

## Research Article

# Preserving Resource Handiness and Exigency-Based Migration Algorithm (PRH-EM) for Energy Efficient Federated Cloud Management Systems

R. Karthikeyan <sup>1</sup>, B. Sundaravadivazhagan <sup>2</sup>, Robin Cyriac <sup>2</sup>,  
Praveen Kumar Balachandran <sup>3</sup> and S. Shitharth <sup>4</sup>

<sup>1</sup>Department of CSE (AI&ML), Vardhaman College of Engineering, Hyderabad, TS 501218, India

<sup>2</sup>Department of IT, University of Technology and Applied Sciences, Al Mussanah, Oman

<sup>3</sup>Department of EEE, Vardhaman College of Engineering, Hyderabad, TS 501218, India

<sup>4</sup>Department of CSE, Kebri Dehar University, Somali, Ethiopia

Correspondence should be addressed to S. Shitharth; shitharths@kdu.edu.et

Received 30 August 2022; Revised 28 October 2022; Accepted 25 November 2022; Published 17 February 2023

Academic Editor: Shuai Zhao

Copyright © 2023 R. Karthikeyan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

On-demand computing ability and efficient service delivery are the major benefits of cloud systems. The limitation in resource availability in single data centers causes the extraction of additional resources from the cloud providers group. The federation scheme dynamically increases resource availability in response to service requests. The dynamic increase in resource count leads to excessive energy consumption, maximum cost, and carbon footprints emission. Hence, the reduction of resources is the major requirement to construct the optimized cloud source models for profit maximization without considering energy mix and CO<sub>2</sub>. This paper proposes the novel migration method to reduce carbon emissions and energy consumption. The initial stage in the proposed work is the categorization of data centers based on the MIPS and cost prior to job allocation offers scalable and efficient services and resources to the cloud user. Then, the job with the maximum size is allotted to the VM only if its capacity is less than the cumulative capacity of data centers. A novel migration based on overutilized and underutilized levels provides the services to the user even if the particular VM fails. The proposed work offers efficient maintenance of resource availability and maximizes the profit of the cloud providers associated with the federated cloud environment. The comparative analysis of the proposed algorithm with the existing methods regarding the response time, accuracy, profit, carbon emission, and energy consumption assures the effectiveness in a confederated cloud environment.

## 1. Introduction

Cloud computing is an architectural framework that helps to construct service-based applications and provide the cost-effective outsourcing to dynamic service environments. Among these service models, IAAS allows resource sharing to the customers in virtual machine (VM) form. The resource provisioning scheme greatly impacts on increasing the profit of IAAS providers. The assurance of quality of service (QoS) constraints with the service level agreement (SLA) agreed upon by customers depends on the effective resource management policies. Depending on the resource

performance level, the relaxation of QoS constraints is required to process a number of requests simultaneously. The trade-off between the QoS guarantee and the minimum number of requests require a dynamic increase of available resources. The increasing scenario requires the attainment of coordination between the providers that refers to cloud federation. During situations such as the reception of on-demand requests and the absence of idle resources in a federated cloud environment, the provider makes the decision between the spot price increment or spot VM termination. Research studies address the several policies to govern the decision-making process.

The extension of cloud computing introduces the cloud data center as the new research area regarding minimum energy consumption and revenue improvement. The rise of service demand by the data center management unit increases the complexity, size, and energy. The virtualization concept evolution in research studies reduces energy consumption effectively. The social and political pressure forces the analysis of CO<sub>2</sub> emissions to protect the environment from excessive energy consumption. The processor time and the energy consumption are interrelated and such relationship plays a major role in energy consumption measurement. Research studies introduce energy-aware approaches such as power-efficient server prediction, optimal workload placement, and application scheduling. During the implementation of energy-aware models, the estimation of energy consumption is based on the assumption such that the power consumption is proportional to the processor time. But the energy-aware VM model depicts that the processor time is not a required criterion to estimate the power accurately.

The VM management with the established rules maintains the energy and footprint as low as possible. Based on the energy objectives, the ad-hoc VM placement algorithm includes the data center and workload properties to reduce the consumption level. The workload characteristics and the hardware refreshing make the data center as unique which is not appropriate in an ad-hoc VM placement algorithm since the ad-hoc nature prevents upgrading according to the data center properties. Alternatively, the VM allocation algorithm is flexible to address the complexities and dynamic changes. Based on multispecific hardware and SLAs, the energy-aware algorithm provides additional energy savings by using the fine-tuning process. The requirements for the implementation of fine-tuning process are as follows: (i) VM is extensive with new SLA declaration for new service and (ii) adaptability with the data center. The utilization of constraint programming (CP) and its relevant algorithms addresses the user requirements, namely, performance and fault tolerance. But the lack of energy models counteracts the explicit energy-related concerns that play a major role in data center upgradation and capacity improvement.

The adaptability of hosts in data centers with another host in VM depends on the resource profile creation based on load management schemes. Each load management scheme should include heterogeneity, hardware diversity, fluctuations in load patterns, and energy consumption. The centralized approaches proposed in traditional research studies are not enough scalable since they require multiple distributed host monitoring which is complex during the stressful condition of the data center. The increase in resource utilization level efficiently reduces the cloud operational cost. The lack of proper management during virtualization within cloud data centers degrades the cloud operating performance. The rise of the VM migration concept supports effective resource management with the elimination of human supervision. The major categories of migration patterns are live and nonlive. VM live migration enables dynamic load management on data centers and allows the migration of one VM to another VM without

suspension of application services compared to nonlive migration. The energy consumption of the VM during the idle state and the data degradation are the major issues in a federated cloud environment. Hence, the research work proposed in this paper reduces energy consumption and improves profit with the novel VM migration model.

- (i) The grouping of data centers based on their MIPS and cost prior to job allocation supports the scalable service delivery to the cloud users
- (ii) The migration based on the VM status (overutilized/underutilized) proposed in this work is responsible for service provision during the failure condition
- (iii) The novel migration and federated cloud environment creation in this paper maintain the resource availability to increase the profit

This paper is organized as follows. Section 2 describes the related works on energy-efficient scheduling and migration techniques for the federated cloud environment. Section 3 discusses the proposed workload-based VM placement/migration and priority-based job allocation algorithm. Section 4 presents the performance analysis of the proposed algorithm over the existing migration techniques. Finally, Section 5 presents the conclusion.

## 2. Related Work

Efficient service delivery by using the resource provisioning on demand conditions governed by the paradigm called cloud computing. The coordination between the service providers increases resource availability dynamically and it plays a major role in the construction of a federated cloud environment. Toosi et al. [1] illustrated the policies to increase resource utilization and profit. The providers who participated in the federated cloud environment had diverse choices that make the decision policies not suitable. The large-size data transmitted in the federated cloud environment consumed more time. Celesti et al. [2] analyzed the potentiality of cloud federated architectures and proposed a more efficient cloud service provisioning strategy with the consideration of the WEB TV test case. The challenges observed from satellite applications are the consideration of application deployment and the migrating services for spanning clouds. Paraiso et al. [3] presented the federated multicloud environment that addressed three foundations: open service model, configurable architecture, and infrastructure service. Cloud federations depend on the aggregation of IAAS providers with their own capabilities. Kertesz et al. [4] introduced the cloud management solution based on the utilization level of cloud brokers. They utilized an integrated monitoring approach that enabled provider selection enhancement and cloud service executions. The provision of elastic on-demand computing ability to support the service delivery. Petri et al. [5] evaluated the federation establishment and determined the impact of policies on the system status by using comet cloud-based implementation. The utilization of special gateways supported the comet cloud implementation. The large deployment of data centers

to deliver various services to the user consumed more energy and power. Hence, development techniques were required to construct eco-friendly cloud computing.

Wadhwa and Verma [6] proposed the carbon-di-oxide ( $\text{CO}_2$ ) emission rate-based new technique to reduce energy consumption in data centers. The distributed data centers used in this technique have different energy sources and carbon footprint rates that affected the VM placement. Wadhwa and Verma [7] proposed the carbon-efficient VM placement and migration (CEPM) algorithm for the optimization of the VM placement problem. The current utilization level of servers was considered to improve efficiency. The metrics such as efficiency/scalability of data center and the performance of hosted applications are dependent on resource allocation. Ferdous et al. [8] presented the necessary background and characteristics of various components for VM placement and migration techniques. The VM management by rule-based approaches was responsible for the minimization of energy consumed by each data center. The rule-based approaches were not applicable for additional energy saving. Dupont et al. [9] proposed the energy-aware VM placement algorithm called Plug4Green that is used to compute the suitable location for the VM and the state of the servers depending on large size constraints. The QoS constraints ensure caused maximum energy consumption. Hence, a suitable technique was required to provide the balance between the minimum energy consumption and an accurate assurance of QoS constraints. Horri and Dastghaibifard [10] proposed a novel study of QoS-aware VM consolidation to provide the necessary trade-off between QoS constraints and energy consumption. They adopted the new method based on historical data existing in CPU and VM memory. The increase in scalability of cloud data centers maximized energy consumption, and a suitable resource allocation strategy was required to reduce the energy consumption.

The traditional data center models have not included energy consumption as a key parameter for their configuration. Dupont et al. [11] presented the energy-aware framework for VM reallocation. They employed the constraint programming (CP) and entropy-based approaches to achieve the flexibility of the model. The evolution of high-throughput computing resource consolidation models effectively reduced energy consumption with idle resource problems. Ding [12] discussed the particle swarm optimized tabu (PSO-T) search for resource utilization level improvement in order to reduce energy consumption. The PSO-T algorithm turn-off the sparse servers to reduce the power and time consumption level. The sustainability of the cloud model was the major concern to develop energy efficiency and  $\text{CO}_2$  emissions. Wajid et al. [13] considered the usefulness of sustainability of the cloud models to allow the conception and development of new techniques. The optimization of assets associated with sustainability cloud models was an important requirement to improve energy efficiency. Volk et al. [14] discussed the energy-efficient approach to Eco2Clouds projects under minimum energy consumption and  $\text{CO}_2$  emissions. The cloud environment monitoring was based on data gathering regarding energy

consumption with workload variations. Based on the processor's time and VM instances, the cost of virtualization varied since the cooling and energy costs for data centers exceeded the purchasing cost. Kim et al. [15] measured the energy consumption level based on the processing events. They utilized a scheduling algorithm to provide the necessary resources with an energy budget basis. The cloud service utilization level improvement required the resource optimization approach in addition to scheduling.

The simultaneous multiple cloud service requests were handled by parallel processing and required suitable resource allocation and scheduling of tasks. Li et al. [16] proposed resource allocation algorithms with preemptable tasks that adjusted the allotted resource dynamically. The periodical update regarding the task execution was required for the dynamic operation. The makespan and energy consumption were more in pre-emptive models. Neschachnow et al. [17] introduced meta-broker (level I) and local providers (level II) to schedule all the received tasks. They investigated the energy consumption and capacity of the resources from the multicore processing systems. The advertisement of the illusion of resources to the customers required higher quality and reliability levels leads to excessive energy consumption. Benyi et al. [18] proposed the pliant system-based VM scheduling approach to reducing energy consumption. They designed the cloud-sim-based simulation environment to evaluate the performance of the pliant-system. The nonadaptability of scheduling algorithms to uncertain and the dynamic nature made the job scheduling as a problematic task. Miranda et al. [19] presented the scheduling problems and reviewed the scheduling algorithms to provide the solution to uncertainty. The evolution of VM migration techniques from one physical machine to another in order to reduce time and service degradation. Soni and Kalra [20] discussed the various migration techniques (offline and live migration) to reduce the overall time consumption. They also presented the various live migration scenarios for better performance with minimum bandwidth. The distinct participants with their own objectives required the multiagent system with negotiation capabilities.

Leite et al. [21] proposed the server consolidation approach called federated application provisioning (FAP) strategy to manage the power consumption level in virtualized federated cloud environments. The major objective of the consolidation approach was to provide the trade-off between the QoS constraints satisfaction and minimum energy consumption compared to the trivial approach. The increase in processors provided the chance for failure occurrence. The application running in a cloud environment was represented by the workflow. But the inclusion of link failure and service provision failure violated the robustness of the computing environment. Singh and Kinger [22] discussed the failure removal with the fault tolerant mechanism (FTM). Failure detection was considered as an important stage in the virtualization process, and it required an effective control scheme with parametric guidance. The balance and consolidation of workloads with energy minimization required problem-solving techniques.

Garcia and Nafarrate [23] proposed the novel load-balancing heuristic algorithm to migrate the VM from the overutilized host to the underutilized host. The increase in cloud resource utilization level reduced the operational cost effectively. Liaqat et al. [24] surveyed the migration techniques and policies to represent future challenges in the VM domain. They proposed the queue-based migration model for memory page migration. The incorporation of virtualization and consolidation approaches were not provided the necessary balance between minimum energy consumption and better execution performance. Kaur and Chana [25] proposed the green cloud scheduling model (GCSM) that exploited the heterogeneity property of tasks and resources by using the scheduler unit. The GCSM facilitated energy efficiency and prevented the degradation on provider perspective. Alternatively, the execution of tasks within the time limit was achieved by GCSM under the client perspective. Efficient resource utilization was the challenging issue in the task execution performance improvement. Rathor et al. [26] provided the best fit and worst fit techniques to improve the resource utilization level at a high cost. Kruekaew and Kimpan [27] discussed the artificial bee colony algorithm with three scenarios first come first serve (FCFS), shortest job first (SJF), and longest job first (LJF) to improve the resource management performance. The migration and placement techniques in traditional approaches were not considered the utilization scenarios (overutilization/underutilization) and the job priority levels that lead to excessive energy consumption and less profit. Recent advances in cloud data centers and their energy-efficient methods were discussed in the following references [28–31].

### 3. Preserving Resource Handiness and Exigency-Based Migration (PRH-EM)

This section discusses the implementation of the novel techniques to maintain resource availability in order to deliver services to cloud users efficiently. Figure 1 shows the flow of the proposed preserving resource handiness and exigency-based migration (PRH-EM) algorithm for profit maximization with less energy consumption.

The (PRH-EM) algorithm contains successive processes such as federated environment creation, grade-based VM placement, job allocation, and exigency-based migration. Initially, the data centers are federated into four groups based on MIPS and cost. Then, the workload on the FDC and the capacity of the VM are measured. The capacity of the VM is compared with the FDC workload and place the VM into FDC if its capacity is less than the FDC. Then, the jobs are allotted to the respective VM in the updated host/VM list within the same and different FDC. When the jobs are allotted to the VM in the next FDC, the threshold value to indicate the utilization level is computed. Then, based on the threshold, the overutilized and underutilized VM are separated. Finally, the jobs are migrated from one VM to another according to their capacity.

**3.1. Federation.** The service provider is assumed to be autonomous with its own customers. During the demand conditions, the federation mechanism helps the providers identify the overloads. The federation model contains the cloud exchange service as the center. The providers send the necessary query to the exchange service to identify the available resources. The cloud exchange service generates the list of providers with the MIPS and cost value. The redirection of the requests helps to identify the suitable providers that use the MIPS and price list.

The component responsible for the decision-making regarding the allocation of additional resources is called the cloud coordinator. The simultaneous measurement of MIPS and cost for each data centers available in a cloud environment is responsible for federated environment creation. For each data center, the proposed algorithm computes the cost for million instruction (MI) execution. The overall cost of the MIPS for each data center is measured. The computed cost and MIPS are separated into three limits: low, medium, and high ( $L, M,$  and  $H$ ). Depending on the MIPS and cost, the federation comprises four types as follows:

- (i) Type I—More than high MIPS and less than low cost
- (ii) Type II—More than high MIPS and less than medium cost
- (iii) Type III—More than medium MIPS and less than medium cost
- (iv) Type IV—More than low MIPS and less than medium cost

Algorithm 1 for the federation scheme is listed as follows:

The federated data centers are assumed to have enough resources to handle the various jobs. But the changes in resource availability and the virtualization process makes the job assignment as uncertain. The proposed algorithm addresses this issue to maintain resource availability and deliver services during failure conditions.

**3.2. Grade-Based VM Placement.** The VM placement in federated data centers is the second stage of the proposed PRH-EM algorithm. After all the data centers are federated into geographic locations, the carbon footprint for each data center is computed as follows:

$$CF = \sum_{t=1}^T \sum_{j=1}^d PUE_i \times \sum_{j=1}^c \left( cf_j \times \sum_{k=1}^h P(vm_{i,j,k,t} \times ht) \right), \quad (1)$$

where CF represents the carbon footprint of the cloud and  $PUE_i$  describes the power usage effectiveness which is the ratio of the overall power consumption of the data centers ( $d$ ) to the power consumed by IT devices within holding time ( $ht$ ). The arrival of new  $vm$  request initiates its allocation to the host depending on the carbon level. The estimated carbon level is considered as the workload for federated cloud data centers. Algorithm 2 proposed in this paper is referred to as grade-based VM placement since it follows the descending order of workload.

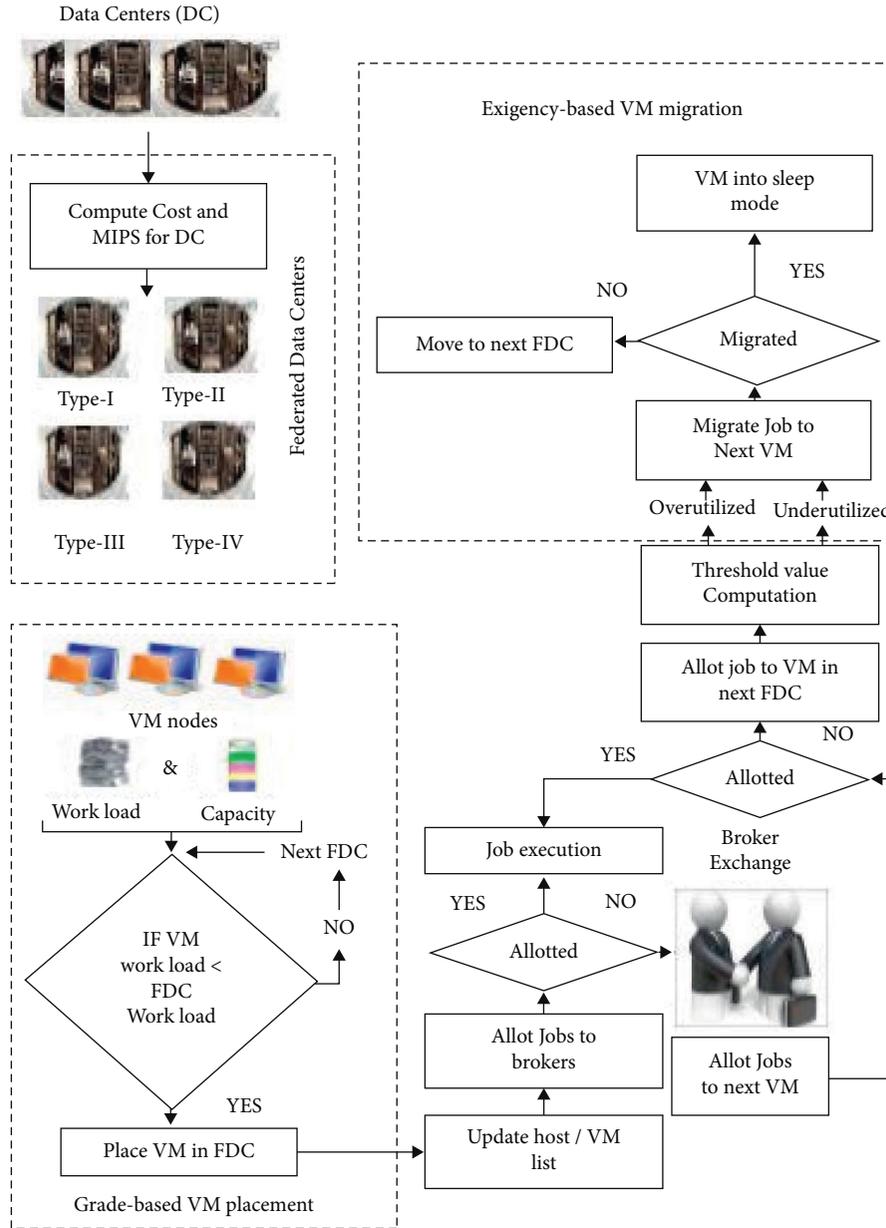


FIGURE 1: Overall flow of PRH-EM method.

- (1) Collect the data centers (DC)
- (2) Compute MIPS of each data center and cost/one MI
- (3) Compute the cost of MIPS of each data center
- (4) Split the MIPS and cost as follows:  
 $MIPS = \{MIPS_L, MIPS_M, MIPS_H\}$   
 $Cost = \{Cost_L, Cost_M, Cost_H\}$
- (5) If  $(MIPS_{executed} \geq MIPS_H \ \&\& \ Cost < Cost_L)$
- (6)  $FDC_1 \leftarrow$  Type 1 DCs
- (7) If  $(MIPS_{executed} \geq MIPS_H \ \&\& \ Cost \leq Cost_M)$
- (8)  $FDC_2 \leftarrow$  Type 2 DCs
- (9) If  $MIPS_{executed} \geq MIPS_M \ \&\& \ Cost \leq Cost_M$
- (10)  $FDC_3 \leftarrow$  Type 3 DCs
- (11) If  $(MIPS_{executed} \geq MIPS_L \ \&\& \ Cost \leq Cost_M)$
- (12)  $FDC_4 \leftarrow$  Type 4 DCs

ALGORITHM 1: Federation.

- (1) Compute the workload for all the FDCs  
(FDC 1, FDC 2, FDC 3, FDC 4)
- (2) Load the VM\_list and compute the carbon level using (1)
- (3) Arrange the Host\_list in descending order
- (4) If ( $cap_{vm} < cap_{FDC1}$ )
- (5) Place  $vm$  into host
- (6) Set  $status_{vm} \leftarrow Ready$
- (7) Compute the remaining capacity of VM
- (8) Update Host\_list and VM\_list
- (9) Else
- (10) Select the next data center (FDC)
- (11) Repeat from step 2
- (12) Update the VM\_list and Host\_list
- (13) Place all  $vm$  to the host

ALGORITHM 2: Grade-based VM placement.

The workload computation for each data center is the initial stage of the VM placement algorithm. Then, the carbon level for each VM is estimated for each machine in the VM list. Based on the workload and the carbon level estimation, the host list is created with the high-capacity host in the first place and the low-capacity host in the last place. Then, compare the capacity of the VM with the workload of the selected FDC to locate VM to the specific host. If the capacity of the selected VM is less than the FDC workload, then the corresponding VM is allotted to the selected host. Otherwise, the next FDC is selected and repeated from the list formation. During the comparison, Host\_list and VM\_list are periodically updated to know the specific host for each VM. The grade-based VM placement proposed in this paper is responsible for the effective service delivery on failure conditions. But the execution of the job is also a major concern in the energy-efficiency cloud environment.

**3.3. Job Allocation.** The major assumption for an energy-efficient cloud model is to allocate the job to one cloud only without replication. If there are multiple jobs arise under the specific instance, high-priority jobs are to be executed initially. Then, according to the capacity of the VM, the other jobs are allotted to the other VM. The particular data center includes the routers and switches responsible for the transportation of traffic between the servers and the outside world. The interconnection of processors is static in nature, and their utilization levels are periodical changes depending on the size of a job executed. The large size of I/O data transfer between them causes the overload condition. The brokers associated with VM monitor the capacity and job size to allocate the job to the maximum capacity VM. During the overload conditions (either VM capacity limit is reached or not), the broker will exchange their monitoring control with each other to provide constant resource availability.

The algorithm 3 for job allocation is described as follows:

Initially, the jobs to be executed are collected and their size is estimated. Then, the jobs are arranged in descending order with maximum size as the first one and minimum size as the last one. Then, the priority level is assigned to each job

based on the size value. The capacity of the VM is estimated parallel to the priority assignment. Then, the job size is compared with the VM capacity level in the initial round of execution. If the capacity of the VM is greater than the job size, then the corresponding job is allotted to the VM from the VM\_list. Then, the remaining capacity and jobs are estimated and placed to the nonallotted list for further processing. The job allocation algorithm 4 for a nonallotted list is described as follows:

The jobs from the nonallotted list are collected and the corresponding broker and host are identified. Then, switch the job to the next broker within the same FDC if there is any VM available with the necessary capacity until the last job on the list. If there is any job available in the list for the capacity limit reached state, then the next FDC is selected for job allocation. Finally, VM\_list and  $J$  are updated after allocation. During the demand conditions (capacity limit is reached) for job execution, the jobs executed on VM is migrated from one VM to another VM in order to provide the service delivery to the user effectively.

**3.4. Exigency-Based Migration.** The traditional VM migration schemes violated resource availability. Alternatively, the exigency-based migration algorithm proposed in this paper estimates the utilization level of VM by the job. Based on the utilization level, two conditions arise such as overutilization and underutilization. The migration of jobs in underutilized status converts the idle VM into a sleep mode that reduces unnecessary energy consumption. Similarly, the jobs in overutilized status are migrated to the maximum capacity VM provides the immediate response that reduces the time delay. Algorithms 5 and 6 for overutilized and underutilized conditions are described as follows:

The VMs allotted for the execution of the job are extracted from the VM\_list. The threshold values for the categorization of VM are computed to identify the status whether it is overutilized or underutilized. If the capacity of the VM is exceeded the maximum threshold value, then the corresponding VM is placed in overutilized list. Then, the job executed by the corresponding VM is migrated to the

- (1) Collect the job list  $J = \{J_1, J_2, \dots, J_N\}$
- (2) Compute the size of each job (size)
- (3) Arrange the jobs in descending order based on size
- (4) Compute  $\max_{\text{size}}$  and  $\min_{\text{size}}$
- (5) Assign  $(J_{\max_{\text{size}}}) \leftarrow \text{High}_{\text{priority}}$  and  $(J_{\min_{\text{size}}}) \leftarrow \text{Low}_{\text{priority}}$
- (6) Compute VM capacity ( $\text{cap}_{vm}$ )
- (7) Load high priority job to the VM with maximum capacity
- (8) For  $J_i, i = 1, 2, \dots, N$
- (9) Collect the VMs from VM\_list
- (10) If ( $\text{cap}_{vm} > J_{\max_{\text{size}}}$  && status)
- (11) Allocate the job to VM
- (12) Compute the remaining capacity of VM
- (13) Update the VM\_list and  $J$
- (14) Else
- (15) Allocate the job to nonallotted list

ALGORITHM 3: Job Allocation.

- (1) Collect the nonallotted jobs
- (2) Identify the broker correspond to the job
- (3) Identify the host from the Host\_list associated to broker in same FDC ( $B_{\text{FDC}_i}$ )
- (4)  $B_{\text{FDC}_i} \leftarrow \text{non - allotted Job}$
- (5) Allot the job to VM monitored by  $B_{\text{FDC}_i}$
- (6) If (Job exist in list)
- (7) Move the job to next FDC ( $\text{FDC}_{i+1}$ )
- (8) Update VM\_list and  $J$
- (9) Else
- (10) Execute the job

ALGORITHM 4: Job allocation for non-allotted list.

- (1) Load VM\_list
- (2) Compute the threshold values (min, max) for each VM
- (3) If ( $\text{cap}_{vm} > \max$ )
- (4) Assign VM as overutilized
- (5) List the remaining VMs with sufficient capacity ( $\text{VM}_R$ )
- (6) If ( $\text{cap}_{\text{vm}_R} > \max$ )
- (7) Migrate the over utilized job to the  $\text{VM}_R$
- (8) Else
- (9) Migrate job to next type  $\text{FDC}_{i+1}$
- (10) Compute remaining capacity of VM
- (11) Update VM\_list and migration list

ALGORITHM 5: Overutilized.

- (1) Identify underutilized VMs
- (2) List the remaining VMs with sufficient capacity in FDC ( $\text{VM}_R$ )
- (3) If ( $(\text{cap}_{\text{vm}_R} > \min) \&\& (\text{VM not in underutilized list})$ )
- (4) Migrate the underutilized job to the  $\text{VM}_R$
- (5) Else
- (6) Migrate job to next type  $\text{FDC}_{i+1}$
- (7) Set VM free and sleep mode
- (8) Update VM\_list and migration list

ALGORITHM 6: Underutilized.

remaining VM for the specific FDC. Alternatively, if the capacity of the selected VM is less than the minimum threshold value, it can be regarded as underutilized. The jobs executed by the underutilized VM are migrated to the next VM with sufficient capacity. The migration based on the utilization level maintains resource availability and reduces energy consumption effectively. VM failure detection and service failover are the two main principles in failure handling.

In general, failure detection is the process of identifying when something went wrong and informing the appropriate administrator so they can address it.

System monitoring and failure detection are two different things. There are frequent and quick failure detection checks (for example, every 5 seconds). They are typically more constrained in what they check as a result.

Failure tests in API Connect simply examine the availability of the web server and database. Contrarily, monitoring checks are often performed less regularly and are more likely to examine factors like CPU, RAM, and disc space utilization. These checks' outcomes can then be kept track of for historical trend analysis to find memory leaks.

#### 4. Performance Analysis

This section discusses the performance of proposed preserving resource handiness and exigency-based migration (PRH-EM) regarding the accuracy, response time variations for various jobs, and VMs. The comparative analysis of proposed PRH-EM with the best/worst fitness, greedy [26], and ABC approaches [27] assures the effectiveness in energy-efficient cloud environment. Besides, the profit variations for nonfederated totally in-house (NFTI), federation aware outsourcing oriented (FAOO), federated aware profit oriented (FAPO), [1] and proposed PRH-EM also presented.

**4.1. Response Time.** The overall time consumption from the rise of a new request to the response from the particular machine refers to the response time. The comparison between the proposed PRH-EM with the existing best/worst fitness and greedy [26] regarding the response time depicts that the hybrid processes such as VM placement, scheduling, and migration in the proposed approach reduce the response time efficiently.

Figure 2 shows the response time variations for a various number of VMs. The increase in VM reduces the response time effectively in traditional approaches. But grade-based placement and exigency-based migration provide a constant resource availability level that reduces the response time compared to other. The comparison shows that the PRH-EM offers a 28 and 50% reduction in response time compared to greedy approaches.

**4.2. Accuracy.** The measure of how the data centers are utilized for the overall time limit refers to the accuracy of the proposed system. The accuracy of the FDCs in the proposed

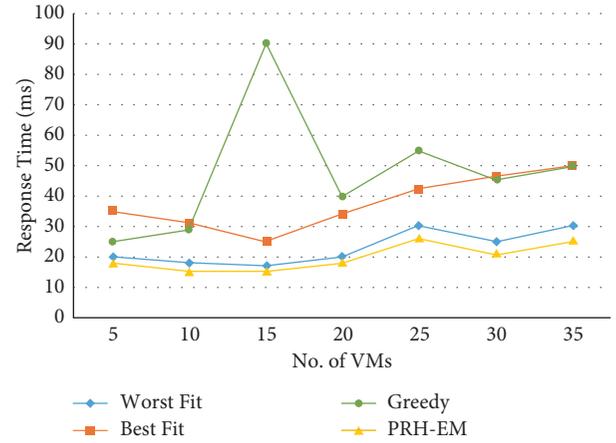


FIGURE 2: Response time analysis.

PRH-EM validated for various VM shows better performance compared to traditional approaches. Table 1 shows the accuracy variations for each FDC over the various VM count.

For the minimum value of VM (25), the FDC 4 shows a better performance than the other FDCs. Similarly, the FDC 3 provides better performance for higher VMs. The increase in VMs reduces the accuracy of FDC within the considerable level of the constant resource availability and efficient delivery to the user during the failure conditions are achieved by PRH-EM.

**4.3. Makespan.** The time consumption for the number of tasks executed within FDC is represented by makespan. The increase in tasks requires more VM that consumes more time for service delivery. But the migration based on exigency conditions reduces the makespan effectively.

Figures 3 and 4 show the makespan performance for a number of jobs and VMs, respectively. The increase in job size and VM size will increase the makespan value. But the provision of migration and grade-based VM placement reduces the makespan effectively. The comparative analysis shows that the PRH-EM scheme reduces the makespan by 40 and 8.3% of minimum and maximum jobs respectively compared to the ABC-SJF approach. Similarly, the PRH-EM reduces the makespan by 11.76 and 25% compared to the ABC-LJF approach respectively.

**4.4. Energy Consumption.** The energy model creation in this paper is based on the assumption that processor utilization and energy consumption are directly proportional to each other. The utilization level for the particular resource is expressed as

$$U_i = \sum_{j=1}^n u_{i,j}, \quad (2)$$

where  $n$ —number of VMs,  $u_{i,j}$ —utilization level for particular VM<sub>j</sub>

TABLE 1: Accuracy analysis.

| No. of VMs | Accuracy (%) |      |      |      |
|------------|--------------|------|------|------|
|            | FDC1         | FDC2 | FDC3 | FDC4 |
| 25         | 91           | 89   | 92   | 93   |
| 40         | 84           | 90   | 85   | 80   |
| 55         | 80           | 82   | 86   | 81   |

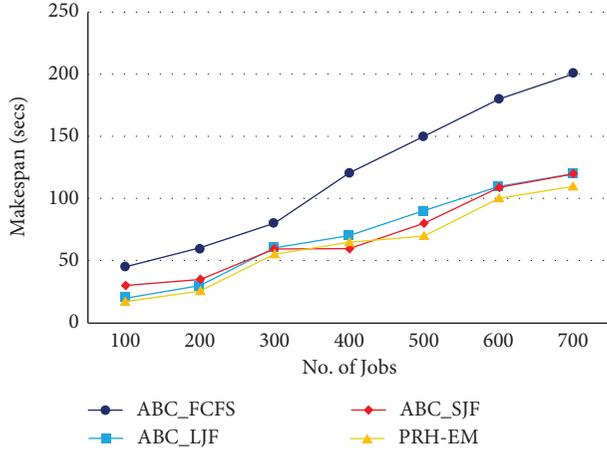


FIGURE 3: Makespan analysis for no. of jobs.

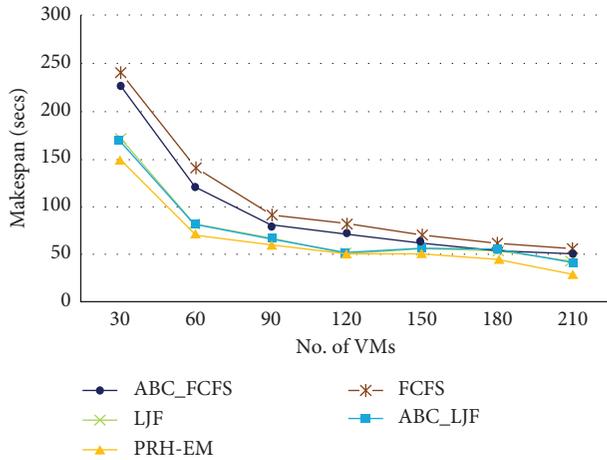


FIGURE 4: Makespan analysis for no. of VMs.

The energy consumption for the particular VM depends on the utilization level, power consumption ( $P_{max}$ ) at the peak overload (100% utilization), and power consumption ( $P_{min}$ ) at the active mode (1% utilization), and it is expressed as

$$E_i = (P_{max} - P_{min}) \times U_i + P_{min}. \quad (3)$$

The energy variations with the utilization level of CPU for proposed PRH-EM and existing MD\_MMT [28] show the effective reduction of energy consumption in PRH-EM.

Figure 5 shows the comparative analysis of PRH-EM with the MD\_MMT regarding energy consumption. The

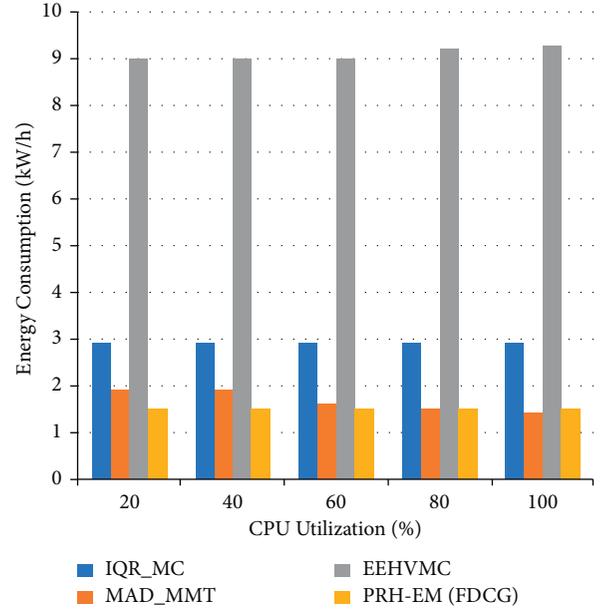


FIGURE 5: Energy consumption analysis.

proposed PRH-EM provides the reduction of energy consumed by the group of data centers by 15.79% for minimum (20%) utilization levels.

**4.5. Carbon Emission.** The carbon emission level depends on the PUE level and the number of VMs used according to equation (1). The analysis of carbon utilization level for proposed PRH-EM and existing carbon efficient placement and migration (CEPM) and round-robin (RR) migration algorithms [6]. Among these methods, the CEPM offers less carbon emission level compared to RR.

Figure 6 shows the comparative analysis of carbon emission level variations for each VM request. The increase in the number of requests increases the carbon footprint level linearly. The comparative analysis shows the PRH-EM reduces the utilization level by 20% for a maximum number of VM requests (200) compared to CEPM, respectively.

**4.6. Profit Analysis.** The deviation of revenue achievement and the cost required to achieve the revenue at the same time refers to profit. The mathematical formulation of profit is expressed as

$$\text{profit}(\Delta t) = \text{Revenue}(\Delta t) - \text{cost}(\Delta t). \quad (4)$$

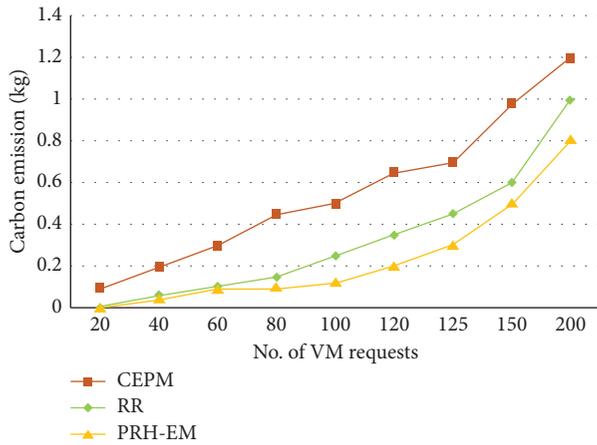


FIGURE 6: Carbon emission analysis.

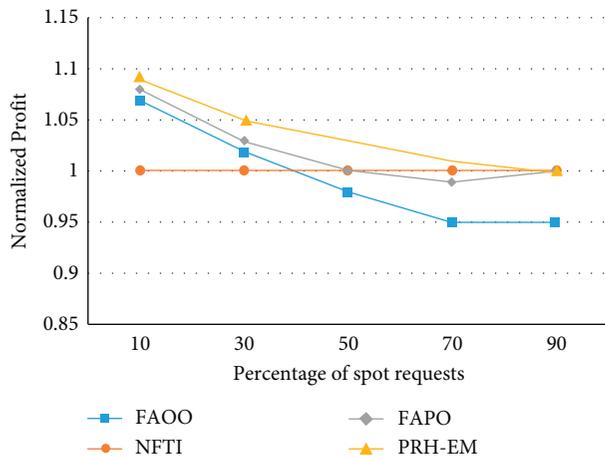


FIGURE 7: Profit analysis.

The experimental analysis of profit variation for the trivial, FAPO strategies [1] shows that the FAPO models provide more profit.

Figure 7 shows the comparative analysis of the profit variation for existing FAPO and PRH-EM models with the different percentages of spot request count. The comparative analysis shows that PRH-EM increases the profit by 0.9% compared to FAPO due to the exigency-based migration and grade-based VM placement for the minimum percentage of spot requests.

## 5. Conclusion

This paper discussed the various issues in the scheduling/migration schemes during the maintenance of resource availability in a confederated cloud environment. A novel VM migration algorithm is proposed to provide the trade-off between the reduction in energy consumption and profit maximization. Initially, the available data centers are categorized based on MIPS and cost value. The prior categorization to job allocation has a great impact on resource availability maintenance. The comparison between the cumulative workload capacity of data centers and the

individual VM capacity offers immediate migration during the demand conditions that prevented resource loss. The overutilized and underutilization-based migration reduced the number of resources that directly reduced energy consumption and carbon emissions. The proposed work maintained the constant resource availability and service delivery to the users during the VM failure conditions that increased the profit level. The comparative analysis of the proposed algorithm with the existing methods assured the effectiveness of the proposed work in a federated cloud environment.

## Data Availability

No data were used to support this study.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## References

- [1] A. N. Toosi, R. N. Calheiros, R. K. Thulasiram, and R. Buyya, "Resource provisioning policies to increase IaaS provider's profit in a federated cloud environment," in *Proceedings of the IEEE 13th International Conference on High-Performance Computing and Communications (HPCC)*, pp. 279–287, Banff, Canada, September 2011.
- [2] A. Celesti, M. Fazio, M. Villari, and A. Puliafito, "Virtual machine provisioning through satellite communications in federated cloud environments," *Future Generation Computer Systems*, vol. 28, no. 1, pp. 85–93, 2012.
- [3] F. Paraiso, N. Haderer, P. Merle, R. Rouvoy, and L. Seinturier, "A federated multi-cloud PaaS infrastructure," in *Proceedings of the IEEE 5th International Conference on Cloud Computing (CLOUD)*, pp. 392–399, Honolulu, HI, USA, June 2012.
- [4] A. Kertész, G. Kecskemeti, M. Oriol et al., "Enhancing federated cloud management with an integrated service monitoring approach," *Journal of Grid Computing*, vol. 11, no. 4, pp. 699–720, 2013.
- [5] I. Petri, T. Beach, M. Zou, J. D. Montes, O. Rana, and M. Parashar, "Exploring models and mechanisms for exchanging resources in a federated cloud," in *Proceedings of the IEEE International Conference on Cloud Engineering (IC2E)*, pp. 215–224, Boston, MA, USA, March 2014.
- [6] B. Wadhwa and A. Verma, "Carbon efficient VM placement and migration technique for green federated cloud data-centers," in *Proceedings of the International Conference on Advances in Computing Communications and Informatics (ICACCI)*, pp. 2297–2302, Delhi, India, September 2014.
- [7] B. Wadhwa and A. Verma, "Energy and carbon efficient VM placement and migration technique for green cloud data-centers," in *Proceedings of the Contemporary Computing (IC3), 2014 Seventh International Conference on*, pp. 189–193, Noida, India, August 2014.
- [8] M. H. Ferdaus, M. Murshed, R. N. Calheiros, and R. Buyya, "Network-aware virtual machine placement and migration in cloud data centers," *Emerging Research in Cloud Distributed Computing Systems*, vol. 42, 2015.
- [9] C. Dupont, F. Hermenier, T. Schulze, R. Basmadjian, A. Somov, and G. Giuliani, "Plug4Green: a flexible energy-aware VM manager to fit data centre particularities," *Ad Hoc Networks*, vol. 25, pp. 505–519, 2015.

- [10] A. R. A. Horri and GH. Dastghaibyfar, "Energy and performance-aware virtual machine consolidation in cloud computing A two-dimensional approach," *Turkish Journal of Engineering, Science and Technology*, vol. 1, pp. 20–35, 2015.
- [11] C. Dupont, G. Giuliani, F. Hermenier, T. Schulze, and A. Somov, "An energy aware framework for virtual machine placement in cloud federated data centres," in *Proceedings of the Third International Conference on Future Energy Systems: Where Energy, Computing and Communication Meet (e-Energy)*, pp. 1–10, Madrid, Spain, May 2012.
- [12] F. Ding, "Energy-aware and revenue-enhancing combinatorial scheduling in virtualized of cloud datacenter," *JCIT*, vol. 7, no. 1, pp. 62–70, 2012.
- [13] U. Wajid, B. Pernici, and G. Francis, "Energy efficient and CO2 aware cloud computing: requirements and case study," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 121–126, Manchester, UK, October 2013.
- [14] E. Volk, A. Tenschert, M. Gienger, A. Oleksiak, L. Sisó, and J. Salom, "Improving energy efficiency in data centers and federated cloud environments: comparison of CoolEmAll and Eco2Clouds approaches and metrics," in *Proceedings of the Third International Conference on Cloud and Green Computing (CGC)*, pp. 443–450, Karlsruhe, Germany, September 2013.
- [15] N. Kim, J. Cho, and E. Seo, "Energy-credit scheduler: an energy-aware virtual machine scheduler for cloud systems," *Future Generation Computer Systems*, vol. 32, pp. 128–137, 2014.
- [16] J. Li, M. Qiu, Z. Ming, G. Quan, X. Qin, and Z. Gu, "Online optimization for scheduling preemptable tasks on IaaS cloud systems," *Journal of Parallel and Distributed Computing*, vol. 72, no. 5, pp. 666–677, 2012.
- [17] S. Nesmachnow, B. Dorronsoro, J. E. Pecero, and P. Bouvry, "Energy-aware scheduling on multicore heterogeneous grid computing systems," *Journal of Grid Computing*, vol. 11, no. 4, pp. 653–680, 2013.
- [18] A. Benyi, J. D. Dombi, and A. Kertész, "Energy-aware VM scheduling in IaaS clouds using pliant logic," *Closer*, vol. 74, pp. 519–526, 2014.
- [19] V. Miranda, A. Tcherynykh, and D. Kliazovich, "Dynamic communication-aware scheduling with uncertainty of workflow applications in clouds," in *High Performance Computer Applications*, pp. 169–187, Springer, 2015.
- [20] G. Soni and M. Kalra, "Comparative study of live virtual machine migration techniques in cloud," *International Journal of Computer Application*, vol. 84, no. 14, pp. 19–25, 2013.
- [21] A. F. Leite and A. C. M. A. De Melo, "Energy-aware multi-agent server consolidation in federated clouds," in *Cloud Computing*, pp. 72–81, Springer, 2012.
- [22] A. Singh and S. Kinger, "Virtual machine migration policies in clouds," *International Journal of Science and Research (IJSR)*, India Online ISSN, pp. 2319–7064, 2013.
- [23] J. O. Gutierrez-García and A. Ramirez-Nafarrate, "Collaborative agents for distributed load management in cloud data centers using live migration of virtual machines," *IEEE Transactions on Services Computing*, vol. 8, no. 6, pp. 916–929, 2015.
- [24] M. Liaqat, S. Ninoriya, J. Shuja, R. W. Ahmad, and A. Gani, "Virtual machine migration enabled cloud resource management: a challenging task," 2016, <http://arxiv.org/abs/1601.03854>.
- [25] T. Kaur and I. Chana, "Energy-aware scheduling of deadline-constrained tasks in cloud computing," *Cluster Computing*, vol. 19, no. 2, pp. 679–698, 2016.
- [26] V. S Rathor, R. K Pateriya, R. K Gupta, M. Shelar, S. Sane, and V. Kharat, "An efficient virtual machine scheduling technique in cloud computing environment," *International Journal of Modern Education and Computer Science*, vol. 7, no. 3, pp. 39–46, 2015.
- [27] B. Kruekaew and W. Kimpan, "Virtual machine scheduling management on cloud computing using artificial bee colony," in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, pp. 12–14, 2014.
- [28] R. Karthikeyan and P. Chitra, "Novel power reduction framework forenhancing cloud computing by integrated GSNM scheduling method," *Cluster Computing*, vol. 21, no. 1, pp. 755–766, 2018.
- [29] R. Karthikeyan and V. Balamurugan, "Energy-aware and SLA-guaranteed optimal virtual machine swap and migrate system in cloud-Internet of Things," *Concurrency and Computation: Practice and Experience*, vol. 33, no. 10, Article ID e6171, 2021.
- [30] M. Shaukat, W. Alasmay, E. Alanazi, J. Shuja, S. A. Madani, and C.-H. Hsu, "Balanced energy-aware and fault-tolerant data center scheduling," *Sensors*, vol. 22, no. 4, p. 1482, 2022.
- [31] S. Bashir, S. Mustafa, and R. W. Ahmad, "Multi-factor nature inspired SLA-aware energy efficient resource management for cloud environments," *Cluster Computing*, vol. 28, p. 188p. 8, 2022.