An Optimal Resource Provisioning Scheme Using QoS in Cloud Computing Based Upon the Dynamic Clustering and Self-Adaptive Hybrid Optimization Algorithm

Article · May 2023

DOI: 10.22266/ijies2022.0630.13

1 author:

Jaspal Singh
Sant Longowal Institute of Engineering and Technology

Received: January 11, 2022. Revised: February 25, 2022.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

An Optimal Resource Provisioning Scheme Using QoS in Cloud Computing Based Upon the Dynamic Clustering and Self-Adaptive Hybrid Optimization Algorithm

Jaspal Singh[1]*

[1]*Department of Computer Sc. & Engineering, Sant Longowal Institute of Engineering and Technology Deemed University, Longowal, Sangrur, Punjab, India*
***** Corresponding author's Email: jaspalsingh@sliet.ac.in

Abstract: The Cloud computing (CC) offers an efficient way of executing IaaS (infrastructure as a service). However, virtualized datacentres still face numerous challenges like imbalanced server loads and low resource utilization of PMs. The key aim of the paper is to diminish the utilization of the energy of datacentres while sustaining the requirements of QoS (quality of service). In this paper, we developed an optimal resource provisioning scheme by QoS in CC based upon the dynamic clustering and self-adaptive hybrid optimization algorithm (SHOA). At first, the k-medoid algorithm is utilized for clustering of tasks and VMs. Then the fuzzy-based multi-dimensional resource scheduling (FMRS) algorithm is used to allocate a group of tasks to a suitable collection of VMs. Finally, the dynamic VM consolidation is improved by the SHOA. This SHOA algorithm merges the integrated particle swarm optimization (iPSO) and teaching and learning-based optimization (TLBO) algorithms. VM consolidation is deliberated as a multi-objective issue, this difficulty is solved by a SHOA. The proposed (KSHOA) is executed on the CloudSim tool and simulation outcomes are estimated based on QoS factors. The results verified that the developed KSHOA method offers a good balance among consumption of energy of Datacentre's and sustained QoS. The simulation results of the proposed scheme achieve energy consumption 104.12 kWh, resource wastage 1.2%, migration rate 5282, ESV 7.286 % and SLA violation 0.0527, when compared to other approaches.

Keywords: Resource management, Dynamic VM consolidation, Quality of service, Energy consumption, Hybrid optimization algorithm, Multi-objective function.

1. Introduction

The cloud computing is the most powerful expertise in recent decades. In a pay-as-you-go mode, the CC offers suitable and cheap services to users with the help of the resource visualization approach. The distribution of anything as a resource or hosted services over the internet is the common term of CC. In the nutshell, particularly the cloud is composed of VMs, resources, hosts, datacentres, etc. To share an application between enterprises and many investors or a single physical occurrence of a resource, CC technology employs the virtualization approach [1 - 3]. Three layers are included in the cloud such as IaaS, PaaS (Platform as a service), and SaaS (Software as a service). These services are mutually supporting one another according to the application execution [4].

SaaS is a distribution scheme that can share any software consumed among several occupants of the service provider [5, 6]. The group of software improvement gradually employ PaaS and CC offers application settings as on-demand services [7]. A compute and storage resources are offered by the IaaS. The consumers purchase units of calculating period of VM occurrences in a flexible pay-as-you-go way by the internet in an IaaS of CC [8, 9].

Due to the minimum consumption of energy and resource sharing features, the CC technology has attracted several investigators compared to the traditional communication method and data sharing

[10]. The resource management (RM) is the main feature of the virtualization and CC approach. The RM handles the gaining and statement of resources. Each resource is virtualized and shared between several users in the cloud surroundings. The metrics for the RM are SLA (service level agreement) violations, system load, hybrid clouds, balancing of load, improvement of profit, mobile CC, and energy efficiency [11]. Due to the extensive range of management objectives, the scale of data centers, and heterogeneity of resources, RM is a major issue in CC technology. For the new applications, a current investigation concentrates on the drawbacks of the centralized CC systems to encounter the QoS factors [12].

At different levels of computing resources, the RM is always a key problem. A cloud consumer demand diversity of services in the CC technology based on the energetically altering requests. Therefore, it is the role of CC to reward all the required services to the cloud customers[13]. The main task in cloud computing is to create a sensible application movement to minimize the consumption of energy. The resource scheduling is an important issue in the CC systems. Because the resource allocation cannot find the usage of VMs. Moreover, due to the usage of VM, the consumption of power of datacentres can be minimized in the CC system[14, 15]. To solve these difficulties, the optimum RM approach is developed in this work.

In CC systems, already there are numerous studies have been presented in the scheduling of tasks to accomplish the high resource utilization and better performance based on the requirements of cloud providers and the users. However, there is still a demanding requirement for a scheduling method and flexible structure that includes significant needs which may have a great influence on network performance. In addition, certain works do not deliberate properties of resources and features of tasks energetically. Others concentrate on improving the response time or execution time without including any other performance metrics like utilization of resources and balancing of load. Waiting time of tasks is the significant parameter and very few works have discussed this metric. Fitness function creation is another noteworthy issue that still wants to deliberate both features of resources and tasks. To address the above-mentioned problems, the efficient approach is designed based on dynamic clustering and meta-heuristic algorithms.

The major contributions can be summarized as follows:

- ❖ To enhance the energy efficiency and assurance the user's QoS, the new approach is designed in IaaS.
- ❖ K-medoid algorithm is used for clustering process and FMRS algorithm is employed for task scheduling.
- ❖ The SHOA is proposed for all the stages of VM consolidation such as overloaded host and underload host detection, selection of VM, and placement of VM.
- ❖ To guarantee the SLA and to fulfill the energy efficiency, a new approach is implemented
- ❖ Dynamic VM consolidation is considered a multi-objective problem to reduce resource wastage of datacentres, consumption of energy, and communication cost of VMs.
- ❖ The developed resource provisioning framework using QoS Attribute structure decreases both SLA violations and energy consumption at a similar time.

The rest of this paper is prepared as trails; the subsequent section defines the recent related works in RM scheme. Section 3 deliberates the proposed method by the KSHOA approach. Section 4 describes the results section and comparison graph of proposed and existing methods. Finally, section 5 defines the conclusions as well as the future scope of the research paper.

2. Background study on resource provisioning in cloud computing and QoS constraints

Resource distribution and computation offloading were discussed in [16] with a min-max fairness guarantee for the CC. To solve the NP-hard issue, the CORA (computation offloading and resource distribution algorithm) method was introduced and the random mining and SDR (semidefinite relaxation) approach is utilized for the offloading judgment creation also. The system can improve the convergence of the algorithm but it does not consider the delay and queue length.

A cloud-based smart grid structure was developed in [17] for effective RM. In a fog, cloud-based surroundings, the recommended HABACO (hybrid artificial bee ant colony optimization) method with four load balancing methods were employed to manage resources effectively. The developed system is not suitable for multiple load balancing applications and also it is only implemented in

Received: January 11, 2022. Revised: February 25, 2022.

simulation scenarios that reduce the system performance.

For blockchain systems, a cloud or fog computing RM and pricing were discussed in [18]. In proof-of-work-based community blockchain systems, the price-based computing resource managing method was introduced to assist in offloading removal tasks to fog or cloud providers. The system does not consider the multiple patches of fog or cloud providers for performance enhancement. Moreover, the optimal schemes of the providers and miners are not included in the developed model.

Power-aware performance investigation for self-adaptive RM was developed in [19] in IaaS clouds. To estimate the performance and consumption of power for the IaaS cloud, the stochastic activity networks (SANs) were utilized. The model does not support heterogeneity by designing various classes of PMs, each one giving various memory, processing, and storage capacities.

By the heterogeneous CC, a RM was presented in [20] in supportable cyber-physical systems. To enhance the sustainability of the scheme, CPS (cyber-physical systems) method was combined with heterogeneous CC. It considered few tasks for allocation and multi-objective are not included in this model. Moreover, it provides low scalability.

By cloud and fog computing, an SNA (social network analysis) based resource optimization was presented in [21] in an optical network. It decreases the makespan time and did not consider the priority of task and deadline constraints.

A workload assignment and effective RM were developed in [22] in IoT (internet of things) based CC by the learning classifier approaches. At the edge of the systems, the learning classifier of XCS (extended classifier system) and BCM-XCS (best classifier memory- extended classifier system) is introduced to decrease the delays and to manage the consumption of power in the workload processing. This approach did not consider other parameters like convergence time and cost.

CC-based an energy-efficient resource provision method was developed in [23] in H-CRAN (heterogeneous cloud radio access network). By the supplementary consumption of power model and cooperative transmission, the benefits of energy efficiency is accomplished in the suggested method. The system attained high computational complexity and fewer resources are considered.

An energy-efficient resource optimization was discussed in [24] for H-CRANs. To improve profit by reducing the consumption of power considering the essential system uncertainties, the suggested scheme investigates an ideal resource distribution for C-RANs. The system attained low scalability and the minimum number of tasks is considered.

A self-characteristics-based energy-efficient RM was discussed in [25] for the cloud. To recognize the faulty resources to avoid misleading allocation and to enhance the utilization of resources, the suggested method provides the ideal solutions. ALO (ant lion optimization) method is used to discover the ideal resources. The malicious workloads and QoS parameters are not considered in this study.

The resource allocation and catalog management were presented in [26] based on the SLA for ideal resource utilization in the cloud. According to the BW and frequency components of SLA, the suggested work develops an SLA-based design to successively employ the resources of cloud. But it considers only a few QoS metrics and attained high implementation costs.

A coupling RM was presented in [27] in smart city systems based on fog computing. A sensor-cloud system is a famous method in several fields along with the integration of CC and CPS. To accomplish high-performance computing, the physical nodes can be shared with multiple users. The computational complexity and execution are increased in this model.

A novel bio-inspired hybrid algorithm (NBIHA) was developed in [28] for effective RM-based fog computing. This algorithm combines the modified PSO (MPSO) and modified CSO (cat swarm optimization) to decrease the average response time and to control the consumption of resources by successively assigning the jobs. Only a few resources, few VMs, and PMs are considered.

For the USVs (unmanned surface vehicles) group of CC systems, a combined computation offloading and resource allocation was presented in [29]. This article develops an adaptive upper confidence bound (AUCB) algorithm and extends it with occurrence-awareness by reforming the service function of the MAB methods. The performance is degraded in this approach and low scalability is achieved.

A distributed market-based resource provisioning was introduced in [30] in edge computing systems. According to the economic and pricing models, the DMRM (distributed market-based resource management) is introduced for the dynamic resource provisioning of IoT systems. The approach did not consider the real-world applications and provided a high computational time.

By analyzing the above existing works, it can be determined that the majority of the existing works performed the resource provisioning and VM consolidation in cloud computing to improve the

Received: January 11, 2022. Revised: February 25, 2022.

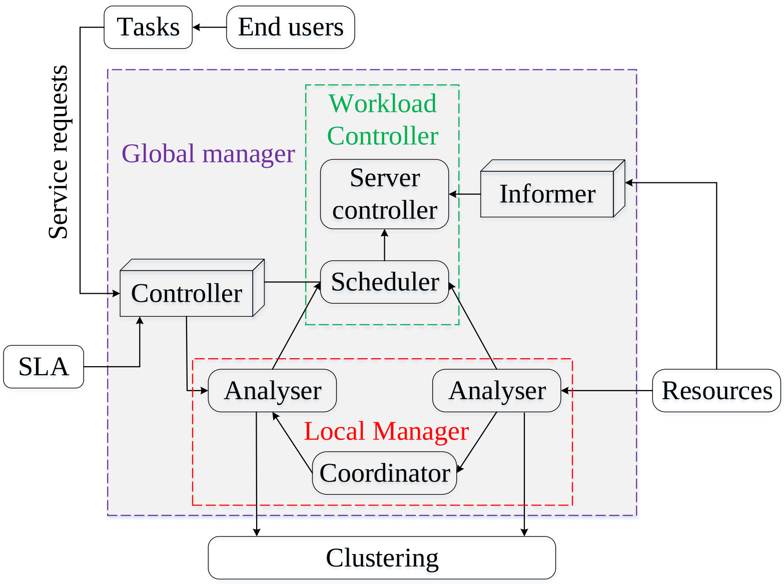

Figure. 1 KSHOA structure [31]

performances. But, these existing works have some drawbacks. Because most of the works only focused on CPU utilization and also they do not assume the minimization of balance between SLA violations and energy consumption to verify the effectiveness of their approaches and to satisfy both user and provider needs. To solve these issues, the proposed model utilizes the optimal resource provisioning mechanism in the cloud based on the QoS parameters to enhance the minimization balance among energy consumption and SLA violations.

3. Proposed methodology

In this paper, the optimal resource provisioning scheme is introduced based on the k-medoid clustering method, and the self-adaptive hybrid optimization procedure is termed as KSHOA to obtain the energy-QoS method. In the initial step of the proposed method, the number of tasks and VMs are grouped through k-medoid method for allocating user needs to proper resources. The FMRS algorithm is utilized in every cluster for assigning tasks to resources. After the task allocation to VMs, the dynamic consolidation of VM is estimated by the SHOA in the second stage of the proposed approach. The VM consolidation follows the four scopes: (i) Overloading host detection, (ii) Underloading host detection, (iii) Selection of VM, and (iv) VM placement.

3.1 KSHOa structure

The steps of KSHOA technique is described and this system contains the workload controller, Global manager, and local manager as exposed in Fig. 1 [31].

A coordinator and two analyzers are the modules for the local manager. Resource information of cloud providers' and users' tasks are received by the analyzer modules. The k-medoid clustering method is executed by the coordinator module. Through the server controller, the workload controller is consolidating the VMs and switch idle servers off. The scheduler is being assigned the collection of tasks to the proper resource clusters with the help of the FMRS approach. In KSHOA method, the workload controller plays a key role to decrease the consumption of power of datacentres. Both workload controllers and local managers are controlled by the global manager. Based on the present datacentre's capacity, the module of the controller considers the user's SLA. The cloud provider information is updated by the module of informer that comprises the utilization of VMs, power status of resources, overheads, and physical resources at run time. It would directly support as an equilibrium of the resource provisioning and the system workload.

3.2 K-medoid algorithm for clustering of tasks and VMs [31, 32]

Using the k-medoid algorithm, the independent tasks and VMs are grouped according to their length and CPU processing power. The k-medoid algorithm[32] is a separating grouping mechanism that is based on medoid computing by reducing the total distance among the designated centroid and the points. In this method, the data points are denoting every cluster. These points are called cluster medoid. Arbitrarily choose the $k$data objects as medoids that

Algorithm 1: Pseudo code for the k-medoid clustering method

Input: Set of input tasks($TaskList = \{T_1, \dots, T_n\}$), set of existing resources ($VMsList = \{R_1, \dots, R_n\}$), $P = inputdata T_i or R_i$ in each cluster, number of clusters ($j = 1,2,\dots,k$), and number of data ($i = 1,2,\dots\dots,n$)

Variety number of resources ($VMsList$) → Find Cluster number (K)
Arbitrarily choose the initial medoids
$k = MedoidsList_{size}$
Find_init_medoids($VMsList$) → List<Medoids$_t$>medoids
Find_init_medoids($TaskList$) → List<Medoids$_v$>medoids

while: NoObjectMoveGroup () **do**
  Compute the distance using Eq. (1)
  Allocate task T_i to Cluster$_t$
  Allocate task R_i to Cluster$_v$
end

Output: Cluster of tasks $\{Cluster_t = C_1, C_2, \dots\dots, C_k\}$ based on their length
Cluster of VMs $\{Cluster_v = C_1, C_2, \dots\dots, C_k\}$ based on their processor power

Table 1. Linguistic variable

	Fuzzy linguistic variables	Fuzzy linguistic states		
Inputs	Memory (Mem)	L	M	H
	Bandwidth (BW)	L	M	H
	Length (Len)	L	M	H
Output	Effective resource scheduling	L	M	H

denotes

$$E = \sum_{c=1}^{k} \sum_{i=1}^{n_c} ||P_{ic} - O_c|| \quad (1)$$

Here, n_c signifies the number of objects in the cluster c, P_{ic} is the non-medoid object i in the cluster c and O_c denotes the medoid values in the cluster c. Pseudocode for k-medoid clustering is shown in algorithm 1 [31].

3.3 Task scheduling based on FMRS

FMRS procedure is employed to allocate every group of tasks to the proper VM groups. The steps for the FMRS [33] approach is fuzzification, defuzzification, fuzzy inference, and fuzzy rule. In this method, the input and output variables are obtained to design the model of fuzzy. Here, memory, bandwidth, and length are considered as the linguistic variables. The effective resource allocation is the output variable 'η'. The linguistic variables are shown in Table 1 and are conveyed as below:

$$In \to \{BW, Mem, Len\} \quad (2)$$

$$BW \to \{L, M, H\}; \quad Mem \to \{L, M, H\}; \quad Len \to \{L, M, H\} \quad (3)$$

Here, 'L' is low, 'M' is medium, and 'H' is high.

After finding the input and output linguistic variables, the trapezoidal fuzzification function is used in the FMRS approach to present an associate observation for every single input linguistic variable. The actual scalar value is changed using the process of fuzzification into a fuzzy value. The trapezoidal function can be given as:

$$f_d : \{BW, Mem, Len\} \to R \quad (4)$$

Here, the fuzzy sets are denoted as R, the trapezoidal fuzzification function is defined by f_d. Using the fuzzy square, the FMRS scheme measures the fuzzy inferences with the resultant trapezoidal fuzzification. The fuzzy inference by the fuzzy square is given as:

$$If In = A then \eta = B \quad (5)$$

Here, 'A' and 'B' are the fuzzy numbers that are denoted by the linguistic variables and effective resource allocation individually.

At last, the defuzzification step is implemented that discovers a distinct number well-matched with the association function that creates the output in the procedure of defuzzification. To convert the fuzzy inferences output values into a fuzzy set, the FMRS approach employs a centroid scheme and is specified below [33]:

$$Fuzzyset(y) = \frac{\mu_A(y) y dx}{\mu_A(y) dx} \quad (6)$$

Here, the equivalent aggregated membership function is denoted as $\mu_A(y)$. Let Y_L, Y_M, and Y_H be the low, medium, and high y value that obtains low, medium, and high task scheduling efficiency η. The pseudo-code for the FMRS algorithm is shown in algorithm 2.

3.4 Dynamic VM consolidation using SHOA

In this section, the SHOA is used for dynamic VM consolidation. This SHOA combines the iPSO and TLBO algorithms. The dynamic VM consolidation is a challenging scheme for decreasing

Algorithm 2: Pseudocode for the FMRS method

Input: Group of tasks, group of VMs, Bandwidth, Memory, length, output resource scheduling 'η', trapezoidal fuzzification function 'f_d', and linguistic variables.
Begin
for cloud users and resources to be assigned
Establish linguistic variables using Eq. (3) for different resources
Obtain 'f_d' for scheduling the multidimensional resources of BW, Mem and Length in cloud using Eq. (4)
if $In = A then \eta = B$
Execute centroid defuzzification for effective resource scheduling using Eq. (6)
Allocate each task clusters to appropriate resources clusters
else
Resource scheduling is not performed
end if
end for
end
Output: Effective resource scheduling

the consumption of energy by energetically altering the number of active machines to match resource demands. By minimizing the consumption of power of datacentres, the cloud providers can improve their resource utilization with the help of VM consolidation. The following problems are considered in the VM consolidation mechanisms:

- *Host overloading detection:* To minimize the utilization of a host, one or more VMs are reassigned to other hosts if the host is overloaded.
- *Host underloading detection:* Here, all the VMs are consolidated to other hosts if the host is underloaded, then the host is switched to the sleep mode.
- *VM selection:* From the overloaded hosts, the most favorable VMs are nominated to be migrated.
- *VM placement:* For the nominated VMs, the most favorable destination host is identified.

The above-mentioned problems are solved by the SHOA approach. The iPSO [34] is a population-based optimization approach that merges the standard PSO scheme and effective idea of the weighted particle to enhance the performances. The weighted gravity center of all particles are called as a weighted particle is available in the swarm and it can be computed. The location of the particle is updated by the weighted particle is given as:

$$if rand_{0i} \le \alpha \rightarrow {}^{t+1}v_i = 0$$
$${}^{t+1}x_i = {}^{t}x_i + \phi_{4i}({}^{t}x^W - {}^{t}x_i) i \ge 1$$
$$\phi_{4i} = C_4 \times rand_{4i} \quad (7)$$

$$if rand_{0i} > \alpha \rightarrow {}^{t+1}v_i = w_i \times {}^{t}v_i + (\phi_{1i} + \phi_{2i} + \phi_{3i})({}^{t}x_j^P - {}^{t}x_i) + \phi_{2i}({}^{t}x^G - {}^{t}x_j^P) + \phi_{3i}({}^{t}x^W - {}^{t}x_j^P)$$
$${}^{t+1}x_i = {}^{t}x_i + {}^{t+1}v_i i \ge 1, j \le M, \phi_{1i} = C_1 \times rand_{1i},$$
$$\phi_{2i} = C_2 \times rand_{2i}, \phi_{3i} = C_3 \times rand_{3i} \quad (8)$$

Here, the current and the next step is defined by the upper right superscripts of t and $t+1$ individually. The updated velocity is denoted as ${}^{t+1}v_i$, the inertia coefficient for the current velocity is denoted as w_i and the current velocity of the i^{th} particle is denoted as ${}^{t}v_i$. The accelerator coefficients are defined as the elements of C1, C2, C3, and C4 that can be deliberated as $-(\phi_{2i} + \phi_{3i})$, 2, 1 and 2 individually. In every iteration, w is arbitrarily designated from a range of [0.5, 0.55] and $\alpha = 0.4$. The arbitrary number is denoted as $rand_{ki} k \in \{0,1,2,3,4\}$ which is chosen from the range of [0, 1] and the arbitrarily designated particle from current $Pbest$ is defined as ${}^{t}x_j^P$. The current location of the particle is represented as ${}^{t}x_i$ and the global best particle is represented as ${}^{t}x^G$. The current and updated location of the i^{th} particle is represented as ${}^{t}x_i$ and ${}^{t+1}x_i$. The weighted particle for the present step is denoted by ${}^{t}x^W$. At the end of the maximum iteration, the optimal solution is determined. The procedure is repeated if the optimal is not found. The learner stage of the TLBO [35] algorithm is employed to improve the searching capability of swarms. The two stages are included in the proposed method. Using iPSO, the position and velocity are updated in the first stage. The TLBO is employed to determine the global solutions with high consistency and low computational effort in the second stage. The flowchart of hybrid algorithm is shown in Fig. 2.

3.4.1. Objective function formulation

The set of processing components is included in the CC platform that is mounted on various PMs which can be shared by the numerous VMs. In our work, multi-objective VM consolidation is utilized. The consumption of energy, the communication cost of VMs, and the resource wastage are minimized by expressing the consolidation of VMs' issue as a multi-objective optimization issue. The goal of multi-objective consolidation of VM is to create balances

Received: January 11, 2022. Revised: February 25, 2022.

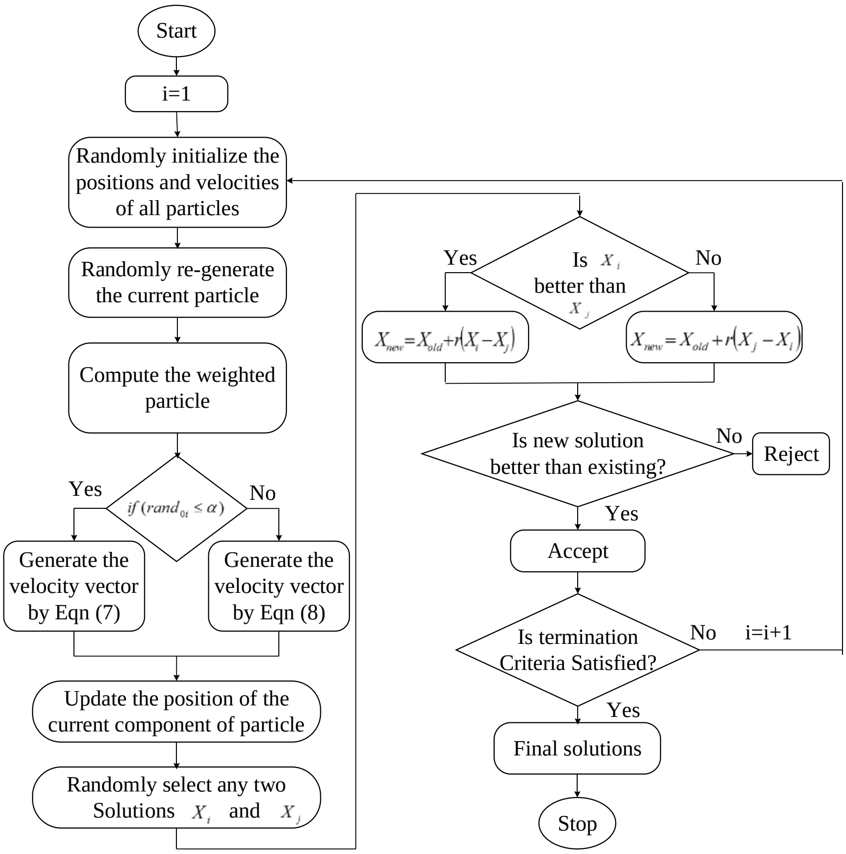

Figure. 2 SHOA flowchart

among these elements. As related to a single objective mechanism, the multi-objective formulation can accomplish improved global performance.

❖ ***Energy consumption***

In the cloud datacentres, the PMs are the largest energy consumers. The active PM consumes energy that has a better direct association with its utilization of the CPU. The PMs are switched off in the idle state, the utilization of energye_j of PM can be given as [36]:

$$e_j = \begin{cases} (e_j^{full} - e_j^{idle}).U_j^p + e_j^{idle}, U_j^p > 0 \\ 0, otherwise \end{cases} \quad (9)$$

Here, the consumed energy by PM$j$is denoted as e_j^{full}if it is entirely burdened. In the idle state, the consumed energy by PM$j$is defined ase_j^{idle} and the normalized utilization of current CPU is denoted asU_j^p. In the idle state, $U_j^p = 0$ and $U_j^p = 100\%$ for the fully loaded. Here, the consumption of energy for an idle and entirely burdened PM are set to 120 w and 185 w individually. The initial purpose is to reduce the consumed energy by each PMs and it can be given as:

$$Energyconsumption = \sum_{j=1}^{n} e_j \quad (10)$$

Here, the amount of each PMs is represented by$n$.

Communication cost of VMs (ComCost)

The Second objective aims to decrease the communication cost of VMs. In data center, a joint communication among VMs reasons considerable inner flow of traffic and placing VM which interconnect with every single closed will try to decrease the traffic and response time of applications. The communication cost of V_j and V_k can be computed [37]. The amount of communication cost among all VMs pairs are termed as the communication cost and it can be given as:

$$ComCost(X) = \sum_{j=1}^{M-1} \sum_{k=j+1}^{M} cos\,t\,(j,k) \quad (11)$$

❖ ***Resource wastage***

The resource wastage of PM$j$can be computed [36]. Here, the standardized remaining resources of CPU and memory are denoted as L_j^p andL_j^m, which are fractions of the equivalent remaining resources to the entire resources. The standardized utilization of CPU and memory are represented as U_j^p and U_j^m, which are the proportion of the assigned resources to the aggregate resources. $\eta$is a constant value that is set to be 0.0001. The third purpose is to diminish the wastage of resources for all PMs. It can be designed as:

$$Re\,s\,ourcewastage = \sum_{j=1}^{n} \frac{\left|L_j^p - L_j^m\right|}{U_j^p + U_j^m + \eta} \quad (12)$$

3.4.2. Identification of overload and underload detection

In this part, the problem of overload and under load host detection is solved by the SHOA. The subject of this difficulty is to govern whether a host is overloaded or underloaded and when to transfer VMs from the host. According to the utilization of the CPU, the host is determined whether it is overloaded or underloaded. Here, the fitness function of SHOA and CPU utilization are compared for the identification of overload and underload host detection. If the CPU utilization is exceeding the fitness value, the host is overloaded or else it is underloaded.

2.4.3. VM selection

The selection of VM process is take place if the overloaded host is identified. Here, the VM is select to offload the host to eliminate the performance degradation. After the VM selection, the host is again tested. If it is still deliberated as being overloaded, the selection of VM is utilized to choose a new VM to transfer from the host. This procedure is repetitive until the host is deliberated as being not overloaded. In this part, based on the cost function the target VM is nominated in the consolidation of VM. The minimum cost value of VM is nominated for the selection of VM. The cost function is already described in the objective function.

3.4.4. VM placement

Consider the amount of all VMs are denoted as$m$, the total number of all PMs are defined as$n$, the set of PMs are denoted as P, and the set of VMs are represented as$V$. In the problem of VM employment, we undertake that any single PM can be fulfilled the highest VM request. For all PM $j \in P$, the standardized thresholds of memory and CPU utilization are represented by $T_{mj}$and T_{pj}. For each PMs, $T_{pj} = T_{mj} = 90\%$. The standardized memory and CPU requests of each VM$i$can be represented byR_{mi} andR_{pi}. Moreover, the decision variable is represented as $x_{i,j} \in [0,1]$ which denotes whether VMi has been allocated to PMj.

In the proposed method, the different objectives are considered for the dynamic VM consolidation. The objectives are the number of VM migration, consumption of energy, and SLA violation. By minimizing the fitness function, the objectives are accomplished through the VM consolidation process. The fitness function can be defined as:

$$fitness = min \begin{cases} \sum_{j=1}^{n} e_j \\ ComCost(X) \\ \sum_{j=1}^{n} w_j \end{cases} \quad (13)$$

Constraints:

$$\sum_{i=1}^{n} x_{i,j} = 1\,, \forall i \in V \quad (14)$$

$$\sum_{i=1}^{m} R_{pi} x_{i,j} \le T_{pj}, \forall j \in P \quad (15)$$

$$\sum_{i=1}^{m} R_{mi} x_{i,j} \le T_{mj}, \forall j \in P \quad (16)$$

$$x_{i,j} \in [0,1], \forall i \in V, j \in P \quad (17)$$

Every VM is positioned at one time is defined by Eq. (14). The capacity of the CPU is defined by Eq. (15) and the capacity of memory is signified by Eq. (16). The decision space is denoted by Eq. (17). The proposed work simultaneously reduces energy consumption and SLA violation.

4. Simulation results & discussions

The developed (KSHOA) technique is executed in the CloudSim simulation tool. For the execution of RM and consolidation VM approaches, the CloudSim is a distinct event tool for various applications. The different performance metrics are assessed depending on the QoS factors. Various performance factors are employed to estimate the performance of consumption of energy, SLA violation, number of VM migration, ESV (energy and SLA violations), wastage of resources, and cost of communication. The performance of the developed method is related to the existing approaches [38] of LR-MMT, IQR-MMT, LR-MC, IQR-MC, IQR-RS, LR-RS, MAD-MMT, MAD-MC, MAD-RS, RE-VMC, MOABC-VMC in terms of energy consumption, ESV, SLA violation, and VM migration. Moreover, the performance of the resource wastage is compared to the existing methods [38] of LR-MMT, RE-VMC, and MOABC-VMC. With the help of the logistic

Received: January 11, 2022. Revised: February 25, 2022.

Table 2. Simulation parameters

Parameters	Values
Datacentre	RAM: 10 GB Storage: 1 TB MIPS: 1000,2000,4000 BW:100 GB OS: Windows System Architecture:x86 Hypervisor: Oracle
Host	Storage:1TB RAM:10 GB BW:100 GB
VMs	RAM:128 MB MIPS: 250,500,750,1000 BW:2500 MB Storage: 1 GB Hypervisor: Oracle
Cloudlet	Storage: 1 GB Length:10000

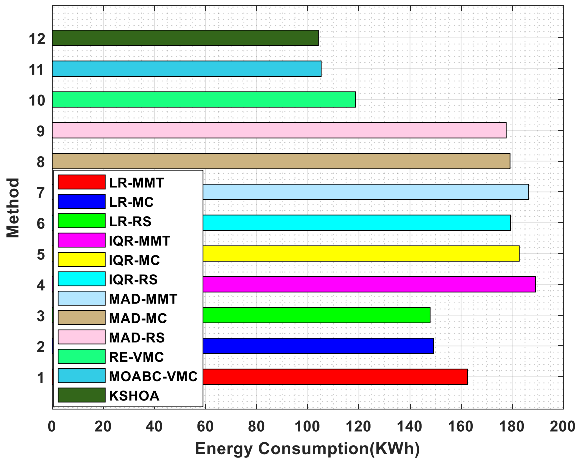

Fig. 3 Energy consumption

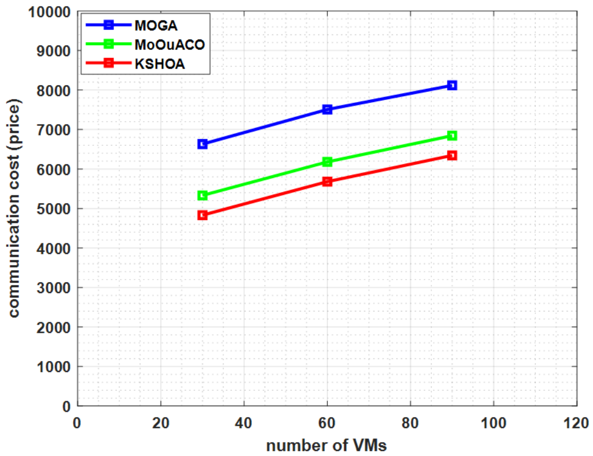

Fig. 4 Communication cost

regression, the utilization of CPU has been predicted in LR approach. By estimating the interquartile range of CPU utilization and the median absolute deviation, the workload stability has been predicted by the IQR and MAD. MMT desired the VMs with the least migration time to migrate because it is considered as a VM assortment policy. According to a homogenously distributed discrete arbitrary variable, VMs has been chosen by the RS (random policy). MC policy chooses those VMs that have the maximum correlation of the CPU utilization with other VMs. These are some conventional approaches. The communication cost of the developed approach is equated to the existing approaches [39] of MOGA and MoOuACO. The detailed configuration parameters for the proposed approach are presented in Table 2.

The physical resources utilized the energy is termed as consumption of energy for performing the tasks in kW. The amount of computational resources which does not employ by any VM is known as wastage of resources. The VMs are nominated for placement if the overloaded or underloaded hosts are determined for consolidation of VM. By decreasing the amount of VM migrations, the time for VM migration can be reduced in the placement procedure. The SLA violation metric provides the SLA violation rate. Consumption of energy and SLA violation are destructively interconnected. Therefore, by improving the level of SLA violations, the energy can be reduced. RM aims to lessen both SLA and energy. Therefore, the ESV (Energy and SLA violation) metric combines both SLA and consumption of energy.

The Fig. 3 is demonstrating the consumption of energy performance for the proposed (KSHOA) and existing algorithms. The KSHOA approach devours the minimum number of energy (104.12 kWh) and decreases the utilization of energy as compared to other algorithms. The IQR-MMT approach consume more energy (189.13 kWh) so it is a worst technique and other algorithm are consume more energy than our proposed model. From the graph, the proposed method achieved the minimum consumption of energy compared to other techniques.

The Fig. 4 displays the communication cost evaluation for the different algorithms of MOGA and MoOuACO. The proposed method utilized the minimum cost (6344 for 90 VMs) compared to other existing methods. MOGA has obtained a high communication cost (8000 for 90 VMs) so it is the worst method. The communication cost is improved with the number of VM count is increased. The VM location that assumes the communication between the VMs will decrease system traffic, decrease the data transmission times and finish the jobs quicker. From the analysis, the proposed scheme obtained the lowest cost when compared to other existing techniques.

Received: January 11, 2022. Revised: February 25, 2022.

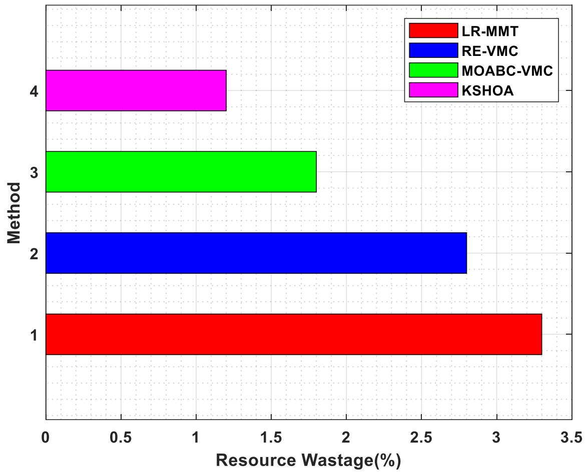

Figure. 5 Resource wastage

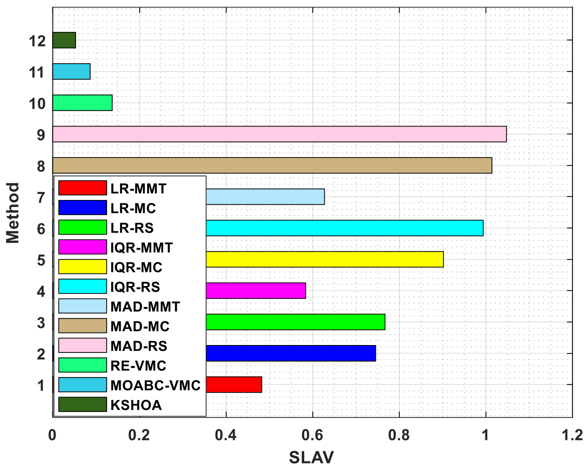

Figure. 6 SLA violation

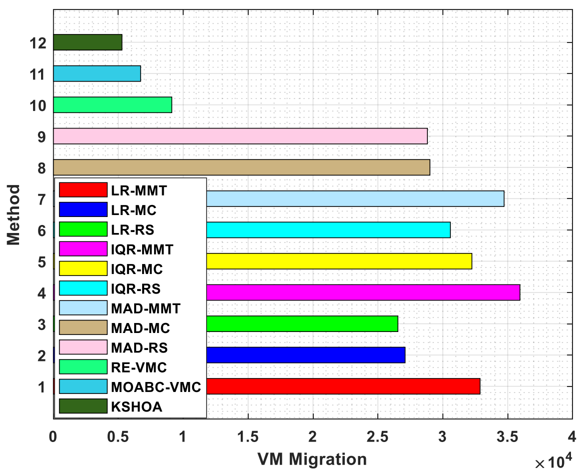

Figure. 7 Number of VM migration

The Fig. 5 displays the resource wastage performance for the proposed and existing techniques. The minimum resource wastage (1.2%) is obtained by the proposed (KSHOA) algorithm when compared to other existing methods of LR-MMT, RE-VMC, and MOABC-VMC. The results denoted that the resources of the running PMs such as memory and CPU are correctly employed. Accordingly, more VMs can be lodged in a minimum amount of PMs. The LR-MMT approach has obtained high resource wastage (3.3%) hence it is worst approach. Therefore, the proposed scheme given the best performance rate compared to other methods.

The Fig. 6 shows the SLA violation for different algorithms. The proposed (KSHOA) method obtained the lowest violation rate than other algorithms. SLA violation is the most significant metric because it shows the algorithm's capability to assurance the QoS. The results show that the KSHOA and MAD-RS approaches have the lowest (0.0527) and the highest value (1.0473) of SLA violation, individually. From the graph, the developed technique accomplished a good performance related to other techniques. Hence, the KSHOA is more effective in assuring QoS.

The Fig. 7 displays the amount of VM migration rate performances for the proposed and existing algorithms. VM migration is the most significant metric in VM consolidation. The developed model obtained the minimum migration rate (5282) than other existing approaches. The energy consumption is increased by the additional migrations, so the amount of migrations interrupts the QoS and energy costs. KSHOA contains the lowest amount of VM migrations than other approaches. Generally, the process of dependable PMs Selection as the relocation destinations and conditionally selecting migrated VMs based on multiple criteria resulted in a considerable decrease in the number of migrations. Such a reduction in the number of migrations will enhance the QoS. It can be observed that the amount of migrations prompted by KSHOA is minimum than other algorithms. This can be stated that the developed model concurrently assumes various factors in VM migration. From the analysis, the proposed method obtained improved performance related to other procedures.

The Fig. 8 displays the ESV (energy and SLA violations) performance for the proposed and existing algorithms. The proposed method obtained minimum ESV (7.286%) compared to other existing algorithms. The multiplication of the energy and SLA violation gives the ESV performance. ESV is a comprehensive factor in evaluating QoS, energy utilization, and the number of migrations. The minimum ESV gives better algorithm performance in energy saving and assuring QoS. The worst ESV (186.06%) has obtained in MAD-RS approach. Hence, it can be stated that the proposed (KSHOA) method obtained

Received: January 11, 2022. Revised: February 25, 2022.

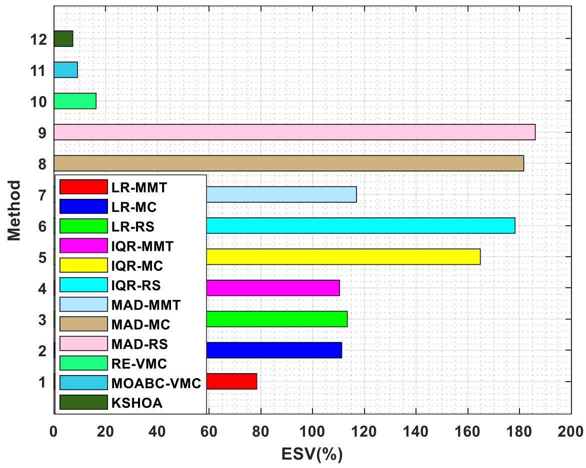

Figure. 8 ESV performance

a low ESV rate when compared to other methods. Therefore, the proposed scheme is the best method compared to other methods.

In the above figures, the performance of the developed structure is improved related to other methods. The developed RM system reduced both consumptions of energy and SLA violation with multi-objective functions. Moreover, by minimizing the fitness function, the proposed method achieved the objectives of our work. As compared to other techniques, the proposed method accomplished better performances in terms of consumption of energy (104.12kWh), communication cost (6344 for 90VMs), VM migrations (5282), SLA violations (0.0527), and wastage of resources (1.2%). Therefore, the proposed scheme is the best technique for the RM-based QoS in the CC field.

5. Conclusions

In this paper, an optimal resource provisioning scheme using QoS in cloud computing based upon dynamic clustering and SHOA has been proposed. In the proposed (KSHOA) technique, the SHOA algorithm is used which combines the iPSO and TLBO algorithms for enhancing dynamic consolidation of VM to obtain the least consumption of energy. The K-medoid method is used to make the group of tasks and VMs, then the FMRS approach is employed to assign the collection of tasks to the collection of VMs. The KSHOA aims to balance the trade-off among the consumption of energy of hosting servers and SLAs in the CC environment. The performances are estimated based on the QoS metrics. The simulation outcomes are proving that the developed (KSHOA) approach provides better performances in terms of energy consumption, SLA violations, number of VM migrations, resource wastage, and cost of communication when equated to the current approaches. In the future, deep learning methods might be applied for more complex VM consolidation processes to enhance the beauty of the concepts. Hence, the proposed framework of Modelling and Simulation measures the Quality of Service (QoS) and performance in Data-Centre along-with the resource utilization policy

Conflicts of Interest

The author declares no conflict of interest as all the references has been cited with regards to all the concerned other authors, respectively.

Author Contributions

Conceptualization, Methodology, Software, Analysis, Resources & Investigations: Jaspal Singh. An Optimal Resource Provisioning Scheme Using QoS in

References

[1] A. Abdelaziz, M. Elhoseny, A. S. Salama, and A. M. Riad, "A machine learning model for improving healthcare services on cloud computing environment", *Measurement. 119*, pp. 117–128, 2018. https://doi.org/10.1016/j.measurement.2018.01.022.

[2] N. Garg, D. Singh, and M. S. Goraya, "VM Selection and Allocation Policy to Optimize VM Migration in Cloud Environment", *Int. J. Recent Technol. Eng. 8*, pp. 3444–3449, 2019. https://doi.org/10.35940/ijrte.B2700.078219.

[3] P. Li, J. Li, Z. Huang, C. Z. Gao, W. B. Chen, and K. Chen, "Privacy-preserving outsourced classification in cloud computing", *Cluster Comput.*, Vol. 21, pp. 277–286, 2018. https://doi.org/10.1007/s10586-017-0849-9.

[4] M. Rath, "Resource provision and QoS support with added security for client side applications in cloud computing", *Int. J. Inf. Technol. 11*, pp. 357–364, 2019. https://doi.org/10.1007/s41870-017-0059-y.

[5] L. S. Revadi, X. Zheng, and Y. Jiang, "Investigating Key User Experiencing Engineering Aspects in Software-as-a-Service Service Delivery Model", *Human-Computer Interact.*, (n.d.). https://doi.org/arXiv:1812.03660v1.

[6] D. R. B, A. Cretu, J. Yi, C. Avram, and A. Adina, "Advances in Internet, Data & Web Technologies", *Lect. Notes Data Eng. Commun. Technol.*, Vol. 17, pp. 675–686, 2018. https://doi.org/10.1007/978-3-319-75928-9.

[7] O. Krancher, P. Luther, and M. Jost, "Key

Affordances of Platform-as-a-Service: Self-Organization and Continuous Feedback", *J. Manag. Inf. Syst.*, Vol. 35, pp. 776–812, 2018. https://doi.org/10.1080/07421222.2018.1481636.

[8] N. Garg and M. S. Goraya, "Energy Aware Hardware and Software Approaches in Cloud Environment", Vol. 7, pp. 66–69, 2017.

[9] N. Garg and M. S. Goraya, "A Survey on Energy-Aware Scheduling Techniques in Cloud Computing Environment", *Int. J. Comput. Sci. Inf. Secur.*, Vol. 14, pp. 523–528, 2016.

[10] J. Shen, T. Zhou, X. Chen, J. Li, and W. Susilo, "Anonymous and Traceable Group Data Sharing in Cloud Computing", *IEEE Trans. Inf. Forensics Secur.*, Vol. 13, pp. 912–925, 2018. https://doi.org/10.1109/TIFS.2017.2774439.

[11] S. Mustafa, B. Nazir, A. Hayat, A. U. R. Khan, and S. A. Madani, "Resource management in cloud computing: Taxonomy, prospects, and challenges", *Comput. Electr. Eng.*, Vol. 47, pp. 186–203, 2015. https://doi.org/10.1016/j.compeleceng.2015.07.021.

[12] H. K. M. Shehab, A. Ragaa, and M. Taher, "Resource Management Challenges in the Next Generation Cloud based Systems: A Survery and Research Directitons", In: *Proc. of 2018 13th Int. Conf. Comput. Eng. Syst.*, pp. 139–144, 2018. https://ieeexplore.ieee.org/document/8639220.

[13] S. M. Parikh, N. M. Patel, and H. B. Prajapati, "Resource Management in Cloud Computing: Classification and Taxonomy", 2017. http://arxiv.org/abs/1703.00374.

[14] 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", *IEEE Trans. Serv. Comput.*, Vol. 11, pp. 78–89, 2018. https://doi.org/10.1109/TSC.2016.2528246.

[15] Y. Sharma, W. Si, D. Sun, and B. Javadi, "Failure-aware energy-efficient VM consolidation in cloud computing systems", *Futur. Gener. Comput. Syst.*, Vol. 94, pp. 620–633, 2019. https://doi.org/10.1016/j.future.2018.11.052.

[16] J. Du, L. Zhao, J. Feng, and X. Chu, "Computation Offloading and Resource Allocation in Mixed Fog/Cloud Computing Systems With Min-Max Fairness Guarantee", *IEEE Trans. Commun.*, Vol. 66, pp. 1594–1608, 2018. https://doi.org/10.1109/TCOMM.2017.2787700.

[17] S. Zahoor, S. Javaid, N. Javaid, M. Ashraf, F. Ishmanov, and M. Afzal, "Cloud–Fog–Based Smart Grid Model for Efficient Resource Management", *Sustainability.*, Vol. 10, p. 2079, 2018. https://doi.org/10.3390/su10062079.

[18] Z. Xiong, S. Feng, W. Wang, D. Niyato, P. Wang, and Z. Han, "Cloud/Fog Computing Resource Management and Pricing for Blockchain Networks", *IEEE Internet Things J.*, Vol. 6, pp. 4585–4600, 2019. https://doi.org/10.1109/JIOT.2018.2871706.

[19] E. Ataie, R. E. Maleki, S. E. Etesami, B. Egger, D. Ardagna, and A. Movaghar, "Power-aware performance analysis of self-adaptive resource management in IaaS clouds", *Futur. Gener. Comput. Syst.*, Vol. 86, pp. 134–144, 2018. https://doi.org/10.1016/j.future.2018.02.042.

[20] K. Gai, M. Qiu, H. Zhao, and X. Sun, "Resource Management in Sustainable Cyber-Physical Systems Using Heterogeneous Cloud Computing", *IEEE Trans. Sustain. Comput.*, Vol. 3, pp. 60–72, 2018. https://doi.org/10.1109/TSUSC.2017.2723954.

[21] S. K. Sood and K. D. Singh, "SNA Based Resource Optimization in Optical Network using Fog and Cloud Computing", *Opt. Switch. Netw.*, Vol. 33, pp. 114–121, 2019. https://doi.org/10.1016/j.osn.2017.12.007.

[22] M. Abbasi, M. Yaghoobikia, M. Rafiee, A. Jolfaei, and M. R. Khosravi, "Efficient resource management and workload allocation in fog–cloud computing paradigm in IoT using learning classifier systems", *Comput. Commun.*, Vol. 153, pp. 217–228, 2020. https://doi.org/10.1016/j.comcom.2020.02.017.

[23] X. Zhang, M. Jia, X. Gu, and Q. Guo, "An Energy Efficient Resource Allocation Scheme Based on Cloud-Computing in H-CRAN", *IEEE Internet Things J.*, Vol. 6, pp. 4968–4976, 2019 https://doi.org/10.1109/JIOT.2019.2894000.

[24] T. Kim and J. M. Chang, "Profitable and Energy-Efficient Resource Optimization for Heterogeneous Cloud-Based Radio Access Networks", *IEEE Access*, Vol. 7, pp. 34719–34737, 2019. https://doi.org/10.1109/ACCESS.2019.2904766.

[25] B. K. Dewangan, A. Agarwal, M. Venkatadri, and A. Pasricha, "Self-characteristics based energy-efficient resource scheduling for cloud", *Procedia Comput. Sci.*, Vol. 152, pp. 204–211, 2019. https://doi.org/10.1016/j.procs.2019.05.044.

[26] S. S, B. S. P, "Service Level Agreement Based Catalogue Management and Resource Provisioning in Cloud for Optimal Resource Utilization", *Mob. Networks Appl.*, Vol. 24, pp. 1853–1861, 2019.

https://doi.org/10.1007/s11036-019-01382-9.

[27] T. Wang, Y. Liang, W. Jia, M. Arif, A. Liu, and M. Xie, "Coupling resource management based on fog computing in smart city systems", *J. Netw. Comput. Appl.*, Vol. 135, pp. 11–19, 2019. https://doi.org/10.1016/j.jnca.2019.02.021.

[28] H. Rafique, M. A. Shah, S. U. Islam, T. Maqsood, S. Khan, and C. Maple, "A Novel Bio-Inspired Hybrid Algorithm (NBIHA) for Efficient Resource Management in Fog Computing", *IEEE Access*, Vol. 7, pp. 115760–115773, 2019. https://doi.org/10.1109/ACCESS.2019.2924958.

[29] K. Cui, W. Sun, and W. Sun, "Joint Computation Offloading and Resource Management for USVs Cluster of Fog-Cloud Computing Architecture", In: *Proc. of 2019 IEEE Int. Conf. Smart Internet Things, IEEE*, pp. 92–99, 2019. https://doi.org/10.1109/SmartIoT.2019.00023.

[30] M. Katsaragakis, D. Masouros, V. Tsoutsouras, F. Samie, L. Bauer, J. Henkel, and D. Soudris, "DMRM: Distributed Market-Based Resource Management of Edge Computing Systems", In: *Proc. of 2019 Des. Autom. Test Eur. Conf. Exhib., IEEE*, pp. 1391–1396, 2019. https://doi.org/10.23919/DATE.2019.8714930.

[31] M. A. Haghighi, M. Maeen, and M. Haghparast, "An Energy-Efficient Dynamic Resource Management Approach Based on Clustering and Meta-Heuristic Algorithms in Cloud Computing IaaS Platforms", *Wirel. Pers. Commun.*, Vol. 104, pp. 1367–1391, 2019. https://doi.org/10.1007/s11277-018-6089-3.

[32] S. Nirmal, "Comparative Study between K-Means and K-Medoids Clustering Algorithms", *Int. Res. J. Eng. Technol.*, Vol. 839, pp. 839–844, 2008. https://www.irjet.net/archives/V6/i3/IRJET-V6I3154.pdf.

[33] V. Priya, C. S. Kumar, and R. Kannan, "Resource scheduling algorithm with load balancing for cloud service provisioning", *Appl. Soft Comput. J.*, Vol. 76, pp. 416–424, 2019. https://doi.org/10.1016/j.asoc.2018.12.021.

[34] A. Mortazavi and V. Toğan, "Simultaneous size, shape, and topology optimization of truss structures using integrated particle swarm optimizer", *Struct. Multidiscip. Optim.*, Vol. 54, pp. 715–736, 2016. https://doi.org/10.1007/s00158-016-1449-7.

[35] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems", *Comput. Des.*, Vol. 43, pp. 303–315, 2011. https://doi.org/10.1016/j.cad.2010.12.015.

[36] Y. Qin, H. Wang, S. Yi, X. Li, and L. Zhai, "Virtual machine placement based on multi-objective reinforcement learning", *Appl. Intell.*, Vol. 50, pp. 2370–2383, 2020. https://doi.org/10.1007/s10489-020-01633-3.

[37] W. Qiu, Z. Qian, and S. Lu, "Multi-Objective Virtual Machine Consolidation", In: *Proc. of 2017 IEEE 10th Int. Conf. Cloud Comput.*, pp. 270–277, 2017. https://doi.org/10.1109/CLOUD.2017.42.

[38] M. H. Sayadnavard, A. T. Haghighat, and A. M. Rahmani, "A multi-objective approach for energy-efficient and reliable dynamic VM consolidation in cloud data centers", *Engineering Science and Technology, an International Journal*, 2021.

[39] Karmakar, Kamalesh, R. K. Das, and S. Khatua. "An ACO-based multi-objective optimization for cooperating VM placement in cloud data center", *The Journal of Supercomputing*, pp. 1-29, 2021.

[40] J. Singh and M. S. Goraya, "Multi-Objective Hybrid Optimization based Dynamic Resource Management Scheme for Cloud Computing Environments", In: *Proc. of 2019 Int. Conf. Smart Syst. Inven. Technol., IEEE*, pp. 386–391, 2019. https://doi.org/10.1109/ICSSIT46314.2019.8987760.