Arabian Journal for Science and Engineering (2019) 44:9455–9469
https://doi.org/10.1007/s13369-019-04048-6

RESEARCH ARTICLE - COMPUTER ENGINEERING AND COMPUTER SCIENCE

Multi-criteria-Based Energy-Efficient Framework for VM Placement in Cloud Data Centers

Nagma Khattar[1] · Jaiteg Singh[1] · Jagpreet Sidhu[2]

Received: 7 June 2019 / Accepted: 22 July 2019 / Published online: 30 July 2019

Abstract
Energy efficiency has become a prime concern in this evolving era of cloud computing. In recent years, several VM consolidation techniques were proposed. However, they involve single-criterion-based host selection methods, which are unsuitable for the dynamic cloud environment. Cloud data centers need an efficient utilization of physical host resources. The selected hosts for VMs deployment must ensure the least energy consumption and service -level agreement (SLA) violations simultaneously. There is a lack of multi-criteria-based solution which considers multiple resources to select hosts for the deployment of VMs. This paper presents a VM deployment framework that considers multiple criteria, i.e., reducing energy consumptions and SLA violations simultaneously while determining target hosts for VMs placement. The article proposes an Improved Technique for Order of Preference by Similarity to Ideal Solution-based Host Selection Policy for VM deployment on the target hosts in terms of energy efficiency. It manages the trade-off between energy consumption and SLA compliance. A case study is demonstrated to prove the usefulness and appropriateness of the proposed framework. The results show that our framework is promising and provides more efficiency in the use of cloud resources and maintaining SLA.

Keywords Cloud computing · Energy efficiency · SLA · Improved TOPSIS · Host selection

1 Introduction

Cloud computing has revolutionized the information technology (IT) industry by providing on-demand computing services to users. Virtualization made it possible for different users to use infrastructural, software and platform-related services rather than to own them individually [1]. Although it has overwhelmed the users by providing high profits, the harmful effects it is causing are putting a burden on society [2]. Cloud data centers consume tremendous power and release toxic gases to the atmosphere. The report by Smart 2020 states that the emission by the IT industry will increase to 180% from 2002 to 2020 [3]. According to Greenpeace, the demand for electricity will increase by 60% or more in 2020 [4]. The World Wide Fund for Nature (WWF) community suggests increasing the expenditure on research and development in achieving energy efficiency twice to avoid catastrophic climate failure [5]. Therefore, the focus of researchers should not be only on performance optimization, but also on energy efficiency in cloud computing. Researchers have addressed the issue of achieving energy efficiency by proposing infrastructure-based solutions, hardware-based solutions and software-based solutions. Software-based solutions are considered more promising in contrast to hardware and infrastructure [6]. These include consolidation techniques [7–13]. VMs from overutilized and underutilized servers are migrated to optimal servers by scheduling mechanisms to optimize energy efficiency [14, 15]. Figure 1 represents the VM consolidation process.

VM consolidation involves host under-load detection, overload detection, VM selection and VM placement algorithms [16–18]. Resources can be consolidated onto lesser machines by migrating VMs from overutilized machines and halting underutilized machines. The selection of effective hosts for deploying the VMs from under-loaded and overloaded hosts plays a significant role in VM consolidation. The ultimate goal of VM consolidation is to select a target

✉ Jagpreet Sidhu
jagpreet.sidhu@juit.ac.in

1 Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

2 Department of Computer Science and Information Technology, Jaypee University of Information Technology, Waknaghat, Himachal Pradesh, India

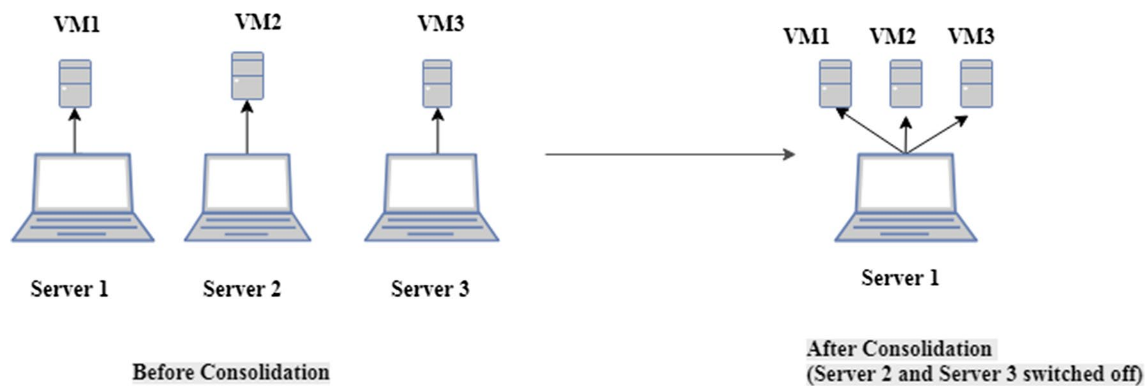


Fig. 1 VM consolidation process

host for placing VMs to switch off underutilized hosts for efficient resource utilization.

Thus, VM consolidation involves migration which is the shifting of VMs from one host to another for optimal resource utilization [19] by switching off idle machines.

VMs are to migrate for optimum resource utilization, energy efficiency, load balancing, maintenances and failure of a server and to remove hotspots. However, in some cases migrations degrade the system performance, so the number of migrations should be minimized as possible. VM migrations demand special requirements in order to maintain SLA. For example, in the case of video streaming, the interconnection network may create a bottleneck. Thus, to maintain performance a feedback mechanism should be incorporated between a scheduler and switches to decide on placement. As the workload in case of the cloud is dynamic, so VM migration which is a part of VM placement must consider the type of workloads to maintain performance.

The existing literature shows that the problem of VM placement is a multi-dimensional bin packing problem [20, 21]. However, the authors in [22] report that it is similar to 3D bin packing but not precisely the same. The VM placement problem can actually be referred to as vector bin packing problem or an NP hard problem. The work found in the literature on VM placement has various anomalies and drawbacks. For example, the heuristics such as First Fit checks the machines sequentially and assigns VM to the first suitable machine. However, this creates a major unbalancing of the load. The other 1-D best-fit heuristics also create load unbalancing as hosts may use all the CPU capacity but storage bandwidth can be underutilized. The literature also lacks techniques which ensure the elasticity of resources during VM placement. Thus, multi-dimensional optimization is a solution which tries to use parameters considering all these dimensions unlike single-criterion-based approaches.

In single-criterion-based strategy, the mono-objective function is used which focuses on a single-parameter minimization or maximization. Whereas, in the case of multi-criteria-based strategies, the multi-objective function is used which considers multiple parameters resulting in the fulfillment of multiple goals. For example, considering increase in power consumption, multiple resources capacities, resource availability, the delay caused due to migration (network) while placing a VM makes the proposed scheme in this paper a multi-criteria decision-making (MCDM) solution. The formulation of VM placement solution considering multiple criteria is the only feasible strategy in the real scenario because of dynamic cloud workloads. The criteria for VM placement in the cloud data centers can be: (1) minimizing the energy consumption, (2) cost optimization, (3) network traffic minimization, (4) balancing resource utilization and (5) ensuring high quality of service (QoS). The multi-criteria-based approaches consider multiple parameters which try to optimize two or more criteria.

In large-scale cloud data centers, selection of an appropriate target hosts for VM placement in a very challenging job. The overall system performance can be improved with efficient placement. The data transfer time can be too long in large data centers [23]. Moreover, large data centers incur high costs in terms of energy consumption. The problem of resource wastage can be to a greater extent as different VMs can be launched on a large number of hosts which may impact VM placement. Similarly, excessive VM migrations incur high overhead costs and performance degradation. Thus, in large-scale data centers, multi-objective functions which involve different parameters related to maintaining QoS, energy efficiency, efficient resource usage and reducing network delays should be incorporated into the placement strategy. In addition to this, it should also deal with scalability issues by minimizing transmission traffic considering constraints associated with servers, dependencies among VMs and applications. Scalability is one of the most attractive features provided by the cloud environment. Autoscaling and elasticity ensure efficient use of resources and help to maintain QoS [24]. VM placement must be carried out in such a way which provides resource scalability.

The optimization approach must address the issue of reducing the data transfer rate between applications to ensure network scalability [25].

Most of the solutions in the literature for selecting hosts include bin packing heuristics which are based on a single criterion. They only consider energy consumption and not energy efficiency (reducing energy consumption and SLA violations simultaneously).

In this paper, an effort is made to design a multi-criteria (energy consumption and SLA violations reduction)-based VM deployment framework for finding an effective host for VM placement considering CPU, RAM and network-related parameters. It manages the trade-off between energy consumption and SLA compliance. A case study is demonstrated to prove the usefulness and appropriateness of the proposed framework. The results show that our framework is promising and provides more efficiency in the use of cloud resources and maintaining SLA.

Significant contributions are as follows:

- It presents a design of a VM deployment framework considering CPU, RAM and network-related parameters to determine the target hosts for VM deployment while reducing the energy consumption and SLA violations simultaneously (multi-criteria).
- Improved Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS)-based Host Selection Policy (ITHSP) is proposed to find energy-efficient hosts considering various scheduling parameters.
- It categorizes hosts into five classes ranging from extremely energy efficient to minimal energy efficient.
- A case study-based approach evaluates the efficiency of the proposed algorithm by ranking the hosts.

The paper is organized as follows: Sect. 2 defines the VM consolidation literature for cloud computing. Section 3 discusses the design of the proposed VM deployment framework. In Sect. 4, the results are illustrated, analyzed and discussed. The conclusion is presented in Sect. 5.

2 Related Literature

VM consolidation is an efficient technique for achieving energy efficiency in cloud computing. There have been numerous host selection algorithms for VM deployment. This section discusses frameworks that derive energy efficiency based on these algorithms.

Nathuji and Schwan [17] designed a VM placement framework for energy-efficient management of resources by using both global and local policies. The drawback of work is that the framework considered a single criterion, and resources were not automatically managed at a global level.

Tarighi et al. [18] proposed a framework for effective resource utilization by migrating VMs between cluster nodes using Fuzzy TOPSIS. VMs were migrated from underutilized hosts to overutilized hosts. However, the VM deployment algorithm was not disclosed.

Beloglazov and Buyya [21] designed an energy-efficient system for resource management of cloud data centers. Authors used bin packing heuristics for VM deployment and did not consider multiple attributes. Algorithms were not appropriately defined, and the system was not generic. It was tested on a random dataset.

Sharifi et al. [26] proposed a scheduling algorithm for mapping of VMs to hosts such that power consumption is minimum. Authors designed power and migration models and presented the consolidation fitness metric. Although authors considered multiple resources like CPU and disk utilization, the parameters were given equal weights which are not possible in the real scenario.

Beloglazov and Buyya extended [21] by proposing an architectural framework using VM consolidation algorithms based on double and fixed thresholds for detection of overloaded hosts [27]. However, VM deployment was based on a modified best-fit decreasing heuristic. It considered only a single criterion that is the increase in power consumption. The system was tested on a random dataset.

Beloglazov and Buyya [20] used adaptive double thresholds for host overload detection; bin packing heuristics were used for VM deployment as in [21]. Planet Lab data were used to check the efficiency of the designed model. This framework also considered a single criterion for VM placement.

Cao and Dong [28] reformed the framework designed in [20] by introducing an algorithm for detecting those overloaded hosts which violate SLA. For VM deployment, minimum power maximum utilization (MPMU) was used. Planet Lab dataset was used to evaluate its performance. Horri et al. [29] proposed a VM consolidation approach that considered QoS factors also for designing energy-efficient system. The algorithm proposed for VM deployment examined just two parameters—host utilization and the correlation between applications. Experiments are conducted in CloudSim. The results revealed that a trade-off exists between energy consumption and QoS. Arianyan et al. [30] designed a resource management framework. Authors considered multiple criteria for selecting under-loaded hosts. However, for VM placement modified best-fit decreasing heuristic was used.

Ding et al. [31] designed an energy-aware scheduling algorithm for VMs, considering timing constraints. For allocation of VMs to hosts, hosts were arranged in decreasing order of their optimal performance power ratio and the first host that satisfies VM's requirements was selected. The drawback is that some assumptions were made which did not apply for practical systems.

Zhou et al. [8] proposed a three threshold-based energy-aware algorithm for VM consolidation. VM consolidation algorithms were based on triple and fixed thresholds. Authors proposed a minimization of migration policy based on three thresholds for VM placement. Planet Lab dataset was used to test the validity of the proposed model. Kansal and Channa [32] focused on live VM migration to reduce energy consumption. This work was based on Firefly algorithm. In this technique, the highly loaded VM was migrated to the least loaded node while taking care of performance and energy efficiency of the data center. However, this approach only considered a single parameter, i.e., the load of the node. Goyal et al. [33] designed an energy-efficient resource management system for IaaS clouds. A scheduling algorithm was used that performed VM consolidation using VM admission control approaches. The results showed that the acceptance rate of VMs was increased and overall energy consumption was reduced. However, the system was not generic.

In [34], the authors proposed energy-aware heuristics to maximize resource utilization and considered VM placement as a bin packing problem. The algorithms were based on a single criterion, and the authors considered that all the hosts have similar computing power.

Kafali and Salah [35] performed modeling and analysis of energy consumption in cloud data centers. For effective resource utilization, authors developed a model for predicting the number of VMs needed according to the given workload and for meeting QoS constraints. The model considered a single criterion, i.e., the number of VMs to maintain QoS. However, we have proposed a solution based on multiple criteria to maintain energy efficiency because workloads are not static in the heterogeneous cloud environment.

A recent survey by Arunarani et al. [36] indicates that single-criterion-based resource allocation heuristics like Min–Min algorithm [37], Max–Min algorithms [37] are not optimal. There is a need to design multi-criteria-based novel solutions which are compatible with the dynamic cloud environment. This survey states that there are a lot of possibilities to enhance the existing scheduling algorithm as researchers have been not able to cater to multiple scheduling aspects simultaneously. Authors suggested combining scheduling algorithms with VM consolidation strategies to enhance popular and classic techniques.

In [38], a study of resource management techniques for maintaining energy efficiency is performed. It states that resource management algorithms are the core of energy-efficient solutions. It also suggests formulating resource allocation problem as a multi-objective optimization problem considering multiple parameters such as power consumption, RAM utilization, migration delay and number of VMs.

The literature implies that there are several efforts for developing VM consolidation algorithms, but very few attempts considered multiple resources for VM deployment. Deployment is a significant phase which helps to optimize the whole process [39]. Multiple scheduling parameters are vital for maintaining performance. There is a need to design a VM deployment algorithm considering multiple parameters for selecting hosts. Thus, this work uses Improved TOPSIS technique for VM deployment that considers multiple parameters related to CPU, RAM and network, which reduce both energy consumption and SLA violations.

3 The Proposed VM Deployment Framework

VM consolidation involves (1) detecting overloaded and under-loaded hosts, (2) selecting VMs, which are to be migrated to optimal hosts and (3) placing VMs on target hosts (VM deployment). The process results in lesser power consumption as idle hosts are switched off after migration. The proposed VM deployment framework employs ITHSP for selecting the target hosts for VMs deployment considering scheduling parameters for maintaining energy efficiency. Figure 2 shows the designed framework.

VM deployment (placement) is an MCDM problem. Decision-making includes handling trade-offs or negotiations between various conflicting criteria. Hwang and Yoon developed the TOPSIS method [40]. It is established on the idea that the selected alternative should be the minimum distance (Euclidean distance) apart from the optimal solution and should have maximum distance from the non-optimal solution. For the assumed ideal solution, all the attributes have maximum values in the dataset of satisfying solutions, and for the case of the supposed negative ideal solution, all the attributes have minimum values in the dataset. Therefore, the solution generated by TOPSIS is not only nearest to optimum, but it is also farthest from non-optimum. In TOPSIS, all the parameters are given equal or random weights.

In Improved TOPSIS technique, the decision maker decides the relative importance of weights systematically using analytical hierarchy processing (AHP) technique [41]. AHP is the most extensively used technique for solving MCDM problems. It is a multi-attribute decision-making (MADM) method, which makes ill-structured difficult problems easier by structuring decision factors in a hierarchal way. The benefits of AHP which provide over other methods are its flexibility, spontaneous appeal to the decision makers and its capability to find irregularities. Moreover, AHP creates a hierarchy of criteria by splitting the decision problem into constituent parts. In addition to these advantages, it also decreases biases in decision-making [42]. Because of these benefits of TOPSIS with AHP (Improved TOPSIS), the designed VM deployment framework has applied ITHSP for mapping of VMs on hosts.

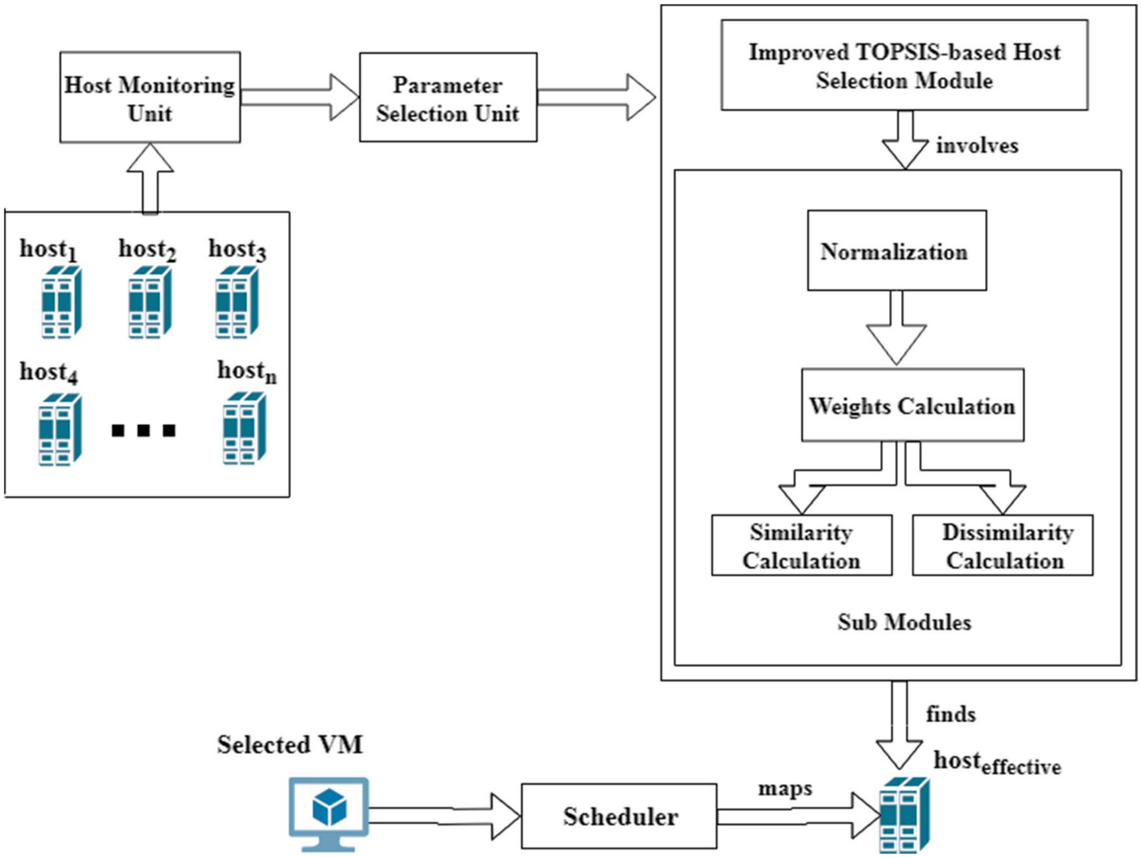


Fig. 2 Design of proposed framework

3.1 Parameters for Host Selection

Power consumption is a vital parameter considered in this work. In addition to it, various other parameters are identified for selection of target hosts in terms of energy efficiency. The selection of criteria considered by ITHSP is motivated from [30], and parameters are described in Table 1. ITHSP policy benefits from the MCDM-Improved TOPSIS algorithm. Scores of all candidate hosts (PMs) (which can host VMs) are calculated using the procedure described in Sect. 3.2.

The host (PM) with the maximum score is chosen. Criteria can be of benefit or cost type in ITHSP policy. The higher value of benefit-type criteria and the lowest value of cost non-benefit type-criteria indicates solution is near to the optimum point. Scores are calculated in ITHSP based on the following conditions:

(1) The host (PM) that is selected should have the maximum available capacity in terms of RAM.
(2) The host (PM) that is selected should have the maximum available computation capacity (power) in MIPS.
(3) The number of VMs in the selected host (PM) should be minimum.
(4) The delay incurred by VM on allocating it to selected host (PM) should be minimum.
(5) Increase in power consumption of the selected host (PM) should be least.

Table 1 Criteria considered by ITHSP

Criteria	Meaning
Available capacity in terms of RAM (AC RAM)	The PM should have the maximum available capacity in terms of RAM
AC capacity in terms of MIPS (AC MIPS)	The PM should have the maximum available computation capacity (power) in MIPS
Number of VMs allocated (VMs)	The total VMs on the selected PM should be minimum
Migration delay (MD)	The delay incurred by VM on allocating it to selected PM should be the minimum
Increase in power consumption (PI)	Increase in power of selected PM should be least

So, it chooses the best alternative from a set of options considering these criteria.

3.2 Proposed Improved TOPSIS-Based Host Selection Policy

ITHSP takes into account multiple resources related to CPU, RAM and network for decision-making. The proposed policy is energy efficient as it considers parameters for reducing energy consumption and SLA violations both. It selects the host that has the minimum number of VMs. So, competent for shared resources becomes less, leading to lesser SLA violations. Further, choosing hosts with the highest available RAM and computation power (MIPS) certifies that VMs will be allocated with ease and reduce SLA violation time per active host. Moreover, considering migration delay decreases violations in SLA during migration and therefore lowers down the performance degradation due to migration. The procedure of ITHSP is:

Step 1 Make an evaluation matrix containing m hosts (options) and n attributes (criteria). The intersection of each option and parameter is called as host_{ij}, $i=1, 2,\ldots, m$ and $j=1, 2,\ldots,n$. Thus, the matrix is $(\text{host})_{m*n}$ (Eq. 1), PH_1, PH_2, … PH_m are the m hosts or PMs (alternatives) and $P_1, P_2, \ldots P_n$ are the different parameters considered for the selection of hosts. $P_{i,j}$ denotes the value of jth parameter of ith host.

Step 2 $(\text{Nhost})_{m*n}$ matrix is constructed by normalizing the matrix $(\text{host})_{m*n}$ as shown in Eq. (2). The values in the formed matrix vary from 0 to 1, where 1 indicates the utmost relevant parameter, and 0 indicates the least relevant parameter

$$\text{Nhost} = \left(\text{host}_{ij}\right)_{m*n}, \tag{1}$$

using the normalization method.

$$\text{Nhost}_{ij} = \frac{\text{host}_{ij}}{\sqrt{\sum_{i=1}^{m} \text{host}_{ij}^2}}, \tag{2}$$

where $i=1, 2\ldots, m$ and $j=1, 2\ldots, n$.

Step 3 A set of weights WT_j (for $j=1, 2,\ldots, n$) such that $\text{WT}_j=1$ is found for parameters (criteria). AHP technique was used to calculate the weights in an orderly manner [41]. The steps of AHP [43] using radical root technique for estimating weights are:

Step 3(i) A pair-wise assessment matrix (Fig. 3) is created using the relative significance scale of criteria shown in Table 2 [43]. As there are n parameters, the pair-wise contrast of ith parameters with jth parameters produces a square matrix $\text{PAR}_{n\times n}$ where PAR_{ij} indicates the relative significance of parameters i in comparison with parameters j. In this matrix, $\text{PAR}_{ij}=1$ when $i=j$ and $\text{PAR}_{ji}=1/\text{PAR}_{ij}$.

Step 3(ii) Calculate the weight (WT_j) of all parameters as follows: (1) the geometric mean of the ith row is calculated (Eq. 3), and (2) the geometric means of the rows in comparison matrix are normalized in Eq. (4).

$$\text{GMean}_j = \left[\prod_{j=1}^{n} \text{PAR}_{ij}\right]^{1/n} \tag{3}$$

$$\text{WT}_j = \text{GMean}_j / \sum_{j=1}^{n} \text{GMean}_j \tag{4}$$

Step 3(iii) Calculate normalized PAR matrix as shown in Eq. (5).

$$\text{NP}_{n*1} = \text{PAR}_{n*n} * \text{WT}_{n*1} \tag{5}$$

Table 2 Scale of the relative significance

Values	Level of importance
1	Similar significance
3	Moderate significance
5	High significance
7	Very high significance
9	Absolute significance
2, 4, 6, 8	For negotiating the values

Fig. 3 Comparison matrix

$$\text{PAR}_{n*n} = \begin{array}{c|cccccc|}
\textit{Parameters} & \text{PAR}_1 & \text{PAR}_2 & \text{PAR}_3 & \cdots & \text{PAR}_{(n-1)} & \text{PAR}_n \\
\text{PAR}_1 & 1 & \text{par}_{12} & \text{par}_{13} & \cdots & \text{par}_{1(n-1)} & \text{par}_{1n} \\
\text{PAR}_2 & \text{par}_{21} & 1 & \text{par}_{23} & \cdots & \text{par}_{2(n-1)} & \text{par}_{2n} \\
\vdots & \vdots & \vdots & \vdots & & \vdots & \vdots \\
\text{PAR}_{n-1} & \text{par}_{(n-1)1} & \text{par}_{(n-1)2} & \text{par}_{(n-1)3} & \cdots & 1 & \text{par}_{(n-1)n} \\
\text{PAR}_n & \text{par}_{n}1 & \text{par}_{n}2 & \text{par}_{n}3 & \cdots & \text{par}_{n(n-1)} & 1
\end{array}$$

Step 3(iv) Calculate relative normalized PAR matrix as shown in Eq. (6).

$$\mathrm{RNP}_{n*1} = \frac{\mathrm{NP}_{n*1}}{\mathrm{WT}_{n*1}} \tag{6}$$

Step 3(v) Define the largest eigenvalue $\lambda_{\max}$ which is calculated by finding the mean of matrix $\mathrm{RNP}_{\mathrm{nx1}}$ and computes the CI (consistency index) as calculated in Eq. (7). The greater is the CI, the greater it deviates from consistency. Therefore, it should be less.

$$\mathrm{CI} = \frac{\left(\lambda\max - n\right)}{(n-1)} \tag{7}$$

Step 3(vi) Find the RI (Random Index) for the used number of attributes in the process of decision-making (Table 3).

Step 3(vii) Find the ratio of consistency (CR) as shown in Eq. (8). CR = 0.1 or less is satisfactory.

$$\mathrm{CR} = \frac{\text{Consistency Index}}{\text{Random Index}} \tag{8}$$

Step 4 Construct the weighted normalized decision matrix as shown in Eqs. (9) and (10).

$$D = \left(d_{m*n}\right) = \left(\mathrm{wt}_j \mathrm{NVMP}\right)_{m*n} \tag{9}$$

$i = 1, 2,\ldots, m$ and $j = 1, 2,\ldots, n$ where

$$\mathrm{wt}_j = \mathrm{WT}_j / \sum_{j=1}^{n} \mathrm{WT}_j \tag{10}$$

$j = 1, 2,\ldots, n$ so that $\sum_{j=1}^{n} \mathrm{WT}_j = 1$, where WT_j represents the original weight assigned to the criteria.

Step 5 Obtain the ideal/optimal alternative ($\mathrm{Al_I}$) and the non-ideal/non-optimal alternative ($\mathrm{Al_N}$) for every criterion as shown in Eqs. (11) and (12)

$$\mathrm{Al_I} = \left\{ \begin{array}{l} \left\langle \min\left(d_{ij} | i = 1,2,\ldots,m\right) \middle| j\epsilon J_-, \right\rangle \\ \left\langle \max\left(d_{ij} | i = 1,2,\ldots,m\right) \middle| j\epsilon J_+ \right\rangle \end{array} \right\} \equiv \left\{ d_{Ij} | j = 1,2,\ldots,n \right\} \tag{11}$$

Table 3 Random index values

Attributes	Values of random index
3	0.52
4	0.89
5	1.11
6	1.25
7	1.35
8	1.4
9	1.45
10	1.49

$$\mathrm{Al_N} = \left\{ \begin{array}{l} \left\langle \max\left(d_{ij} | i = 1,2,\ldots,m\right) \middle| j\epsilon J_-, \right\rangle \\ \left\langle \min\left(d_{ij} | i = 1,2,\ldots,m\right) \middle| j\epsilon J_+ \right\rangle \end{array} \right\} \equiv \left\{ d_{Nj} | j = 1,2,\ldots,n \right\} \tag{12}$$

where,

$$J_+ = \left\{ j = 1,2,\ldots,n \middle| \begin{array}{r} j \text{ is related to positively} \\ \text{impacting criteria} \end{array} \right\},$$

$$J_- = \left\{ j = 1,2,\ldots,n \middle| \begin{array}{r} j \text{ is related to negatively} \\ \text{impacting criteria} \end{array} \right\}$$

Step 6 Calculate the separation measures. The separation of each alternative from the ideal one is represented in Eqs. (13) and (14):

$$\mathrm{Se}_{i+} = \left\{ \sum_{j=1}^{n} \left(d_{ij} - d_{Ij}\right)^2 \right\}^{0.5} \tag{13}$$

$$\mathrm{Se}_{i-} = \left\{ \sum_{j=1}^{n} \left(d_{ij} - d_{Nj}\right)^2 \right\}^{0.5} \tag{14}$$

The values are $i = 1, 2,\ldots, m$ and $j = 1, 2,\ldots, n$.

Step 7 The comparative nearness of a specific alternative to the optimal/ideal solution is called as $\mathrm{Opt\_Host}_i$ and represented in Eq. (15).

$$\mathrm{Opt\_Host}_i = \frac{\mathrm{Se}_{i-}}{\left(\mathrm{Se}_{i+} + \mathrm{Se}_{i-}\right)} \tag{15}$$

The value of $i = 1, 2,\ldots, m$.

Step 8 Rank the choices with respect to $\mathrm{Opt\_Host}_i$ ($i = 1, 2,\ldots, m$), where $\mathrm{Opt\_Host}_i$ signifies the efficiency of an ith host.

Thus, the ITHSP method ensures that the selected host is least distance (Euclidean distance) apart from the optimal solution and the maximum distance apart from the non-optimal solution.

3.3 Case Study: Host Selection for VM Deployment Based on Planet Lab Dataset

Although many scheduling approaches are proposed in the literature, they have limited applicability to dynamic cloud computing environment as they do not consider multiple resources. In this case study, ITHSP is used to find the optimal host for VM deployment. In this process, compliance is generated considering five parameters. A sample dataset generated from Planet Lab data of CPU utilization and real cloud VM

Table 4 Description of planet lab data

Workload trace name	Hosts	VMs
20110303	800	1052
20110306	800	898
20110309	800	1061
20110322	800	1516
20110325	800	1078
20110403	800	1463
20110409	800	1358
20110411	800	1233
20110412	800	1054
20110420	800	1033

Table 5 Configuration of hosts and VMs used in the experiment

Configuration	Host	VMs
Types	2	4
MIPS	{1860, 2660} MHz	{2500, 2000, 1000, 500} EC2 compute units
PES	{2, 2}	{1, 1, 1, 1}
RAM	{4, 4} GB	{0.85, 1.7, 1.7, 0.6} GB
BW	1 Gbit/s	100 Mbit/s

and host configurations is used in this case study. Planet Lab dataset is already provided with CloudSim [44]. The data are associated with CoMon project. CoMon was used to monitor the nodes of Planet Lab. The workload contains CPU utilization of VMs collected after every 5 min. The data were collected on ten different days having a diverse number of VMs as described in Table 4.

CloudSim [44] was used during implementation. It allows us to experiment on large-scale virtualized data centers. The data center has 800 heterogeneous hosts having 400 HP ProLiant ML110 G4 and 400 HP ProLiant ML110 G5. Four types of VMs that correspond to Amazon EC2 instances are used. The hosts' and VMs' configuration is given in Table 5. Power models employed are taken from SPEC power benchmark [20] as shown in Table 6.

Throughout the simulations, each VM is arbitrarily given a workload trace from one of the VMs from the corresponding day. For the demonstration of the results, sample dataset comprises of performance of 40 hosts accessed on five parameters. The parameters are available capacity in terms of RAM (AC RAM), available capacity in terms of MIPS (AC MIPS), number of VMs allocated on a host, migration delay and an increase in power consumption. AHP method is used to calculate relative weights that are to be assigned to these parameters. The parameters AC RAM and AC MIPS are presumed to be beneficial attributes in the case study. From the dataset generated by using Planet Lab workload trace and real hosts and VMs configuration, the normalized decision matrix (Table 7) is formed. It is a 40×5 matrix, which represents 40 physical hosts on five attributes.

The relative significance among parameters is given in Table 8. The steps described in Sect. 3.2 are used to calculate relative normalized weights of all attributes. The relative normalized weights of parameters are AC RAM = 0.1786, AC MIPS = 0.2143, number of VMs allocated = 0.1786, migration delay = 0.1071 and increase in power consumption = 0.3214. The value of $\lambda_{max} = 5.4222$, CI = 0.1056 and CR = 0.0950, which is smaller than the allowed CR value of 0.1. It denotes the good reliability in the judgments for evaluating the weights of attributes.

According to the normalized weights, effective host for VM deployment is found. They are based on the ranks generated according to compliance of parameters, as shown in Table 9. From Table 9, it is evident that PH_401 is evaluated to be the most effective host for VM deployment with compliance of parameters (AC RAM = 0.1353, AC MIPS = 0.3219, number of VMs allocated = 0.0576, migration delay = 0.0265 and increase in power consumption = 0.0275). PH_612 is assessed to be the most non-optimal host with compliance of parameters (AC RAM = 0.1832, AC MIPS = 0.0089, number of VMs allocated = 0.0576, migration delay = 0.2592 and increase in power consumption = 0.5104).

4 Results and Discussion

During VM placement, various hosts are investigated to find an energy-efficient target host. Because of being a dynamic technology, the problem of host selection in cloud computing data centers needs a solution based on multiple criteria. The designed system selects a target host in terms of energy efficiency and efficient resource utilization considering multiple criteria.

Table 6 Power consumption (W)

Server	Idle	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
HP Proliant G4	86	89.4	92.6	96	99.5	102	106	108	112	114	117
HP Proliant G5	93.7	97	101	105	110	116	121	125	129	133	135

Table 7 Normalized decision matrix

Physical host	AC RAM	AC MIPS	Number of VMs allocated	Migration delay	Increase in power consumption
PH_1	0.0620	0.1799	0.0576	0.0265	0.0183
PH_2	0.0508	0.1138	0.1153	0.0596	0.0356
PH_3	0.0396	0.0476	0.1729	0.1022	0.0356
PH_4	0.0421	0.1138	0.0576	0.0735	0.0183
PH_5	0.0309	0.0476	0.1153	0.1124	0.0183
PH_201	0.0620	0.2858	0.0576	0.0265	0.0178
PH_204	0.0171	0.0212	0.2882	0.2384	0.0394
PH_207	0.0110	0.0873	0.1153	0.1653	0.0394
PH_209	0.0197	0.0873	0.1729	0.1625	0.0178
PH_210	0.0084	0.0212	0.2306	0.2294	0.0394
PH_211	0.0046	0.0873	0.0576	0.1620	0.0178
PH_212	0.0577	0.0212	0.0576	0.0367	0.0178
PH_401	0.1353	0.3219	0.0576	0.0265	0.0275
PH_402	0.1241	0.2557	0.1153	0.0596	0.0275
PH_403	0.1129	0.1896	0.1729	0.1022	0.0275
PH_404	0.1016	0.1234	0.2306	0.1589	0.0577
PH_405	0.0904	0.0573	0.2882	0.2384	0.0275
PH_406	0.1154	0.2557	0.0576	0.0735	0.0275
PH_407	0.0843	0.1234	0.1153	0.1653	0.0275
PH_408	0.0778	0.1234	0.0576	0.1620	0.0275
PH_409	0.1310	0.0573	0.0576	0.0367	0.0577
PH_410	0.1042	0.1896	0.1153	0.1124	0.0577
PH_411	0.0929	0.1234	0.1729	0.1625	0.0275
PH_412	0.0817	0.0573	0.2306	0.2294	0.0577
PH_413	0.0730	0.0573	0.1729	0.2229	0.0275
PH_414	0.0666	0.0573	0.1153	0.2121	0.0275
PH_601	0.2819	0.3396	0.0576	0.0265	0.2137
PH_602	0.2706	0.2735	0.1153	0.0596	0.2137
PH_603	0.2594	0.2073	0.1729	0.1022	0.2137
PH_604	0.2482	0.1412	0.2306	0.1589	0.3216
PH_605	0.2369	0.0750	0.2882	0.2384	0.2137
PH_606	0.2369	0.0750	0.3458	0.2980	0.3216
PH_607	0.2619	0.2735	0.0576	0.0735	0.2137
PH_608	0.2308	0.1412	0.1153	0.1653	0.2137
PH_609	0.1997	0.0089	0.1729	0.2833	0.2137
PH_610	0.2244	0.1412	0.0576	0.1620	0.2137
PH_611	0.2775	0.0750	0.0576	0.0367	0.2137
PH_612	0.1832	0.0089	0.0576	0.2592	0.5104
PH_613	0.2507	0.2073	0.1153	0.1124	0.2137
PH_614	0.2395	0.1412	0.1729	0.1625	0.2137

4.1 Dataset

A standard dataset (Planet Lab) containing CPU utilization values is investigated during the literature survey. A simulation run of 800 hosts for the data center is performed. Improved TOPSIS method evaluates the compliance of hosts as described in Sect. 3.2. The efficiency of hosts is classified in Table 10.

4.2 Results on Planet Lab Dataset

Figure 4 illustrates the evaluation of physical hosts on various parameters. Figure 5 presents the relative importance of parameters utilized in this study of host selection. Figure 5 indicates the comparative significance of parameter AC RAM relative to other parameters (AC MIPS, number of VMs allocated, migration delay and increase in power

Table 8 Relative importance of parameters

	AC RAM	AC MIPS	Number of VMs allocated	Migration delay	Increase in power consumption
AC RAM	1.00	0.83	1.00	1.67	0.56
AC MIPS	1.20	1.00	1.20	2.00	0.67
Number of VMs allocated	1.00	0.83	1.00	1.67	0.56
Migration delay	0.60	0.50	0.60	1.00	0.33
Increase in power consumption	1.80	1.50	1.80	3.00	1.00

Table 9 Ranks generated according to compliance

Physical host	Rank (normalized)	Rank	Physical host	Rank (normalized)	Rank
PH_1	0.8820	0.7687	PH_409	0.8053	0.7018
PH_2	0.8195	0.7143	PH_410	0.8765	0.7639
PH_3	0.7627	0.6647	PH_411	0.8267	0.7205
PH_4	0.8303	0.7236	PH_412	0.7425	0.6471
PH_5	0.7784	0.6784	PH_413	0.7735	0.6741
PH_201	0.9343	0.8143	PH_414	0.7831	0.6825
PH_204	0.7041	0.6137	PH_601	0.7944	0.6923
PH_207	0.7785	0.6785	PH_602	0.7632	0.6652
PH_209	0.7829	0.6823	PH_603	0.7146	0.6228
PH_210	0.7157	0.6238	PH_604	0.4913	0.4282
PH_211	0.7941	0.6921	PH_605	0.5976	0.5208
PH_212	0.7817	0.6813	PH_606	0.4241	0.3696
PH_401	1.0000	0.8715	PH_607	0.7711	0.6721
PH_402	0.9524	0.8300	PH_608	0.6836	0.5958
PH_403	0.8834	0.7699	PH_609	0.5854	0.5102
PH_404	0.7940	0.6920	PH_610	0.6944	0.6052
PH_405	0.7473	0.6513	PH_611	0.6792	0.5919
PH_406	0.9548	0.8321	PH_612	0.2940	0.2562
PH_407	0.8357	0.7283	PH_613	0.7271	0.6337
PH_408	0.8406	0.7326	PH_614	0.6710	0.5848

Table 10 Efficiency classification

S. no.	Efficiency class	Value
1	Extremely energy efficient	≥ .8
2	Energy efficient	.7999– .6
3	Moderately energy efficient	.5999–.4
4	Base line energy efficient	.3999–.2
5	Minimal energy efficient	< .2

consumption) is 0.83, 1.00, 1.67 and 0.56. Similarly, the comparative significance of parameter increase in power consumption compared to other parameters (AC RAM, AC MIPS, number of VMs allocated and migration delay) is 1.80, 1.50, 1.80 and 3.00.

Figure 6 shows the normalized comparative weights for parameters, and an increase in power consumption has the maximum relative normalized weight (increase in power consumption = 0.3214). In comparison, migration delay has the minimum relative normalized weight (migration delay = 0.1071).

In other words, an increase in power consumption has been considered the most significant parameter and migration delay as the least significant parameter.

Figure 7 shows the ranks of physical hosts according to compliance (Fig. 4) and the normalized comparative weights (Fig. 6). It is seen that PH_401 is evaluated to be the most effective host. In contrast, PH_612 is assessed to be the least effective host for VM placement.

The grouping of hosts concerning the classification pattern given in Table 10 is illustrated in Table 11.

The correctness of framework is verified as it (1) uses TOPSIS-a standard MCDM technique, (2) calculated weights using AHP which reduces biasness and (3) considered parameters related to CPU, memory, network and multiple criteria, i.e., energy efficiency and SLA efficiency.

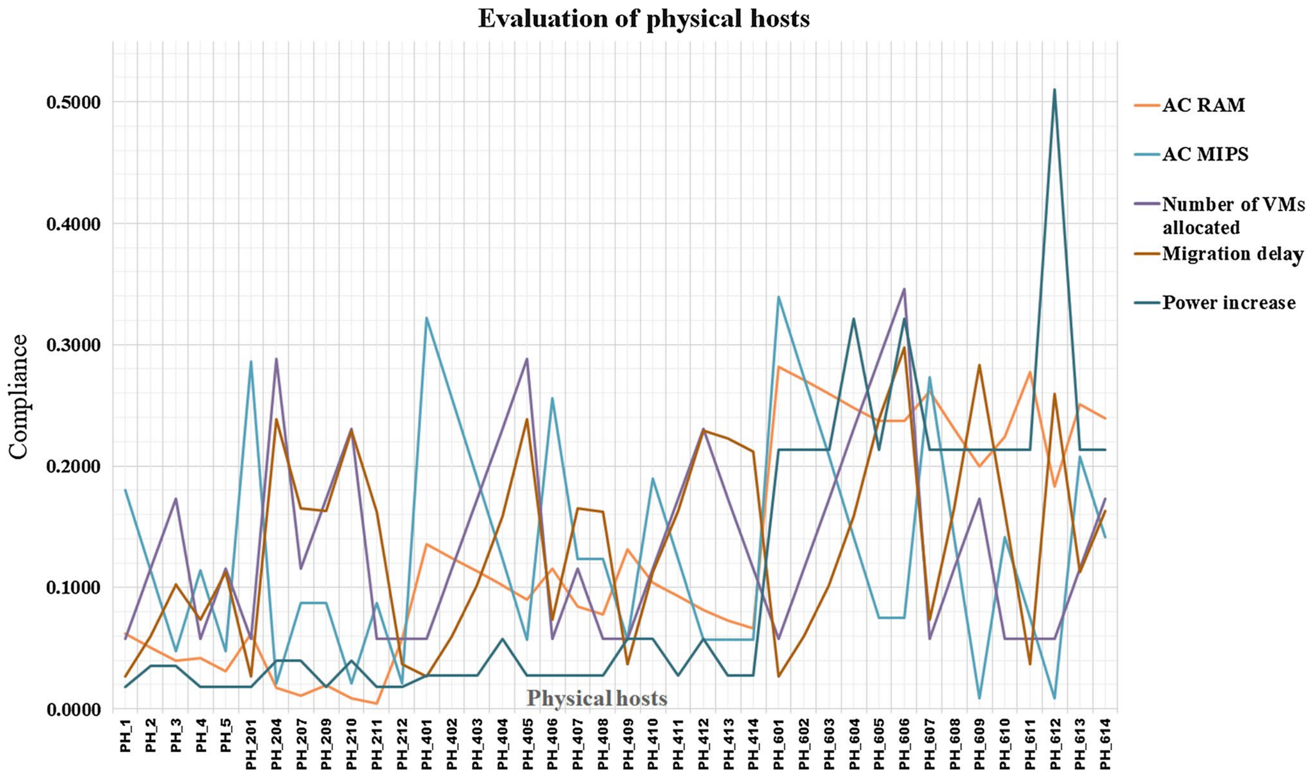
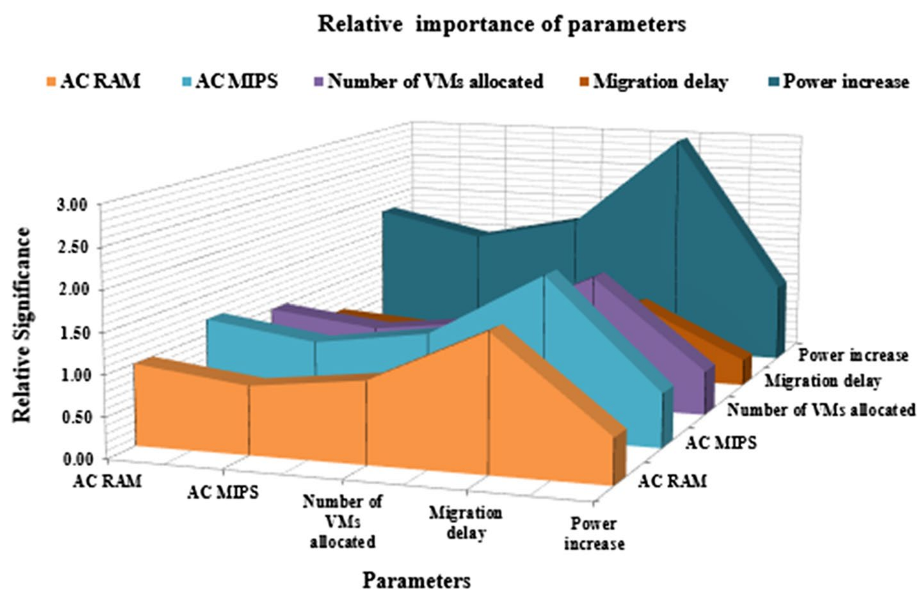


Fig. 4 Compliance of the various attributes for physical hosts

Fig. 5 Relative significance of the parameters for physical hosts

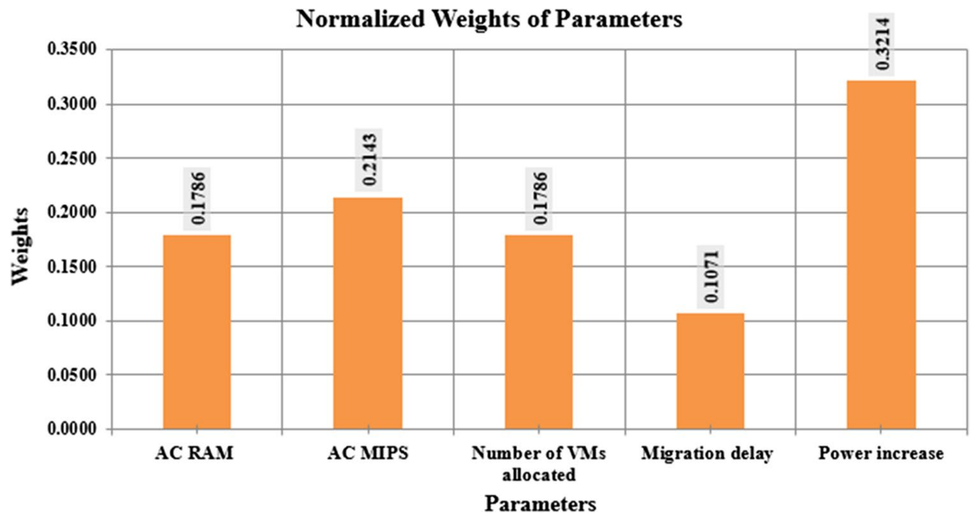


The results are validated as the ranking of hosts are done based on a real dataset, real hosts and VMs' configuration because the case study is an in-depth, detailed, qualitative validation of proposed work representing the living reality [45]. Moreover, the case study approach is widely used in the literature [46–48] for validation of the proposed work.

4.3 Comparison with Other Methods

This section illustrates the comparison between our framework and others in terms of energy efficiency. We have made the comparison qualitatively, as quantitatively comparing the frameworks is not possible. The frameworks

Fig. 6 Relative normalized weights of the parameters for physical hosts



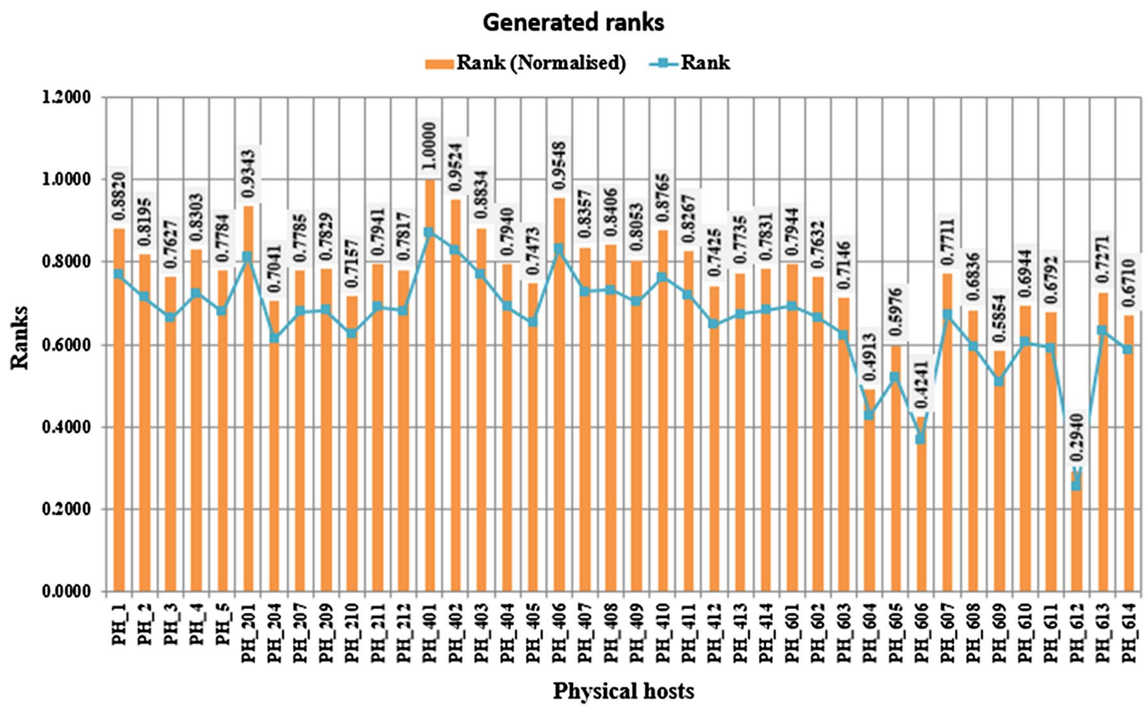


Fig. 7 Efficiency of the hosts generated according to ranks

Table 11 Hosts classification according to efficiency class

S. no.	Efficiency class	Value	Hosts
1	Extremely energy efficient	≥ .8	PH_401, PH_406, PH_402, PH_201
2	Energy efficient	.7999–.6	PH_403, PH_1, PH_410, PH_408, PH_407, PH_4, PH_411, PH_2, PH_409, PH_601, PH_211, PH_404, PH_414, PH_209, PH_212, PH_207, PH_5, PH_413, PH_607, PH_602, PH_3, PH_405, PH_412, PH_613, PH_210, PH_603, PH_204, PH_610
3	Moderately energy efficient	.5999–.4	PH_608, PH_611, PH_614, PH_605, PH_609, PH_604
4	Base line energy efficient	.3999–.2	PH_606, PH_612
5	Least energy efficient	< .2	–

Table 12 Qualitative comparison of the proposed framework

Work	Type	Parameters	Weights decision	Technique	Summary
[18]	Multi-criteria	CPU, network and RAM-related parameters	User	Fuzzy TOPSIS for detecting over-loaded/under-loaded hosts	In this work, Fuzzy TOPSIS was used to find overloaded and under-loaded servers using real case study. However, they did not propose a solution for VM placement. Moreover, the authors did not evaluate the framework using a cloud environment
[21]	Single criterion	Considered CPU only	Not applicable	Bin packing heuristics	Authors proposed an energy-efficient resource management system, but it was not or multiple resources. Further, it was tested for the synthetic dataset and SLA-related metrics were not considered
[26]	Multi-criteria	CPU, disk utilization	Equal weights	Integer nonlinear programming, simulated annealing algorithm	The authors proposed a scheduling model using integer nonlinear programming and considered equal weights for parameters. However, this is not realistic in the dynamic cloud environment
[30]	Multi-criteria	CPU, network and RAM-related parameters	Equal weights	TOPSIS	The proposed resource management framework used TOPSIS for allocation of VMs to hosts and considered equal weights for parameters. However, this is not realistic in the dynamic cloud environment
[32]	Single criterion	Load of the node	Not applicable	Bio-inspired technique	In this work, authors proposed techniques for live VM migration mainly focusing on the load of the node
This work	Multi-criteria	AC RAM, AC MIPS, Number of VMs allocated, migration delay, increase in power consumption	AHP Technique	Improved TOPSIS	We have used Improved TOPSIS technique (decided parameter weights using AHP) for selecting the target host for the deployment of migrated VMs. The proposed framework considers multiple CPU, network and memory-related parameters, and its efficiency is proved with a real case study

developed in [18, 21, 26, 30, 32] are considered for comparison as they have also made efforts to tackle the same issue. The model [18] used TOPSIS for migration of VMs to utilize the resources optimally. However, we have systematically decided the weights of parameters employing AHP technique (MADM), i.e., we have used Improved TOPSIS for VM deployment. It makes our framework suitable for the real environment. In [21], VMs were mapped to PMs using bin packing heuristics rather than following the MADM approach, which may not be suitable for a dynamic environment. The framework in [26] used a scheduling algorithm for mapping of VMs to hosts such that power consumption is minimum. However, attributes were given equal weights which are too far from reality. Therefore, this may not be a realistic framework. In framework [30], authors considered multiple criteria and used TOPSIS for selecting under-loaded hosts. However, for VM placement modified best-fit decreasing heuristic was used. In [32], only a single criterion, i.e., the load of the node, was considered to decide on placement. Thus, this is not suitable for the dynamic cloud environment and is not compliant with features provided by cloud computing [49]. The framework proposed in this paper is more robust, energy efficient and considers multiple resources like CPU, RAM and network. It can be used to evaluate the compliance of hosts in the real cloud environment.

Table 12 illustrates the comparison of the proposed framework with other frameworks.

5 Conclusion

The issue of maintaining energy efficiency in cloud data centers should be given utmost priority. This article proposes a VM deployment framework, which selects the target hosts for VM placement considering energy-efficient use of physical resources. Ranks are generated according to optimal hosts. The proposed ITHSP scheme uses Improved TOPSIS to evaluate the energy efficiency of hosts normalized in the range 0–1. Parameters for maintaining energy efficiency (scheduling parameters for energy-efficient deployment of VMs) are also described, which help to maintain the SLA. The energy efficiency estimated by ITHSP is categorized into five classes, ranging from extremely energy efficient to minimal energy efficient. The designed framework is an effort in the direction of research on energy-efficient VM deployment. The parameters considered in this work lead to a reduction in SLA violations, performance degradation due to migration and SLA violation time per active host. In the future, we are planning to implement Improved TOPSIS algorithm in other phases of VM consolidation like VM selection.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Buyya, R.: Market-oriented cloud computing: vision, hype, and reality of delivering computing as the 5th utility. In: 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, 2009, vol. 25, no. 6, pp. 1
2. Altomare, A.; Cesario, E.; Vinci, A.: Data analytics for energy-efficient clouds: design, implementation and evaluation. Int. J. Parallel Emerg. Distrib. Syst. (2018)
3. The Climate Group: SMART 2020 : Enabling the low carbon economy in the information ag (2008). https://www.theclimategroup.org/sites/default/files/archive/files/Smart2020Report.pdf. Accessed 24 June 2018
4. Cook, G.; Dowdall, T.; Pomerantz, D.; Wang, Y.: Clicking Clean: How Companies are Creating the Green Internet. Greenpeace Inc., Washington, DC (2014)
5. Living planet report (2014). https://www.wwf.or.jp/activities/data/WWF_LPR_2014.pdf. Accessed 24 June 2018
6. Kaur, T.; Chana, I.: Energy efficiency techniques in cloud computing: a survey and taxonomy. ACM Comput. Surv. **48**(2), 1–46 (2015)
7. Zhou, Z.; et al.: Minimizing SLA violation and power consumption in cloud data centers using adaptive energy-aware algorithms. Future Gener. Comput. Syst. **86**, 836–850 (2018)
8. Zhou, Z.; Hu, Z.; Li, K.: Virtual machine placement algorithm for both energy-awareness and SLA violation reduction in cloud data centers. Sci. Program. **2016**(i), 15 (2016)
9. Castro, P.H.P.; Barreto, V.L.; Corrêa, S.L.; Granville, L.Z.; Cardoso, K.V.: A joint CPU-RAM energy efficient and SLA-compliant approach for cloud data centers. Comput. Netw. **94**, 1–13 (2016)
10. Wang, Y.H.; Wu, I.C.: Achieving high and consistent rendering performance of java AWT/Swing on multiple platforms. Softw. Pract. Exp. **39**(7), 701–736 (2009)
11. Cao, Z.; Dong, S.: Dynamic VM consolidation for energy-aware and SLA violation reduction in cloud computing. In: Parallel Distrib. Comput. Appl. Technol. PDCAT Proc., pp. 363–369 (2012)
12. Ashraf, A.; Byholm, B.; Porres, I.: Distributed virtual machine consolidation: a systematic mapping study. Comput. Sci. Rev. **28**, 118–130 (2018)
13. Ashraf, A.; Porres, I.: Multi-objective dynamic virtual machine consolidation in the cloud using ant colony system. Int. J. Parallel Emerg. Distrib. Syst. **33**, 103–120 (2018)
14. Verma, A.; Kaushal, S.: A hybrid multi-objective Particle Swarm Optimization for scientific workflow scheduling. Parallel Comput. **62**, 1–19 (2017)
15. Hussain, H.; et al.: A survey on resource allocation in high performance distributed computing systems. Parallel Comput. **39**, 709–736 (2013)
16. Li, K.; Zheng, H.; Wu, J.; Du, X.: Virtual machine placement in cloud systems through migration process. Int. J. Parallel Emerg. Distrib. Syst. **30**, 393–410 (2015)
17. Nathuji, R.; Schwan, K.: Virtualpower: coordinated power management in virtualized enterprise systems. In: Proc. Twenty-First ACM SIGOPS Symp. Oper. Syst. Princ.—SOSP'07, pp. 265 (2007)
18. Tarighi, M.; Motamedi, S.A.; Sharifian, S.: A new model for virtual machine migration in virtualized cluster server based on fuzzy decision making. J. Telecommun. **1**(1), 40–51 (2010)
19. Calcavecchia, N.M.; Biran, O.; Hadad, E.; Moatti, Y.: VM placement strategies for cloud scenarios. In: 2012 IEEE 5th Int'l Conference on Cloud Comput., pp. 852–859. IEEE
20. Beloglazon, A.; Buyya, R.: Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. Concurr. Comput. Pract. Exp. **24**(13), 1397–1420 (2012)
21. Beloglazov, A.; Buyya, R.: Energy efficient resource management in virtualized cloud data centers. In: 10th IEEE/ACM Int'l Conference on Cluster, Cloud and Grid Comput., pp. 826–831 (2010)
22. Mishra, M.; Sahoo, A.: On theory of VM placement: anomalies in existing methodologies and their mitigation using a novel vector based approach. In: 2011 IEEE CLOUD, pp. 275–282
23. Attaoui, W.; Sabir, E.: Multi-criteria virtual machine placement in cloud computing environments: a literature review. arXiv preprint arXiv:1802.05113 (2018)
24. Al-Haidari, F.; Sqalli, M.; Salah, K.: Impact of cpu utilization thresholds and scaling size on autoscaling cloud resources. In: 2013 IEEE 5th Int'l Conference on Cloud Comput. Tech. and Sci., vol. 2, pp. 256–261. IEEE.
25. Calyam, P.; Rajagopalan, S.; Seetharam, S.; Salah, K.; Ramnath, R.: VDC-analyst: design and verification of virtual desktop cloud resource allocations. Comput. Netw. **68**, 110–122 (2014)
26. Sharifi, M.; Salimi, H.; Najafzadeh, M.: Power-efficient distributed scheduling of virtual machines using workload-aware consolidation techniques. J. Supercomput. **61**(1), 46–66 (2012)
27. Beloglazov, A.; Abawajy, J.; Buyya, R.: Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. Future Gener. Comput. Syst. **28**(5), 755–768 (2012)

28. Cao, Z.; Dong, S.: An energy-aware heuristic framework for virtual machine consolidation in Cloud computing. J. Supercomput. **69**(1), 429–451 (2014)
29. Horri, A.; Mozafari, M.S.; Dastghaibyfard, G.: Novel resource allocation algorithms to performance and energy efficiency in cloud computing. J. Supercomput. **69**(3), 1445–1461 (2014)
30. Arianyan, E.; Taheri, H.; Sharifian, S.: Novel energy and SLA efficient resource management heuristics for consolidation of virtual machines in cloud data centers. Comput. Electr. Eng. **47**, 222–240 (2015)
31. Ding, Y.; Qin, X.; Liu, L.; Wang, T.: Energy efficient scheduling of virtual machines in cloud with deadline constraint. Future Gener. Comput. Syst. **50**, 62–74 (2015)
32. Kansal, N.J.; Chana, I.: Energy-aware virtual machine migration for cloud computing—a firefly optimization approach. J. Grid Comput. **14**(2), 327–345 (2016)
33. Goyal, S.; Bawa, S.; Singh, B.: Energy optimised resource scheduling algorithm for private cloud computing. Int. J. Ad Hoc Ubiquitous Comput. **23**(1/2), 115 (2016)
34. Lee, Y.C.; Zomaya, A.Y.: Energy efficient utilization of resources in cloud computing systems. J. Supercomput. **60**, 268–280 (2012)
35. El Kafhali, S.; Salah, K.: Modeling and analysis of performance and energy consumption in cloud data centers. Arab. J. Sci. Eng. **43**, 7789–7802 (2018)
36. Arunarani, A.R.; Manjula, D.; Sugumaran, V.: Task scheduling techniques in cloud computing: a literature survey. Future Gener. Comput. Syst. **91**, 407–415 (2019)
37. He, X.S.; Sun, X.H.; Von Laszewski, G.: QoS guided min–min heuristic for grid task scheduling. J. Comput. Sci. Technol. **18**, 442–451 (2003)
38. Bhattacherjee, S.; Das, R., Khatua, S.; Roy, S.: Energy-efficient migration techniques for cloud environment: a step toward green computing. J. Supercomput. (2019). https://doi.org/10.1007/s11227-019-02801-0
39. Csorba, M.J.; Meling, H.; Heegaard, P.E.: A bio-inspired method for distributed deployment of services. New Gener. Comput. **29**, 185 (2011)
40. Yoon, K.P.; Hwang, C.-L.: Multiple Attribute Decision Making: An Introduction. Sage publications, Thousand Oaks (1995)
41. Rao, R.V.: Improved multiple attribute decision making methods. In: Pham, D.T. (ed.) Decision Making in Manufacturing Environment Using Graph Theory and Fuzzy Multiple Attribute Decision Making Methods, vol. 2. Springer, Berlin (2013)
42. Ramanathan, R.: A note on the use of the analytic hierarchy process for environmental impact assessment. J. Environ. Manag. **63**, 27–35 (2001)
43. Saaty, T.L.; Vargas, L.G.: Economic, political, social and technological applications with benefits, opportunities, costs and risks. In: Price, C.C. (ed.) Decision Making with the Analytic Network Process, vol. 195. Springer, Berlin (2006)
44. Calheiros, R.N.; Ranjan, R.; Beloglazov, A.; De Rose, C.A.F.; Buyya, R.: CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Softw. Pract. Exp. **41**, 23–50 (2011)
45. Hodkinson, P.; Hodkinson, H.: The strengths and limitations of case study research. In: 2001 Learning and Skills Development Agency Conference at Cambridge, vol. 1, no. 1, pp. 5–7
46. Xu, J.; Fortes, J.A.: Multi-objective virtual machine placement in virtualized data center environments. In: 2010 IEEE/ACM Int'l Conference on Green Comput. and Communications & Int'l Conference on Cyber, Physical and Social Comput., pp. 179–188. IEEE
47. Janpan, T.; Visoottiviseth, V.; Takano, R.: A virtual machine consolidation framework for CloudStack platforms. In: 2014 Int'l Conference on Information Networking, pp. 28–33. IEEE
48. Corradi, A.; Fanelli, M.; Foschini, L.: VM consolidation: a real case based on OpenStack Cloud. Future Gener. Comput. Syst. **32**, 118–127 (2014)
49. Wang, L.; et al.: Cloud computing: a perspective study. New Gener. Comput. **28**, 137–146 (2010)