Survey on Energy Efficient-Load Balancing in Cloud

Shilpa B. Kodli
Research Scholar
Department of Computer Science and Engineering
PDA College of Engineering, Kalaburagi,
Karnataka, India

Sujata Terdal
Professor and Head of the Department
Department of Computer Science and Engineering
PDA College of Engineering, Kalaburagi,
Karnataka, India

ABSTRACT

Cloud computing is a popular technology which delivers virtualized computer resources via the internet. Numerous load balancing considerations substantially determine the performance of the cloud. Load balancing (LB) distributes a dynamic workload among cloud systems & evenly shares resources such that no database server is overloaded or underloaded. Consequently, an active load balancing strategy in the cloud may improve dependability, services, and resource usage.Load balancing task scheduling is a significant issue in cloud systems that directly impacts resource usage. Load balancing scheduling is important for its significant influence on the cloud research industry's back and front end. If an appropriate load balancing is accomplished in the cloud, useful resource utilization is obtained. Therefore, this survey aims to review the recent research papers on existing techniques based on cloud VM migration & load balancing. The literature study examines the various techniques for VM migration & load balancing approaches in the cloud. It analyzes various research articles and provides a detailed analysis. The analytical examination also considers the maximum performance attainments in various contributions. Furthermore, the chronological review and the tools used in the analyzed works are also examined. Furthermore, the survey includes a variety of research problems and gaps that might help researchers to enhance the future study on VM migration & load balancing approaches in cloud technology.

Keywords

VM Migration; Load Balancing; Cloud; Maximum Performance Achievements; Research Gaps

1. INTRODUCTION

Recently, cloud computing [26] has developed as a heterogeneous computing paradigm to provide computing resources like storage, bandwidth & computing power for delivering IT services to customers through the Internet in self-service, metered ways, and dynamically scalable. In the submitted tasks, the cloud user uses the available services and resources to achieve the greatest efficiency (e.g., maximum availability, minimum execution time) at computation time & cheapest cost. Taking full advantage of cloud computing, it's critical to satisfying the needs of both cloud providers and users. Therefore, the service provider executes load balancing & task scheduling to meet the needs of cloud providers and consumers, resulting in greater usage of available resources. Therefore, load balancing [27] is the process of dispersing the load over different compute nodes in such a way that nodes would not have heavy loads (bottleneck) or small loads (waste of resources). Its benefits involve enhancing cloud system scalability, growing resource utilisation, preventing overloaded & loaded resources, optimising makespan time, growing resource availability, minimising energy consumption, lowering resource usage costs, and protecting cloud computing elasticity. Task scheduling focuses on finding appropriate VMs (Virtual Machines) between heterogeneous VMs for scheduling suitable heterogeneous tasks or specialised workloads to decrease task execution time & monetary costs.

LB(Load Balancing) [28] & task scheduling are 2 difficult problems in cloud resource management that must be addressed simultaneously to suit the need of cloud customers & cloud providers. With the increasing count of cloud customers, the tasks count is expanding exponentially, whereas the amount of VMs(Virtual Machines) remains the same. The amount of VMs(Virtual Machines) is restricted by the capability of actual computers resulting in energy consumption constraints. To equalize load & decrease energy usage, cloud data centres employ live VM(Virtual Machine) migration. VMs are allocated dynamically in hosts during a live migration to lower the host's smaller utilization & increase the amount of the host's larger utilization.

The VM resource management [29] issue is a bin packing issue. However, numerous approaches have been developed, including Best Fit, First Fit, Best Fit Decreasing, GA(Genetic Algorithm)-based and Simulated Annealing-based techniques. Nevertheless, the preceding approaches provide optimal outcomes to issues but fail to provide a global optimum. Such strategies do not regularly modify the VM structure to identify the best solution, and they also expand the number of active servers in terms of moving overloaded servers.While compared to VMs with lower bandwidth utilisation, those VMs with higher bandwidth utilization are positioned nearer to the host system. Virtual machines [30] are portrayed as objects that must be packaged into small bins representing existing technology to address the VM mapping issue. Evolutionary techniques employ meta-heuristic optimization to pick a solution of potential solutions from a search space. Gas(Genetic Algorithms), PSO(Particle Swarm Optimization), & ACO(Ant Colony Optimization) are instances of techniques that use fitness computing to optimise parameters for feasible solutions. On a range of various solutions, the GA applies evolutionary operators such as crossover, mutation, as well as selection. To accommodate VMs [31] in physical hosts, the VMs are plotted to physical hosts via pheromone trails & updated at each iteration. The following is the survey's primary contribution.

1. It provides an extended analysis based on cloud VM migration & load balancing and an analysis of the various approaches used in each article.
2. It reviews performance metrics & their maximum achievements for each work.
3. In addition, the chronological review & evaluation tools utilized are examined in the analyzed works.
4. Provides research gaps & challenges that enable researchers to develop the cloud-based VM migration & load balancing.

The remaining sections are discussed in this article: The relevant works based on VM migration & load balancing in the cloud are shown in Section 2. The review of developed schemes, performances, and maximum attainments are determined in section 3. The chronological review & simulation tool utilised in the reviewed articles is described in Section 4. Finally, section 5 depicts the research gaps & challenges, while Section 6 depicts the conclusion.

Abbreviation	Description
VMs	Virtual Machines
GAs	Genetic Algorithms
ECG	Electrocardiogram
CMODLB	Clustering-based Multiple Objective Dynamic Load Balancing Technique
LB	Load Balancing
ACSO	Adaptive Cat Swarm Optimization
IMPSO	Improved Multi-Objective Particle Swarm Optimization
FF	Firefly
EA-LB	Energy Aware Load Balancing
SLA	Service-Level Agreement
IoT	Internet of Things
AES	Advanced Encryption Standard
PSO	Particle Swarm Optimization
PMHEFT	Predictive Priority-based Modified Heterogeneous Earliest Finish Time
ANN	Artificial Neural Network
ML	Machine Learning
ACO	Ant Colony Optimization
ANN-LB	Artificial Neural Network-based Dynamic Load Balancing
BOEK-means	Bayesian Optimization-based Enhanced K-Means
MVC	MC-aware VM Consolidation
ADA	Adaptive Dragonfly Algorithm
MADRL-DRA	Multi-agent Deep Reinforcement Learning-Dynamic Resource Allocation
LUA	Local User Agent
KDE	Kernel Density Estimation
WAVMCM	Workload Aware Virtual Machine Consolidation Method
INS-SGM	Intermediate Node Selection in Scatter-Gather Migration
MSSP	Minimum Subset Sum Problem
VN	Virtual Network
HLBZID	Heuristic Load Balancing based Zero Imbalance Mechanism
ABSO	Adaptive Beetle Swarm Optimization
BSO	Beetle Swarm Optimization
NVMC	Normalization-based VM Consolidation
CATR	Cumulative Available-to-Total Ratio
FIMPSO	Firefly and IMPSO
MRLB	Main Resource Load Balancing
TB	Time Balancing
PFTF	Proactive Fault Tolerant Framework
LBCO	Load Balancing and Computation Offloading
DFTM	Dynamic Fault Tolerant VM Migration
HHO-PIO	Harries Hawks Optimization-Pigeon Inspired Optimization
PBPSO-DCO	Pearson-BPSO Dynamic VM Consolidation

2. LITERATURE REVIEW

In this section, a few recent techniques are reviewed. In addition, the pros and cons of all the traditional methods are also reviewed.

In 2021, Balaji *et al.* [1]developed an energy-efficient, safe VM migration approach based on optimum cryptography methodology. This research used the ACSO (Adaptive Cat Swarm Optimization) methodology to establish a load balancing system. Load balancing was mostly utilized in the cloud to ensure high availability, hardware maintenance, workload balance, & fault assessment. The VM has been transmitted from 1 physical server to another during the LB procedures. Compared with the conventional technique, the suggested technique has the shortest makespan & uses the least amount of energy.

In 2020, Francis *et al.* [2] adopted FIMPSO (Firefly and Improved Multi-Objective Particle Swarm Optimization) a novel load balancing method that would combine the FF (Firefly)& IMPSO (Improved Multi-Objective Particle Swarm Optimization) techniques. The FF(Firefly) method reduces the search space, while the IMPSO approach has been used to find the increased response in this approach. The IMPSO method selects the global best (gbest) particle with the shortest point-to-line distance. As per the simulation results, the FIMPSO algorithm produced an excellent performance with the reliability, highest CPU usage, throughput, shortest average reaction time, memory utilization, and makespan that was excellent to other comparable approaches.

In 2020, Mandeep *et al.* [3] developed an EA-LB (Energy Aware-Load Balancing)approach for experimental processes in a fog-cloud computing environment. In addition, an LB technique was presented in fog environments. In a fog situation, the EA-LB schemes assist in minimizing energy usage. Data transmission seems more important in scientific process applications, necessitating additional hardware and raising energy requirements. The findings were comparable to other approaches. Furthermore, LB aids in lowering latency, efficiently using resources & improving service quality at the fog layer.

In 2019, Lin *et al.* [4] have determined primary resource load balancing & time balancing as 2 IoT-aware multi-resource task scheduling techniques for heterogeneous cloud environments. The methods were designed to improve load balance, SLA(Service Level Agreement),& IoT response time of a task when reducing energy usage to the greatest extent feasible. Herein, load balancing maintains the efficient balance between loads, whereas time balancing saves energy and time. Furthermore, it executes more admirably in response time than other models.

In 2019, Tamilvizhi*et al.* [5] proposed a novel approach to implementing a fault-tolerant method. That covers the development of cloud servers with cloud selection to reduce network congestion & monitoring systems for fault detection with a migration strategy to handle faults adaptively. The data unavailability was decreased efficiently in the cloudlets owing to the network traffic of the cloud server. The suggested ECB outperforms the traditional honeybee foraging system by 20 to 30 percent.

In 2021, Zhang *et al.* [6] suggested a combined load balancing & compute offloading approach for MEC networks, and also created a new security layer to address any security concerns. The first step was to develop a load balancing method for effective MDU redistribution between BSs. Furthermore, a novel AES cryptographic approach infused with ECG signal-based decryption/encryption key was offered as a security layer to protect data during transmission. Findings demonstrated that the strategy with or without the extra security layers might save around 68.2 & 72.4 per cent of system utilization, respectively, compared to local implementation.

In 2021, Sohani*et al.* [7] introduced the PMHEFT(Predictive Priority based Modified Heterogeneous Earliest Finish Time) method that might predict future resource demands for an application. This study adds to the development of a prediction-based framework for effective & flexible resources in system setting to meet end users' needs. This study presents the PMHEFT method, which improves LBin all VMs to reduce the makespan of a workflow management system. According to empirical observations, the power consumption, efficiency, &makespan of the suggested method were superior to previous methods.

In 2021, Negi *et al.* [8] described the CMODLB(Clustering Based Multiple Objective Dynamic Load Balancing Technique) approach, which seems to be a mix of supervised (ANN-Artificial Neural Network), unsupervised (clustering) ML (Machine Learning), as well as soft computing (interval type 2 fuzzy logic system)-based load balancing techniques. Originally, the BOEK-means (Bayesian Optimization based Enhanced K-means)method was used to cluster into overloaded &underloaded VMs employing the described ANN-LB approaches previously. The user tasks were assigned for underloading VMs in stage 2 to optimize load balancing & resource consumption. The CMODLB approach incorporated 31.067 percent & 71.6 percent less time to complete than TaPRA& BSO (Beetle Swarm Optimization), correspondingly, according to experimental measurements.

In 2019, Xu *et al.* [9] investigated the MVC (MC-aware VM Consolidation)problem and formulated it as a multi-constraint optimization method that considers migration costs & leftover VM runtime. A heuristic method named the MVC method was constructed based on the presented concept. Tests show that, when compared to several popular methods, the MVC method may effectively reduce migration costs while guaranteeing a minimal energy usage level.

In 2020, Neelima*et al.* [10] used ADA (Adaptive Dragonfly Algorithm) to attack a revolutionary load-balancing work scheduling method in the cloud. Moreover, ADA seems to be a combination of DA (Dragon Fly Algorithm) &FF (Firefly) models. Furthermore, a multi-objective function depending on 3 criteria, including completion time, processing costs, & load, was designed to achieve enhanced results. Furthermore, the effectiveness of the suggested technique was assessed using two parameters: execution cost & execution time. The test findings show that the adopted method achieves a better load balancing result than other methods.

In 2021, Mapetu*et al.* [11 adopted a dynamic VM consolidation model - based load balancing in a heterogeneous network to minimize the tradeoff among energy consumption, SLA violations, and VM migrations while maintaining minimal host shutdowns & lesser time difficulty. Furthermore, the Pearson correlation coefficient and imbalance degree in each host & VM correspond to CPU, RAM, & bandwidth. The achievement results of the suggested strategy for the NPproblem have been demonstrated through comprehensive analysis and experimental studies using random workloads and real PlanetLab.

In 2020, Kong *et al.* [12] have suggested a quick heuristic methodology based on the zero imbalance method as a novel notion in a diverse environment. This strategy seeks to reduce the variation in completion times across diverse VMs without using priority techniques or complex scheduling decisions that typically subject optimization algorithms to the cloud configuration. The experimental outcomes reveal that the suggested method addresses the NP-hard optimization issue more successfully than previous heuristic methods while meeting the needs of cloud providers and consumers.

In 2021, Annie *et al.* [13] used HHO-PIO(Harries Hawks Optimization-Pigeon Inspired Optimization)Algorithms to develop an effective load balancing method that provides optimal resource usage & task response time. The developed technique was implemented in the JAVA Net beans IDE, which would be integrated into the cloudsim architecture and evaluated using a variety of tasks to determine efficiency. Compared to other current approaches, the suggested technique was 97 percent efficient.

In 2019, Filiposka*et al.* [14] created a hierarchical method for migrations relying on a mix of network community & efficient packing techniques. It considers hierarchical VM migrations as an expansion to network consciousness for lesser latency communication, community-based placement technique, effective packing for maximum resource utilization, & excellent consolidation via migrations. The findings suggest that although initial placement was critical for optimal system performance, continual energy efficiency could only be accomplished by periodical consolidation through migrations.

In 2020, Jyoti *et al.* [15] have presented a novel strategy depending on load balancing & service brokering to offer dynamic resource provisioning. The MADRL-DRA(Multi-agent Deep Reinforcement Learning-Dynamic Resource Allocation) would be first utilized in the LUA (Local User Agent) to forecast the user task's environmental activities & assign the employment to VM priority. Then, cloud brokers act as middlemen among users and suppliers in the global agent. The suggested work demonstrated the traditional methodologies based on makespan, energy efficiency, throughput, resource usage, waiting time, and execution time.

In 2018, Joseph *et al.* [16] have suggested a neural network-based adaptive selection of VM consolidation techniques that dynamically picks the best algorithm based on the objective priority & environment factors of the cloud provider. Tests on PlanetLab VMs workload traces were used to assess the efficiency of the suggested systems for different assessment priorities. The findings reveal that the suggested scheme generates satisfactory adaptive outcomes based on assessment priority, as evidenced by a higher average score than independent techniques.

In 2018, Yogesh *et al.* [17] suggested a failure-aware VM consolidation technique that executes VM consolidation and considers the probability of failures and the hazard rate of physical resources. Furthermore, they suggested the smoothing-based failure prediction approach to activate 2 fault tolerance schemes (checkpointing & migration of VM). The findings show that by combining checkpointing and VM migration with the suggested technique, dependability was enhanced by 12%, the lowered energy consumption, and the frequency of failures decreased by 14%.

In 2020, Hamid *et al.* [18] have suggested a SLAV-minimizing energy-aware VM consolidation technique. There seem to be three stages to dynamic VM consolidation: i) predicting under -&over -utilized hosts; ii) choosing 1 or many VMs migration for such hosts; &iii) identifying target hosts for chosen VMs. Consequently, separate models for each stage were included in adopted VM consolidation. They created separate fine-tuned ML prediction schemes for specific VMs during the first step to anticipate the ideal moment to start migrations from hosts. According to the analysis, the VM consolidation technique reduced energy usage and SLAVs.

In 2018, Tarek *et al.* [19] used VM consolidation strategies focused on estimating desired resources & VM migration traffic in the future. They use KDE's (Kernel Density Estimation) strength as a strong tool for forecasting each VM's future resource utilization and the AKKA toolkit's strength as an actor-based model for communicating vital information about the host's states. Throughout the overall amount of migrations & energy usage, the acquired statistics demonstrate the strategy's success.

In 2018, Irfan *et al.* [20] used a VM consolidation approach to put idle physical servers into sleep mode, reducing power consumption. As a consequence, they call the method WAVMCM (Workload Aware Virtual Machine Consolidation Method). While comparing to a GA-based solution, experimental findings show that the suggested WAVMCM lowers active servers by 9% while saving 15% of electricity.

In 2020, Chakravarthy *et al.* [21] have suggested defining and proving that the issue of intermediate node selection in Scatter-Gather migration was NP-complete. Moreover, by lowering the 0–1 knapsack issue to MSSP (Minimum Subset Sum Problem)& subsequently MSSP to INS-SGM (Intermediate Node Selection in Scatter-Gather Migration), they show that the INS-SGM issue was NP-hard. To overcome the issue, two heuristic techniques have been suggested: least-increase-in-energy & maximum-decrease-in-eviction-time, while its effectiveness was evaluated using three parameters: overall migration, eviction time, & energy.

In 2018, Sivagami*et al.* [22] deployed a novel DFTM(Dynamic Fault Tolerant VM Migration) for enforcing cloud data centre infrastructure dependability using an improved VN (Virtual Network)demand recovery mechanism. For assessing route traffic, an integer linear programming model was used, incorporating all connected numerical aspects to select the best VM in the best way. Investigations show that the suggested fault recovery technique improves the VM-based cloud data centre's durability.

In 2019, Yashwant *et al.* [23] have used improved DTP & wireless PTP clock synchronization techniques to obtain higher accuracy in intra-cloud & inter-cloud data centre networks. In addition, messages were transferred directly between any 2 data centres rather than being forwarded because there was a direct link. Researchers also demonstrate the impact of different performance characteristics on data centre networking topologies using simulations & real-time trials.

In 2021, Hariharan *et al.* [24] suggested a multi-objective energy-efficient VM consolidation using the ABSO (Adaptive Beetle Swarm Optimization) method. The ABSO suggested was a cross between PSO & BSO. The suggested technique includes PSO & BSO operators, efficient solution representation & fitness function. The approach's efficacy was assessed using several assessment metrics, and its efficacy was evaluated to other techniques.

In 2021, Khan *et al.* [25] have considered an NVMC(Normalization based VM Consolidation) approach aimed at bringing VM online whilst reducing energy usage, SLA violations, and the amount of VM migrations. The CATR (Cumulative Available-to-Total Ratio)has been used to locate under-utilized hosts, whereas the comparative capacity of VM and hosts was determined for detecting the over-utilized hosts. Further, the findings show that the NVMC technique surpasses other well-known techniques in terms of energy usage, SLA breaches, and the amount of VM migrations by a large margin. Table 1 shows the review o f conventional methods.

Table 1 Reviews on conventional techniques

Author	Deployed schemes	Features	Challenges
Balaji *et al.* [1]	ACSO Algorithm	❖ Shortest makespan ❖ Uses less amount of energy	❖ Real time
Francis *et al.* [2]	FIMPSO	❖ Better response ❖ Better memory utilization ❖ Excellent makespan ❖ Reliability ❖ Throughput ❖	❖ Data deduplication algorithm
Mandeep *et al.* [3]	EA-LB algorithm	❖ Minimize energy usage ❖ Improve service quality ❖ Lower latency	❖ Nature-inspired load balancing approaches
Lin *et al.* [4]	MRLB & TB algorithm	❖ Saves energy and time ❖ Enhanced load balancing	❖ Online and dependent task scheduling ❖ Workload prediction for resizing containers
Tamilvizhi*et al.* [5]	PFTF algorithm	❖ Reduce network congestion	❖ Fault masking technique to improve tolerance even after a fault occurs. ❖ Improve Qos parameters
Zhang *et al.* [6]	Secure LBCO algorithm	❖ Enhanced security ❖ System utilization	❖ Enhance automated security related decision making
Sohani*et al.* [7]	PMHEFT algorithm	❖ Reduce makespan ❖ Power consumption ❖ Efficiency	❖ More efficient resource provisioning system for end users ❖ Proactive resource provisioning technique
Negi *et al.* [8]	CMODLB technique	❖ Resource consumption ❖ Optimized load balancing	❖ Improve energy efficiency ❖ Load balancing with machine learning
Xu *et al.* [9]	MVC algorithm	❖ Reduce migration cost ❖ Minimal level of energy usage	❖ Needs to focus on utilization
Neelima*et*	ADA-based multi-objective	❖ Better load balancing	❖ Need to focus on the insecure

al. [10]	load balancing method		interface
Mapetu*et al.* [11]	PBPSO-DCO algorithm	❖ Minimum host shutdowns ❖ Lesser time difficulty	❖ Security threats ❖ Real-world cloud computing
Kong *et al.* [12]	HLBZID algorithm	❖ Effective task scheduling ❖ Optimal completion time ❖ Earliest finish time ❖ Reduce makespan	❖ Power consumption in data centre ❖ VM live migration
Annie *et al.* [13]	Hybrid HHO-PIO load balancing method	❖ Optimal resource usage ❖ Task response time	❖ Needs to reduce the degree of imbalance
Filiposka*et al.* [14]	Network aware community based hierarchical approach	❖ Lesser latency ❖ Energy efficiency ❖ Reduce the carbon footprint	❖ Focus on Secure
Jyoti *et al.* [15]	MADRL-DRA algorithm	❖ Less makespan ❖ Energy efficiency ❖ Waiting time ❖ Resource usage	❖ Secure scheduling ❖ Needs to focus on mapping resources
Joseph *et al.* [16]	Neural network based adaptive selection based VM consolidation algorithm	❖ Higher average score ❖ Better assessment priority	❖ Better metric on SLA violation ❖ Data deduplication
Yogesh *et al.* [17]	Best fin bin packing algorithm	❖ Decrease in frequency of failures ❖ Enhanced dependability	❖ Failure correlation ❖ Fault tolerance in energy efficient manner
Hamid *et al.* [18]	ML models	❖ Reduced energy usage ❖ Reduce SLAVs	❖ Enhance the accuracy of VM CPU usage prediction
Tarek *et al.* [19]	KDE technique	❖ Reduce energy usage	❖ Reconfigurabl e prediction model ❖ Need to analyse on available information on each VM such as type, size, resource intensiveness etc
Irfan *et al.* [20]	WAVMCM	❖ Less power consumption ❖ Lower active server	❖ Needs to focus on efficiency
Chakravarthy *et al.* [21]	Max-Decrease in Eviction & Least increase in energy algorithm	❖ Eviction time ❖ Energy	❖ Minimize the excess bandwidth allocation for SGM ❖ Routing mechanism
Sivagami*et al.* [22]	DFTM algorithm	❖ Minimal complexity ❖ Better efficiency	❖ Rollback feature requirements
Yashwant *et al.* [23]	Enhanced DTP & wireless PTP based clock synchronization algorithms	❖ Higher accuracy ❖	❖ Security issues
Hariharan *et al.* [24]	ABSO algorithm	❖ High efficiency	❖ VM consolidation adaptive selectors in real cloud such as Open stack or Cloud stack ❖ Needs to focus on security issues
Khan *et al.* [25]	NVMC algorithm	❖ Energy usage ❖ SLA breaches ❖ Amount of VM migrations by a large margin	❖ Trade-off between performance and estimation overhead ❖ Threshold value needs to adapt dynamically

3. REVIEW ON DEVELOPED SCHEMES, PERFORMANCES & MAXIMUM ATTAINMENTS

3.1 Review of Adopted Techniques

This section discusses the methods used throughout the evaluated works based on VM migration &LB in the cloud, and also its schematic diagram is illustrated in Fig. 1. Moreover, the adopted techniques are classified based on LB and VM migration or VM consolidation schemes in the cloud. Here, the adopted techniques based on the load balancing scheme is determined in [1-8] [10] [12,13] and [15]. From the review, it was noted that the ACSO algorithm was used in [1]; the purpose of the proposed algorithm is to reduce power, migration cost and memory utilization while the load is balanced. Only with the support of ACSO are the methods effective just after fitness computation. The implemented FIMPSO method accomplished an effective average load for generating and enhancing the key parameters such as suitable resource use and task response time in [2].

EA-LB algorithm was exploited in [3], MRLBalgorithm and TB algorithms were used in [4], PFTF algorithm was exploited in [5], and secure LBCO algorithm was exploited in [6]. For multi-user, multi-task, multi-tier, mobile-edge cloud computing systems, the secure LBCO method defines the best LB and offloading decision.PMHEFT algorithm was adopted in [7], and the CMODLB technique was adopted in [8]. ADA-based multi-objective Load balancing method was used in [10], and the main objective of the proposed methodology is to assign the task to

VM using ADA, which minimizes the total execution time and cost while balancing the load. HLBZID (Heuristic Load Balancing based Zero Imbalance Mechanism) algorithm was exploited in [12]. In addition, a hybrid HHO-PIO Load Balancing Method was used in [13], and the MADRL-DRA algorithm was deployed [15]. Moreover, the adopted techniques based on VM migration or VM consolidation scheme is determined in [9] [11] [14] [16-25]. Furthermore, MVC Algorithm was exploited in [9] it decreases the migration cost and, at the same time, guarantees the energy consumption within a certain low level. Moreover, the PBPSO-DCO algorithm was deployed in [11]. Moreover, a network-aware community-based hierarchical approach was adopted in [14], and a neural network-based adaptive selection-based VM consolidation algorithm was deployed in [16]. In addition, the Best Fit Bin Packing algorithm, ML models, KDE technique, WAVMCM Method, and Max-Decrease-in-Eviction & Least-Increase-in-Energy algorithm were deployed in [17] [18] [19] [20] and [21], respectively. Consequently, the DFTM algorithm was adopted in [22], and enhanced DTP and wireless PTP-based clock synchronization algorithms were used in [23]. DFTM [22] algorithm-based Load Monitoring and Balancing Mechanism is implemented to monitor the workload of virtual links and distribute it to avoid overloads at any resources under VM employment and VM association. ABSO algorithm was deployed in [24], and the NVMC algorithm was adopted in [25]. The suggested ABSO [24] method combines the PSO & BSO algorithms. To conserve energy, the suggested ABSO moves the VM from the under-loaded PM towards the overloaded PM and then turns off the undesired PM.

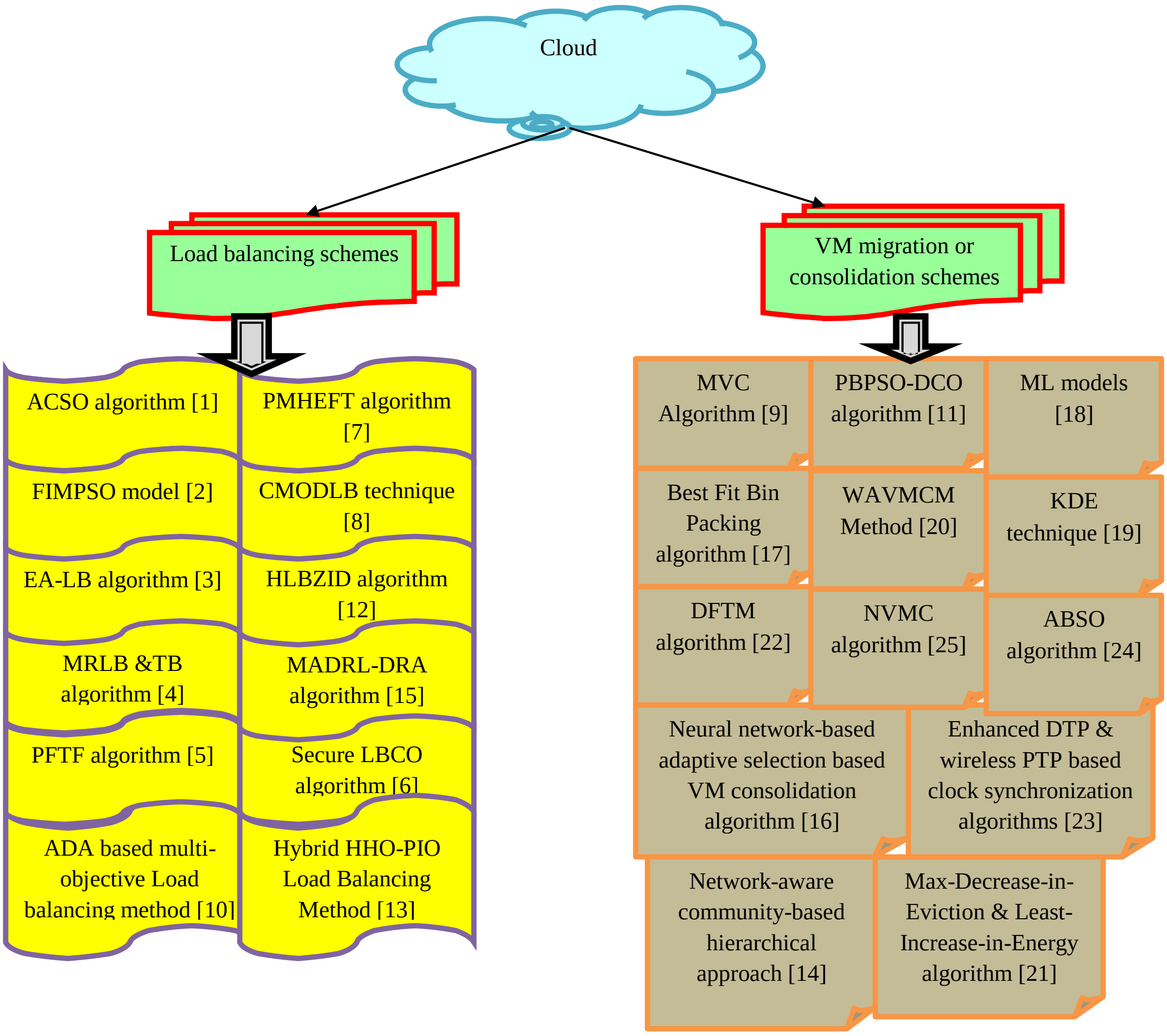

Fig 1 Pictorial representation of area coverage maximization models in WSNs

3.2 Analysis of Performance metrics

The performance metrics utilised for VM migration &LB in the cloud are listed in Table 2. Table 2 shows that 9 articles examined energy consumption, accounting for approximately 36% of the total contribution. In comparison, 7 articles explored bandwidth, accounting for about 28% of the total contribution. In addition, five studies investigated time consumption, accounting for around 20% of the total contribution. Similarly, makespan & response time was examined in 16% (4 articles) of the total contribution. In addition, 3 articles account for around 12% of the total contribution focused on CPU utilisation and execution time. The number of migrations, processing speed, efficiency, time delay, & cost was evaluated in two articles that contributed around 8%. Furthermore, the reliability, sensitivity, specificity, scalability, power consumption, storage capacity, transmission time, migration cost, SLA violation, execution cost, memory utilisation, throughput, run time, resource utilisation, as well as waiting time each contributed about 4% of the total contribution (1 article). In addition, one article accounting for almost 4% of the total contribution examined the prediction accuracy, link speed, SLAVs value, average 1st rank occurrence, error rate, saved power, number of active sensors, average relative score, VM size, migration frequency, simulation time, average rank, convergence time gain, and memory usage.

Table:2 Review of different performance metrics for VM migration and load balancing in the cloud

Measure	Citation
Energy consumption	[1] [3] [4] [11] [14] [18] [19] [24] [25]
Bandwidth	[3] [6] [7] [8] [12] [15] [25]
Time consumption	[1] [4] [20] [21] [24]
Makespan	[1] [4] [7] [15]
Response time	[2] [4] [13] [22]
CPU utilization	[2] [16] [19]
Execution time	[5] [10] [22]
Number of migrations	[11] [18]
Processing speed	[12] [15]
Efficiency	[13] [15]
Time delay	[3] [6]
Cost	[5] [24]
Reliability	[2]
Sensitivity	[5]
Specificity	[5]
Scalability	[5]
Power consumption	[7]
Storage capacity	[8]
Average relative score	[16]
Migration cost	[9]
SLA violation	[9]
Execution cost	[10]
Memory utilization	[2]
Throughput	[2]
Run time	[4]
Resource utilization	[15]
Waiting time	[15]
Error rate	[19]
Transmission time	[8]
Convergence time gain	[23]
Prediction accuracy	[17]
Link speed	[17]
SLAVs value	[18]
Average 1st rank occurrence	[16]
Saved power	[20]
Number of active sensors	[20]
VM size	21]
Migration frequency	[22]
Simulation time	[23]
Average rank	[16]
Memory usage	[24]

3.3 Analysis of Energy Consumption

This section's describes the maximum energy consumption performance attained in the reviewed works, which is illustrated in Table 3. From Table 3, it can be noticed that minimal energy of 6.235J is attained in [24], when compared with other works in [1], [3], [4], [11], [18], [19] and [25].

Table:3 Maximum performance of energy consumption in the reviewed papers

Sl. no	Citation	Maximal performance
1	[1]	6.1%
2	[3]	7.225J
3	[4]	1178.37(KWh)
4	[11]	110(KWh)
6	[18]	$6.8 \times 10{-}4$
7	[19]	413KW h
8	[24]	6.235J
9	[25]	91.58 kWh

3.4 Analysis of Makespan

This section's describes the performance of makespan attained in the reviewed works, as shown in Table 4. From Table 4, it is clear that the makespan of 178sec is attained in [1], which is the minimum value when compared with other works in [1], [4], [7] and [15].

Table:4 Makespan obtained in the examined papers

S.no	Citation	Maximal performance
1	[1]	178sec
2	[4]	900 sec
3	[7]	300 sec
4	[15]	589 sec

3.5 Analysis of Maximum Performance

The maximum performance based on VM migration and load balancing in the cloud obtained in the reviewed works is represented in Table 5. From the review, area coverage measured in [24] has obtained a minimal value of 6.235J and bandwidth analysed in [8] has attained a higher value of 1, 00, 000 Mbps. Moreover, Makespan measured in [1] has obtained a larger value of 178secand CPU utilization examined in [16] [19] has obtained a higher value of 100%. ASCO method attains the maximum makespan than using CSO-based LB& GA-based LB. Similarly, time consumption, response time, memory utilization, reliability, throughput, time delay, run time, execution time, cost, sensitivity, specificity, and scalability has attained the values of 200-300sec, 13.58ms, 93%, 67%, 72%, 15ms, 14sec, 109ms, 1.4$, 92.91%, 81.23%, and 93.75% and it has been examined in [14] [2] [2] [2] [2] [6] [4] [22] [24] [5] [5] and [5] correspondingly. The FIMPSO algorithm [2] yielded an effective result with the least average response time, memory utilization, reliability and throughput, which was superior to all the other compared methods. The measures such as power consumption, storage capacity, transmission time, migration cost, SLA violation, execution cost, number of migrations, processing speed, efficiency, resource utilization, waiting time, average 1st rank occurrence, average relative score, average rank, and prediction accuracy have attained better values of 150000Kwh, 11TB, 0.02s, 113469, 0.41%, 0.0824, 2803, 2000-16000MIPS, 97%, 99%, 98sec, 49.08%, 92.09%, 1.88, and 71% and they have been analysed in [7] [8] [8] [9] [9] [10] [18] [12] [13] [15] [15] [16] [16] [16] and [17], respectively. Compared to the PSO &TOPSIS-PSO method, the CMODLB [8] transmission time is 1.30 s. The total relative score of a technique is divided by the total of simulation runs of the technique to get the average relative score. Several existing algorithms yield lower average 1st rank occurrence, average relative score, and average rank than the neural network-based adaptive selection-based VM consolidation algorithm [16].Also, the link speed, SLAVs value, error rate, saved power, number of active sensors, VM size, migration frequency, simulation time, convergence time gain, and memory usage were exploited in [17] [18] [19] [20] [20] [25] [22] [23] [23] and [24], and they have attained optimum values of 1Gbps, 0.188×10-5, 0.1%, 33%, 70, 2500MB, 11.26, 5000sec, 59%, and 12326MB, correspondingly. The proposed WAVMCM [20] reduces active servers and saves power compared to the GA-based method. The recommended ABSO [24] model takes less memory for scheduling certain tasks than the BSO-based scheduling, PSO-based scheduling, and GA-based scheduling.

Table: 5 Maximum performance obtained in the examined papers

Sl. no	Citation	Performance metrics	Maximal performance
1	[24]	Energy consumption	6.235J
2	[8]	Bandwidth	1,00,000Mbps
3	[1]	Makespan	178sec
4	[14]	Time consumption	200-300sec
5	[2]	Response time	13.58ms
6	[16] [19]	CPU utilization	100%
7	[2]	Memory utilization	93%
8	[2]	Reliability	67%
9	[2]	Throughput	72%
10	[6]	Time delay	15ms
11	[4]	Run time	14sec
12	[22]	Execution time	109ms
13	[24]	Cost	1.4$
14	[8]	Storage capacity	11TB
15	[5]	Specificity	81.23%
16	[9]	Migration cost	113469
17	[7]	Power consumption	150000Kwh
10	[5]	Sensitivity	92.91%
11	[8]	Transmission time	0.02s
12	[5]	Scalability	93.75%
13	[9]	SLA violation	0.41%
14	[10]	Execution cost	0.0824
15	[18]	Number of migrations	2803
16	[12]	Processing speed	2000-16000MIPS
17	[13]	Efficiency	97%
18	[15]	Resource utilization	99%
19	[15]	Waiting time	98sec
20	[16]	Average 1st rank occurrence	49.08%
21	[16]	Average relative score	92.09%
22	[16]	Average rank	1.88
23	[17]	Prediction accuracy	71%
24	[17]	Link speed	1Gbps
25	[18]	SLAVs value	0.188×10^{-5}
26	[19]	Error rate	0.1%
27	[20]	Saved power	33%
28	[20]	Number of active sensors	70
29	[25]	VM size	2500MB
30	[22]	Migration frequency	11.26
31	[23]	Simulation time	5000sec
32	[23]	Convergence time gain	59%
33	[24]	Memory usage	12326MB

4. CHRONOLOGICAL REVIEW AND SIMULATION TOOL USED IN REVIEWED PAPERS

4.1 Chronological Review

The possible papers based on VM migration and LB in the cloud have been collected and published in different years. The depiction of a chronological review is shown in Fig. 2. In 2021, 32% of publications (8 articles) were examined. Furthermore, in 2020, 28% of publications (7 articles) will be assessed. In addition, 2019 contributed 20% of the total evaluated papers (5 articles). Likewise, in 2018, 20% of publications (5 articles) on VM migration &LB in the cloud were analyzed.

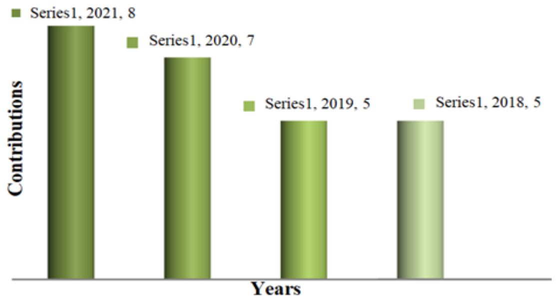

Fig 2 Representation of chronological review

4.2 Review on a simulation tool

Based on simulation tools used in each article, recent studies regarding VM migration & LB in the cloud are examined. The simulation tools utilised in each article are represented in Fig. 3. Initially, 10 papers [5] [7] [8] [10] [12] [13] [18] [19] and [24] has used cloudsim (Java) as a simulation tool. Further, 7 papers [1] [4] [9] [14] [16] [22] has [25] has contributed normal cloud sim simulator tool. Moreover, three papers [1] [6] and [20] have adopted MATLAB as a simulation tool. Likewise, two papers [15] and [17] have adopted the clouds (MatLab) as a simulation tool. Moreover, 1 paper [21] has adopted Java as a simulation tool. Also, 2 papers [3] and [23] have used other simulation tools

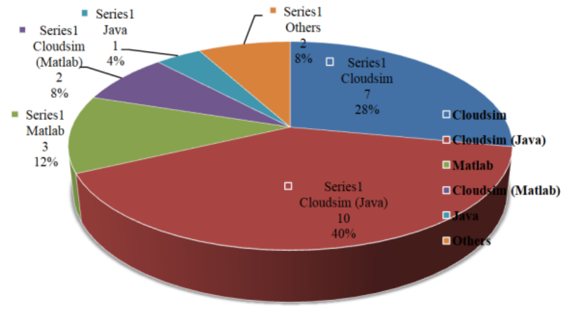

Fig 3 Review of Simulation Tool

5. RESEARCH GAPS AND CHALLENGES

Cloud computing research [32] is still in its early stages, and many technical issues in the research community remain unknown. With the advent of computer technology, cloud computing has developed an innovative approach to user services, permitting consumers to access Information Systems on a compensation basis at any location and any time. Because of the flexibility of cloud services, more organizations are migrating to the cloud, and service providers are expanding their data centres to customer services. Nevertheless, providing cost-effective job execution and optimal resource use is critical. Numerous approaches depending on resource management, workload management, quality of service, job scheduling, &LB have been established in review to optimize performance and resource consumption. Load balancing in the cloud [33] helps data centers to prevent virtual machine overloading/underloading, which is a difficulty in and of itself in the field of cloud computing. Consequently, researchers & developers must implement & construct an LB appropriate for distribution as well as parallel cloud systems.

In terms of practical and theoretical aspects, cloud computing [34] technology is becoming the focus of increasingly sophisticated study in data and computation. However, cloud computing research is beset by problems, with LBas the most pressing difficulty that requires particular concentration. In order to identify the best potential solution for enhancing cloud resource utilization, additional concerns such as virtual machine (VM) migration [35], VM security, user QoS satisfaction, and resource use must also be addressed. For computational purposes, data centers are generally scattered globally in the cloud. Certain load-balancing strategies are built for a narrower region. They do not consider aspects like distance among distributed computing nodes, communication delay, network latency, and distance among users& resources, among others. Researchers thought about homogenous nodes throughout early LB studies in the cloud. However, user needs vary continuously in cloud computing, necessitating its execution on diverse nodes to less response time & higher resource utilization. Thus, researchers have a problem developing effective load-balancing strategies [36] for diverse environments. Furthermore, cloud services' on-demand availability and scalability allow customers to access the services scale up or down fast at any time. To support these changes efficiently, a competent LB must consider a largermodification in demands based onsystem structure, storage, processing power, etc.

6. CONCLUSION

This survey has reviewed VM migration and load balancing in the cloud. Finally,

- The analysis has reviewed the performance measures, and its maximum achievements were contributed by various VM migration and load balancing schemes in the cloud.
- Further, the chronological reviews and simulation tools used in the existing works were analyzed.
- Finally, this paper has described the research gaps & challenges that may be helpful for researchers to carry out further work on VM migration and load balancing in the cloud.

7. REFERENCES

[1] K. BalajiP. Sai KiranM. Sunil Kumar, "An energy efficient load balancing on cloud computing using adaptive cat swarm optimization", Materials Today: ProceedingsAvailable online 7 January 2021In press, corrected proof

[2] A. Francis Saviour DevarajMohamed ElhosenyK. Shankar, "Hybridization of firefly and Improved Multi-Objective Particle Swarm Optimization algorithm for energy efficient load balancing in Cloud Computing environments", Journal of Parallel and Distributed Computing11 April 2020Volume 142 (Cover date: August 2020)Pages 36-45

[3] Mandeep KaurRajni Aron, "Energy-aware load balancing in fog cloud computing", Materials Today: ProceedingsAvailable online 17 December 2020In press, corrected proof

[4] Lin, W., Peng, G., Bian, X. et al. Scheduling Algorithms for Heterogeneous Cloud Environment: Main Resource Load Balancing Algorithm and Time Balancing Algorithm. J Grid Computing 17, 699–726 (2019). https://doi.org/10.1007/s10723-019-09499-7.

[5] Tamilvizhi, T., Parvathavarthini, B. A novel method for adaptive fault tolerance during load balancing in cloud computing. Cluster Comput 22, 10425–10438 (2019). https://doi.org/10.1007/s10586-017-1038-6.

[6] W. -Z. Zhang et al., "Secure and Optimized Load Balancing for Multitier IoT and Edge-Cloud Computing Systems," in IEEE Internet of Things Journal, vol. 8, no. 10, pp. 8119-8132, 15 May15, 2021, doi: 10.1109/JIOT.2020.3042433.

[7] M. Sohani and S. C. Jain, "A Predictive Priority-Based Dynamic Resource Provisioning Scheme With Load Balancing in Heterogeneous Cloud Computing," in IEEE Access, vol. 9, pp. 62653-62664, 2021, doi: 10.1109/ACCESS.2021.3074833.

[8] Negi, S., Rauthan, M.M.S., Vaisla, K.S. et al. CMODLB: an efficient load balancing approach in cloud computing environment. J Supercomput 77, 8787–8839 (2021). https://doi.org/10.1007/s11227-020-03601-7

[9] Xu, H., Liu, Y., Wei, W. et al. Migration Cost and Energy-Aware Virtual Machine Consolidation Under Cloud Environments Considering Remaining Runtime. Int J Parallel Prog 47, 481–501 (2019). https://doi.org/10.1007/s10766-018-00622-x

[10] Neelima, P., Reddy, A.R.M. An efficient load balancing system using adaptive dragonfly algorithm in cloud computing. Cluster Comput 23, 2891–2899 (2020). https://doi.org/10.1007/s10586-020-03054-w

[11] Mapetu, J.P.B., Kong, L. & Chen, Z. A dynamic VM consolidation approach based on load balancing using Pearson correlation in cloud computing. J Supercomput 77, 5840–5881 (2021). https://doi.org/10.1007/s11227-020-03494-6

[12] Kong, L., Mapetu, J.P.B. & Chen, Z. Heuristic Load Balancing Based Zero Imbalance Mechanism in Cloud Computing. J Grid Computing 18, 123–148 (2020). https://doi.org/10.1007/s10723-019-09486-y

[13] Annie Poornima Princess, G., Radhamani, A.S. A Hybrid Meta-Heuristic for Optimal Load Balancing in Cloud Computing. J Grid Computing 19, 21 (2021). https://doi.org/10.1007/s10723-021-09560-4

[14] Filiposka, S., Mishev, A. & Gilly, K. Multidimensional hierarchical VM migration management for HPC cloud environments. J Supercomput 75, 5324–5346 (2019). https://doi.org/10.1007/s11227-019-02799-5

[15] Jyoti, A., Shrimali, M. Dynamic provisioning of resources based on load balancing and service broker policy in cloud computing. Cluster Comput 23, 377–395 (2020). https://doi.org/10.1007/s10586-019-02928-y

[16] Joseph Nathanael WitantoHyotaek LimMohammed Atiquzzaman, "Adaptive selection of dynamic VM consolidation algorithm using neural network for cloud resource management", Future Generation Computer Systems3 May 2018Volume 87 (Cover date: October 2018)Pages 35-42

[17] Yogesh SharmaWeisheng SiBahman Javadi, "Failure-aware energy-efficient VM consolidation in cloud computing systems", Future Generation Computer Systems21 December 2018Volume 94 (Cover date: May 2019)Pages 620-633

[18] Seyedhamid Mashhadi MoghaddamMichael O'SullivanCharles Peter Unsworth, "Embedding individualized machine learning prediction models for

energy efficient VM consolidation within Cloud data centers", Future Generation Computer Systems15 January 2020Volume 106 (Cover date: May 2020)Pages 221-233

[19] Tarek MahdhiHaithem Mezni, "A prediction-Based VM consolidation approach in IaaS Cloud Data Centers", Journal of Systems and Software28 September 2018Volume 146 (Cover date: December 2018)Pages 263-285

[20] Irfan MohiuddinAhmad Almogren, "Workload aware VM consolidation method in edge/cloud computing for IoT applications", Journal of Parallel and Distributed Computing9 October 2018Volume 123 (Cover date: January 2019)Pages 204-214

[21] Chakravarthy Sudarshan AnnadanamSudhakar ChapramRamesh T, "Intermediate node selection for Scatter-Gather VM migration in cloud data center", Engineering Science and Technology, an International Journal2 March 2020Volume 23, Issue 5 (Cover date: October 2020)Pages 989-997

[22] V. M. SivagamiK. S. Easwarakumar, "An Improved Dynamic Fault Tolerant Management Algorithm during VM migration in Cloud Data Center", Future Generation Computer Systems23 November 2018Volume 98 (Cover date: September 2019)Pages 35-43

[23] Yashwant Singh PatelAditi PageSajal K. Das, "On demand clock synchronization for live VM migration in distributed cloud data centers", Journal of Parallel and Distributed Computing19 December 2019Volume 138 (Cover date: April 2020)Pages 15-31

[24] Hariharan, B., Siva, R., Kaliraj, S. et al. ABSO: an energy-efficient multi-objective VM consolidation using adaptive beetle swarm optimization on cloud environment. J Ambient Intell Human Comput (2021). https://doi.org/10.1007/s12652-021-03429-w

[25] Khan, M.A. An efficient energy-aware approach for dynamic VM consolidation on cloud platforms. Cluster Comput 24, 3293–3310 (2021). https://doi.org/10.1007/s10586-021-03341-0

[26] RahimiZadeh, K., Dehghani, A. Design and evaluation of a joint profit and interference-aware VMs consolidation in IaaS cloud datacenter. Cluster Comput 24, 3249–3275 (2021). https://doi.org/10.1007/s10586-021-03310-7

[27] Khattar, N., Singh, J. & Sidhu, J. An Energy Efficient and Adaptive Threshold VM Consolidation Framework for Cloud Environment. Wireless Pers Commun 113, 349–367 (2020). https://doi.org/10.1007/s11277-020-07204-6

[28] Mohammadi Bahram Abadi, R., Rahmani, A.M. & Hossein Alizadeh, S. Self-adaptive architecture for virtual machines consolidation based on probabilistic model evaluation of data centers in Cloud computing. Cluster Comput 21, 1711–1733 (2018). https://doi.org/10.1007/s10586-018-2806-7

[29] Tarafdar, A., Debnath, M., Khatua, S. et al. Energy and quality of service-aware virtual machine consolidation in a cloud data center. J Supercomput 76, 9095–9126 (2020). https://doi.org/10.1007/s11227-020-03203-3

[30] Afzal, S., Kavitha, G. Load balancing in cloud computing – A hierarchical taxonomical classification. J Cloud Comp 8, 22 (2019). https://doi.org/10.1186/s13677-019-0146-7

[31] Pourghebleh, B., Aghaei Anvigh, A., Ramtin, A.R. et al. The importance of nature-inspired meta-heuristic algorithms for solving virtual machine consolidation problem in cloud environments. Cluster Comput 24, 2673–2696 (2021). https://doi.org/10.1007/s10586-021-03294-4

[32] Sayadnavard, M.H., Toroghi Haghighat, A. & Rahmani, A.M. A reliable energy-aware approach for dynamic virtual machine consolidation in cloud data centers. J Supercomput 75, 2126–2147 (2019). https://doi.org/10.1007/s11227-018-2709-7

[33] Li, S., Zhai, D., Du, P. et al. Energy-efficient task offloading, load balancing, and resource allocation in mobile edge computing enabled IoT networks. Sci. China Inf. Sci. 62, 29307 (2019). https://doi.org/10.1007/s11432-017-9440-x

[34] Seyedhamid Mashhadi MoghaddamMichael O'SullivanCameron Walker, "Metrics for improving the management of Cloud environments — Load balancing using measures of Quality of Service, Service Level Agreement Violations and energy consumption", Future Generation Computer Systems24 April 2021Volume 123 (Cover date: October 2021)Pages 142-155.

[35] Kansal, N.J., Chana, I. An empirical evaluation of energy-aware load balancing technique for cloud data center. Cluster Comput 21, 1311–1329 (2018). https://doi.org/10.1007/s10586-017-1166-z.

[36] M. Ala'anzy and M. Othman, "Load Balancing and Server Consolidation in Cloud Computing Environments: A Meta-Study," IEEE Access, vol. 7, pp. 141868-141887, 2019, doi: 10.1109/ACCESS.2019.2944420