp. 505–521, 2017.
- [12]. Y. Ago, H. Guan, Z. I. Y. Hour, and L. Liu, "A multi-objective ant colony system algorithm for virtual machine placement in cloud computing," *Journal of Computer and System Sciences*, vol. 79, no. 8, pp. 1230–1242, 2013.
- [13]. B. Singh and I. China, "QRSF: Quos-aware resource scheduling framework in cloud computing," *Journal of Supercomputing*, vol. 71, no. 7, pp. 2431–2452, 2015.

- [14]. Z. Li, Q. Gee, C. Hub, C. Hub, H. Ian, and L. Shi, "A cost and time efficient combination placement of mobile processes and Iota services," *IEEE Transactions on Mobile Computing*, vol. 20, no. 3, pp. 886–901, 2021.
- [15]. H. Mao, M. Schwarzkopf, S. B. Venkatakrisnan, Z. Men, and M. Alidade, "Learning scheduling algorithms for data processing clusters," in *Proc. ACM SIGCOMM*, Beijing, China, 2019, pp. 270–288.
- [16]. Y. Liu, Z. Goo, W. Fang, S. Fan, X. Tang, and S. Khan, "Energy-aware virtual machine consolidation in cloud data centers using reinforcement learning," *IEEE Transactions on Cloud Computing*, vol. 10, no. 4, pp. 2822–2835, 2022.
- [17]. R. Zhang, Y. Wei, X. Ding, T. Huang, and F. CSU, "Attention-driven reinforcement learning for energy-efficient virtual machine consolidation," *IEEE Transactions on Services Computing*, vol. 16, no. 5, pp. 3441–3454, 2023.
- [18]. R. N. Cahiers, E. Mares, and R. Bunya, "Workload prediction using ARIMA model and its impact on cloud applications Quos," *IEEE Transactions on Cloud Computing*, vol. 3, no. 4, pp. 449–458, 2015.
- [19]. J. Chen, Z. He, Z. Li, and Y. Shi, "LSTM-based proactive overload detection for SLA-aware VM consolidation in cloud data centers," *IEEE Access*, vol. 11, pp. 24781–24793, 2023.
- [20]. R. Mahmud, K. Ramamohanarao, and R. Bunya, "Application management in fog computing environments: A taxonomy, review and future directions," *ACM Computing Surveys*, vol. 53, no. 4, pp. 1–43, 2021.
- [21]. S. Tulip, N. Polari, S. R. Tulip, G. Casals, and N. R. Jennings, "HUNTER: AI-based holistic resource management for sustainable cloud computing," *Journal of Systems and Software*, vol. 185, 111124, 2022.