

## Research Article

# Intelligent SLA-Aware VM Allocation and Energy Minimization Approach with EPO Algorithm for Cloud Computing Environment

Jitendra Kumar Samriya <sup>1</sup>, Subhash Chandra Patel <sup>2</sup>, Manju Khurana <sup>3</sup>,  
Pradeep Kumar Tiwari <sup>4</sup>, and Omar Cheikhrouhou <sup>5</sup>

<sup>1</sup>Department of Information Technology, Dr. B. R. Ambedkar National Institute of Technology, Jalandhar, India

<sup>2</sup>School of Computer Science and Engineering, VIT Bhopal University, Bhopal, India

<sup>3</sup>Computer Science and Engineering Department, Thapar Institute of Engineering and Technology (TIET), Patiala, India

<sup>4</sup>Manipal University Jaipur, Jaipur, India

<sup>5</sup>College of Computers and Information Technology, Taif University, Taif 21944, Saudi Arabia

Correspondence should be addressed to Omar Cheikhrouhou; o.cheikhrouhou@tu.edu.sa

Received 25 March 2021; Revised 13 April 2021; Accepted 7 May 2021; Published 30 May 2021

Academic Editor: Robertas Damaševičius

Copyright © 2021 Jitendra Kumar Samriya 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.

Cloud computing is the most prominent established framework; it offers access to resources and services based on large-scale distributed processing. An intensive management system is required for the cloud environment, and it should gather information about all phases of task processing and ensuring fair resource provisioning through the levels of Quality of Service (QoS). Virtual machine allocation is a major issue in the cloud environment that contributes to energy consumption and asset utilization in distributed cloud computing. Subsequently, in this paper, a multiobjective Emperor Penguin Optimization (EPO) algorithm is proposed to allocate the virtual machines with power utilization in a heterogeneous cloud environment. The proposed method is analyzed to make it suitable for virtual machines in the data center through Binary Gravity Search Algorithm (BGSA), Ant Colony Optimization (ACO), and Particle Swarm Optimization (PSO). To compare with other strategies, EPO is energy-efficient and there are significant differences. The results of the proposed system have been evaluated through the JAVA simulation platform. The exploratory outcome presents that the proposed EPO-based system is very effective in limiting energy consumption, SLA violation (SLAV), and enlarging QoS requirements for giving capable cloud service.

## 1. Introduction

Promptly, cloud computing (CC) has been established as one of the prevailing network technologies in the past. In today's world, data is turned into a vital resource of information due to the arrival of big data. For mining data value, CC is very efficient, valuable, and inexpensive. It creates a significant effect on social and economic development [1–4].

In CC environment, the main keystone is virtualization. In this, each physical machine (PM) or host's hardware resources are splitted into various execution environments which are represented as a virtual machine (VM). In PM, the computing request from the user is organized by the VM

with fulfilling the computational resource constraints such as the size of the memory, time for computing, and CPU resources [5, 6].

In general, the cloud collects the data from VMs, resources, hosts, data centers, and so forth [7]. Cloud services are provided by data centers in cloud environments. However, a large amount of energy is utilized by these data centers in their tasks. The yearly energy consumption of data centers in the United States is 91 billion kWh, which is assumed to rise by 140 billion kWh in 2020. Like that, in the data center, a quick rise in power consumption leads to a key financial problem and also environmental trouble. And also, the energy cost of amazon's data center is approximately 42% of the entire operational price [8, 9].

In future, this energy requirement is rising by more than 80% as expected and the data centers are an important reason for this enhancement. Compared to the old office systems, 100 times more energy is used in data centers. This statement is based on the report of the Environmental Protection Agency [10, 11].

Based on the computational capacity of PM and computational necessity of allocated VMs, the power consumption of particular PM is differed. For physical servers which are idle, the power utilization is exceeding two-thirds of the used server [12–14].

To overcome this energy inefficiency, efficient usage of physical machine resources using virtualization is a promising solution. Specifically, virtual machine placement (VMP) is an effective solution. In a data center, VMP refers to how VMs are allocated on PMs. Due to this, the usage of physical servers is enhanced and the number of underloaded servers is minimized. For power efficiency purpose, the load of the underloaded server is allocated to another available server and turn the underloaded server to sleep mode [15–17].

However, due to the following factors, providing an effective solution to this problem is not simple. First, VMs and PMs have heterogeneous features. Next, the improper VMP causes server exhaustion. Due to this, the server is unable to host any more VMs. Third, the complexity is increased by searching the optimum allocation among hundreds of candidates (sometimes, a lot of requests are received by data centers) [18, 19].

Therefore, in this paper, we suggest an optimal multi-objective VM allocation scheme. This scheme will allocate the VMs in an efficient way with energy efficiency than the other mechanisms because it is easy to implement and it utilizes less memory space for saving the best optimal solutions.

The important contributions of the proposed VM placement method are as follows:

- (i) We proposed a multiobjective Emperor Penguin Optimization (EPO) algorithm that successfully allocates the VMs to the least number of active PMs in cloud data centers.
- (ii) EPO has more efficiency than BSGA, ACO, and PSO algorithms in terms of energy consumption and resource utilization and also the execution time and the quantity of active server is also reduced.
- (iii) We analyzed and tested the proposed work by creating the computer-based simulation using the CloudSim toolkit. Compared with other methods, this method has a competitive advantage.

The remaining portion of the paper is structured as follows: we analyzed recent studies about VM placement in Section 2; the detailed problem formulation approach is presented in Section 3; the proposed EPO methodology is presented in Section 4; Section 5 provides the obtained results and its explanation; and finally, Section 6 describes the conclusion.

## 2. Related Works

Development in the cloud area has opened the door to a comprehensive study. VM placement strategies largely led to

the efficiency of cloud computing. In cloud data centers, various scholars from industry and institutions have tried to give the best VM placement recently [20, 21].

Mohammadhosseini et al. [22] suggested a balance-based cultural algorithm for virtual machine placement (BCAVMP) with less energy consumption and efficient resource management. The authors evaluated the solutions using a new fitness function which is based on the total balance vector lengths for every VM allocation. The balanced resource usage is also studied by authors and the result is compared with PABFD that contains IQR-MMT, LRR-MMT, THR-MMT, MAD-MMT, and LR-MMT algorithms.

A combination of multiobjective Sine–Cosine Algorithm and Salp Swarm Algorithm is used by Gharehpasha et al. [23] for efficient VM allocation and their objectives are to decrease power consumption, waste of resources, and SLA violations. The authors compared results with Ant Colony System, First Fit, and Modified Best Fit Decreasing algorithm which are existing ones.

Abdessamia et al. [24] proposed an optimization method for VM allocation in the data center which is based on a gravitational search algorithm named Binary Gravitational Search Algorithm (BGSA). They implemented their solutions in MATLAB for energy efficiency and evaluated the result with existing strategies.

An energy-efficient algorithm named GATA for VM allocation is designed by Zhao et al. [25]. They generated the proposed algorithm with a combination of genetic algorithm and tabu search algorithm. Load balance of the data center is maximized to decrease energy consumption and results compared with the simulated annealing algorithm (SA), traditional genetic algorithm, and ant colony algorithm.

Wei et al. [26] have presented the mixed-integer linear program (MILP) considering makespan, energy consumption, and idle energy of active PMs for VM allocation problem. The authors used multiple capacity constraints to address the multiobjective problem and also used Branch and Bound algorithms for the MILP program to attain a solution.

Abohamama and Hamoudaa [8] have performed research for reducing the quantity of physical server which are active and shutting down the idle server to improve energy consumption rate by hybrid VMP algorithm in data centers. Besides, they reduced the resource wastage rate by balanced usage of CPU, RAM, and Bandwidth of servers which are active. For a hybrid VM allocation algorithm, the authors combined an enhanced permutation-based GA (IGA-POP) and resource-aware best fit strategy. The results were compared with deterministic algorithms and stochastic optimization algorithms.

Gharehpasha et al. [27] have analyzed the optimal placement of VMs with resource organization and avoidance of resource wastage in a cloud environment. For this, they introduced a new method based on the grouping of a multiverse optimizer with chaos function and a hybrid discrete multiobjective whale optimization algorithm, which can be optimally placed in a cloud data center. The results were compared with other algorithms such as MBFD, VMPACS, and First Fit.

Ruan et al. [28] have presented a “PPRGear” strategy for VM placement and migration with energy efficiency in the cloud based on host utilization. They calculated the performance-to-power ratios (PPR) and estimated features of computing nodes at different usage levels, called gears. They compared the solutions with IqrMc, MadMmt, and ThrRs algorithms.

Feng et al. [29] have developed a two-step SAG algorithm to minimize the energy consumption in cloud data centers. In the first step, Simulated Annealing (SA) algorithm was utilized to minimize the server and cooling energy consumption. The Greedy algorithm was used in the second step to reduce the energy consumption of the network. For performance evaluation, XINT-GA, power-aware and performance-guaranteed VMP (PPVMP), CAC, and task scheduling algorithm based on thermal-aware (TSTD) techniques are used.

For energy-efficient and SLA-aware VM placement problems, Elsedimy and Algarni [30] suggested a technique based on Particle Swarm Optimization (PSO) technique. To optimize the VM placement, the crowding entropy method was used. The existing algorithms named VMPACS, FDD, and MGGA were used for performance estimation. The authors analyzed their proposed approach via two experiments. In the first experiment, they evaluate SLA violation, power consumption, and wastage resource. In the second experiment, Spacing and Generational Distance (GD) was used as the performance metric.

### 3. Problem Formulation

In this work, our challenges are considered as NP-hard problem and our objective is VM placement with an energy-aware approach in servers. In this effort, we utilized a bioinspired metaheuristic algorithm termed Emperor Penguin Optimization (EPO) for VM placement. We used some notations to explain the VM placement in a cloud environment that is presented in Table 1.

Consider  $n$  number of PMs and  $m$  number of VMs in a cloud data center and  $P$  denotes a group of PMs, where  $P_i$  denotes the  $i^{\text{th}}$  PM,  $i$  belongs to  $[1 \dots n]$ , and  $P_i \in P$ .

Similarly, a set of VMs are represented by  $V$ , where  $V_j$  represents the  $j^{\text{th}}$  VM,  $j$  belongs to  $[1 \dots m]$ , and  $V_j \in V$ .

Power consumption is an important aspect of this study that is related to memory resources and CPU utilization.  $V_{\text{cpu}_j}$  and  $V_{\text{ram}_j}$  are the required computational power and memory of  $V_j$ . Similarly,  $P_{\text{cpu}_i}$  and  $P_{\text{ram}_i}$  are the capacities of  $P_i$  for CPU and memory.

**3.1. Multiobjective Optimization.** As the multiobjective optimization problem, VM placement is considered in this work. To optimize a number of possibly conflicting objects, a pool of VMs is placed in a minimum number of PMs [31]. A multiobjective optimization problem generally can be stated as follows:

$$\begin{aligned} MO &= [f_1(u), f_2(u), \dots, f_h(u)], \\ u &= [u_1, u_2, \dots, u_k] \in U, \end{aligned} \quad (1)$$

TABLE 1: Notations and its descriptions.

Notation	Definition
PM	Physical machine
VM	Virtual machine
$P$	Set of PMs
$V$	Set of VMs
$P_i$	Physical machine $i$
$V_j$	Virtual machine $j$
$V_{\text{cpu}_j}$	Required computational power of VM $j$
$V_{\text{ram}_j}$	Required memory of VM $j$
$P_{\text{cpu}_i}$	CPU capacity of physical machine $i$
$P_{\text{ram}_i}$	Memory capacity of physical machine $i$
PC	Power consumption
$P_{\text{idle}}$	Power of CPU in idle state
$P_{\text{full}}$	Power of CPU in utilized state
$T_{\text{rm}}$	Total requested MIPS
$T_{\text{am}}$	Total allocated MIPS
$O_v$	Set of optimal value

where the number of objectives is denoted by  $h \geq 2$ , a set of feasible solutions are denoted by  $U$ , a feasible solution is denoted by  $u = [u_1, u_2, \dots, u_k]$ , and the number of active PMs are represented by  $k$ . Decreasing the power consumption and SLA violation are two important key objectives which are analyzed in this paper.

**3.2. Objective Function.** We assume that each PM has sufficient ability to host any single VM. Thus, only one PM can host a single VM. Here,  $s$  denotes the placement solution and that is represented by a zero-one adjacency matrix, where  $x_{ij} = 1$  if  $VM_j$  is assigned to  $P_i$ ; otherwise,  $x_{ij} = 0$ . Similar to constraints, best VM placement can be expressed with the least active PM in a cloud data center, shown as follows:

$$s = \min \sum_{i=1}^n y_i, \quad (2)$$

subject to

$$x_{ij} = \begin{cases} 1, & \text{if } V_j \text{ is assigned to } P_i, \quad \forall i \in P \text{ and } \forall j \in V, \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

$$y_i = \begin{cases} 1, & \text{if } \sum_{i=1}^n x_{ij} \geq 1 \quad \forall i \in P, \\ 0, & \text{otherwise,} \end{cases} \quad (4)$$

$$\sum_{i=1}^n x_{ij} = 1 \quad \forall j \in V, \quad (5)$$

$$\sum_{i=1}^m V_{\text{cpu}_j, x_{ij}} \leq P_{\text{cpu}_i, y_i}, \quad \forall i \in P, \quad (6)$$

$$\sum_{i=1}^m V_{\text{ram}_j, x_{ij}} \leq P_{\text{ram}_i, y_i}, \quad \forall i \in P. \quad (7)$$

According to equation (3),  $y_i = 1$  when  $P_i$  is active and  $y_i = 0$  when  $P_i$  is idle. Equation (4) ensures that  $V_j$  is allocated to only one PM in the group. Equations (6) and (7) relate to CPU and memory utilization respectively that must not exceed the value of capacity constraints ( $P_i$ ). The power consumption of active PM without load is approximately 50% to 70% of fully utilized PM [32]. Therefore, turning off inactive PMs are essential to decrease the total power consumption of a cloud data center.

**3.3. Power Consumption.** Therefore, the power consumption of the CPU utilization is formulated as follows:

$$PC(P_{cpu_j}) = \begin{cases} P_{idle} + (P_{full} - P_{idle}) \times P_{cpu_j}, & \text{if } P_{cpu_j} > 0, \\ 0, & \text{otherwise,} \end{cases} \quad (8)$$

where  $P_{idle}$  denotes idle state power consumption,  $P_{full}$  denotes fully utilized CPU power consumption, and  $P_{cpu_i} \in [0, 1]$ . As per our assumption, the idle state CPU power consumption is 60% of the energy that is consumed in a fully utilized state. Efficient resource usage and decrease in the number of active PMs are effective ways to lower power consumption. Thus, maximum utilization of CPU would contribute to the total power consumption of active PMs, regardless of other resource types. Therefore, we consider the CPU resource which is vital to decrease the power consumption in our model. The CPU utilization of  $i$ th PM is calculated by

$$P_{cpu_i} = \sum_{j=1}^m x_{ij} \cdot C_{ij}^c \quad (9)$$

Here,  $C_{ij}^c$  is the CPU demand of every VM.

The following equation (10) calculates the fitness function for power consumption:

$$\text{fitness} = \frac{1}{\left(\sum_1^N PC(P_{cpu_i})\right)} \quad (10)$$

**3.4. SLA Violations.** The SLA violations (SLAVs) arise either overuse of the PM or the requested processor and memory for VM is not allocated by the PM. For better assignment and satisfying QoS requirements, lowering the total SLA violation is important. The overall SLA violations are represented in the following equations:

$$\text{SLAV} = \frac{T_{rm} - T_{am}}{T_{rm}} \quad (11)$$

Here,  $T_{rm}$  is the total requested MIPS and  $T_{am}$  is the total assigned MIPS to VMs based on the resource demand.

#### 4. Proposed Emperor Penguin Optimizer (EPO) in VM Allocation

This section presents a multiobjective VM placement model with the Emperor penguin optimizer algorithm and the proposed system architecture is shown in Figure 1.

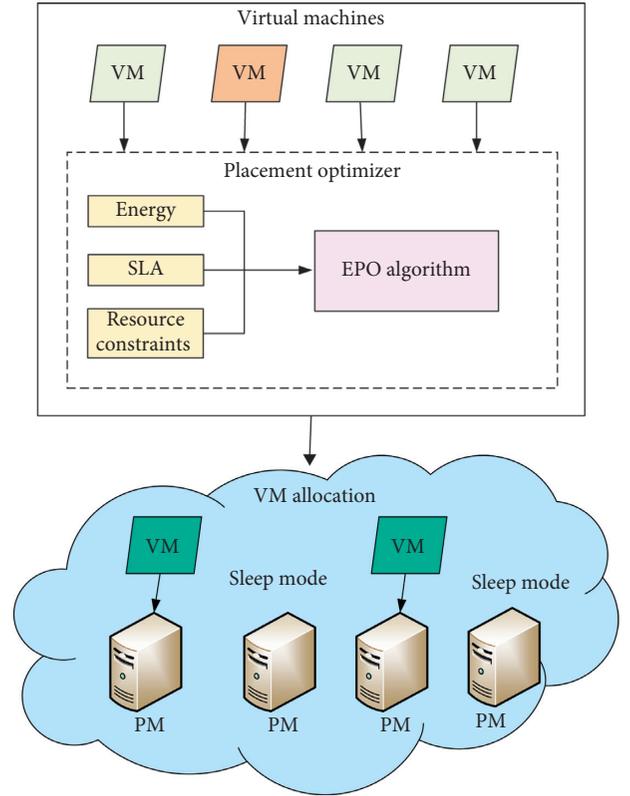


FIGURE 1: System architecture of the proposed model.

Our proposed algorithm, Emperor Penguin Optimizer (EPO) is created based on the hunting behavior of penguins producing the best global result over concerted coordination in their dives during collective hunting and nutritional activities. The vocalization of penguins is close to the distinctive detection and identification of penguins, and this is difficult to recognizing penguins from a large number of colonies providing great similarity. Penguins are clustered in groups, the quantity of penguins may differ depending on the availability of food at a given spot, and they prefer to search in groups and go around randomly. The significance of the EPO algorithm is to obtain a global optimum solution with superior convergence. EPO provides a global solution even when the amount of penguin is high because EPO is strong, and it provides a clear compromise between the global and local minima.

By reducing the number of active PMs, the energy consumption of a cloud data center can be decreased efficiently. Hence, we changed the EPO algorithm for the VMP problem and gain a solution that maps the VMs to the least number of active PMs. We assume that list VM requests are collected and search for the PM. Initially, the EPO randomly assigns VM and calculates power consumption and SLAV of PM. If the values are least, then assign the VM to a particular PM. Before assignment, the CPU and memory capacity of PM is also considered. This is the iteration process to get the perfect PM for VM. Once the VM is assigned, the location of the PM is updated and the other inactive PMs are going to sleep mode state.

Assume that allocating the number of physical machines  $n = 5$  and number of virtual machines  $m = 4$  like the penguins determine the huddle boundary, collect the VM request, and randomly initialize the VM1 or penguin to PM1 and calculate power consumption and SLAV using equations (7) and (9) which is similar to the temperature calculation of penguins. The penguins are fixed huddle boundary based on temperature.

After that, collect optimal power consumption and SLA violation values of PMs using the following equation:

$$O_v = \sum_{i=1}^n [PC(P_{cpu_i}); SLAV]. \quad (12)$$

Here,  $PC(P_{cpu_i})$  is the power consumption value of PM and SLAV is the service level agreement violation value. Assume that the set  $O_v$  contains a set of optimum values  $O_v = [O_{w1}, O_{z1}; O_{w2}, O_{z2} \dots O_{wn}, O_{zn}]$  and the results are arranged in ascending order of the number of active PMs.

Then, find the best minimum optimal solution  $E$  from the collection of values  $O_v$ :

$$E = \min(O_v). \quad (13)$$

Based on equation (11), find the best optimal solution with the least power consumption and SLA violation.

And also to assign the VM to a particular PM, resource availability is very important. In our proposed method, the memory and CPU capacity of PM is calculated from the following equation:

$$P_i = \begin{cases} \sum_{j=1}^m x_{ij} \cdot V_{cpu_j} + V_{cpu_i} \leq P_{cpu_i} \text{ and,} \\ \sum_{j=1}^m x_{ij} \cdot V_{ram_j} + V_{ram_i} \leq P_{ram_i}. \end{cases} \quad (14)$$

In  $P_i$ , the value of  $i$  is  $1 \leq i \leq n$  and  $V_{cpu_j}$  and  $V_{ram_j}$  are the sum of CPU and memory capacities of previously submitted VMs on  $P_i$ , correspondingly.  $V_{cpu_i}$  and  $V_{ram_i}$  are the CPU and memory capabilities of unallocated VM.  $P_{cpu_i}$  and  $P_{ram_i}$  are the capabilities of PM. This equation is used to select the suitable PM for VM.

According to the best optimal solution and resource availability, the VM is placed to a suitable PM which is similar to the penguin that moved based on the temperature.

Sometimes, the collision is occurring when more than one PM can host a single VM. To avoid this problem, equation (4) is used. If the VM is allocated to one PM, then the value of that VM is changed to 1 and other entries are changed to zero. This method is used to prevent unnecessary confusion.

**4.1. Algorithm.** The step-by-step methods of VM allocation based on the EPO algorithm would be described as follows:

Step 1: initialization

- (i) We term the list of VMs (penguins) as  $m$  in the data center

- (ii) We term the list of PMs as  $n$  in the data center
- (iii) The total number of repetitions
- (iv) VM requests

Step 2: choose the PM randomly that must be in the search space.

Step 3: calculate the objective values (power consumption and SLAV) of PM.

Step 4: the fitness function is used to determine how well a result can provide an effective answer. In every iteration, we calculate the fitness of power consumption value and also SLAV for every PM, then analyze the collected values, and sort them in ascending order.

Step 5: find the minimum optimum value from a set of collected values which is considered as the best optimum value.

Based on the best optimum value and a set of resource availability, the penguin is placed in PM.

Step 6: to save power consumption, the remaining unallocated PMs are moved to a sleep mode state.

In Figure 2, all the algorithm steps are described in the form of a flow chart.

**4.2. Computational Complexity.** The computational complexity of the EPO algorithm is discussed in this section. Space and time complexities of the proposed algorithm are given below.

**4.2.1. Space Complexity.** During the initialization process of the proposed algorithm, the highest amount of space utilized at any one time is called space complexity. Therefore, the space complexity is  $O(m \times d)$ . Here,  $m$  denotes the population size and  $d$  denotes the dimension of the given problem.

**4.2.2. Time Complexity.** In the EPO algorithm, the population initialization process takes  $O(n \times d)$  time. Each search agent's fitness takes  $O(\text{high}_{\text{iter}} \times m \times d)$  time. Here,  $\text{high}_{\text{iter}}$  denotes the maximum number of equations to simulate the proposed algorithm. It takes  $O(n)$  time, where  $n$  denotes EPO's huddling behavior for better exploration and exploitation. Therefore, the total time complexity is  $O(\text{high}_{\text{iter}} \times m \times d \times n)$ .

## 5. Simulation Results

The proposed EPO algorithm is analyzed in this section and it is compared with BSGA, ACO, and PSO-based algorithms in Windows 10 operating system with 4 GB RAM and Intel i5 2.60 GHz processor. For simulation, we used CloudSim tool with JAVA. In addition to the compared algorithms, all the tasks required in the algorithms have been applied in Java.

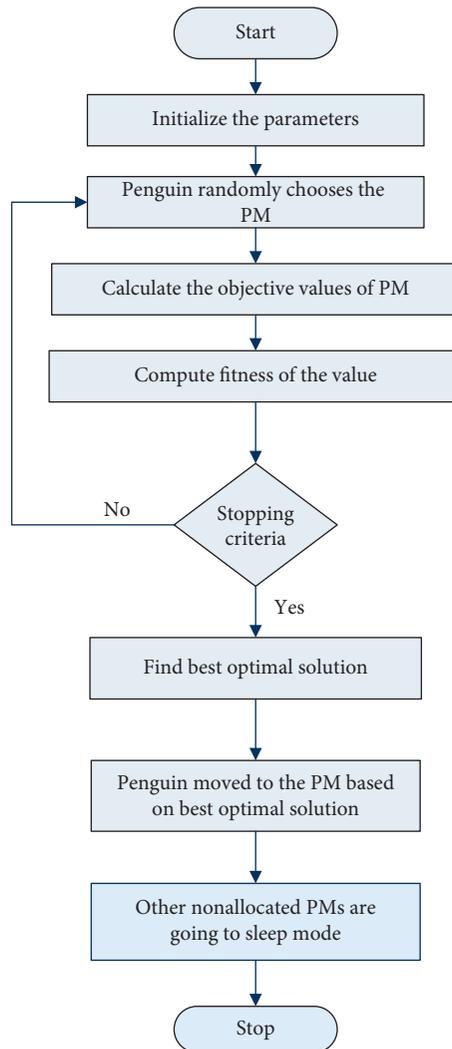


FIGURE 2: Flowchart of the proposed system.

## 5.1. Algorithms for Comparison

**5.1.1. Particle Swarm Optimization (PSO).** It is based on fitness and groups [33]. Every member of the swarm is referred to as a particle, and each one represents a possible solution to the problem. Position and velocity are the two parameters of each particle. The position of each particle is linked with the value of fitness. The quality of the result is evaluated based on the fitness value. The PSO starts by assigning a group of random particles and then iterates until it finds the best solution. It imitates the interactive behavior of birds foraging.

**5.1.2. Ant Colony Optimization (ACO).** ACO [34] is a metaheuristic that was inspired by actual ant colonies and is focused on their collective foraging behavior. Ants live in colonies and are social insects. Ants often move between their nest and food sources during the food search process. Generally, ants explore the environment in a random manner around their nest. They leave unique substances termed pheromones along their paths. When deciding where

to go, they are more likely to select routes with high pheromone concentrations. Once the ant finds a source of food, it evaluates the quantity and quality of the food and brings some of it back to the nest. During the return journey, the amount of pheromone left by the ants on the ground may depend on the quality and quantity of the food. Other ants will follow these pheromone trails to the food source. Moreover, it finds the shortest routes by these pheromone trails among their food sources and nest.

**5.1.3. Binary Gravitational Search Algorithm (BGSA).** The gravitational search algorithm is based on the mass interactions and law of gravity [24]. In the Binary Gravitational Search Algorithm (BGSA), the search agents are a set of masses interacting according to Newtonian gravity and motion laws. The performance of search agents is measure by their masses. These search agents get attracted to each other and move towards the search agents of heavy mass by the gravitational force. As a result, an object with a large mass has a superior solution to other objects. The movement

of heavier masses is slower than that of lighter masses guarantees exploitation.

**5.2. Parameters.** The effectiveness of the proposed VM placement solution is assessed using four efficiency metrics which are energy consumption, overall SLA violations, the number of active servers, and execution time.

**5.2.1. Energy Consumption.** In a cloud data center, the total amount of energy consumed by all running PMs is represented by energy consumption [35–38]. To reduce expenditure, the least amount of energy consumption is used. For better assignments, the energy consumption should be lower. Most of the papers discussed in the literature review take this parameter. In [22], they try to achieve less energy consumption. However, the low convergence speed affects the performance of the algorithm.

**5.2.2. Overall SLA Violations.** Overusage of PM causes SLA violations. For better assignments, the overall SLA violation should be lower. In [8], they mainly concentrate to reduce resource wastage. During this process, when there is high usage of CPU, SLA violations may occur.

**5.2.3. Number of Active Servers.** To reduce the power usage of a data center, the number of active servers should be decreased. In [27], resource utilization is unbalanced. Hence, the number of active servers is increased.

**5.2.4. Execution Time.** To obtain the result in a short time, the execution time of the algorithm should be reduced. In [27], the algorithm takes a long time for fitness evaluation. Hence, the execution time of this approach is increased.

**5.3. Experimental Evaluation.** The same amounts of PMs and VMs are used in all algorithms for efficient comparison. We simulated a group of heterogeneous servers with various energy consumption and they use static resources such as CPU and memory. When we use different server tools in the data center, the energy consumption ratio will be different. In this research, we used two types of PMs as shown in Table 2. Three types of VMs are used for the simulation as shown in Table 3.

For the simulations, the power consumption and SLA violation have been calculated through the proposed EPO technique. To analyze the energy consumption of all PMs, we tried to allocate under 200 VM. As our algorithm has low complexity, a small number of VMs allocated in the algorithm give similar results. But the higher number of VMs gives an alternate result. Although our proposed algorithm gives better results with minor variations, as per the low complexity of our algorithm, we get the results in a very short time and simulation results are very similar. And also, it gives better results for the VM allocation

problem compared to other algorithms with a high number of VMs.

The energy consumption is depending on the number of active servers. After VM allocation using the EPO algorithm, the amount of active server is reduced and that is shown in Figure 3. The proposed approach attempts to make the most efficient use of a small number of hosts while allocating a large number of jobs to the active hosts. Since the number of active hosts in a data center is directly proportional to energy consumption, an increase in the number of host shutdowns will result in a decrease in energy consumption.

The values of energy consumption and SLAV is calculated and evaluated to the other existing algorithms. Figure 4 and Table 4 explain the energy comparison of EPO with BGSA, ACO, and PSO. From the graph and table, it is observed that if the number of VM is increased then the energy consumption of the data center is also increased. The attained values show that the EPO approach helps to reduce the power consumption even if the number of VMs increases.

Figure 5 and Table 5 explain the comparison of SLAV with the abovementioned algorithms. In terms of SLAV, the compared techniques have parallel performance [39, 40]. However, compared to other approaches, PSO has the worst result. Compared with BGSA and ACO, BGSA's performance is better. But in most of the iterations, the performance of EPO is superior. Hence, it is observed that energy consumption and SLA violation of the proposed approach is lower than other algorithms.

The overall improvement of our proposed algorithm over the other 3 algorithms is shown in Figure 6 and it shows the consistency and fitness of our proposed technique.

Based on the comparison of EPO with the BGSA technique, 65.53% of energy is saved and 56.81% of SLA violation is reduced on average and that is shown in Figure 7.

In Figure 8, the comparison of EPO over ACO is shown; an average of 50.29% of energy is saved; and 45.27% of SLA violation is reduced. Figure 9 shows the comparison of EPO with PSO, and the average values for saving energy and SLA reduction are 52.07% and 39.29%, respectively.

Figure 10 shows the execution time of all four approaches. In comparison, the execution time of the BGSA algorithm is higher because it has a high computational complexity than the other three algorithms. The execution time of the PSO algorithm is shorter than the ACO algorithm; the effect of energy consumption by using the PSO algorithm is larger than the ACO algorithm. The ACO algorithm is a simple conventional ant colony algorithm in which the heuristic factors do not alter with dynamic virtual machine allocation, ants prefer to travel directly to resource-rich hosts, and its effect deteriorates as the number of tasks increases. Moreover, we analyzed that the execution time of the EPO algorithm is better than BGSA, ACO, and PSO techniques and the convergence rate of EPO is very high.

The overall complexity of a system is reduced by the lower execution time of the proposed technique.

TABLE 2: PM parameters.

Physical machine type	kWh	Memory	MIPS
PM1	150	16	125000
PM 2	130	8	11000

TABLE 3: VM parameters.

Virtual machine type	Memory (GB)	MIPS
VM 1	3	11500
VM 2	2	1500
VM 3	600	3 500

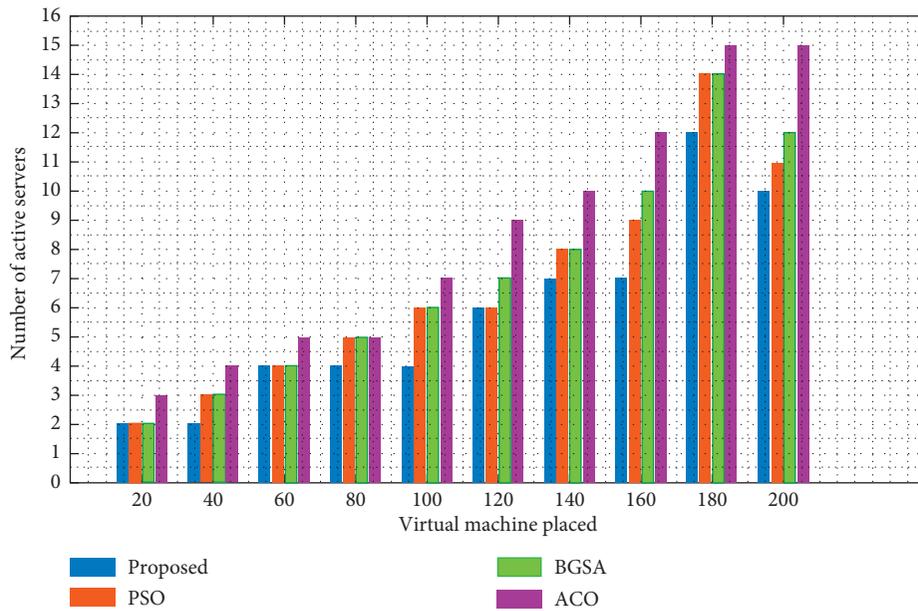


FIGURE 3: Active server number in every VM placement.

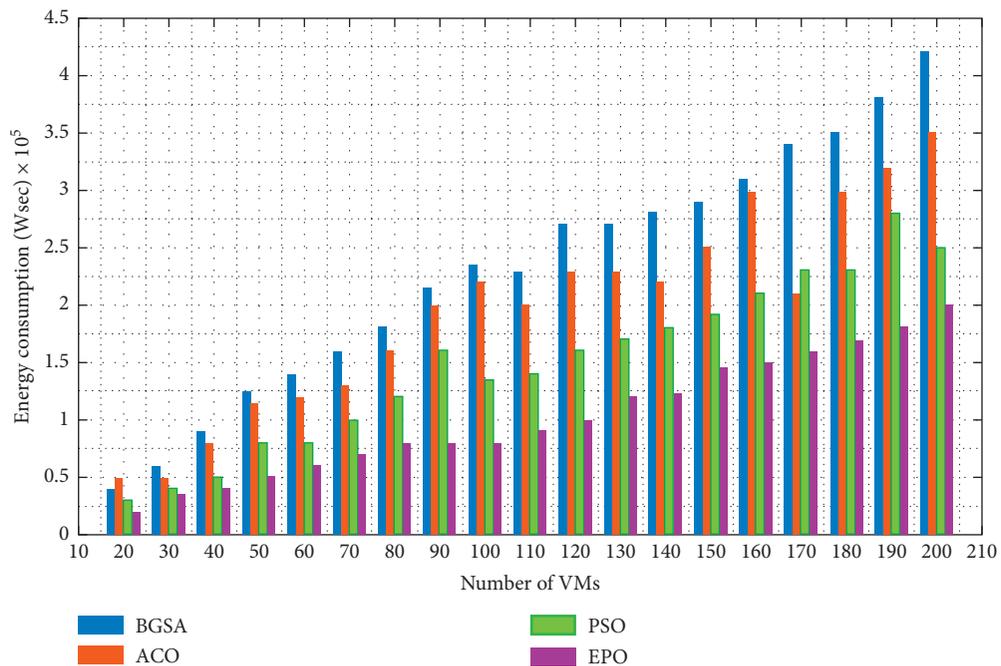


FIGURE 4: Comparison of energy consumption rate with other algorithms.

TABLE 4: Comparison of energy consumption.

Number of VMs	BGSA	ACO	PSO	Proposed
40	$0.9 \times 10^5$	$0.7 \times 10^5$	$0.5 \times 10^5$	$0.4 \times 10^5$
80	$1.8 \times 10^5$	$1.6 \times 10^5$	$1.2 \times 10^5$	$0.8 \times 10^5$
120	$2.7 \times 10^5$	$2.2 \times 10^5$	$1.6 \times 10^5$	$1.0 \times 10^5$
160	$3.2 \times 10^5$	$3.0 \times 10^5$	$2.1 \times 10^5$	$1.5 \times 10^5$
200	$4.3 \times 10^5$	$3.5 \times 10^5$	$2.5 \times 10^5$	$2 \times 10^5$

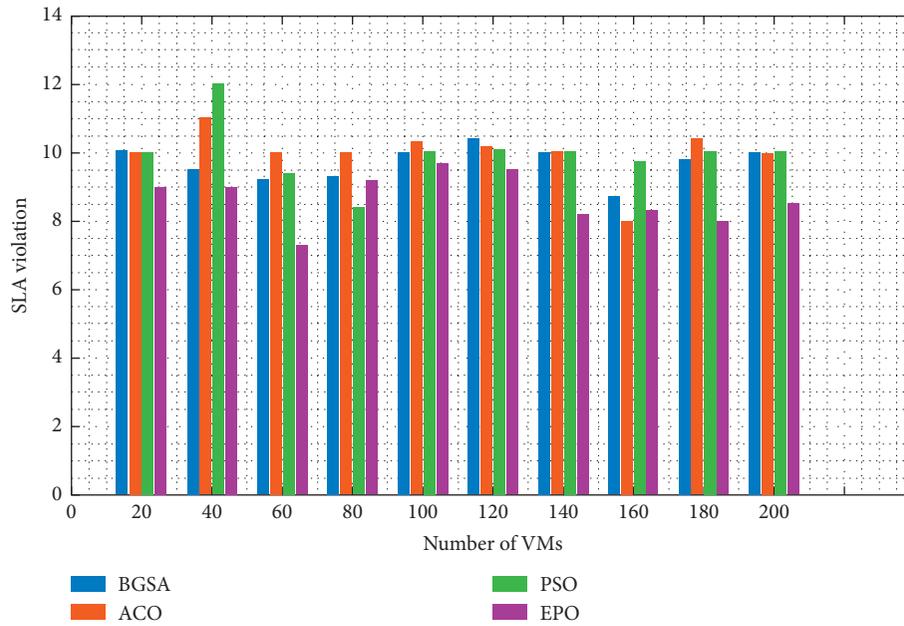


FIGURE 5: Comparison of SLA violation rate with other algorithms.

TABLE 5: Comparison of SLAV.

Number of VMs	BGSA	ACO	PSO	Proposed
40	9	11	12	7
80	9.2	10	8.2	9
120	10.2	10.1	10	9.3
160	8.8	8	9.6	8.2
200	10	10	10	8.5

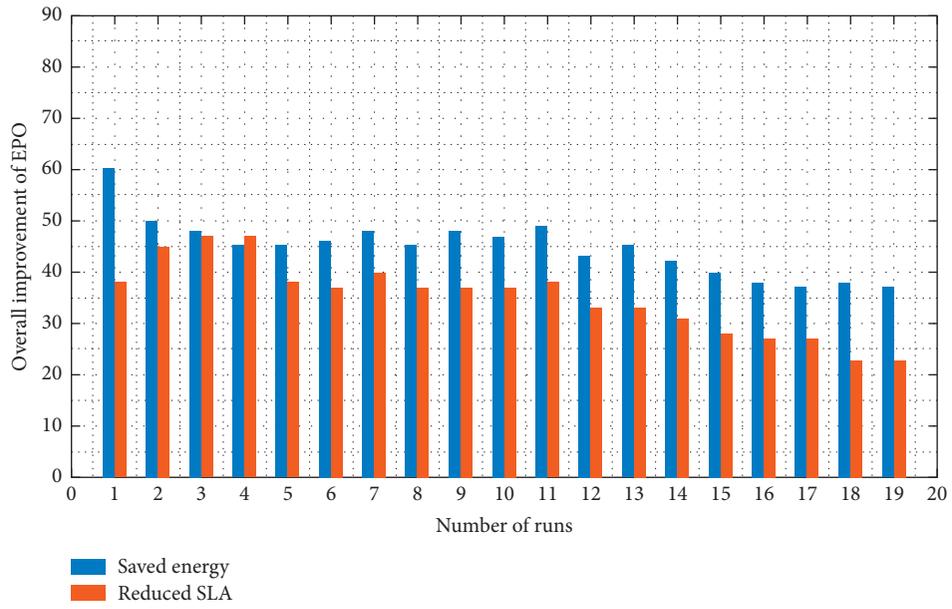


FIGURE 6: Overall improvement of EPO algorithm.

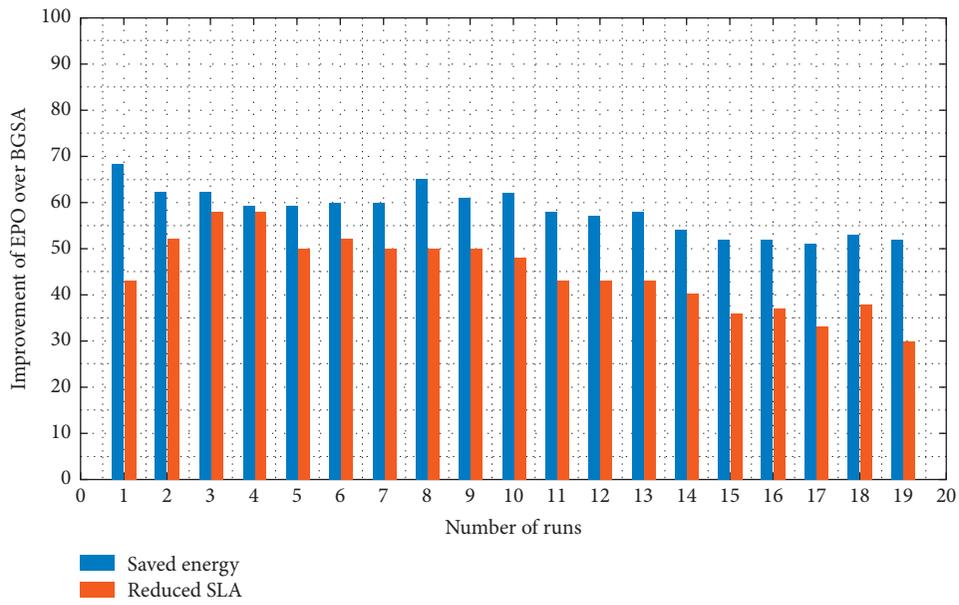


FIGURE 7: Comparison of EPO versus BGSA.

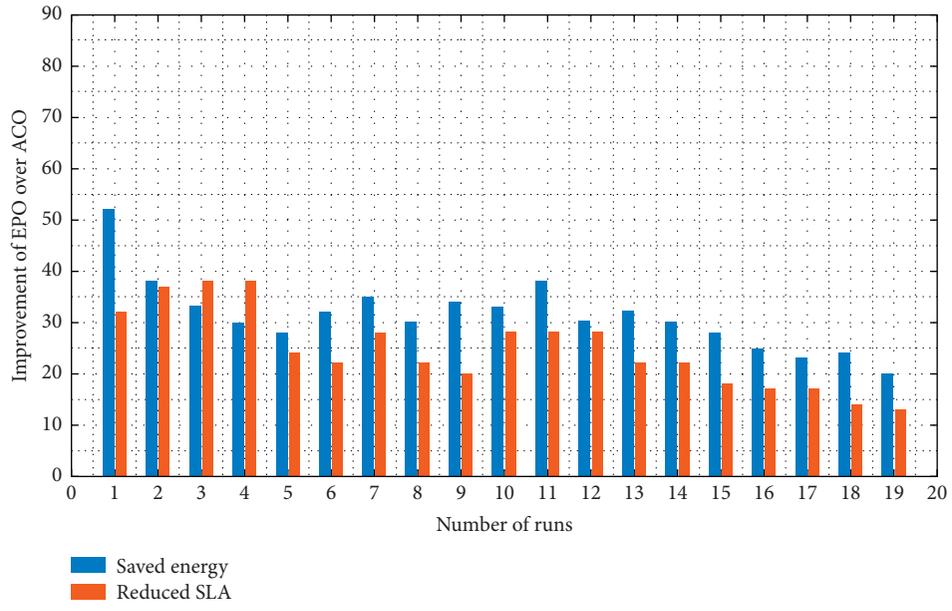


FIGURE 8: Comparison of EPO versus ACO.

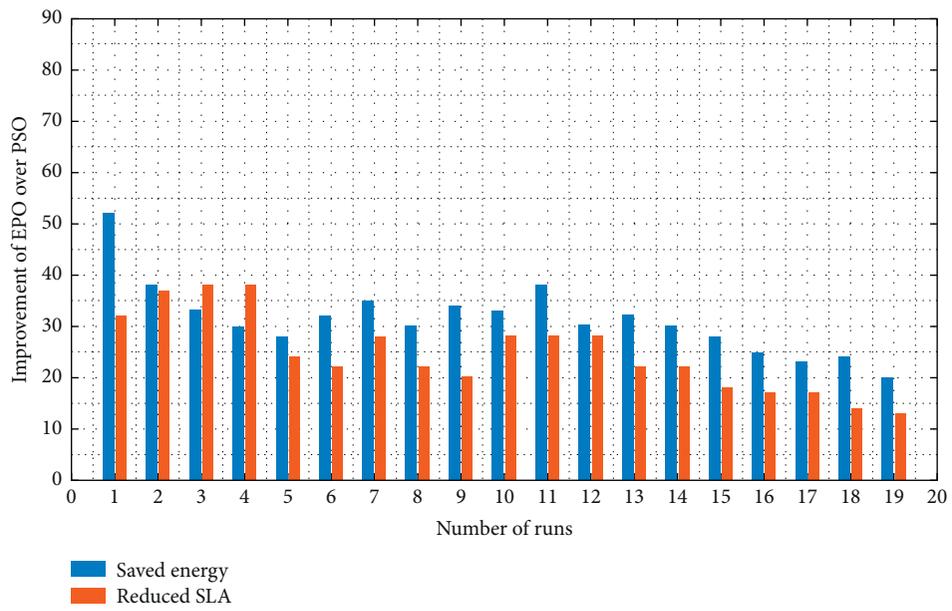


FIGURE 9: Comparison of EPO versus PSO.

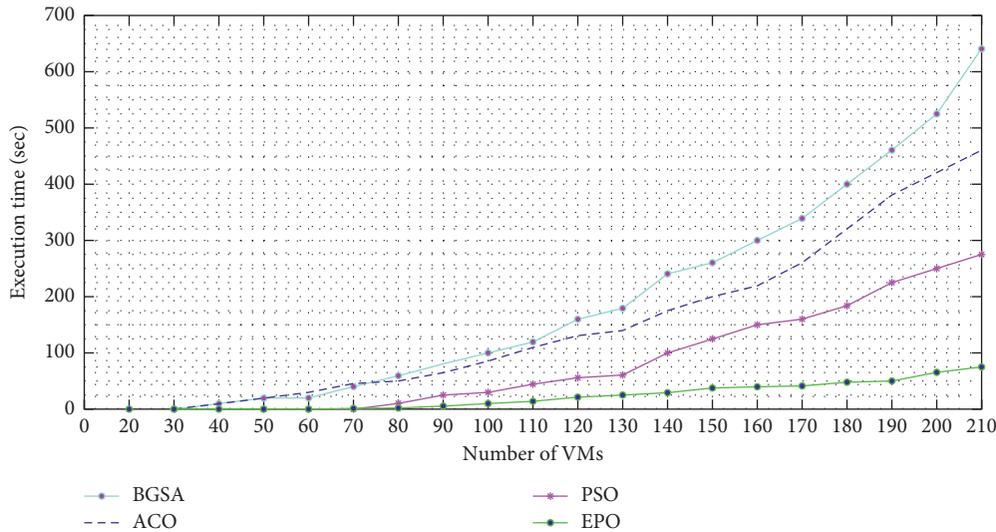


FIGURE 10: Execution time analysis of the proposed algorithm.

## 6. Conclusion and Future Work

In recent years, the important requirement of a computing system is energy efficiency and cloud data centres consume a high amount of electricity. It creates many problems like economic and environmental issues. Hence, the effective energy management is crucial for data centers. To find the solution for the best energy consumption, we proposed a multiobjective Emperor Penguin Optimization algorithm for VM allocation in this paper and the EPO is more appropriate for this problem. We analyze our results with Binary Gravitational Search Algorithm, Ant Colony Optimization, and Particle Swarm Optimization. Finally, our proposed method is better than other methods and it conserves an average of 55.96% energy and reduces an average of 47.12% SLA violations. Therefore, this method decreases the energy consumption of cloud data centers and leads to green computing. In future, we can extend this work with the combination of other methods for better energy efficiency. Moreover, it can also be extended to consider other objectives like resource wastage of cloud computing VM placement.

### Data Availability

The data used to support the findings of the study are available within the article.

### Conflicts of Interest

The authors declare that they have no conflicts of interest.

### Acknowledgments

The authors thank Taif University for its support under the Taif University Researchers Supporting Project no. (TURSP-2020/55), Taif University, Taif, Saudi Arabia.

## References

- [1] X. Fu, Q. Zhao, J. Wang, L. Zhang, and L. Qiao, "Energy-aware vm initial placement strategy based on bpsa in cloud computing," *Scientific Programming*, vol. 2018, p. 10, 2018.
- [2] M. Masdari, S. Gharehpusha, M. Ghobaei-Arani, and V. Ghasemi, "Bio-inspired virtual machine placement schemes in cloud computing environment: taxonomy, review, and future research directions," *Cluster Computing*, vol. 23, no. 4, pp. 2533–2563, 2019.
- [3] H. S. Basavegowda and G. Dagnev, "Deep learning approach for microarray cancer data classification," *CAAI Transactions on Intelligence Technology*, vol. 5, no. 1, pp. 22–33, 2020.
- [4] Z. Li, Y. Li, T. Yuan, S. Chen, and S. Jiang, "Chemical reaction optimization for virtual machine placement in cloud computing," *Applied Intelligence*, vol. 49, no. 1, pp. 220–232, 2019.
- [5] A. Abdelaziz, M. Elhoseny, A. S. Salama, and A. M. Riad, "A machine learning model for improving healthcare services on cloud computing environment," *Measurement*, vol. 119, pp. 117–128, 2018.
- [6] D. He, N. Kumar, M. K. Khan, L. Wang, and J. Shen, "Efficient privacy-aware authentication scheme for mobile cloud computing services," *Institute of Electrical and Electronics Engineers Systems Journal*, vol. 12, no. 2, pp. 1621–1631, 2018.
- [7] A. Tripathi, I. Pathak, and D. P. Vidyarthi, "Modified dragonfly algorithm for optimal virtual machine placement in cloud computing," *Journal of Network and Systems Management*, vol. 28, no. 4, pp. 1316–1342, 2020.
- [8] A. S. Abohamama and E. Hamouda, "A hybrid energy-Aware virtual machine placement algorithm for cloud environments," *Expert Systems with Applications*, vol. 150, Article ID 113306, 2020.
- [9] M. Tarahomi, M. Izadi, and M. Ghobaei-Arani, "An efficient power-aware VM allocation mechanism in cloud data centers: a micro genetic-based approach," *Cluster Computing*, vol. 2020, p. 6, 2020.
- [10] M. Kaur, D. Singh, and R. Singh Uppal, "Parallel strength pareto evolutionary algorithm-II based image encryption," *IET Image Processing*, vol. 14, no. 6, pp. 1015–1026, 2019.

- [11] M. A. Reddy and K. Ravindranath, "Virtual machine placement using JAYA optimization algorithm," *Applied Artificial Intelligence*, vol. 34, no. 1, pp. 31–46, 2020.
- [12] S. Ghosh, P. Shivakumara, P. Roy, U. Pal, and T. Lu, "Graphology based handwritten character analysis for human behaviour identification," *CAAI Transactions on Intelligence Technology*, vol. 5, no. 1, pp. 55–65, 2020.
- [13] A.-M. Ammar, J. Luo, Z. Tang, and O. Wajdy, "Intra-balance virtual machine placement for effective reduction in energy consumption and SLA violation," *Institute of Electrical and Electronics Engineers Access*, vol. 7, pp. 72387–72402, 2019.
- [14] A. Aryania, H. S. Aghdasi, and L. M. Khanli, "Energy-aware virtual machine consolidation algorithm based on ant colony system," *Journal of Grid Computing*, vol. 16, no. 3, pp. 477–491, 2018.
- [15] B. Gupta, M. Tiwari, and S. Singh Lamba, "Visibility improvement and mass segmentation of mammogram images using quantile separated histogram equalisation with local contrast enhancement," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 2, pp. 73–79, 2019.
- [16] M. Kaur, D. Singh, and V. Kumar, "Color image encryption using minimax differential evolution-based 7D hyper-chaotic map," *Applied Physics B*, vol. 126, no. 9, pp. 1–19, 2020.
- [17] V. D. Reddy, G. R. Gangadharan, and G. S. Rao, "Energy-aware virtual machine allocation and selection in cloud data centers," *Soft Computing*, vol. 23, no. 6, pp. 1917–1932, 2019.
- [18] W. Wei, X. Fan, H. Song, X. Fan, and J. Yang, "Imperfect information dynamic stackelberg game based resource allocation using hidden Markov for cloud computing," *Institute of Electrical and Electronics Engineers Transactions on Services Computing*, vol. 11, no. 1, pp. 78–89, 2018.
- [19] Y. Sharma, W. Si, D. Sun, and B. Javadi, "Failure-aware energy-efficient VM consolidation in cloud computing systems," *Future Generation Computer Systems*, vol. 94, pp. 620–633, 2019.
- [20] K. M. Baalamurugan and S. Vijay Bhanu, "A multi-objective krill herd algorithm for virtual machine placement in cloud computing," *The Journal of Supercomputing*, vol. 76, no. 6, pp. 4525–4542, 2020.
- [21] S. Azizi, M. Shojafar, J. Abawajy, and R. Buyya, "Grvmc: a greedy randomized algorithm for virtual machine placement in cloud data centers," *Institute of Electrical and Electronics Engineers Systems Journal*, vol. 2020, p. 12, 2020.
- [22] M. Mohammadhosseini, A. Toroghi Haghighat, and E. Mahdipour, "An efficient energy-aware method for virtual machine placement in cloud data centers using the cultural algorithm," *The Journal of Supercomputing*, vol. 75, no. 10, pp. 6904–6933, 2019.
- [23] S. Gharehpasha, M. Masdari, and A. Jafarian, "Power efficient virtual machine placement in cloud data centers with a discrete and chaotic hybrid optimization algorithm," *Cluster Computing*, vol. 2020, p. 2, 2020.
- [24] F. Abdessamia, W.-Z. Zhang, and Y.-C. Tian, "Energy-efficiency virtual machine placement based on binary gravitational search algorithm," *Cluster Computing*, vol. 23, no. 3, pp. 1577–1588, 2020.
- [25] D.-M. Zhao, J.-T. Zhou, and K. Li, "An energy-aware algorithm for virtual machine placement in cloud computing," *Institute of Electrical and Electronics Engineers Access*, vol. 7, pp. 55659–55668, 2019.
- [26] C. Wei, Z.-H. Hu, and Y.-G. Wang, "Exact algorithms for energy-efficient virtual machine placement in data centers," *Future Generation Computer Systems*, vol. 106, pp. 77–91, 2020.
- [27] S. Gharehpasha, M. Masdari, and A. Jafarian, "Virtual machine placement in cloud data centers using a hybrid multi-verse optimization algorithm," *Artificial Intelligence Review*, vol. 2020, p. 37, 2020.
- [28] X. Ruan, H. Chen, Y. Tian, and S. Yin, "Virtual machine allocation and migration based on performance-to-power ratio in energy-efficient clouds," *Future Generation Computer Systems*, vol. 100, pp. 380–394, 2019.
- [29] H. Feng, Y. Deng, and J. Li, "A global-energy-aware virtual machine placement strategy for cloud data centers," *Journal of Systems Architecture*, vol. 116, Article ID 102048, 2021.
- [30] E. I. Elsedimy and F. Algarni, "Toward enhancing the energy efficiency and minimizing the SLA violations in cloud data centers," *Applied Computational Intelligence and Soft Computing*, vol. 2021, 2021.
- [31] K. Braiki and H. Youssef, "Fuzzy-logic-based multi-objective best-fit-decreasing virtual machine reallocation," *The Journal of Supercomputing*, vol. 76, no. 1, pp. 427–454, 2020.
- [32] A. Al-Moalimi, J. Luo, A. Salah, and K. Li, "Optimal virtual machine placement based on grey wolf optimization," *Electronics*, vol. 8, no. 3, p. 283, 2019.
- [33] S. E. Dashti and A. M. Rahmani, "Dynamic VMs placement for energy efficiency by PSO in cloud computing," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 28, no. 1-2, pp. 97–112, 2016.
- [34] Y. Gao, H. Guan, Z. Qi, Y. Hou, and L. Liu, "A multi-objective ant colony system algorithm for virtual machine placement in cloud computing," *Journal of Computer and System Sciences*, vol. 79, no. 8, pp. 1230–1242, 2013.
- [35] M. Khurana, J. Sharma, M. K. Ranjana, Y. Jha, and L. Liu, "Direction determination in wireless sensor networks using grid topology," *Journal of Emerging Technologies in Web Intelligence*, vol. 5, no. 2, pp. 166–170, 2013.
- [36] M. Thalore, J. Sharma, M. Khurana, and M. K. Jha, "QoS evaluation of energy-efficient ML-MAC protocol for wireless sensor networks," *AEU-International Journal of Electronics and Communications*, vol. 67, no. 12, pp. 1048–1053, 2013.
- [37] M. Khurana, R. Thalore, V. Raina, and M. K. Jha, "Improved time synchronization in ML-MAC for WSN using relay nodes," *AEU-International Journal of Electronics and Communications*, vol. 69, no. 11, pp. 1622–1626, 2015.
- [38] M. Khurana, S. Shivani, S. Tiwari, M. K. Sharma, M. S. Obaidat, and K. F. Hsiao, "Optimized time synchronized multilayer MAC protocol for WSN using relay nodes," *Adhoc & Sensor Wireless Networks*, vol. 48, 2020.
- [39] S. Tiwari, K. Kaur, Y. Pathak, S. Shivani, and K. Kaur, "Computed tomography reconstruction on distributed storage using hybrid regularization approach," *Modern Physics Letters B*, vol. 33, no. 6, Article ID 1950063, 2019.
- [40] Y. Pathak, K. V. Arya, and S. Tiwari, "An efficient low-dose CT reconstruction technique using partial derivatives based guided image filter," *Multimedia Tools and Applications*, vol. 78, no. 11, pp. 14733–14752, 2019.