Design and Development of an Energy Efficient Multimedia Cloud Data Center with Minimal SLA Violation

Nirmal Kr. Biswas[1], Sourav Banerjee[2]*, Utpal Biswas[3]

[1] Global Institute of Management & Technology, Krishnagar, 741102 (India)
[2] Kalyani Government Engineering College, Kalyani, 741235 (India)
[3] University of Kalyani, Kalyani, 741235 (India)

Received 21 October 2020 | Accepted 13 January 2021 | Published 20 April 2021

unir LA UNIVERSIDAD EN INTERNET

Abstract

Multimedia computing (MC) is rising as a nascent computing paradigm to process multimedia applications and provide efficient multimedia cloud services with optimal Quality of Service (QoS) to the multimedia cloud users. But, the growing popularity of MC is affecting the climate. Because multimedia cloud data centers consume an enormous amount of energy to provide services, it harms the environment due to carbon dioxide emissions. Virtual machine (VM) migration can effectively address this issue; it reduces the energy consumption of multimedia cloud data centers. Due to the reduction of Energy Consumption (EC), the Service Level Agreement violation (SLAV) may increase. An efficient VM selection plays a crucial role in maintaining the stability between EC and SLAV. This work highlights a novel VM selection policy based on identifying the Maximum value among the differences of the Sum of Squares Utilization Rate (*MdSSUR*) parameter to reduce the EC of multimedia cloud data centers with minimal SLAV. The proposed *MdSSUR* VM selection policy has been evaluated using real workload traces in CloudSim. The simulation result of the proposed *MdSSUR* VM selection policy demonstrates the rate of improvements of the EC, the number of VM migrations, and the SLAV by 28.37%, 89.47%, and 79.14%, respectively.

Keywords

Energy Consumption (EC), VM Migrations, Multimedia Cloud (MC), SLA Violation (SLAV), VM Selection.

DOI: 10.9781/ijimai.2021.04.004

I. Introduction

Nowadays, multimedia is emerging as a service over the Internet. Multimedia applications [1], like image searching, sharing, editing, video conferencing, multimedia content delivery, multimedia streaming, video retrieval, etc. required a massive amount of storage and computation power. Thus Cloud Computing [2] technology can provide multimedia application services to the users on demand. In Multimedia Cloud (MC) [3], cloud service providers deploy the cloud resources to process multimedia demands and provide the necessary service to the users. The user can store and process the requisite multimedia application data in the cloud in a distributed manner in the modern paradigm of Multimedia Cloud (MC). The need for a full installation of user's media applications in the user's computer is over. The biggest challenge is to optimally allocate the resources to maintain the Quality of Service (QoS) [4] of the various multimedia applications.

The demand for multimedia services has increased rapidly day by day. The MC data centers required a substantial amount of energy to provide services to the users. However, it's a challenge to the researchers to give the MC services to the users with satisfactory Quality of Service (QoS). A large-scale MC data centers consist of millions of servers. It consumes a considerable amount of energy and emitting a massive amount of carbon dioxide into the environment. The electricity consumed by global data centers in 2018 was an estimated 198 terawatt-hours (TWh), which is almost 1% of the demand for global electricity [5]. Because of the energy consumption of global data centers, the average electricity emission rate at each data center is about 4.4 kilogram of carbon dioxide per kilowatt-hour [6]. The Yale School of the Environment estimates that global data centers have a gross emission of carbon dioxide compared to the aviation industry [7] around the world, amounting to around 900 billion kilograms of carbon dioxide [8]. So, global data centers' energy consumption reduction with satisfactory QoS to the MC users becomes a key concern to the researchers. The virtualization approach is used to address these issues. In the MC environment, Virtual machines (VMs) are created in physical machines (PM) using virtualization [9]–[11] technology, depending on the user's request. PMs are encapsulated different applications in the form of VMs by separating with each VM. Each VM required some resources like CPU, Memory, Storage, Band Width, etc. To run the VM, the sum of the required resources must always be lesser than the host capacity. VM Consolidation (VMC) [12]–[14] is an approach which can efficiently utilize the resources with satisfactory QoS. The Service Level Agreement (SLA) [15] between MC users and service providers define QoS. VMC can help to reduce the energy

* Corresponding author.

E-mail address: mr.sourav.banerjee@ieee.org

consumption of MC data centers with optimal SLA violations. It has four main steps. Firstly, detect the overloaded hosts and then detect the underloaded hosts. Now select some VMs from overloaded hosts and all VMs from underloaded hosts. The selected VMs must eventually migrate to the medium loaded hosts, and all of the underloaded hosts are shut down. MC data center's required energy can be minimized by using VMC and VM migration [16], [17]. In live VM migration, an entire running VM can move from one host to another host without any interruption of the user's service. But, too much VM migration may increase the SLA violation and cost of the operation.

VM selection is an essential part of VMC. It creates a bunch of VMs from an overloaded host, which should migrate to moderately loaded hosts. The selection of proper VMs from an overloaded host may control the number of migrations. It can reduce the operational cost with reduced energy consumption and SLA violation. Herein, we have proposed a novel VM selection policy based on Maximum value among the differences in the Sum of Squares Utilization Rate (*MdSSUR*). The proposed *MdSSUR* VM selection policy selects VMs from overload hosts and performs the VM migration with reduced EC and SLA violation. It also reduced the number of migrations. The proposed *MdSSUR* VM selection policy has been executed and evaluated in CloudSim 3.0 [18]. The performance of the proposed *MdSSUR* VM selection policy has been assessed with some established VM selection approaches [19]–[21]. The proposed *MdSSUR* VM selection policy significantly improved the EC, SLAV, and VM migration.

The remaining section of the paper is organized as follows. Section II gives a brief description of the literature survey and state of the art. The proposed *MdSSUR* VM selection policy is shown in section III. The VM consolidation based on *MdSSUR* VM selection policy is shown in section IV. The detailed analysis of the proposed *MdSSUR* VM selection policy is shown in section V. Finally, Section VI concludes the research work.

II. Literature Survey

MC's rapid growth has got more attention from MC providers for the MC data center's cost and efficiency. MC providers of-fer high-quality services at the lowest price to mesmerize the MC users. The number of MC users are increasing exponentially. So, efficient approaches must be adopted by the service providers to satisfy the user's requirements with minimum energy consump-tion and SLA violation. Virtualization [11] is an approach where VMs are into the servers to provide the services to the MC users.

In [22]–[24], the authors successfully tried to address the EC issue with a DVFS approach in cloud computing. DVFS is an approach in which the server load is balanced dynamically with the CPU's voltage and frequency. The energy consumption has reduced for lower voltage and frequency. But, lower CPU frequency may decrease the CPU performance. Beloglazov et al. [19], proposed a system based on VM consolidation and VM migration to improve energy consumption. They developed the following policies for VM selection: 1) Minimum migration time (MMT), 2) Maximum correlation (MC), 3) Minimum utilization (MU), and 4) Random selection (RS). The MMT prefers VMs whose migration time is minimum, and MC selects VMs with full correlation. In the MU selection policy, underutilized VMs are selected, and RS selects the VMs randomly.

Yadav et al. [20] implemented a proposal referred to as the maximum utilization minimum size (MuMs). MuMs is based on CPU utilization and VM's RAM size. It selects highly utilized VMs with minimal RAM, and as an essential parameter, selects the ratio between CPU and VM's RAM size. Akhter et al. [21] proposed a policy to reduce the EC of cloud data centers. Their proposed VM selection policy is known as Maximum migration time (MxMT). The MxMT select VMs with maximum migration time. It reduces EC by 19%. But, MxMT had undoubtedly experienced a severe effect of SLA violation. Lin et al. [25] proposed a model for the task of sequence labeling. Natural language processing (NLP) is used for managing text and speech. In NLP, one of the essential tasks is sequence labeling to define and allocate category label to each unit in the particular entry. These traditional models' efficiency is heavily dependent on manufactured features and task-specific intelligence, which are very time-consuming. The authors developed an attention segmental recurrent neural network (ASRNN) for the task of sequence labeling. The model depends on an ordered recognition neural semi-Markov condition random fields. A hierarchical structure uses to incorporate character level and word level information. The proposed model takes advantage of the hierarchical structure, with many data that achieve competitive efficiency.

R. Mandal et al. [26] developed a VM selection policy known as the Power-Aware VM selection policy. It selects the maximum utilized VM and adds it to the migration list. The utilization of a VM is computed by the ratio of current VM utilization and VM's allocated resources. R. Yadav et al. [27] proposed a Bandwidth-Aware VM selection policy. This policy chooses a VM from a host that is overloaded with a minimum current utilization and total migration time. The Bandwidth Transfer Component (BTC) has been computed by dividing the VM size and its current utilization by available bandwidth. Now, the migration time is calculated for all VMs using BTC and ping time (PT), and finally, a VM with minimum migration time is selected and added to the migration list. Lin et al. [28] proposed an efficient approach namely HUIM-BPSOsig to mine high-utility itemsets. The proposed approach is based on discrete particle swarm optimization (PSO). C. Zhang et al. [29] developed a VM selection policy to reduce the energy consumption and SLA violation of the cloud dissenters. The VM selection aimed to minimize the number and cost of migration. The authors refer to the VM selection as the Minimize Number and Cost of Migrations (MNCM) policy. In MNCM, a VM with maximum VM resource occupancy (VRO) is selected for migration. H. Toumi et al. [30] develop a cooperative framework between Hybrid Intrusion Detection System (Hy-IDS) based upon Mobile Agents and virtual firewalls. The possibility of intrusion rises in occurrence due to the massive use of the cloud. Security, accountability, and stability in the cloud model are essential for customer satisfaction. The minimization of the effect of any penetration into this area is one of the security concerns. The proposed cooperative framework system makes for quicker and more productive detection and resolution of new distributed threats.

Y. Wen et al. [31] proposed a VM selection policy known as minimum migration (MM) policy. The authors calculate the Euclidean distance between the VMs load pattern and PMs load pattern. The distance has been sorted, and depending on a threshold value, select the VMs which consume more resources. In [32], the authors proposed a policy based on Minimum Utilization Gap (MUG). The authors computed the difference between the utilization of overloaded host and upper utilization threshold value as Δ. The relative utilization of each VM was computed by the ratio between the required MIPS and total capacity. The Utilization Gap is the absolute difference between Δ and relative utilization of each VM. In the migration list, VMs with Minimum Utilization Gap are added. Lin et al. [33] developed a model to secure secret and sensitive information. The 6G networking based on Terahertz offers the absolute highest efficiency and reliability but faces new man-in-the-middle attacks. The main challenge of such extremely vulnerable environments is the security and confidentiality of the data. The authors proposed an ant colony optimization (ACO) method to secure 6G IoT networks. The proposed method has multiple targets and the deletion of a transaction to ensure data security.

Every ant in the population is represented as a set of possible deletion transactions for hiding sensitive information. The authors claim that the proposed method reaches a negligible side effect with a low average computational cost. V.K. Solanki et al. [34] have developed a module that integrates new technical peripherals for simple energy-saving trends and modernizes the module in IoT. Owing to irresponsible officials' most resources like water and electricity have been wasted in different cities. The developed module can significantly save the wastage of these resources.

S. B. Melhem et al. [35] evolved a VM selection policy known as Minimum Migration Time Minimum VM Migrated Count (MmtMiMc). MmtMiMc first selects VMs with a minimal quantity of memory and sort them in growing order. Then, from the selected VMs, the policy finds the VMs with a minimum number of VM migrated count and adds them to the migration list to perform the VM migration. They claimed that MmtMiMc decreases the number of VM migration maximum of up to 52.11%. S.M. Moghaddam et al. [36] proposed a VM selection policy based on predictive maximum CPU usages and minimum migration time. The ratio between the memory of VM and available bandwidth of a host is computed, and its minimum value is multiplied with a maximum predicted CPU usage of the VM. H. Peng et al. [37] developed a gradual gradable neural language learning structure. It can be used in the Continuous Bag-of-Words (CBOW) and skip-gram model. The authors extended the classical hierarchical formation from a human tree to a weighted contextual frequency aggregated tree for a long time. S.A. Makhlouf et al. [38] proposed a novel method for data-intensive workflow scheduling applications. Several optimization methods have been developed to improve the cost and efficiency of data-intensive scientific Workflow Scheduling (DiSWS) in cloud computing. Most of the DiSWS techniques are based on an optimization process using heuristic and metaheuristic approaches. The authors explore the task hierarchy in data data-intensive scientific workflows by their proposed method.

J. S. Pan et al. [39] proposed an approach named Multi-group Grasshopper Optimization Algorithm (MGOA). A modern algorithm that imitates Grassley's actions in nature is the Grasshopper Optimization Algorithm (GOA). The MGOA can be used to address the capacitated vehicle routing problem (CVRP). The authors claim that the efficiency of the MGOA is better than the Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). K.W. Huang et al. [40] developed an image recognition framework using the GoogLeNet model. The proposed framework is a convolutionary-neural-network module focused on deep learning. The image recognition framework can enhance the precision of module recognition effectively for preprocessed images. M. El. Ghazouani et al. [41] proposed a block-chain based solution that will maintain the privacy of cloud data checks by deduplicating data. An excellent alternative to ensure data storage reliability in cloud computing is called the deduplication of data. Cloud technology provides many benefits for storage service and poses security problems, notably concerning data privacy, a core component of any cloud system. The proposed method guarantees that consumer data remain confidential for auditors in the course of audits.

P. Xu. et al. [42] developed a VM allocation policy (VMA-ACO) based on Ant Colony Optimization (ACO) [43]. The main aim of the VMA-ACO policy was to maximize resource utilization by balancing the load of physical machines. The utilization of resources has increased by VMA-ACO with minimal SALV. One of the main disadvantages of ACO is its slow convergent. The authors proposed the "PM selection expectation" parameter to overcome the drawback. Yadav et al. [44] proposed the Minimum Sum of CPU Utilization and Memory Size (MSCM) based VM selection policy. In MSCM, the host utilization and the total number of assigned VMs for that host was computed. Then, VMs are sorted in increasing order using their CPU utilization and memory. Now, the authors calculated the host upper utilization threshold, and all the VMs were selected for migration who can bring down its utilization below the upper threshold.

J. Thaman et al. [45] proposed Variance minimization-based selection (Var_Sel) policy. Var_Sel is based on unifying the utilization across the hosts. It selects a VM, which reduced the mean square deviation of the excess load of hosts. The authors proposed a variance-based heuristics approach that selects VMs for migration.

III. Proposed System Model

Consider a large-scale data centers consist of '*m*' overloaded hosts and 1n1 Virtual Machines (VMs). The proposed *MdSSUR* VM selection policy selects some VMs from the overloaded host to perform VM migration with optimal SLAV and reduced EC. The following factors are the base of the proposed *MdSSUR* VM selection policy:

- Compute the Sum of Squares Utilization Rate (*SSUR*) of an Overloaded Host using RAM, Band Width, and MIPS.
- Compute the Remaining Sum of Squares Utilization Rate (*RSSUR*) of an Overloaded Host by excluding one VM re-sources from that Overloaded Host and find out the difference Sum of Squares Utilization Rate (*dSSUR*).
- Select VM with Maximum value among the differences of the Sum of Squares Utilization Rate (*MdSSUR*) to perform VM migration.

Table I sums up the abbreviations of the terms defined in this section.

TABLE I. Abbreviations and Full Names

Abbreviation	Full Name
urRam	Utilization rate of RAM
urBw	Utilization rate of Band Width
urMIPS	Utilization rate of CPU in MIPS
urAvg.	Average utilization
sdurRam	Squared Difference Utilization Rate of RAM
sdurBw	Squared Difference Utilization Rate of Band Width
sdurMIPS	Squared Difference Utilization Rate of CPU in MIPS
ssur	Sum of Squares Utilization Rate
rurRam	Remaining Utilization rate of RAM
rurBw	Remaining Utilization rate of Band Width
rurMIPS	Remaining Utilization rate of CPU in MIPS
rurAvg.	Remaining Average utilization
rsdurRam	Remaining Squared Difference Utilization Rate of RAM
rsdurBw	Remaining Squared Difference Utilization Rate of Band Width
rsdurMIPS	Remaining Squared Difference Utilization Rate of CPU in MIPS
rssur	Remaining Sum of Squares Utilization Rate
dSSRU	Difference Sum of Squares Utilization Rate
udMax	Maximum difference Utilization

A. Sum of Squares Utilization Rate (SSUR) of an Overloaded Host

The Sum of Squares Utilization Rate (*SSUR*) of an overloaded host has computed using the utilization resources like RAM, Band Width, and MIPS. Algorithm 1 is used to find the *SSUR* of an overloaded host. The Utilization Rate (*UR*) of Ram, Band Width, and MIPS of an overloaded host computed using line number 2, 3, and 4 of algorithm 1, respectively. The Squared Difference Utilization Rate (*SDUR*) of Ram, Band Width, and MIPS estimated using line number 6, 7, and 8 of algorithm 1, respectively. Finally, The Sum of Squares Utilization Rate (*SSUR*) of that overloaded host is computed using line number 9 of algorithm 1.

Algorithm 1: Sum of Squares Utilization Rate (SSUR) of an Overloaded Host

Input: $host_j = j^{th}$ overloaded Host

Output: *ssur* = Sum of Squares Utilization Rate

```
1 start
2 urRam ← host_j.usedRam ÷ host_j.totalRam
3 urBw ← host_j.usedBw ÷ host_j.totalBw
4 urMIPS ← 1 – (host_j.availableMIPS ÷ host_j.totalMIPS)
5 urAvg. ← (urRam + urBw + urMIPS) ÷ 3
6 sdurRam ← (urAvg. – urRam)^2
7 sdurBw ← (urAvg. – urBw)^2
8 sdurMIPS ← (urAvg. – urMIPS)^2
9 ssur ← sdurRam + sdurBw + sdurMIPS
10 return ssur
11 stop
```

B. Remaining Sum of Squares Utilization Rate (RSSUR) of an Overloaded Host

The Remaining Sum of Squares Utilization Rate (*RSSUR*) of an Overloaded Host has been computed by excluding one VM's resources from that host. Algorithm 2 is used to find the *RSSUR* of an overloaded host. The utilization rate of Ram, Band Width, and MIPS excluding one VM's resources of the overloaded host is computed by using line number 2, 3, and 4 of algorithm 2, respectively.

Algorithm 2: Remaining Sum of Squares Utilization Rate (*RSSUR*) of an Overloaded Host

Input: $host_j = j^{th}$ overloaded Host, $vm_i = i^{th}$ Virtual Machine of j^{th} overloaded Host

Output: *rssur* = Remaining Sum of Squares Utilization Rate

```
1 start
2 rurRam ← (host_j.usedRam – vm_i.Ram) ÷ host_j.totalRam
3 rurBw ← (host_j.usedBw – vm_i.Bw) ÷ host_j.totalBw
4 rurMIPS ← 1 – ((host_j.availableMIPS – vm_i.MIPS) ÷ host_j.totalMIPS)
5 rurAvg. ← (rurRam + rurBwr + urMIPS) ÷ 3
6 rsdurRam ← (rurAvg. – rurRam)^2
7 rsdurBw ← (rurAvg. – rurBw)^2
8 rsdurMIPS ← (rurAvg. – rurMIPS)^2
9 rssur ← rubfRam + rubfBw + rubfMIPS
10 return rssur
11 stop
```

The Remaining Squared Difference Utilization Rate (*RSDUR*) of Ram, Band Width, and MIPS of that overloaded host excluding one VM's resources computed using line number 6, 7, and 8 of algorithm 2, respectively. Finally, The Remaining Sum of Squares Utilization Rate (*RSSUR*) of an overloaded host excluding one VM's resources computed using line number 9 of algorithm 2.

C. Maximum Difference Sum of Squares Utilization Rate (MdSSUR) VM Selection Policy

The proposed *MdSSUR* VM selection policy described in Algorithm 3. The set of active hosts are the input in the proposed *MdSSUR* VM selection policy. If any active host is overloaded, find out all allocated VMs of that host and add it to the *migratableV ms* list using line number 4. Initially, the *SelectedV m* is set as *NULL* using line number 5, and the Maximum difference utilization (*udMax*) is set by the minimum value using line number 6. Now, compute the *SSUR* of the overloaded host using line number 7. The line number 7 has called algorithm 1 to compute the *SSUR* of that overloaded. Then, for each VM, estimate the *RSSUR* of the overloaded host by excluding each VM's resources using line number 10. The line number 10 has called algorithm 2 to estimate the *RSSUR* of that overloaded host by excluding the resource of each VM. The difference between *SSUR* and *RSSUR* is computed as a difference Sum of Squares Utilization Rate (*dSSUR*) using line number 11. The value of udMax is set as a minimum. Therefore, if the condition of line number 12 becomes true, then the VM for which the value of *dSSUR* is maximum is assigned in *udMax*, and that VM is assigned in *SelectedV* m by line number 13 and 14, respectively. Finally, *SelectedV* m is added to the migration list using line number 18.

Algorithm 3: Proposed MdSSUR VM Selection Policy

Input: *host_list* = set of Active Hosts

Output: *V MsT oMigrateList* = List of Selected VMs needs to be Migrated

```
1 start
2 for each host, in host_list do
3   if (isHostOverloaded(host)) then
4     migratableV ms ← host.AllocatedV Ms()
5     SelectedV m ← NULL
6     udMax ← Double.MinV alue()
7     ssur ← host.SSUR()
8     for each vm, in migratableV ms do
9       if !(isInMigration(vm) then
10        rssur ← RSSUR(host, vm)
11        dSSUR ← ssur – rssur
12        if (dSSUR > udMax) then
13          udMax ← dSSUR
14          SelectedV m ← vm
15        end
16      end
17    end
18    V MsT oMigrateList.add(SelectedV m)
19  end
20 end
```

The difference Sum of Squares Utilization Rate (*dSSUR*) can also be represented by Eq. 1.

$$dSSUR_i = SSUR_{host_j} - RSSUR_{host_j}^{Vm_i} \quad (1)$$

where, $host_j$ is the j^{th} host from the overloaded *host_list*, $V\,m_i$ is the i^{th} VM from the *V m_list* which has been allocated to the $host_i$, $SSUR_{hostj}$ is the Sum of Squares Utilization Rate of $host_j$, $RSSUR_{host_j}^{Vm_i}$ is the Remaining Sum of Squares Utilization Rate of $host_j$ excluding V_{mi} resources, and $dSSUR_i$ is the i^{th} difference of the Sum of Squares Utilization Rate. The VM which has Maximum *dSSUR* is selected form the *V m_list* using Eq. 2.

$$SelectedVm = \max_{\forall Vm_i \in Vm\_list} \{dSSUR_i\} \quad (2)$$

Now, the *SelectedV m* is added to the migration list, and the process will be continued until all overloaded hosts are examined.

IV. Energy Efficient and SLA Aware VM Consolidation Based on *MdSSUR* Policy

VM consolidation comprises the following steps: A) Detection of Overloaded Host, B) Detection of Underloaded Host, C) VM Selection, and D) VM Placement.

A. Overload Host Detection

Overload host detection is the first step of VM consolidation. Initially, the authors [19] set a threshold value of 0.9, and if the CPU utilization of any host is more than the threshold value, then the host is marked as over-utilized. Then, using Linear Regression Robust (LRR), future CPU utilization is predicted. In [19], the authors proposed the LRR prediction model to overcome the disadvantage of Linear Regression (LR) [19] prediction model. The LR prediction model is based on the Loess method proposed by Cleveland [46]. The LR prediction model is vulnerable to outliers due to heavy-tailed distributions. To overcome the Loess method's disadvantage and make it robust, Cleveland proposed the least-squares method [47]. If the multiplication of predicted CPU utilization and safety parameter is greater than or equal to one, the host is marked as over-utilized. For evaluating the proposed *MdSSUR* VM selection policy, the LRR has been used to detect the over-utilized hosts.

B. Underload Host Detection

In [19], the authors proposed an iterative process to determine the underloaded hosts. After the migration of VMs from overloaded hosts to moderately loaded hosts, the underutilized host detection processes start. The system finds the minimum utilized host by comparing it with the other hosts. All the VMs from an underloaded host migrated to the moderately loaded hosts keeping them as not overloaded.

C. VM Selection

Now, select some VMs from overloaded hosts and all VMs from underloaded hosts in this step. The proposed *MdSSUR* VM selection policy described in Algorithm 3 is used to select the VMs from overloaded hosts and added to the migration list. All the VMs from underloaded hosts are selected and added to the migration list to perform the VM migration.

D. VM Placement

After the migration, overloaded hosts will become moderately loaded hosts, and underloaded hosts will be in sleep mode. All the migratable VMs must be placed in some moderately loaded hosts based on some VM placement policy. In this research work, the chosen VM placement policy is the Power-Aware Best Fit Decreasing (PABFD) [19] placement policy for the evaluation of the proposed *MdSSUR* VM selection policy. In PABFD policy, all migratable VMs are sorted decreasingly based on CPU utilization. Each VM has been allocated into a host that required minimum power consumption due to the allocation.

V. Performance Evaluation

A. Experimental Setup

One of the main aspects of the proposed *MdSSUR* VM selection approach is to reduce the total number of VM migrations. The migration list is prepared based on selecting a VM where the difference Sum of Squares Utilization Rate (*dSSUR*) is maximum in an overloaded host and performs VM migration in moderately loaded hosts. It will keep the overloaded hosts undercontrol, and energy consumption by the host will be reduced. The proposed *MdSSUR* VM selection policy will significantly reduce the number of migrations. As a result, it will substantially minimize SLA violations. CloudSim [18], [48], [49] toolkit is the most popular simulator used for large-scale virtualized cloud applications. It provides a stronger virtualized model of cloud architecture compare to other simulators. It supports dynamic resource management and scalability.

A data center containing 800 heterogeneous hosts is used to evaluate the proposed MdSSUR VM selection policy, 50% of them are re HP ProLiant ML110 G4 servers 245 clocked at 1,860 1860MHz, and the remaining are HP ProLiant ML110 G5 servers clocked at 2,660 MHz. Each one has two cores, 4 GB memory, 1 GB/s network bandwidth. Table II is to show the characteristics of the hosts. The hosts' energy consumption characteristics are given in Table III.

TABLE II. Characteristics of Hosts [19]

Host	Clock Speed	Cores	RAM	Bandwidth
G4	1860 MHz	2	4 GB	1 Gbps
G5	2660 MHz	2	4 GB	1 Gbps

The standard Amazon EC2 [51] has been used for the VM instances. Four different types of VM are available. One of the VM instances is created into the host, depending on the requirement of the workload. Table IV is to show the characteristics of the VMs.

TABLE IV. Characteristics of VMs [50]

VM Instances	Clock Speed	Cores	RAM
Micro Instance	500 MHz	1	613 MB
Small Instance	1000 MHz	1	1740 MB
Extra large Instance	2000 MHz	1	1740 MB
High-CPU Medium Instance	2500 MHz	1	870 MB

1. Workload

The experiment has been run using real-life workload traces to make simulation-based approaches more acceptable. Planet-Lab [52] has collected these workload traces from an infrastructure monitoring framework, called CoMon [53]. These traces consist of the CPU utilization data by more than a thousand VMs from many servers located over 500 different places in the world. After every 300 seconds, the utilization values were recorded. During March and April of 2011, ten random dates were chosen from the workload traces. Between them, four days of data is selected for the evaluation of the proposed *MdSSUR* VM selection policy. Table V is to show the characteristics of each workload.

TABLE III. Energy Consumption of PMs At Different Load [19]

PM	Energy Consumption (in Watts) at Different Load on Hosts										
	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
HP ProLiant ML110 G4	86	89.6	92.6	96	99.5	102	106	108	121	114	117
HP ProLiant ML110 G5	93.7	97	101	105	110	116	121	125	129	133	135

TABLE V. Characteristics of Workload [19]

Date	No. of VMs	No. of Hosts	Mean (%)	St. dev. (%)
03-03-2011	1052	800	12.31	17.09
06-03-2011	898	800	11.44	16.83
03-04-2011	1463	800	12.39	16.55
20-04-2011	1033	800	10.43	15.21

B. Result & Analysis

The performance evaluation of the proposed *MdSSUR* VM selection policy has been measured and compared with some classical VM selection algorithms like Minimum Migration Time (MMT) [19], Maximum Correlation (MC) [19], Minimum Utilization (MU) [19], Random Selection (RS) [19], Maximum Utilization Minimum Size (MuMs) [20], Maximum Migration Time (MxMT) [21], and Minimize Number and Cost of Migrations (MNCM) [29]. These policies are previously mentioned in Section II. The Beloglazov et. al.'s [19] proposed metrics have been used to measured and compared the effectiveness of the proposed *MdSSUR* VM selection policy.

1. Performance Degradation Due to Migration (PDM)

Performance degradation due to migration (PDM) is an SLA-based metric. It is represented in Eq. 3. Excessive VM migration may degrade performance. Fig. 1 shows the comparative analysis of PDM of the proposed *MdSSUR* VM selection policy.It indicates that the PDM of the proposed *MdSSUR* VM selection policy is very significantly lesser than other VM selection policies.

$$PDM = \frac{1}{M}\sum_{j=1}^{M}\frac{C_{Deg_j}}{C_{CPU_j}} \quad (3)$$

where,

- M is the number of Virtual machines.
- C_{Deg_j} is the performance degradation of VM j due to migration.
- C_{CPU_j} is the total capacity requested by VM j during its life time.

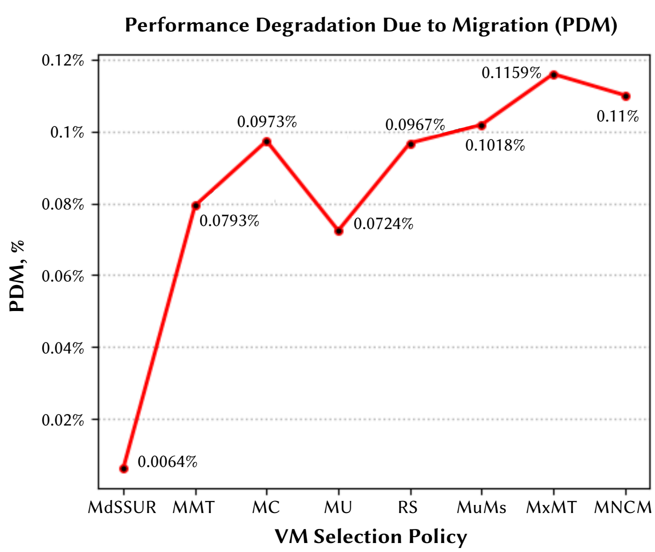

Fig. 1. Performance Degradation due to Migration (PDM) comparison.

2. Service Level Agreement Violation (SLAV)

Service Level Agreement violation (SLAV) is one of the most important metrics. SLAs [54] contains several parameters to satisfy MC users. So, the level of QoS is measured by and reduced by SLA violation. SLAV is calculated by Eq. 4.

$$SLAV = SLATH \times PDM \quad (4)$$

where,

- *SLAV* is a percentage violation of Service Level Agreement.
- *SLATH* is the duration of the 100% CPU use of an active host.
- *PDM* is the performance degradation during VMs migration in percentage.

Fig. 2 is to show the comparative analysis of SLAV, and our proposed VM selection policy has reduced SLAV by 79.14% on an average compare to other approaches.

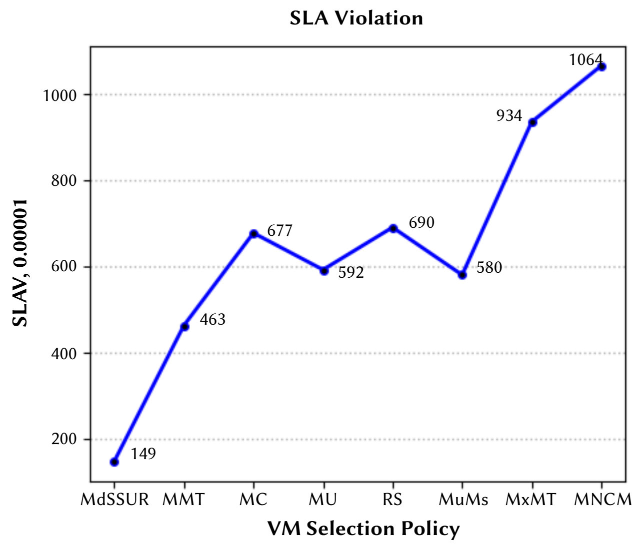

Fig. 2. SLA violation (SLAV) comparison.

3. Number of VM Migrations (NVMG)

Migration of VM is an expensive process. The VM manager initiates VM migration during the VM placement at each time frame. The migrated VMs will possess some CPU time and network bandwidth on both source hosts and targeted hosts. So, VM migration may detrimentally influence the performance of hosts. It may increase the EC and SLA violation of the data center. Therefore, a limited number of VM migration is more desirable. A limited number of VM migration can reduce the total cost of the operation requested by MC users. It can also reduce the total EC and SLA violation of the cloud data centers.

Fig. 3 is to show the comparative analysis of the Number of VM migrations. The total number of migration of the proposed VM selection policy is 2.41. The average number of migration of the other compared policies is 22.89. The proposed VM selection policy has reduced VM migrations by 89.47% on average compared to other approaches mentioned above.

4. Number of Host Shutdowns (HSD)

The number of host shutdowns is a migration based metric. An enormous number of VM migration can increase host shutdowns and the energy consumption of MC data centers. If specific hosts are repeatedly switched on and off, it may increase the MC data centers' EC and operational cost.

Fig. 4 is to show the comparative analysis of the number of host shutdowns of the proposed *MdSSUR* VM selection policy.

From Fig. 4, the proposed VM selection strategy has clearly limited the number of host shutdowns.

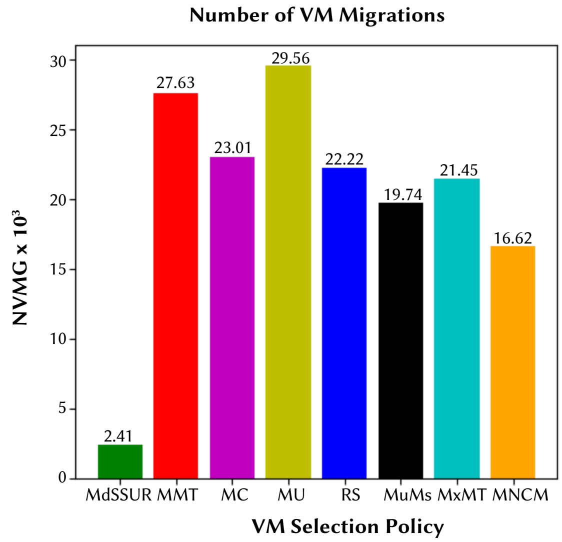

Fig. 3. Number of VM migrations (NVMG) comparison.

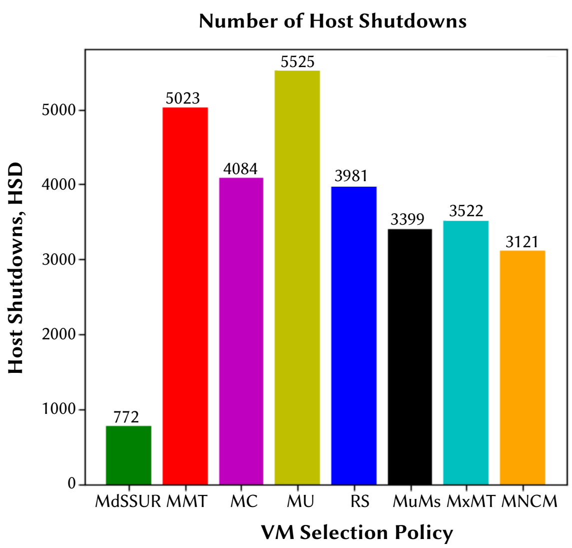

Fig. 4. Number of host shutdowns (HSD) comparison.

5. Total Energy Consumption (EC)

Nowadays, Total energy consumption becomes a key concern to researchers. Reducing MC data centers' energy consumption with optimal SLA violations has become the main objective of the researchers because it has a massive impact on environments.

Fig. 5 is to show the comparative analysis of Total energy consumption. Fig. 5 indicates that our proposed VM selection policy has reduced energy consumption by 28.37% on an average compare to other baseline policies.

6. Energy and SLA Violation (ESV)

The EC and SLA violations of the MC data centers are the essential matrices. However, EC and SLAV are negatively correlated. The EC of MC data centers may typically be minimized by the expense of an increased amount of SLA violations. The resource management system is aimed at reducing EC and the SLA violations of MC data centers. It can be computed by Eq. 5. So, ESV established a relation between two negatively correlated matrices.

$$ESV = EC \times SLAV \quad (5)$$

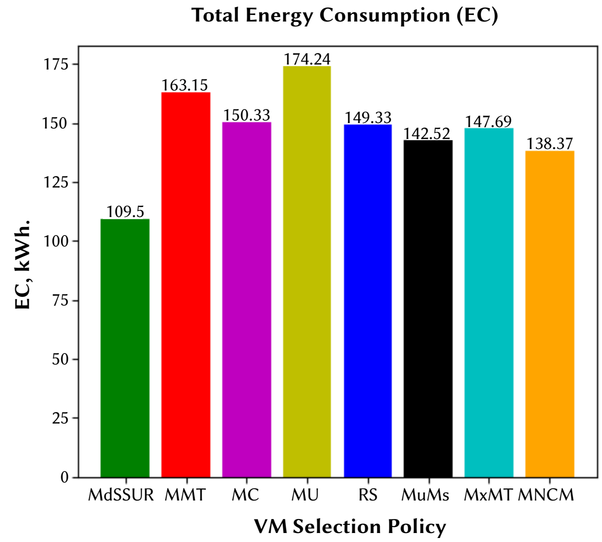

Fig. 5. Total energy consumption (EC) comparison.

Fig. 6 is to show the comparative analysis of ESV. The ESV of the proposed MdSSUR VM selection policy is 1.62, and the average ESV of the other compared policies is 10.75. It indicates that our proposed VM selection has reduced ESV by 84.93% on an average compare to other approaches.

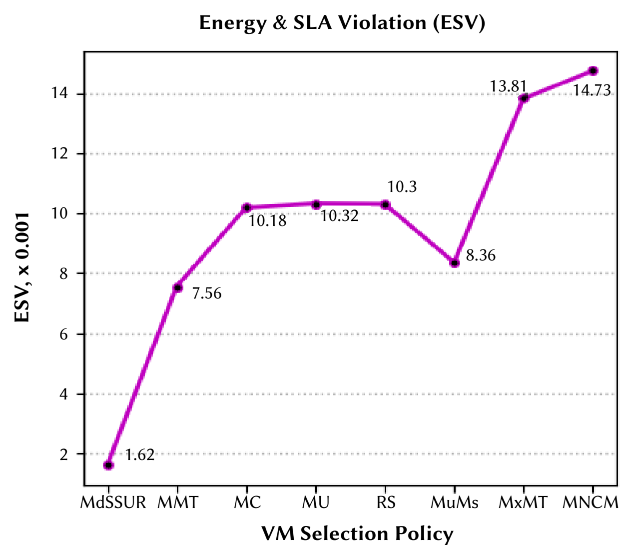

Fig. 6. Energy and SLA Violation (ESV) comparison.

7. Total Execution Time (ET)

Total execution time (ET) is the time to complete an algorithm for a given workload. It determines the efficiency of the algorithms in terms of time. So, the throughput of the MC user's request depends on the total execution time. If the total execution time can be minimized, the throughput of the MC user's request will increase. It can also reduce the operational cost of the MC user's request.

Fig. 7 shows the comparative analysis of total execution time, indicating that the proposed MdSSUR VM selection policy is much faster than other algorithms. The average execution time of the proposed MdSSUR VM selection policy is 50.1 milliseconds, and the average execution time of the other compared policies is 341.1 millisecond. So, the total execution time significantly increased by the proposed MdSSUR VM selection to reduce the MC user's request's operational cost.

TABLE VI. Comparative Analysis of *MdSSUR* VM Selection Policy With Renowned VM Selection Policies

VM Selection Policy	Energy in kWh.	PDM (%)	SLAV $\times 10^{-5}$	Migration $\times 10^{3}$	Host Shutdw.	ESV $\times 10^{-3}$	ExeTime in Milisec.
MdSSUR	109.5	0.0064	149	2.41	772	1.62	50.01
MMT	163.15	0.0793	463	27.632	5023	7.56	409.76
MC	150.33	0.0973	677	23.004	4084	10.18	350.7
MU	174.24	0.0724	592	29.555	5525	10.32	461.77
RS	149.33	0.0967	690	22.223	3981	10.3	318.91
MuMs	142.52	0.1018	580	19.744	3399	8.63	312.51
MxMT	147.69	0.1159	933.5	21.45	3822	13.81	304.91
MNCM	138.37	0.11	1064	16.62	3121	14.73	229.09

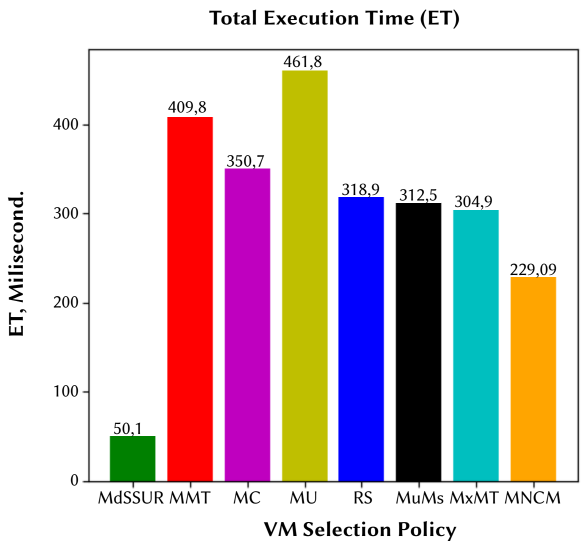

Fig. 7. Total Execution Time (ET) comparison.

Table VI shows a comparative analysis of all the above mention metrics, and Table VII shows the average improvements rate of the proposed MdSSUR VM selection policy based on the above mention metrics.

TABLE VII. Average Improvement Rate of *MdSSUR* Policy

Metric	Avg. Improvement Rate in %
Energy Consumption	28.37
SLA violation	79.14
Number of Migration	89.47
Energy and SLA violation	84.93

8. Workload Based Analysis

In this section, the evaluation of the proposed *MdSSUR* VM selection policy has been done with different workloads like 20110306, 20110403, and 20110420. These workloads result compared with the metrics like Energy consumption, Number of VM migration, and SLA violation. Fig. 8 depicts energy consumption with different workloads. The average reduction in EC of the proposed MdSSUR VM selection policy is 30.28%, 30.7%, and 33.22% compared to the other mentioned policies using 20110306, 20110403, and 20110420 workloads, respectively. The number of VM migrations mainly controls the overall cost of MC users. The number of VM migrations with various workloads shown in Fig. 9. The number of migrations significantly reduced by the proposed MdSSUR VM selection policy. It decreases on average by 87.42%, 90.98%, and 88.08% compared to the other mentioned VM selection policies using 20110306, 20110403, and 20110420 workloads.

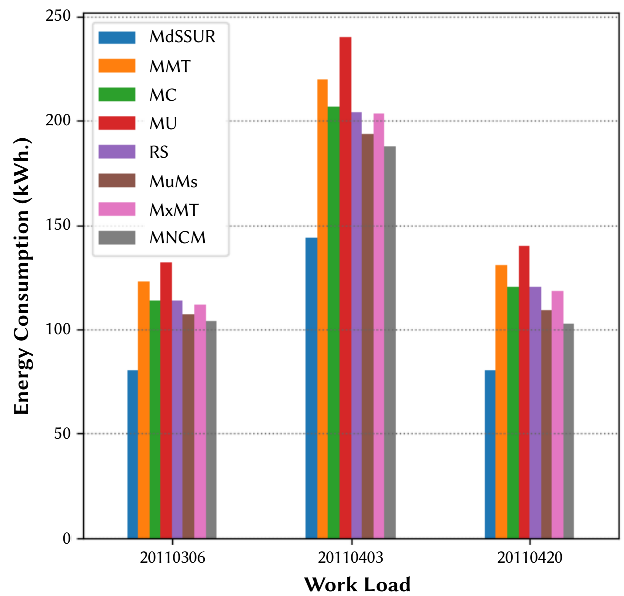

Fig. 8. Total Execution Time (ET) with different workloads.

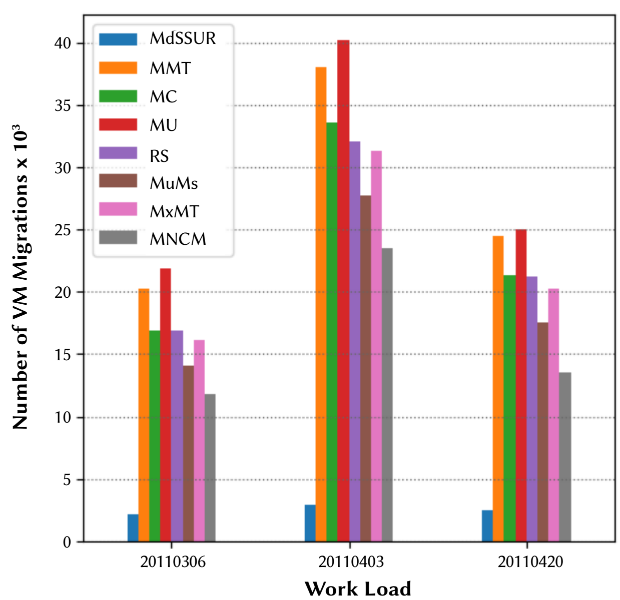

Fig. 9. Number of Migrations with different workloads.

Minimal service level agreement violation increases the QoS provided by the MC service providers. Fig. 10 is to show the SLAV with different workloads. The average reduction of SLAV by the proposed *MdSSUR* VM selection policy is 70.45%, 78.3%, and 79.18% compared to the other mentioned policies using 20110306, 20110403, and 20110420 workloads, respectively.

Thus, the proposed *MdSSUR* VM selection policy defeats other benchmark mentioned VM selection policies.

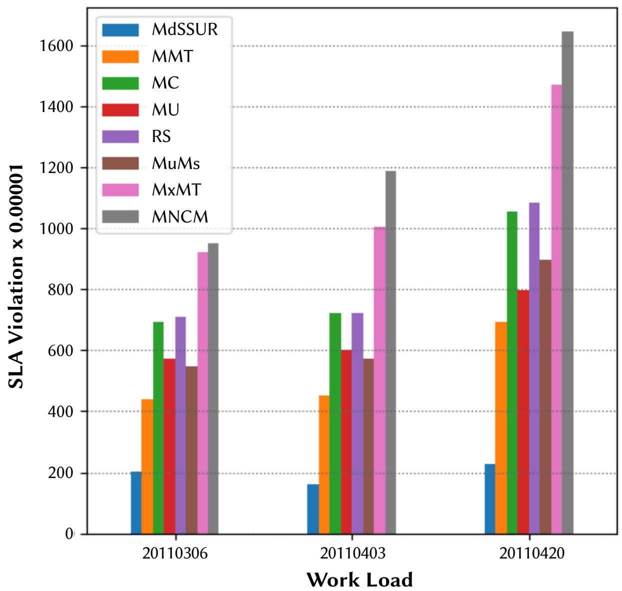

Fig. 10. SLA Violation with different workloads.

VI. Conclusion

Optimal VM selection can decrease VM migrations and increase the throughput of the MC user's request. The emission of greenhouse gases by the MC data centers all over the world needs to be decreased. This paper proposed a novel *MdSSUR* VM selection policy to reduce EC with minimal SLAV. The simulation results have shown that the proposed *MdSSUR* VM selection policy will scale back SLAV and enhance the system performance considerably whereas saving energy. This research plans to incorporate with the Internet of Things (IoT) to enhance the Cloud of Things (CoT) environment.

Acknowledgement

We want to offer our sincere gratitude to Dr. Rajkumar Buyya, one of the renowned researchers behind the inventions and progressive research directions towards cloud computing. One of his excellent contributions to cloud simulation is CloudSim, which has helped thousands of researchers to test different cloud computing algorithms very rapidly. His research work has motivated us to develop the *MdSSUR* VM selection policy.

References

[1] N. Williams, G. S. Blair, "Distributed multimedia applications: A review," *Computer Communications*, vol. 17, no. 2, pp. 119–132, 1994.

[2] M. N. Birje, P. S. Challagidad, R. H. Goudar, M. T. Tapale, "Cloud computing review: concepts, technology, challenges and security," *International Journal of Cloud Computing*, vol. 6, no. 1, pp. 32–57, 2017.

[3] W. Zhu, C. Luo, J. Wang, S. Li, "Multimedia cloud computing," *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 59–69, 2011.

[4] A. Vogel, B. Kerherve, G. von Bochmann, J. Gecsei, "Dis-tributed multimedia and QoS: A survey," *IEEE multimedia*, vol. 2, no. 2, pp. 10–19, 1995.

[5] Masanet, Shehabi, Smith, Lei, "Global Data Center Energy Use: Distribution, Composition, and Near-Term Outlook," *Northwestern University: Evanston*, IL, USA, 2018.

[6] E. Innovation, "How Much Energy Do Data Centers Really Use?." https://energyinnovation.org/2020/03/17/how-much-energy-do-data-centers-really-use/, 2020(ac-cessed on 11th September, 2020).

[7] F. Pearce, "Energy Hogs: Can World's Huge Data Centers Be Made More Efficient?." https://e360.yale.edu/features/energy-hogs-can-huge-data-centers-be-made-more-efficient, 2018(accessed on 18th September, 2020).

[8] ATAG, "Air Transport Action Group: Facts and Figure." https://atag.org/facts-figures.html, 2019(accessed on 17th September, 2020).

[9] F. Lombardi, R. Di Pietro, "Secure virtualization for cloud computing," Journal of network and computer applications, vol. 34, no. 4, pp. 1113–1122, 2011.

[10] Y. Xing, Y. Zhan, "Virtualization and cloud computing," Future Wireless Networks and Information Systems, pp. 305–312, 2012.

[11] N. Jain, S. Choudhary, "Overview of virtualization in cloud computing," *Symposium on Colossal Data Analysis and Networking*, pp. 1–4, 2016.

[12] A. Abdelsamea, E. E. Hemayed, H. Eldeeb, H. Elazhary, "Virtual machine consolidation challenges: A review," International Journal of Innovation and Applied Studies, vol. 8, no. 4, p. 1504, 2014.

[13] M. A. Khan, A. Paplinski, A. M. Khan, M. Murshed, R. Buyya, "Dynamic virtual machine consolidation algorithms for energy-efficient cloud resource management: a review," *Sustainable cloud and energy services*, pp. 135–165, 2018.

[14] H. Wang, H. Tianfield, "Energy-aware dynamic virtual machine consolidation for cloud datacenters," *IEEE Access*, vol. 6, pp. 15259–15273, 2018.

[15] A. Beloglazov, R. Buyya, "System, method and computer program product for energy-efficient and service level agreement (SLA)-based management of data centers for cloud computing," *US Patent 9,363,190, Google Patents*, June 7, 2016.

[16] P. G. J. Leelipushpam, J. Sharmila, "Live VM migration techniques in cloud environment–a survey," *IEEE Conference on Information & Communication Technologies*, pp. 408–413, 2013.

[17] A. Choudhary, M. C. Govil, G. Singh, L. K. Awasthi, E. S. Pilli, D. Kapil, "A critical survey of live virtual machine migration techniques," *Journal of Cloud Computing*, vol. 6, no. 1, p. 23, 2017.

[18] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and experience*, vol. 41, no. 1, pp. 23–50, 2011.

[19] A. Beloglazov, R. Buyya, "Optimal online deterministic algo-rithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers," *Concurrency and Computation: Practice and Experience*, vol. 24, no. 13, pp. 1397–1420, 2012.

[20] R. Yadav, W. Zhang, H. Chen, T. Guo, "Mums: Energy-aware vm selection scheme for cloud data center," *28th International Workshop on Database and Expert Systems Applications, IEEE*, pp. 132–136, 2017.

[21] N. Akhter, M. Othman, R. K. Naha, "Energy-aware virtual machine selection method for cloud data center resource allocation," *arXiv preprint arXiv:1812.08375*, 2018.

[22] A. Beloglazov, R. Buyya, "Energy efficient resource management in virtualized cloud data centers," *10th IEEE/ACM In-ternational Conference on Cluster, Cloud and Grid Comput-ing, IEEE*, pp. 826–831, 2010.

[23] Q. Deng, D. Meisner, L. Ramos, T. F. Wenisch, R. Bianchini, "Memscale: active low-power modes for main memory," *ACM SIGPLAN Notices*, vol. 46, no. 3, pp. 225–238, 2011.

[24] Q. Deng, D. Meisner, A. Bhattacharjee, T. F. Wenisch, R. Bianchini, "MultiScale: memory system DVFS with multiple memory controllers," *Proceedings of the 2012 ACM/IEEE international symposium on Low power electronics and design*, pp. 297–302, 2012.

[25] J. C. W. Lin, Y. Shao, Y. Djenouri, U. Yun, "ASRNN: A recurrent neural network with an attention model for sequence labeling," *Knowledge-Based Systems*, p. 106548, 2020.

[26] R. Mandal, M. K. Mondal, S. Banerjee, U. Biswas, "An approach toward design and development of an energy-aware VM selection policy with improved SLA violation in the domain of green cloud computing," *The Journal of Supercomputing*, pp. 1–20, 2020.

[27] R. Yadav, W. Zhang, O. Kaiwartya, P. R. Singh, I. A. Elgendy, Y.C. Tian, "Adaptive energy-aware algorithms for minimizing energy consumption

and SLA violation in cloud computing," *IEEE Access*, vol. 6, pp. 55923–55936, 2018.

[28] J. C. W. Lin, L. Yang, P. Fournier-Viger, J. M.-T. Wu, T. P. Hong, L. S. L. Wang, J. Zhan, "Mining high-utility itemsets based on particle swarm optimization," *Engineering Applications of Artificial Intelligence*, vol. 55, pp. 320–330, 2016.

[29] C. Zhang, Y. Wang, Y. Lv, H. Wu, H. Guo, "An Energy and SLA-Aware Resource Management Strategy in Cloud Data Centers," *Scientific Programming*, vol. 2019, 2019.

[30] H. Toumi, B. Marzak, A. Talea, A. Eddaoui, M. Talea, "Use Trust Management Framework to Achieve Effective Security Mechanisms in Cloud Environment," *International Journal of Interactive Multimedia & Artificial Intelligence*, vol. 4, no. 3, 2017.

[31] Y. Wen, Z. Li, S. Jin, C. Lin, Z. Liu, "Energy-efficient virtual resource dynamic integration method in cloud computing," *IEEE Access*, vol. 5, pp. 12214–12223, 2017.

[32] X. Wu, Y. Zeng, G. Lin, "An energy efficient VM migration algorithm in data centers," *16th International Symposium on Distributed Computing and Applications to Business, Engineering and Science, IEEE*, pp. 27–30, 2017.

[33] J. C. W. Lin, G. Srivastava, Y. Zhang, Y. Djenouri, M. Alo-qaily, "Privacy Preserving Multi-Objective Sanitization Model in 6G IoT Environments," *IEEE Internet of Things Journal*, 2020.

[34] V. K. Solanki, M. Venkatesan, S. Katiyar, "Conceptual Model for Smart Cities: Irrigation and Highway Lamps using IoT," *International Journal of Interactive Multimedia & Artificial Intelligence*, vol. 4, no. 3, pp. 28–33, 2017.

[35] S. B. Melhem, A. Agarwal, N. Goel, M. Zaman, "Minimizing biased VM selection in live VM migration," *3rd International Conference of Cloud Computing Technologies and Applications, IEEE*, pp. 1–7, 2017.

[36] S. M. Moghaddam, M. O'Sullivan, C. Walker, S. F. Piraghaj, C. P. Unsworth, "Embedding individualized machine learning prediction models for energy efficient VM consolidation within Cloud data centers," *Future Generation Computer Systems*, vol. 106, pp. 221–233, 2020.

[37] H. Peng, L. Liu, L. Ma, W. Zhao, H. Ma, L. Yuntao, "Approximate Error Estimation based Incremental Word Representation Learning," *Data Science and Pattern Recognition*, vol. 4, 2020.

[38] S. A. Makhlouf, B. Yagoubi, "Data-Aware Scheduling Strategy for Scientific Workflow Applications in IaaS Cloud Computing," *International Journal of Interactive Multimedia & Artificial Intelligence*, vol. 5, no. 4, 2019.

[39] J. S. Pan, X. Wang, S.-C. Chu, T. Nguyen, "A multi-group grasshopper optimisation algorithm for application in capacitated vehicle routing problem," *Data Science and Pattern Recognition*, vol. 4, no. 1, pp. 41–56, 2020.

[40] K. W. Huang, C. C. Lin, Y.-M. Lee, Z.-X. Wu, "A deep learn-ing and image recognition system for image recognition," *Data Science and Pattern Recognition*, vol. 3, no. 2, pp. 1–11, 2019.

[41] M. El Ghazouani, E. Kiram, M. Ahmed, L. Er-Rajy, Y. El Khanboubi, "Efficient Method Based on Blockchain Ensuring Data Integrity Auditing with Deduplication in Cloud," *International Journal of Interactive Multimedia & Artificial Intelligence*, vol. 6, no. 3, 2020.

[42] P. Xu, G. He, Z. Li, Z. Zhang, "An efficient load balancing algorithm for virtual machine allocation based on ant colony optimization," *International Journal of Distributed Sensor Networks*, vol. 14, no. 12, p. 1550147718793799, 2018.

[43] M. Dorigo, M. Birattari, T. Stutzle, "Ant colony optimiza-tion," *IEEE computational intelligence magazine*, vol. 1, no. 4, pp. 28–39, 2006.

[44] S. B. S. Yadav, M. Kalra, "Energy-Aware VM Migration in Cloud Computing," *Proceedings of International Conference on IoT Inclusive Life, NITTTR Chandigarh, India*, Springer, pp. 353–364, 2020.

[45] J. Thaman, M. Singh, "SLA conscious VM migration for host consolidation in cloud framework," *International Journal of Communication Networks and Distributed Systems*, vol. 19, no. 1, pp. 46–64, 2017.

[46] W. S. Cleveland, "Robust locally weighted regression and smoothing scatterplots," *Journal of the American statistical association*, vol. 74, no. 368, pp. 829–836, 1979.

[47] W. S. Cleveland, *Visualizing data*. Hobart Press, 1993.

[48] R. Buyya, R. Ranjan, R. N. Calheiros, "Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities," *International conference on high performance computing & simulation, IEEE*, pp. 1–11, 2009.

[49] T. Goyal, A. Singh, A. Agrawal, "Cloudsim: simulator for cloud computing infrastructure and modeling," *Procedia Engineering*, vol. 38, pp. 3566–3572, 2012.

[50] A. Beloglazov, *Energy-efficient management of virtual machines in data centers for cloud computing*. PhD dissertation, 2013.

[51] A. EC, "Amazon EC2 instance types." https: //aws.amazon.com/ec2/instance-types/, 2019(accessed on 21st September 2020).

[52] L. Peterson, A. Bavier, M. E. Fiuczynski, S. Muir, "Experiences building planetlab," *Proceedings of the 7th symposium on Operating systems design and implementation*, pp. 351–366, 2006.

[53] K. Park, V. S. Pai, "CoMon: a mostly-scalable monitoring system for PlanetLab," *ACM Special Interest Group in Operating Systems Review*, vol. 40, no. 1, pp. 65–74, 2006.

[54] R. Sahal, M. H. Khafagy, F. A. Omara, "A survey on SLA management for cloud computing and cloud-hosted big data analytic applications," *International Journal of Database Theory and Application*, vol. 9, no. 4, pp. 107–118, 2016.

Nirmal Kr. Biswas

Nirmal Kr. Biswas is currently working as an Assistant Professor at Department of Computer Science & Engineering of Global Institute of Management and Technology, Krishnagar, West Bengal. He has completed Master of Technology (M.Tech) in Computer Science & Engineering in 2010 from Kalyani Government Engineering College. He has completed his Bachelor of Technology (B.Tech) in Information Technology in 2008 from Murshidabad College of Engineering and Technology. His research interests are in Green Cloud Computing, Cloud Computing, Multimedia Computing.

Dr. Sourav Banerjee

Sourav Banerjee achieved Ph.D degree in Computer Science and Engineering from the University of Kalyani in 2018. He completed his B.E in Computer Science and Engineering in the year 2004 and M.Tech in Computer Science and Engineering in 2006. He is currently an Assistant Professor at Department of Computer Science and Engineering of Kalyani Government Engineering College at Kalyani, West Bengal, India. He has authored numerous reputed non-paid SCI journal articles, book chapters and International conferences. He has edited book on Green Cloud Computing. His research interests include Big Data, Blockchain, Cloud Computing, Green Cloud Computing, Cloud Robotics, Distributed Computing and Mobile Communications, IoT. He is a member of IEEE, ACM, IAE and MIR Labs as well. He is a SIG member of MIR Lab, USA. He has published papers in various reputed journals such as, Wireless Personal Communications, Automatic Control and Computer Sciences, Arabian Journal for Science and Engineering, Journal of King Saud University - Computer and Information Sciences, Service Oriented Computing and Applications, Journal of Super computing, Peer-to-Peer Networking and Applications. He is an Editorial board member of Sustainability Journal (SCI, IF:2.576), Wireless Communication Technology. He is the reviewer of IEEE Transactions on Cloud Computing, Wireless Personal Communications,Journal of Ambient Intelligence and Humanized Computing, Journal of Computer Science,Journal of Super computing, Future Internet,International Journal of Computers and Applications, Personal and Ubiquitous Computing, Innovations in Systems and Software Engineering, Journal of Information Security and Applications, etc. He is connected with various international events, like,Workshop on Security and Privacy in Distributed Ledger Technology (IEEE SP-DLT).He is a lead guest editor of Complex and Intelligent Systems Journal (Springer, SCI, IF: 3.79) for the Special Issue on "Advancement and Trends in Green Cloud Computing, Block chain and IoT for Modern Applications and Systems". He is a guest editor of Mathematical Biosciences and Engineering journal (SCI, IF: 1.285) for the Special Issue on "Recent Advancements in Cognitive Green Computing for Smart Cities". A number of worldwide research scholars are attached with him.

Dr. Utpal Biswas

Utpal Biswas received his B.E, M.E, and Ph.D. degrees in Computer Science and Engineering from Jadavpur University, India in 1993, 2001, and 2008 respectively. From 1994 to 2001, he served as a faculty member in the Department of Computer Science and Engineering at NIT, Durgapur, India. He is currently working as a Professor at the University of Kalyani in the Department of Computer Science and Engineering. At the University of Kalyani, he served as Head of the Department of Computer Science and Engineering. He has also served as Dean, Faculty of Engineering, Technology and Management, University of Kalyani. He has over 200 research articles in different journals, book chapters, and conferences. His research interests include optical communication, ad-hoc and mobile communication, semantic web services, E-governance, Cloud Computing, etc. He is engaged in various state government regulatory committees.