International Journal of Advanced Technology and Engineering Exploration, Vol 10(102)
ISSN (Print): 2394-5443 ISSN (Online): 2394-7454
http://dx.doi.org/10.19101/IJATEE.2021.876483

Research Article

Artificial bee colony optimized VM migration and allocation using neural network architecture

Sudhir Kumar Sharma [1*] and Wiqas Ghai[2]
Research Scholar, Department of Computer Application, RIMT University, Mandi Gobindgarh, Punjab, India[1]
Professor, Department of Computer Application, RIMT University, Mandi Gobindgarh, Punjab, India[2]

Received: 22-July-2022; Revised: 29-May-2023; Accepted: 30-May-2023

Abstract
The concept of cloud computing has emerged to address the challenges posed by advancements in internet technology, which attract internet users to access various online resources using multiple applications managed by third parties. Despite its numerous advantages, the cloud computing environment faces challenges such as service level agreement (SLA) violations and increased energy consumption. In this context, a proposed scheme for optimized virtual machine (VM) allocation and migration aims to be energy efficient and minimize violations while meeting the demand for storage space. The allocation of tasks involves using an artificial bee colony (ABC) optimization approach to reduce the overall computation cost. This information is then fed to the support vector machine (SVM), which sends the optimized feature vector to an artificial neural network (ANN) to complete the migration task. Comparative analysis against existing work demonstrates an overall improvement of 3% to 9% in VM migrations, 1% to 6% in energy consumption, and 1% to 5.5% in SLA violations. Furthermore, the effectiveness of the proposed power-aware ABC-based VM allocation and migration is evaluated based on the success rate, which claims better resource allocation for delivering high-end quality of service (~10%) in terms of the number of delivered packets and (~4%) improvement in response time for completing jobs in minimum time. Additionally, the work demonstrates minimal overall migration cost (~3%) involved in delivering better service using the proposed approach.

Keywords
VM migration, VM allocation, Artificial bee colony (ABC), Artificial neural network (ANN).

1.Introduction
In recent years, cloud computing (CC) has been mostly preferred by users due to its advanced features as compared to other environments. There are many advantages to CC, like security, the privacy of data, fast speed of the internet, reliability, etc. [1–3]. But still, some issues arise in CC, like service attacks, less privacy, etc. The CC architecture is categorized into two parts; the user end and the cloud data center end. The CC architecture, which explains how a user interacts with a cloud data center, is shown in *Figure 1*. The front end defines the process of user interaction with the cloud by using the internet. Users can interact with any internet-connected device, including personal computers, mobile phones, etc., and the device communicates with the cloud data center in the backend.

*Author for correspondence

It describes the processes of the cloud based on user requests. The cloud data center is divided into many sub-parts like security, infrastructure, storage, etc.

1.1Virtual machine allocation
Nowadays, virtual machines (VMs) are used with CC to reduce the server load on the cloud data center. VM allocation is the most important technique that helps to optimize the total energy consumption of cloud data centers. VM allocation is not an easy task that is performed on CC.

It is very difficult and in the current scenario, it has become a big issue in cloud environments. It is very important to map the request of the physical machine (PM) with the application that is stored in the cloud data center [4]. *Figure 2* describes the architecture of VM allocation. The architecture is divided into two layers; the application layer and the hardware layer. The purpose of the application layer is to interact with the users of the cloud services by using the internet. The hardware layer provides information regarding PMs. The cloud data center is responsible for responding to each request that comes from the application layer. The purpose of the VM scheduler is to arrange the number of requests that arise in the VM.

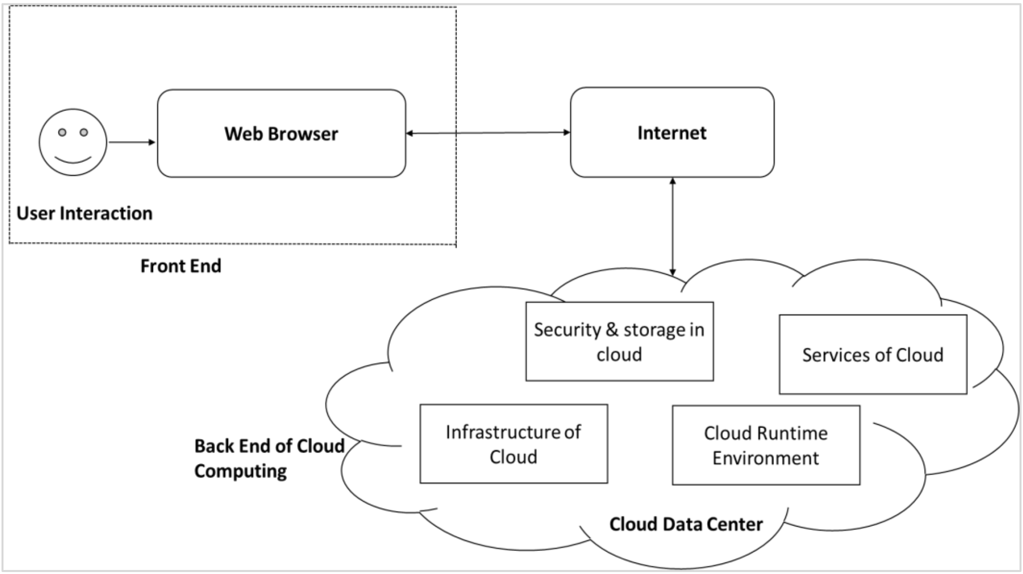


Figure 1 Cloud computing architecture using front-end and back end

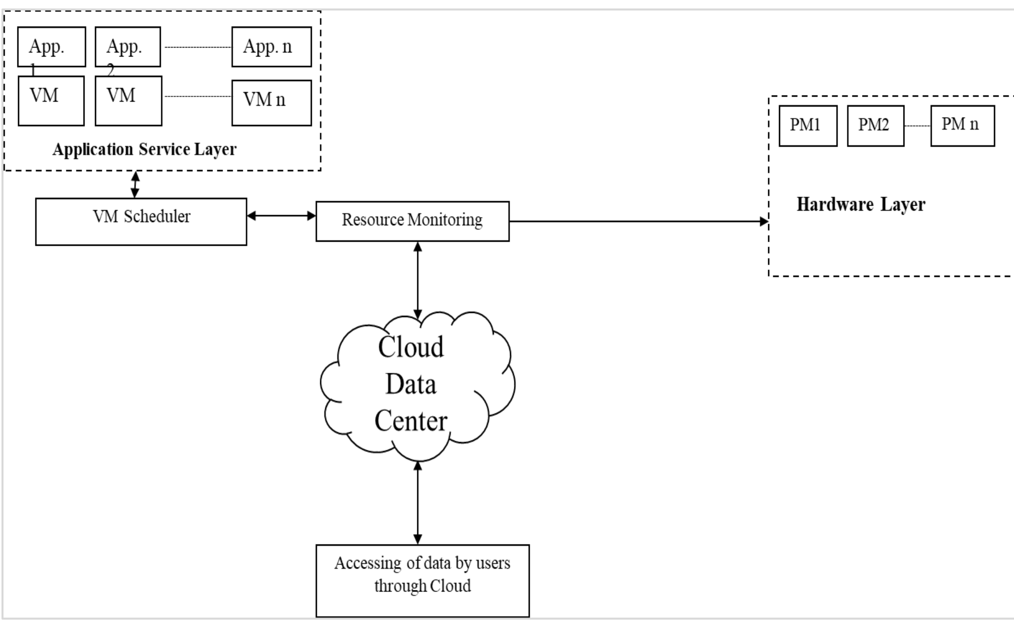


Figure 2 VM allocation

1.2Motivation

Power consumption has been one of the key factors while addressing the issues related to VM allocation and migration policies. The proposed work aims to reduce overall power consumption and maintain service level agreement **(**SLA) policies [5, 6]. The total consumed power in a given interval of time is referred to as energy consumption. When PM is overloaded with tasks, it consumes high power and faced the issue of computation complexity at the same time. As for the example, if execution is attainable under \$100 and the outcome is \$120, then \$20 worth of power is wasted in the time which adds to global warming [7]. The aim is to attain a high

SLA with the least power consumption and minimize the SLA violation and number of migrations while meeting the high-end quality of service (QoS) parameters in terms of packet delivery ratio (PDR), and response time.

1.3Contribution

The main contributions of the paper are listed below**.**

a) Utilize the artificial bee colony (ABC) algorithm to minimize overall power consumption and effectively manage resources.
b) The approach has been validated based on training and testing criteria using machine learning classifier.
c) The features obtained through support vector machine (SVM) have been further trained and tested using a neural network-based Levenberg algorithm to improve the VM selection.
d) A comparative analysis has been conducted with other states of art algorithms/techniques using QoS evaluation parameters.

The main objective of the paper is to overcome the challenges of the existing work that include issues related to energy consumption, VM migration, cost of VM migration, and SLA violations [8, 9]. The proposed technique helps to minimize the energy consumption by optimal VM allocation and migration which is further evaluated in terms of QoS parameters and resolve the limitations of the existing work related to energy consumption, VM migration, and SLA-violations, VM migration cost minimization [10, 11].

Further, the paper is organized into various sections including an introduction to CC and the challenges of VM allocation discussed in section 1. The recent literature addressing the VM allocation and migration challenges is described in section 2. Section 3 discusses the proposed work in the form of algorithmic architecture of the “power-aware artificial bee colony” (PA-ABC) used for resource allocation. Section 4 summarizes the results and discusses the claimed outcomes. Section 5 concludes and is followed by the list of references and abbreviations.

2.Literature survey

The research published in the recent years to address the challenges of secure communication while maintaining the QoS are discussed in this section. Network attacks have arisen with rising popularity of CC and hence cannot be neglected. To address such incidents and to secure the data communication, (Khadka et al., 2015) introduced a tool named intrusion detection system (IDS) that depends upon Snort to detect distributed denial of service (DDoS) attacks. The purpose of the tool is to alert network administrators about attacks and also define the possible nature of attacks. The purpose of work is not only to detect the attacks but also to suspend attackers for some time [12].

Ahmad et al., 2015 have introduced a review that is based on state of art live and non-live virtual migration methods. There are a large number of issues that arise in VM migration, but the author describes that the security threat is a big issue as compared to others. The discussion of the paper is based on various applications and parameters that affect VM migration. Different challenges seen in VM migration are the workload of the system, memory size, etc. [13].

(Singh et al. 2019) have presented a resource scheduling technique that is used with the help of the ant colony optimization (ACO) algorithm. Resource scheduling is a big challenge in CC. As the analysis of the comparative results, the proposed algorithm provides better efficiency as compared to the previous technique [14].

(Arivudainambi and Dhanya 2017) have introduced a technique named cuckoo search (CS) optimization for resource allocation that is responsible to provide a secured and reliable VM to the users. Task scheduling and resource allocation are big issues in CC. The author uses a flow analyzer that monitors jobs and provides resources to the users [15].

(Kakkar and Young 2018) have proposed a machine learning technique that is designed for VM allocation for energy efficiency in CC. The work depends upon the job scheduling process of the VM allocation and modified best fit decreasing (MBFD) is used as a classifier using artificial neural network (ANN) [16].

(Barlaskar et al., 2018) have presented work based on the solving issue of energy consumption on data that is stored on the cloud data center. The author proposed an algorithm named enhanced cuckoo search (ECS) that is implemented. As a result, a comparative result of ECS is evaluated against the genetic algorithm (GA), ant colony (AC) algorithm, and optimized firefly search (OFS) algorithm. The parameters that are used for results evaluations are; energy consumption, workload, the performance of SLA, and VM migration [17].

(Samriya and Kumar, 2020) proposed a work that is related to the (QoS) parameter including makespan, minimization of migration (MM) of tasks, security, and cost. The proposed algorithm defines that the requests are executed without disturbing the SLA. The author used a fuzzy ant bee colony (FABC) for implementation. The result section shows the secure job scheduling and assurance of QoS [18].

(Kumar et al., 2020) presented a QoS based on a resource allocation approach because resource allocation is a very important part of CC. The author used ACO for optimization. CloudSim tool is used for the implementation. As a result of the analysis, 12% of energy-saving increased the performance of resource allocation [19].

(Peter and Indumathi, 2021) presented a work that is used to detect DDoS attacks that are based on VMs. They proposed an improved CS algorithm to detect load distribution and find the reasons for attacks. As the classification, a fuzzy extreme learning machine (FELM) is used. The CloudSim simulator is used for the implementation [20].

(Asghari and Jafari 2021), the ACO method was utilized to address issues like VM migration and allocation for cloud applications that use the least amount of energy. The ACO divided the goals and provided some advice on how to solve the issues. The results showed that the ACO is still having trouble fixing several problems. The local optimal problem is a stochastic issue that needs to be considered and globally optimal solutions are required for efficient migration [21].

(Mangalampalli et al. 2022) have proposed an algorithm that focuses on the job scheduling process from job initialization to job end. The work is implemented by using the CS algorithm that tracks the performance of severing to check the server does not get overloaded. MBFD is used for setting schedule rules that provide migration of VM between cloud data centers [22].

(Tran et al., 2022) had addressed the challenges related to the migration of the machines in CC architecture. The authors proposed a VM migration algorithm based on the concept of Q-learning and Markov decision-making models. The work was comprised of a training phase followed by the extraction phase. The superiority of the proposed algorithm in terms of feasibility and strengths of the extraction phase is demonstrated using comparative analysis against max-min ant system, round-robin, and ant system algorithms [23].

The challenges related to machine migration in CC architecture were addressed by Tran et al. (2022) [23]. An algorithm was proposed for the VM migration. It is based on Q-learning and Markov decision-making models. The work consisted of a training phase followed by an extraction phase. Through comparative analysis against the max-min ant system, round-robin, and ant system algorithms, the superiority of the proposed algorithm was demonstrated in terms of feasibility and the strengths of the extraction phase.

(Peake et al. 2022), the assignment of a VM to PM in a CC environment is the difficult optimization challenge known as the virtual machine placement (VMP) problem. VM placement can have a big impact on how a cluster uses its resources, which affects operating expenses and the environment. In this study, the authors provided a parallel ant colony optimization (PACO)-based improved algorithm for VMP that effectively utilized parallelization strategies and the contemporary processor techniques which were employed in the distributed computation. The author has achieved a solution quality that was approximately parred with or better than those obtained by existing nature-inspired methods, with an increase in efficiency up to 2002x over contemporary consecutive algorithms available in the literature [24].

(Mangalagowri and Venkataraman 2023) have proposed an Adaptive firefly algorithm for resource allocation to ensure that better VM selection was performed to minimize the energy consumption in cloud data centers. The simulation results have shown that the proposed work performed better in terms of PDR with minimal delay and low energy consumption in comparison to the existing studies [9].

(Pushpa and Siddappa 2022) purposed a novel framework named "Fractional Artificial Bee Chicken Swarm Optimisation" (ABCSO) for the optimal placement of VM in the cloud. Initially, VM placement was performed with the help of various system factors including central processing unit (CPU), million instructions per second, migration cost, frequency, power, and QoS". The experimental analysis has shown that the developed algorithm outperformed other existing techniques in terms of load energy consumption and migration cost [11].

Parameter evaluation has always been a part where the researchers have shown keen interest and it is necessary to know the proposed evaluation criteria and the importance of outcome. From the illustration of *Table 1*, energy consumption is one of the major parameters that is referred to by different authors that include recent and significant related work. Hence the proposed algorithm is also energy consumption centric.

Table 1 Comparative analysis of VM migration frameworks

Author	**Method**	**Result**	**Limitation**
Sharma and Guddeti,2016 [25]	Hybrid Genetic Cat Swarm Optimization	Minimize the Energy Consumption	Issue related to SLA violations
Arivudainambi and Dhanya, 2017 [15]	CS	Minimize the average job execution time	Issue of optimal migration observed
Kakkar and Young 2018 [16]	MBFD and ANN	Improved the Energy Efficiency	Migration cost issue
Barlaskar et al., 2018 [17]	ECS+ GA Optimized Firefly	Energy Consumption has been minimized	The problem of SLA in dense Workload time
Kumar et al., 2020 [19]	ACO	Minimize the Energy Consumption	SLA-violation issue in an overloaded condition
Karthikeyan et al. , 2020 [10]	ABC + Bat Algorithm	Lower energy consumption	False VM migration issue
Li et al., 2020 [26]	Machine learning-based Host state 3rd-order Markov chain	Minimize SLA violation	Issue related to false migration and QoS parameters like PDR
Talwani and Singla, 2021 [8]	ECS	Energy Consumption reduced	Issue of unwanted VM migrations
Mangalampalli et al. 2022 [22]	MBFD + CS	Energy Consumption is minimized and SLA is improved	Higher cost of VM migration during network complexity
Najm and Tamarapalli, 2022 [27]	Polynomial-time VM migration algorithm	Minimize VM cost and execution time	Issue of SLA violation and PDR
Arshad et al. , 2022 [28]	Energy Efficiency Heuristic using VM Consolidation	Energy consumption, VM migration, and SLA violations reduced	Issue related to response time and migration cost in an overload condition
Singh et al., 2022 [29]	Modified QoS-aware Task Consolidation algorithm	Reduced SLA violation and time	An issue faced related to energy consumption and execution time of VM migration
Jain and Sharma 2022 [30]	Quantum-inspired Salp Swarm Grey Wolf Algorithm	Improved the response time, SLA violation	Migration cost and energy consumption issues faced in case of overloaded task
Khan and Santhosh 2022 [31]	CS + PSO	Improved QoS parameters and energy consumption	Issue related to SLA violation, execution time, migration cost
Rakrouki and Alharbe 2022 [32]	PSO + gravitational search algorithm	Minimize the SLA violation and energy consumption	Issue related to migration cost and response time
Mangalagowri and Venkataraman, 2023[9]	Adaptive Firefly Algorithm	Secured data transfer and lower energy consumption	Issue of VM migration cost and false migration in complex scenarios
Pushpa and Siddappa 2022 [11]	Fractional Artificial Bee Chicken Swarm Optimisation	Minimize energy consumption and migration cost	Issues related to SLA violations in complex scenarios
Zhao et al., 2023 [33]	Multi-objective ACO	Minimize migration cost, and energy consumption and improved QoS parameters	Issue related to SLA violation, response time
Yao et al., 2023 [34]	Load balancing strategy based on VM consolidation	Minimize energy consumption and SLA-violation	Issue related to migration cost and QoS parameters
Mangalampalli et al., 2023 [35]	Chaotic Social Spider algorithm	Minimize the makespan and energy consumption	Issues related to migration cost and SLA-violation

The existing literature discussed the various issues related to VM migration and allocation, energy consumption, VM migration cost, and SLA violation. The existing technique considered the optimization algorithm, swarm intelligence-based algorithms in order to address the challenges but no approach has been combining the strategies of swarm intelligence, machine learning, and neural network. Hence, the proposed technique has an edge over the existing techniques with superior optimization, feature classification, and selection in order to address the problems related to energy consumption, VM migration, and cost, PDR success rate, response time, and SLA-violation.

3.Proposed methodology

The usage of machine learning has become a common practice in the world of VM allocation and migration. The studied literature says that to cross verify. In the allocation architecture, the best way is to validate the data through a classification algorithm. The proposed work model is inspired by the architecture of neural networks which consist of different layers such as input, hidden, and output which are used to perform different tasks. In the proposed model the neural network approach helps to improve the VM migration, allocation, and selection criteria due to the combined impact of machine learning and swarm intelligence which helps to improve the feature selection and classification ability in neural network-based schemes. The neural network framework is based on the Levenberg algorithm [36] that relies on the gradient variation which improves the training and testing features using previous data and working neurons on the hidden layer. It provides both feed-forward and back-propagation methods for the processing which helps to improve the overall proposed neural network strategy. The proposed work model aims to reduce the computation cost of the allocation of the task over the cloud platform by managing the SLA-violation limit over the physical architecture of the deployed service network. It is assumed that there are several service vendors against different demand requests and varying bandwidth utilization.

As the nature of the cloud environment is always assumed to be dynamic and hence the incoming request utility cannot be measured directly. The learning content will have several types of file formats and structures and hence will have a varying number of bits to be processed. The density of the bit pattern is evaluated using the computer's usage of the CPU in the task which is supplied by the user/student. The overall structure and work done are illustrated using *Figure 3*. It is divided into various layers highlighting the user layer, service layer, and the grouping architecture borrowed from ABC [11]. The service layer shows the communication between service managers and PM This is followed by the coordinated working of employed bees and the onlooker bees. The most commonly used similarity measure approaches are cosine and correlation-based. Under the correlation-based approach, a person correlation measure is used to calculate the similarity between two variables, which can be defined as shown in Equation 1 [10].

$$PC(u_1,u_2) = \frac{\sum_{i=1}^{n}(Rt_{u_1,i}-\overline{Rt_{u_2}})(Rt_{u_2,i}-\overline{Rt_{u_1}})}{\sqrt{(Rt_{u_2,i}-\overline{Rt_{u_1}})^2}\sqrt{\sum_{i=1}^{n}(Rt_{u_1,i}-\overline{Rt_{u_2}})^2}} \quad (1)$$

n→ Total number of items in the user-item space.
The similarity between the two users (u_1, u_2) is represented by the Equation 1. Here, $Rt_{u_1,i}$ is the rating given to item (i) by the mean of user 1 and user 2.

Also, prediction for an item can be measured by integrating the weights of the selected neighbor's rating [37]. The general formula used for prediction is given by Equation 2 as follows.

$$Pr(u_1,i) = \overline{Rt_{u_1}} + \frac{\sum_{i=1}^{n}(Rt_{u_2,i}-\overline{Rt_{u_2}})PC(u_1,u_2)}{\sum_{i=1}^{n}PC(u_1,u_2)} \quad (2)$$

Cosine similarity is a vector space model that works based on linear algebra instead of a statistical approach. Using this approach, the similarity between the two n-dimensional vectors is calculated based on the angle between them. In the proposed work, two types of bees are used; Employed bee and Onlooker bee that is implemented for performance analysis. In the present work architecture, cosine similarity represents the similarity measure between two vectors that are represented by employed bee group (d1) members and onlooker bee group (d2) members. It computes the cosine angle between the two vectors for the measurement of orientation. The steps followed for the computation of cosine similarity are as follows.

- Compute dot product betweent the two vectors $d1_{members}$ and $d2_{members}$.
- Compute the mangitudes L2 norms of $d1_{members}$ and $d2_{members}$.
- Compute cosine similarity between $d1_{members}$ and $d2_{members}$ using Equation 3.

Mathematically, the cosine similarity is represented as follows.

$$Cosine_{similarity} = \frac{\overrightarrow{d1_{members}} \cdot \overrightarrow{d2_{members}}}{|\overrightarrow{d1_{members}}| \ |\overrightarrow{d2_{members}}|} \quad (3)$$

Where, numerator represents the dot product of the space vectors of the two bee group memenbers and the denominator represents the product of L2 norms of the same. Here, $\overrightarrow{d1_{members}}$ denotes the vector representation of employed bee group members and $\overrightarrow{d2_{members}}$ denotes the vector representation of onlooker bee group memebrs.

Now, there are two types of values stored in the following *Figure 4* in which the first and second column indicates the values related to employed bees and the third and fourth indicates values of Onlooker bees. The performance analysis using ABC based on the Employed bee and Onlooker bee has been summarized in *Figure 4*.

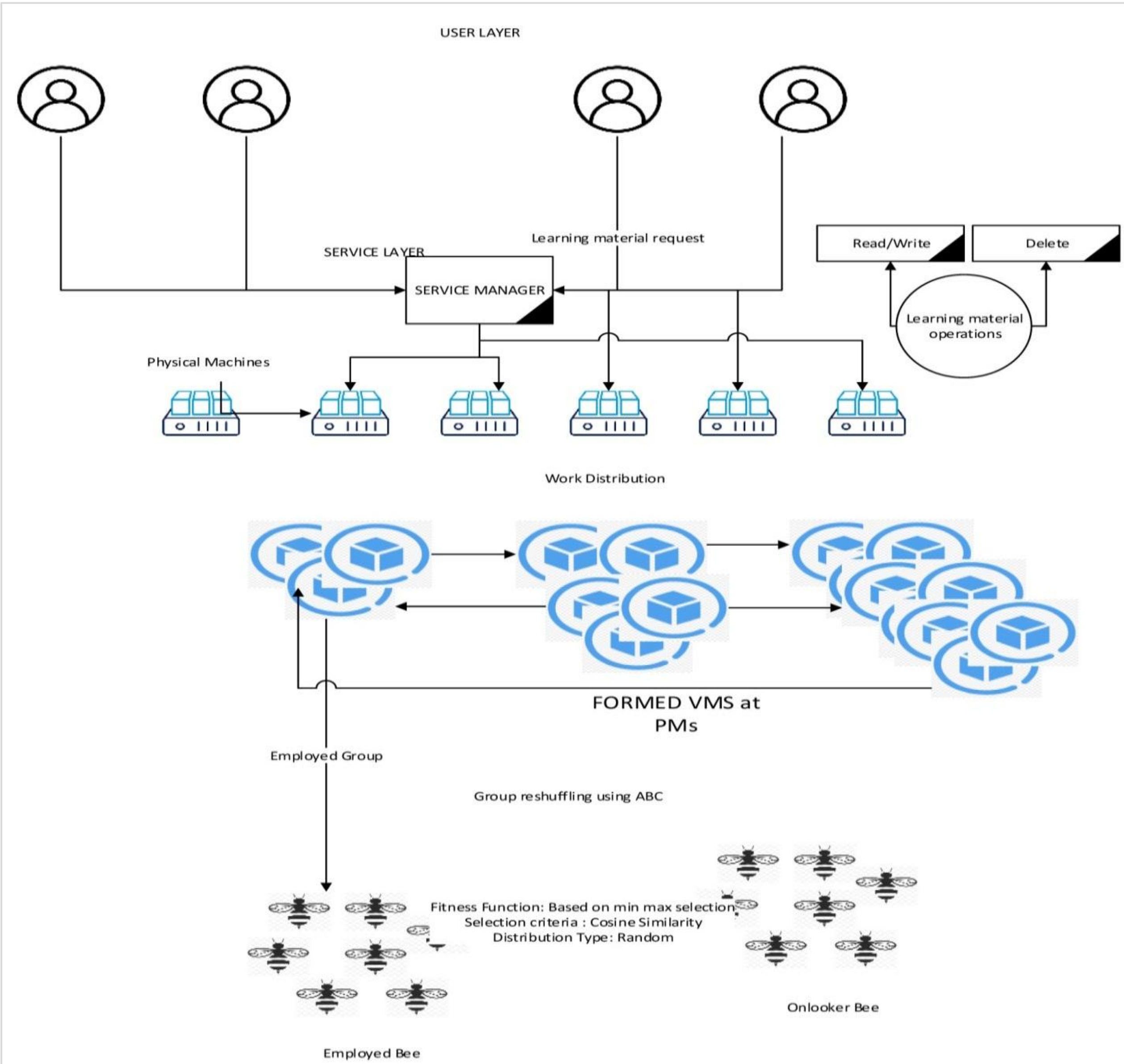


Figure 3 Proposed work diagram

The proposed algorithm uses the VMs to be migrated through the MM algorithm. The proposed algorithm uses Idol power consumption, consumed at referred PM as a primary parameter along with the power consumption in a non-Idol state. For one 'PM,' the VMS will be evaluated.

Algorithm: PA-ABC [Power Aware-ABC]

1.Inputs: $Mig_{VMs}(mVm)$, $Mig_{VM_{Group}}(mVmg\ ,VM_{work_{Architecture}}(Vm_wA)$

Vm_wA will contain all the essential detail of "allocated time (at)", "Idol time (It)", "used-CPU-Idol (UCI)", "Non-Idol time (NIt)", "Non-Idol $(UCnI)$" and Power consumed in both the states as essential elements.

States-Idol and non-idol, $LTL = [\ ]$;

2. Output: Low-Trust-List (LTL).

LTL is the list that contains the information of the VMs that are not suitable for the specific PM for a while.

3. for each, host in mVmG

//for every host that has ever got a VM. //

a) $f_1 = find\ (mVmG == host)$

//**F**ind the allocation indexes of a current host in the list. //

b $f_2 = find\ (mVmG.f_1.VMs\ )$

//find the allocation VMs of an identified host.//

c) $f_{3_{(UCI)}} = find\ (VmwA.f_2\ .UCI$

// find UCI for identified VMS //

d) $f_4 = find\ (VmwA.f_2.UC_nI)$

//find UCI for identified VMs//

e) $f_5 = find\ (VmwA.f_2.Pci)$

f) $f_6 = find\ (VmwA.f_2.Pcni)$

(Where p is the power consumed in the idol state and Pcni is the power consumed non-idol state.)

4. End for$_2$

5. $Employed_{Bee_{group}}(EBg) = \int_{i=1}^{m} \int_{j=1}^{n} (f_3, f_4, f_5, f_6)_{\{2tm,\ nItm\}}\, dt$

6. $On_{looker_{bee}}(OBg = \int_{k=1}^{t} EBg\ dt$

Where

m= total number of hosts.

n= total number of VMS

t= a total number of intervals.

As each parameter is of a different parametric frame cocaine similarity for both the groups EBg and OBg is used.

For$_2$ each Bee, we consider a primary 100 allocations in VmwA. Hence Evaluation of EBg is done based on 100 files.

7. $EBg = \int_0^p EBg_p dt$ *Where p is 100 in the proposed case.*

8. $Evaluate - X_1 = \int_{i=0}^{p} [\int_{j=0}^{n} cos - sim(EBg)_n dt]pdt$

9. $Evaluate - X_2 = \int_{i=0}^{p} [\int_{j=0}^{n} cos - sim(OBg)_n dt]pdt$

10.fx = fitness PA – ABC $\begin{pmatrix} 1 & if & \frac{\sum_{i=1}^{p} x_1}{p} > \frac{\sum_{i=1}^{p} x_2}{p} \\ 0 & & otherwise \end{pmatrix}$

11. if f_x ==1

Accept Bee and reallocation is possible

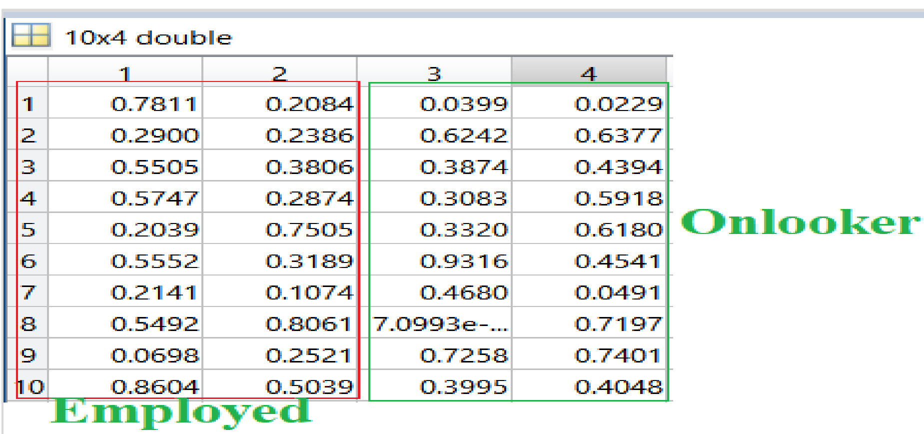
else

Not acceptable, reduce the workload (add to LTL).

end if

12. End for $_2$.

13. Return LTL.



	1	2	3	4
1	0.7811	0.2084	0.0399	0.0229
2	0.2900	0.2386	0.6242	0.6377
3	0.5505	0.3806	0.3874	0.4394
4	0.5747	0.2874	0.3083	0.5918
5	0.2039	0.7505	0.3320	0.6180
6	0.5552	0.3189	0.9316	0.4541
7	0.2141	0.1074	0.4680	0.0491
8	0.5492	0.8061	7.0993e-...	0.7197
9	0.0698	0.2521	0.7258	0.7401
10	0.8604	0.5039	0.3995	0.4048

Figure 4 Performance analysis using ABC

The proposed ABC algorithm enhances the current performance of the allocation strategy by employing a multi-objective fitness function to the desired set of evaluations. The ABC algorithm uses the properties of the VMs marked as {f1, f2, f3, f4, f5, f6} to form and employed a bee. The onlooker bee takes the same set of parameters to justify the onlooker bee. The onlooker bee uses the mean absolute value for the processing of the fitness function. ABC considers each host as a bee hive and each VM as an employed bee. The evaluation parameters are directly connected to power efficacy in order to attain optimal power consumption value. The bee flies are evaluated utilizing an interval ratio which illustrates the time gap in which the user is allocated.

The threshold for the judgment is 100 simulations and hence a matrix of $\{100 \times t\}$ is prepared for each employed bee where t represents the total number of time divisions. Load passed over each employed bee is apprehended to maintain integrity in the proposed system. The fitness function is a binary repeater that either results in 1 or 0 in which 1 is attained in which 0 represents the drop, and 1 represents the selection. The fitness function incorporates a load of ‘t’ simulation to its consumed power which is also referred to as fly position in the proposed work. To validate the proposed algorithm sequence, a SVM is used as a classifier. The overall work can be described using *Figure 5.* In the proposed model, firstly VM and PM are deployed and classify the PMs on the basis of migration cost and allocation of the VM to the minimum cost PM. Then, identify the hotspot in PMs and categorize them in terms of utilization (under or overutilized), accordingly initiating the ABC technique in order to optimize the power consumption based on CPU utilization. Examine fitness function, and whether it meets the minimum energy consumption at the time of VM migration. Once the conditional criteria meet, further machine learning-based SVM classifier has been employed to extract the features which provide the optimal outcomes in terms of energy consumption, minimum migration cost, and SLA violations. Finally, the features obtained through SVM are further trained and tested using ANN based on the Levenberg algorithm which is one of the most efficient training algorithms that relies on the gradient variation and the gradient is generated using existing data and active neurons on the hidden layer. It provides both feed-forward and back-propagation methods for processing [38].

The proposed approach works in order to minimize the response time to complete the job. The rank repository has been created to assist the migration which increases the success rate of PDR and improves the overall efficiency of the proposed model using the Levenberg algorithm.

The ordinal measures of work can be defined using *Table 2* as follows.

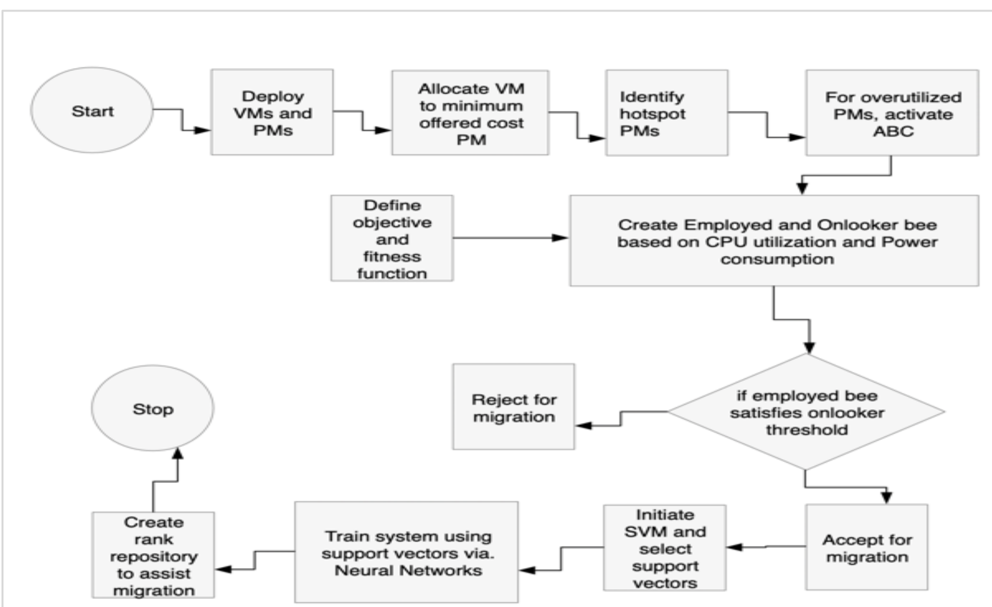


Figure 5 Overall block diagram

Table 2 System and training specifications

Hardware Requirement	a) 1.2 GHz CPU b) 1 Gb HDD c) Windows 10 Or Above /Macintosh 3.1 and above
Training Algorithm	Levenberg
Number of neurons per layer	10
Number of hidden layers	2
Total number of epochs	1000
Stopping criteria	Gradient/Epochs

The evaluations of the proposed work have been done to significantly deny the null hypothesis as follows.

Hypothesis

H0: There is no significant difference in the power consumption with enhanced load distribution and migration scheme.
H1: Alternative
H2: There is no significant impact of the total number of migrants in the evaluation of total power consumption.
H3: Alternative

SVM is capable of efficiently performing both linear and non-linear classification tasks. When it comes to non-linear classification, the concept of kernel plays a crucial role. Kernels are mathematical functions that enable SVM to find a hyperplane capable of distinguishing the training data by mapping the non-linearly separable dataset into high-dimensional feature spaces. By employing kernels, SVM significantly contributes to its performance and reduces the complexity compared to deep learning approaches. These kernels serve the purpose of transforming nonlinearly separable input data into linearly separable data in higher-dimensional space. In the field of machine learning, the term "kernel," also referred to as the "kernel trick," represents a technique that employs a linear classifier to address non-linear problems. It involves the transformation of data that is not linearly separable into linearly separable data. The kernel function is applied to each data instance, facilitating the mapping of the original non-linear observations into a higher-dimensional space where they become separable. The most commonly used kernel function is the radial basis function (RBF). It is to be noted that when the linear kernel is used only the c parameter needed to be optimized. However, the inclusion of the RBF kernel needs to optimize both the gamma parameter and c parameters simultaneously. The idea here is that if gamma is larger the effect of c is negligible and vice versa. Ideally, the values of these two SVM parameters are $0.0001 < gamma < 10$ and $0.1 < c < 100$.

Standard kernels are not able to differentiate some specific types of data sets and hence performance reduces on such types of data sets. These kernel functions have two main disadvantages. First, it only models the inner product between individual feature vectors as opposed to an ensemble of vectors.

Secondly, they are quite generic and do not take advantage of the statistics of the individual signal under consideration. Therefore, multi-layered neural network architecture is also involved in the present work.

4.Results and discussion

The result section describes the simulation results of the proposed work. The comprehensive analysis is performed for various scenarios and against the existing work. The parameters used in the comparative analysis are VM migrations, energy consumption, and some SLA violations to depict the resource scheduling of the proposed work. *Table 3* describes the Comparative analysis of VM migrations by using ABC-SVM, ABC-SVM-ANN, and Karthikeyan et al. 2020 [10] and Talwani and Singla 2021 [8]. The first column of the table describes the number of VMs that lie between 10 and 100. The last three columns describe the VM migrations using ABC-SVM, Karthikeyan et al., 2020 [10], Talwani and Singla 2021 [8], and ABC-SVM-ANN.

The average values of VM migration for ABC-SVM, Karthikeyan et al., 2020 [10], Talwani and Singla 2021 [8] and the proposed ABC-SVM-ANN are 39.4, 37.8, 37.2, and 36.1. The obtained results showed that the proposed technique has a minimum number of VM migrations. The overall %improvement of the proposed technique over Karthikeyan et al. 2020 [10], and Talwani and Singla, 2021 [8] is shown in *Figure 6*. The proposed technique has shown the average improvement against other techniques in terms of percentage i.e., 9.14% over ABC-SVM, 4.70% over Karthikeyan et al. 2020 [10], and 3.04% over Talwani and Singla, 2021 [8] for a number of VM migrations which is due to better resource scheduling achieved with the involvement of trained support vectors for ANN in the proposed work.

Energy consumption is an important parameter to compute resource scheduling efficiency. *Table 4* describes the comparative analysis of energy consumption analyzed for the existing and the proposed work for 100 VMs. The detailed analysis summarized shows that the proposed ABC-SVM-ANN provides better results in comparison to the existing techniques reflecting better resource scheduling efficiency. *Table 4* shows the average values of energy consumption(kW) for ABC-SVM, Karthikeyan et al., 2020 [10], Talwani and Singla 2021[8] and proposed ABC-SVM-ANN are 1.18 kw, 1.14 kw, 1.13 kw, and 1.11 kW.

Table 3 Comparative analysis of VM migrations

Number of VMs	VM Migrations			
	ABC-SVM	**Karthikeyan et al., 2020 [10]**	**Talwani and Singla 2021 [8]**	**ABC-SVM-ANN**
10	10	10	10	9
20	11	11	11	10
30	22	19	18	17
40	27	25	24	23
50	30	28	28	27
60	40	38	37	36
70	52	51	50	49
80	60	58	57	55
90	70	68	67	66
100	72	70	70	69

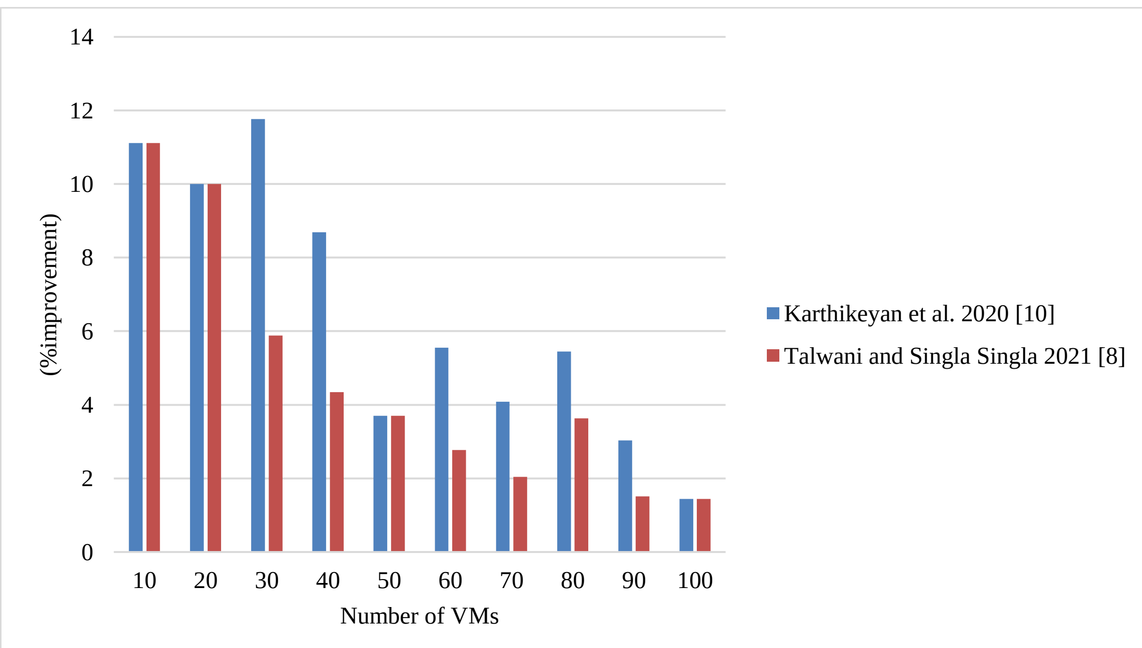


Figure 6 % Improvement in terms of the number of VM migrations

Table 4 Comparative analysis of energy consumption

Number of VMs	Energy consumption(KW)			
	ABC-SVM	**Karthikeyan, 2020 [10]**	**Talwani and Singla 2021 [8]**	**ABC-SVM-ANN**
10	0.26	0.25	0.24	0.23
20	0.34	0.33	0.32	0.31
30	0.75	0.73	0.71	0.69

Number of VMs	Energy consumption(KW)			
	ABC-SVM	**Karthikeyan, 2020 [10]**	**Talwani and Singla 2021 [8]**	**ABC-SVM-ANN**
40	0.94	0.92	0.91	0.9
50	1.05	0.99	0.98	0.97
60	1.28	1.23	1.21	1.18
70	1.51	1.47	1.45	1.42
80	1.54	1.49	1.47	1.46
90	1.98	1.91	1.89	1.88
100	2.17	2.14	2.12	2.11

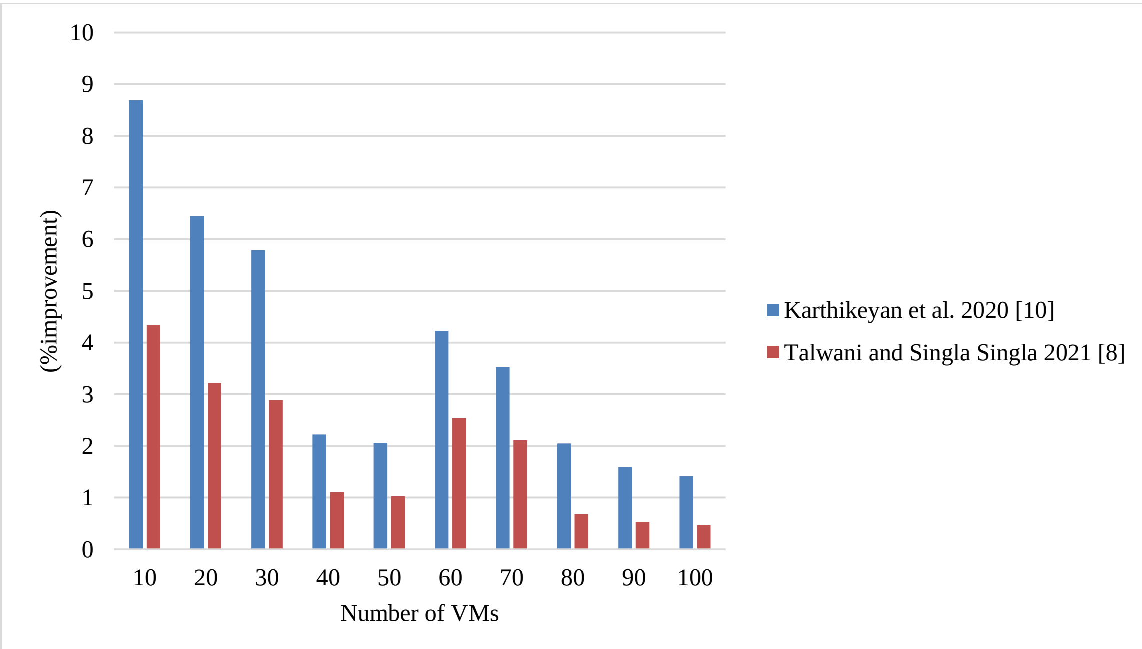


Figure 7 % Improvement in terms of energy consumption

The obtained results showed that the proposed technique has minimum energy consumption. Further, as illustrated in *Figure 7*, the proposed technique has shown the average improvement against other techniques as 6.30% over ABC-SVM, 2.70% over Karthikeyan et al. (2020) [10], and 1.80% over Talwani and Singla (2021)[8] for energy consumption. Thus, a significant improvement is shown by the proposed work owing to the involvement of ABC and two machine learning techniques namely SVM and ANN.

Table 5 describes the comparative analysis performed for the evaluation in terms of SLA violation using ABC-SVM, Karthikeyan et al. (2020) [10], Talwani et al. (2021) [8] and proposed ABC-SVM-ANN. The table lists the observed SLA violation against variation in the number of VMs from 10 to 100. It is observed that for 100 VMs, the average value of SLA violations for ABC-SVM, Karthikeyan et al. (2020) [10], Talwani and Singla (2021) [8] and proposed ABC-SVM-ANN are 1.15, 1.13, 1.11, and 1.09 (*Figure 8*).

Table 5 Comparative analysis of SLA violation

Number of VMs	SLA Violation			
	ABC-SVM	**Karthikeyan et al., 2020 [10]**	**Talwani and Singla 2021 [8]**	**ABC-SVM-ANN**
10	0.24	0.24	0.23	0.22
20	0.26	0.25	0.24	0.23
30	0.71	0.69	0.68	0.67
40	0.93	0.93	0.91	0.9
50	1.04	1.03	1.01	0.99

Number of VMs	SLA Violation			
	ABC-SVM	Karthikeyan et al., 2020 [10]	Talwani and Singla 2021 [8]	ABC-SVM-ANN
60	1.18	1.16	1.14	1.13
70	1.33	1.3	1.29	1.27
80	1.58	1.55	1.51	1.49
90	2.03	1.98	1.95	1.92
100	2.25	2.19	2.17	2.12

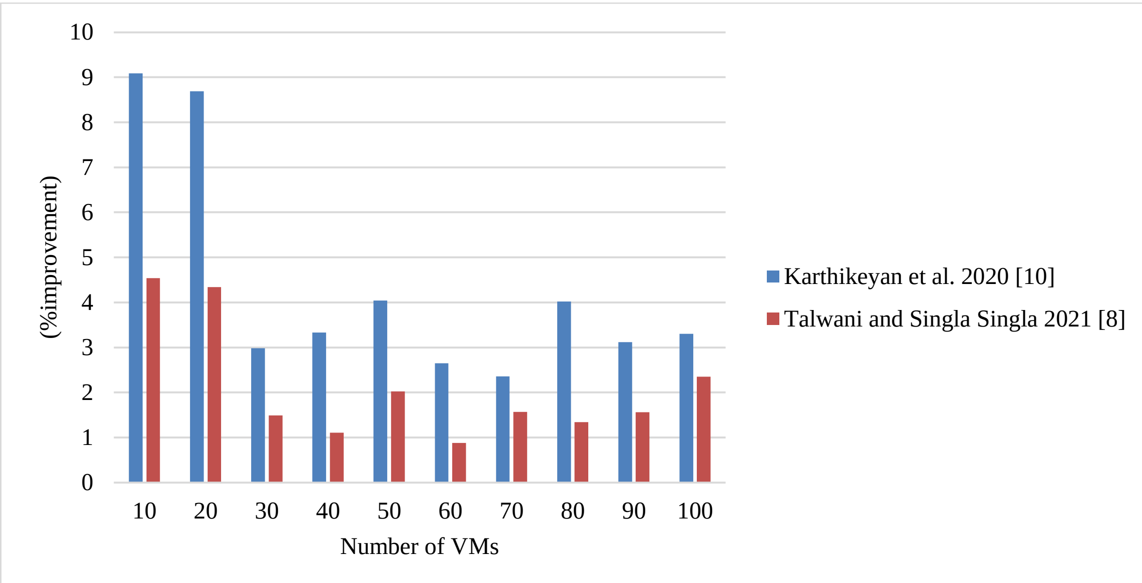


Figure 8 % Improvement in terms of SLA violation

4.1Comparative analysis

The proposed power aware – ABC-SVM-ANN architecture illustrated better performance in delivering quality service using CC. The success rate and extent of better resource scheduling is further analyzed in term of achieved PDR, success rate, and overall migration cost in this section. The proposed technique improves resource scheduling optimally with the help of a neural network which enhances decision-making during the VM allocation in overutilized scenarios and effectively uses the underutilized VMs.

PDR evaluation

The comparative analysis for the PDR is shown in *Figure 9*. The proposed work exhibited an average PDR of 79.103% to reflect the success of the power-aware proposed technique for optimal VM allocation in comparison to 71.557% and, 71.22% demonstrated by Mangalagowri and Venkataraman (2023) [9], and Pushpa and Siddappa (2022) [11]. The average success rate of the proposed algorithm stands higher with the average success rate of the proposed algorithm touching the 80% mark with a lot of variations. Here, the job completion or the success rate of task execution is high due to the involvement of ABC, machine learning, and neural network abilities proposed work.

Migration cost evaluation

Further, lower migration cost is also claimed by the proposed work using power-aware ABC due to better scheduling.

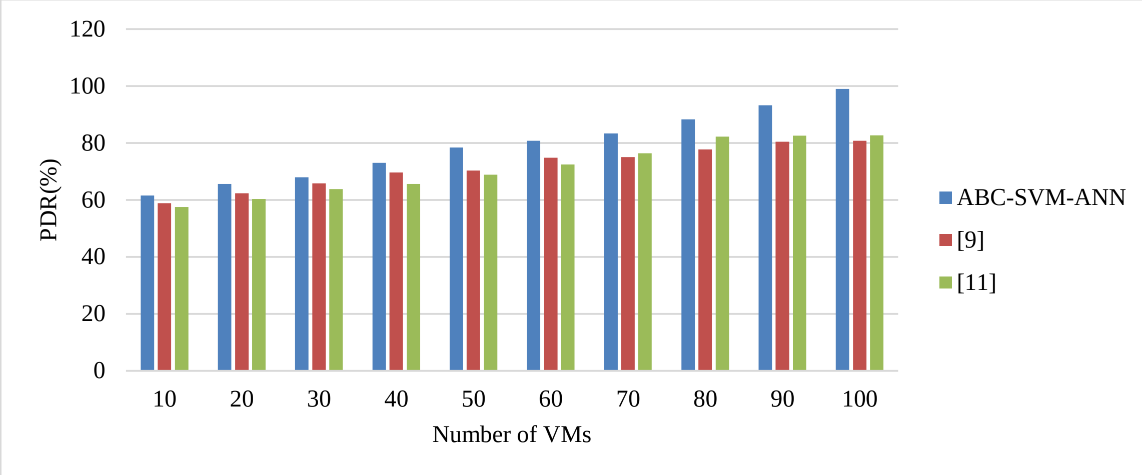


Figure 9 Success rate evaluation

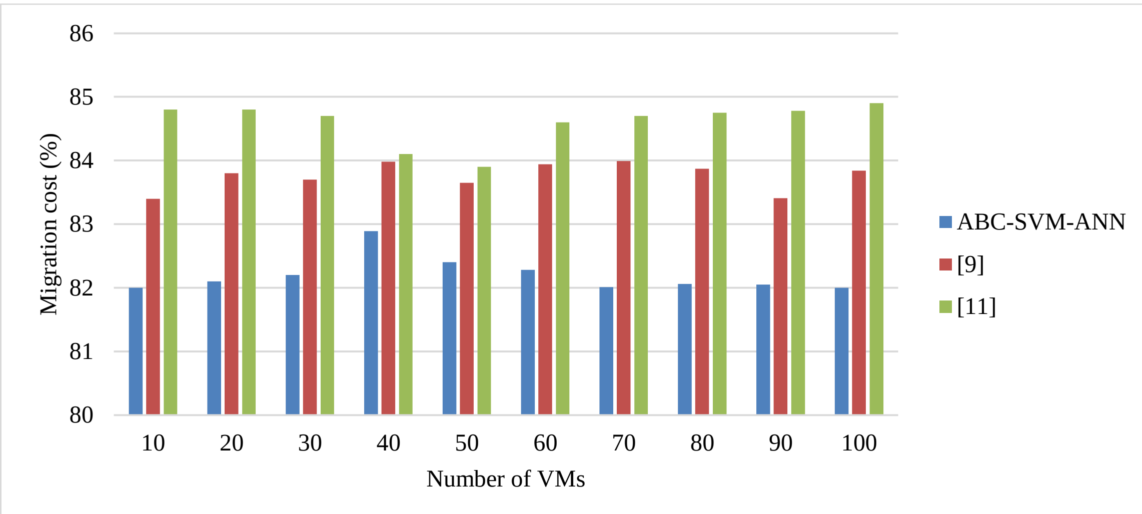


Figure 10 Average migration cost evaluation

The average migration cost for techniques implemented by Pushpa and Siddappa (2022) [11], Mangalagowri and Venkataraman (2023) [9], and proposed are 84.604%, 83.7599%, and 82.1995%. The overall migration cost against the number of VMs by the proposed work against the existing work is shown in *Figure 10*. Thus, it is observed a better selection mechanism is proposed in the paper. Overall, it is observed that the involvement of ABC in performing power-aware VM allocation and migration improves the overall efficiency of CC. The proposed technique showed a reduction in average migration cost by 2.93% from Pushpa and Siddappa (2022) [11] and 1.89% from Mangalagowri and Venkataraman (2023) [9]. This is further, claimed that due better selection mechanism involved for VM selection and resource allocation the minimum migration cost and better QoS are also associated with the proposed work.

Response time Evaluation

The response time also played a crucial role in order to understand the efficacy of the proposed work. The proposed work involved a combination of ABC, SVM, and ANN, and without proposed work scenario involved only ABC and SVM. *Figure 11* shows the response time evaluation with and without proposed work in addition to existing studies in order to examine the effectiveness of the developed framework. The proposed work employed the swarm intelligence-based neural network with a machine learning strategy which helps to make the overall model more responsive with the least execution time in order to complete the job of VM allocation and migration with lower cost and higher PDR.

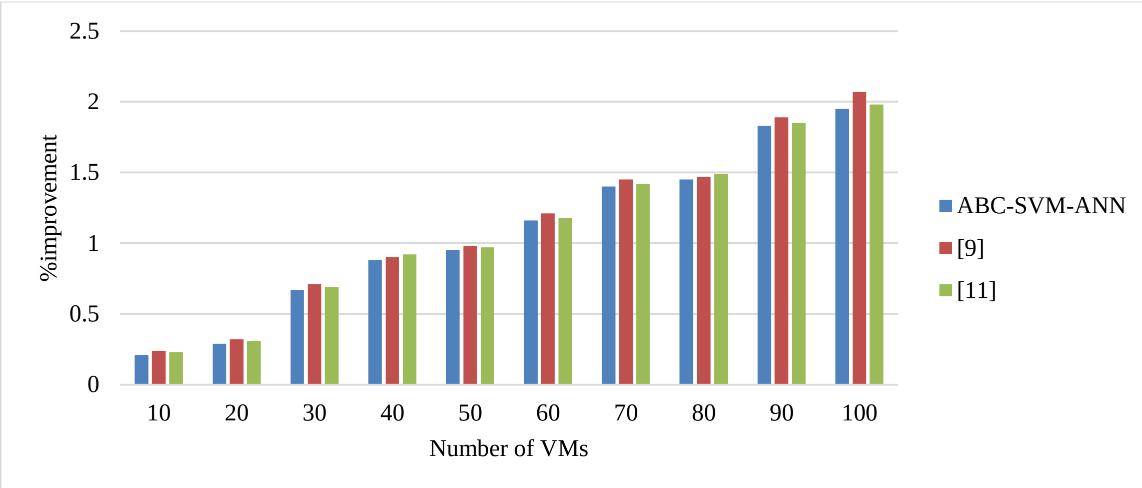


Figure 11 Response time evaluation

The impact of response time has been computed with the proposed ABC-SVM-ANN, and without the proposed work as Pushpa and Siddappa (2022) [11] and Mangalagowri and Venkataraman (2023) [9]. *Figure 11* shows that the average response time of the proposed technique has been improved by 2.31% and 4.13%, over techniques without the proposed i.e., ABC-SVM.

4.2 Discussion

The proposed technique enhances the overall performance of the allocation strategy by utilizing a multi-objective fitness function that calculates the absolute value to measure efficiency. It significantly improves energy consumption, VM migration, and SLA violations compared to other techniques. The proposed work is divided into two scenarios: the first is based on ABC-SVM, and the second is ABC-SVM-ANN. The results of the proposed work are compared to ABC-SVM, Karthikeyan et al. (2020) [10], and Talwani and Singla (2021) [8].

On average, there is a 9.14% improvement in VM migrations compared to ABC-SVM, a 4.70% improvement compared to Karthikeyan et al. (2020) [10], and a 3.04% improvement compared to Talwani and Singla (2021) [8]. Similarly, there is an average improvement of 6.30% in energy consumption compared to ABC-SVM, 2.70% compared to Karthikeyan et al. (2020) [10], and 1.80% compared to Talwani and Singla (2021) [8]. Additionally, there is an average improvement of 5.50% in terms of SLA violations compared to ABC-SVM, 3.66% compared to Karthikeyan et al. (2020) [10], and 1.83% compared to Talwani and Singla (2021) [8].

The existing research identified limitations such as energy wastage due to unwanted/false VM migrations in a CC environment. Furthermore, lower PDR, higher migration cost, and prominent SLA violations were also limitations [8-11]. The proposed technique overcomes these limitations by combining swarm intelligence with machine learning classifiers and neural network approaches, resulting in enhanced performance metrics. It minimizes unwanted/false VM migrations and optimizes VM allocation, reducing overall migration costs and increasing the success rate of the PDR.

Moreover, the proposed work exhibits high responsiveness with the lowest response time and minimum job execution time. The designed architecture has been simulated in a cloud environment for a specific duration, and the rest of the algorithms are arranged similarly. The success rate achieved by the proposed algorithm is the highest within the given timeframe compared to other algorithms.

A complete list of abbreviations is shown in *Appendix I.*

5.Conclusion and future work

The proposed work focuses primarily on optimizing VM migrations in a CC environment. The main objective of the research is to understand the impact on QoS metrics during VM migration. The proposed technique, ABC-SVM-ANN, addresses limitations related to unwanted/false migrations, energy consumption, and SLA violations found in previous works by Karthikeyan et al. (2020) [10] and Talwani and Singla (2021) [8]. Comparative analysis against existing work demonstrates an overall improvement of 3% to 9% in VM migrations, 1% to 6% in energy consumption, and 1% to 5.5% in SLA violations. These improvements significantly contribute to the design of large-scale architectures that offer better services to cloud users in a cost-effective manner. Additionally, the proposed work achieves a better performance with a success rate of up to 10% in terms of packet delivery, a 3% reduction in migration cost through effective resource allocation, and up to 4% improvement in response time for job completion. The overall enhancement in QoS parameters makes the designed system more suitable for practical implications in the future. The designed architecture provides higher reliability for services at a minimum cost on a cloud platform. For future improvements, it is recommended to explore the integration of swarm intelligence with deep learning concepts to enhance resource utilization on a broader scale and include more QoS parameters to provide more realistic outcomes.

Conflicts of interest

The authors have no conflicts of interest to declare.

Author's contribution statement

Sudhir Kumar Sharma: Conceptualization, Investigation, Data curation, Writing – original draft, Writing – review and editing. **Dr. Wiqas Ghai:** Supervision, Investigation on challenges and Draft manuscript preparation**.**

References

[1] Saxena D, Singh AK, Buyya R. OP-MLB: an online VM prediction-based multi-objective load balancing framework for resource management at cloud data center. IEEE Transactions on Cloud Computing. 2021; 10(4):2804-16.

[2] Rakhi, Pahuja GL. An efficient trust-based approach to load balanced routing enhanced by virtual machines in vehicular environment. In proceedings of international conference on intelligent computing and smart communication 2019 (pp. 925-35). Springer Singapore.

[3] Mandal R, Mondal MK, Banerjee S, Biswas U. An approach toward design and development of an energy-aware VM selection policy with improved SLA violation in the domain of green cloud computing. The Journal of Supercomputing. 2020; 76:7374-93.

[4] Ezugwu AE, Buhari SM, Junaidu SB. Virtual machine allocation in cloud computing environment. International Journal of Cloud Applications and Computing. 2013; 3(2):47-60.

[5] Kumar A, Kumar R, Sharma A. Energy aware resource allocation for clouds using two level Ant colony optimization. Computing & Informatics. 2018; 37(1).

[6] Kumar A, Kumar R, Sharma A. Equal: energy and qos aware resource allocation approach for clouds. Computing and Informatics. 2018; 37(4):781-814.

[7] Beloglazov A, Abawajy J, Buyya R. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. Future Generation Computer Systems. 2012; 28(5):755-68.

[8] Talwani S, Singla J. Enhanced bee colony approach for reducing the energy consumption during VM migration in cloud computing environment. In IOP conference series: materials science and engineering 2021 (p. 012069). IOP Publishing.

[9] Mangalagowri R, Venkataraman R. Ensure secured data transmission during virtual machine migration over cloud computing environment. International Journal of System Assurance Engineering and Management. 2023:1-12.

[10] Karthikeyan K, Sunder R, Shankar K, Lakshmanaprabu SK, Vijayakumar V, Elhoseny M, et al. Energy consumption analysis of virtual machine migration in cloud using hybrid swarm optimization (ABC–BA). The Journal of Supercomputing. 2020; 76:3374-90.

[11] Pushpa R, Siddappa M. Fractional artificial bee chicken swarm optimization technique for QoS aware virtual machine placement in cloud. Concurrency and Computation: Practice and Experience. 2022; 35(4): e7532.

[12] Khadka B, Withana C, Alsadoon A, Elchouemi A. Distributed denial of service attack on cloud: detection and prevention. In international conference and workshop on computing and communication 2015 (pp. 1-6). IEEE.

[13] Ahmad RW, Gani A, Ab. Hamid SH, Shiraz M, Xia F, Madani SA. Virtual machine migration in cloud data centers: a review, taxonomy, and open research issues. The Journal of Supercomputing. 2015; 71:2473-515.

[14] Singh H, Bhasin A, Kaveri P. SECURE: efficient resource scheduling by swarm in cloud computing. Journal of Discrete Mathematical Sciences and Cryptography. 2019; 22(2):127-37.

[15] Arivudainambi D, Dhanya D. Scheduling optimized secured virtual machine using cuckoo search and flow analyzer. Journal of computational and Theoretical Nanoscience. 2017; 14(6):2715-24.

[16] Kakkar D, Young GS. Heuristic of VM allocation to reduce migration complexity at cloud server. In proceedings of the international conference on scientific computing 2018 (pp. 60-6). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).

[17] Barlaskar E, Singh YJ, Issac B. Enhanced cuckoo search algorithm for virtual machine placement in cloud data Centres. International Journal of Grid and Utility Computing. 2018; 9(1):1-7.

[18] Samriya JK, Kumar N. Fuzzy ant bee colony for security and resource optimization in cloud computing. In 5th international conference on

computing, communication and security 2020 (pp. 1-5). IEEE.

[19] Kumar A, Sharma A, Kumar R. A swarm intelligence-based quality of service aware resource allocation for clouds. International Journal of Ad Hoc and Ubiquitous Computing. 2020; 34(3):129-40.

[20] Peter Soosai Anandaraj A, Indumathi G. Improved cuckoo search load distribution (ICS-LD) and attack detection in cloud environment. Concurrency and Computation: Practice and Experience. 2021; 33(3): e5226.

[21] Asghari S, Jafari Navimipour N. The role of an ant colony optimisation algorithm in solving the major issues of the cloud computing. Journal of Experimental & Theoretical Artificial Intelligence. 2021:1-36.

[22] Mangalampalli S, Sree PK, Usha Devi N SS, Mallela RB. An effective VM consolidation mechanism by using the hybridization of PSO and cuckoo search algorithms. In proceedings of computational intelligence in data mining 2022 (pp. 477-87). Singapore: Springer Nature Singapore.

[23] Tran CH, Bui TK, Pham TV. Virtual machine migration policy for multi-tier application in cloud computing based on Q-learning algorithm. Computing. 2022; 104(6):1285-306.

[24] Peake J, Amos M, Costen N, Masala G, Lloyd H. PACO-VMP: parallel ant colony optimization for virtual machine placement. Future Generation Computer Systems. 2022; 129:174-86.

[25] Sharma NK, Guddeti RM. On demand virtual machine allocation and migration at cloud data center using hybrid of cat swarm optimization and genetic algorithm. In fifth international conference on eco-friendly computing and communication systems 2016 (pp. 27-32). IEEE.

[26] Li L, Dong J, Zuo D, Ji S. SLA-aware and energy-efficient VM consolidation in cloud data centers using host state 3rd-order Markov chain model. Chinese Journal of Electronics. 2020; 29(6):1207-17.

[27] Najm M, Tamarapalli V. Towards cost-aware VM migration to maximize the profit in federated clouds. Future Generation Computer Systems. 2022; 134:53-65.

[28] Arshad U, Aleem M, Srivastava G, Lin JC. Utilizing power consumption and SLA violations using dynamic VM consolidation in cloud data centers. Renewable and Sustainable Energy Reviews. 2022; 167:1-14.

[29] Singh P, Prakash V, Bathla G, Singh RK. QoS aware task consolidation approach for maintaining SLA violations in cloud computing. Computers and Electrical Engineering. 2022; 99: 107789.

[30] Jain R, Sharma N. A quantum inspired hybrid SSA–GWO algorithm for SLA based task scheduling to improve QoS parameter in cloud computing. Cluster Computing. 2022:1-24.

[31] Khan MS, Santhosh R. Hybrid optimization algorithm for VM migration in cloud computing. Computers and Electrical Engineering. 2022; 102: 108152.

[32] Rakrouki MA, Alharbe N. QoS-aware algorithm based on task flow scheduling in cloud computing environment. Sensors. 2022; 22(7):1-20.

[33] Zhao H, Feng N, Li J, Zhang G, Wang J, Wang Q, Wan B. VM performance-aware virtual machine migration method based on ant colony optimization in cloud environment. Journal of Parallel and Distributed Computing. 2023; 176:17-27.

[34] Yao W, Wang Z, Hou Y, Zhu X, Li X, Xia Y. An energy-efficient load balance strategy based on virtual machine consolidation in cloud environment. Future Generation Computer Systems. 2023; 146:222-33.

[35] Mangalampalli S, Karri GR, Rajkumar KV. EVMPCSA: efficient VM packing mechanism in cloud computing using chaotic social spider algorithm. Procedia Computer Science. 2023; 218:554-62.

[36] Yan Z, Zhong S, Lin L, Cui Z. Adaptive levenberg–marquardt algorithm: a new optimization strategy for Levenberg–Marquardt neural networks. Mathematics. 2021; 9(17):1-17.

[37] Kim H, Kim J, Herlocker J. Feature-based prediction of unknown preferences for nearest-neighbor collaborative filtering. In fourth international conference on data mining 2004 (pp. 435-8). IEEE Computer Society.

[38] Cui W, Qu W, Jiang M, Yao G. The atmospheric model of neural networks based on the improved levenberg-marquardt algorithm. Open Astronomy. 2021; 30(1):24-35.

Sudhir Kumar Sharma is a research scholar at RIMT University in Mandi Gobindgarh, Punjab, India. Currently, he is serving as an assistant professor at the Post Graduate Government College for Girls, Sector-42, Chandigarh. He holds a Bachelor of Computer Applications (BCA) and a Master of Computer Applications (MCA) degree. His primary research interests lie in the areas of Cloud Computing and the Internet of Things (IoT). He has presented and published several research papers on these topics.
Email: nangal_sudhir@yahoo.co.in

Dr. Wiqas Ghai holds a Ph.D. in Computer Science and has 18 years of experience in academia. He currently works as a Professor. His research interests include Artificial Intelligence, Machine Learning, Speech Recognition, and Big Data Analytics. He has published numerous research papers on these topics.
Email: ghaialpha@gmail.com

Appendix I

S. No.	Abbreviation	Description
1	ABC	Artificial Bee Colony
2	AC	Ant Colony
3	ACO	Ant Colony Optimization
4	ANN	Artificial Neural Network
5	CC	Cloud Computing
6	CPU	Central Processing Unit
7	CS	Cuckoo Search
8	DDoS	Distributed Denial of Service
9	ECS	Enhanced Cuckoo Search
10	FABC	Fuzzy Ant Bee Colony
11	FELM	Fuzzy Extreme Learning Machine
12	GA	Genetic Algorithm
13	IDS	Intrusion Detection System
14	MBFD	Modified Best Fit Decreasing
15	MM	Minimization of Migration
16	OFS	Optimized Firefly Search
17	PACO	Parallel Ant Colony Optimization
18	PA-ABC	Power-Aware Artificial Bee Colony
19	PM	Physical Machine
20	QoS	Quality of Service
21	RBF	Radial Basis Function
22	SLA	Service Level Agreement
23	SVM	Support Vector Machine
24	VMs	Virtual Machines
25	VMP	Virtual Machine Placement