**International Journal of Cloud Applications and Computing**
Volume 9 • Issue 1 • January-March 2019

# Towards Green Cloud Computing an Algorithmic Approach for Energy Minimization in Cloud Data Centers

Jenia Afrin Jeba, Jahangirnagar University, Dhaka, Bangladesh
Shanto Roy, Jahangirnagar University, Dhaka, Bangladesh
Mahbub Or Rashid, Jahangirnagar University, Dhaka, Bangladesh
Syeda Tanjila Atik, Jahangirnagar University, Dhaka, Bangladesh
Md Whaiduzzaman, Jahangirnagar University, Dhaka, Bangladesh

## ABSTRACT

The article presents an efficient energy optimization framework based on dynamic resource scheduling for VM migration in cloud data centers. This increasing number of cloud data centers all over the world are consuming a vast amount of power and thus, exhaling a huge amount of $CO_2$ that has a strong negative impact on the environment. Therefore, implementing Green cloud computing by efficient power reduction is a momentous research area. Live Virtual Machine (VM) migration, and server consolidation technology along with appropriate resource allocation of users' tasks, is particularly useful for reducing power consumption in cloud data centers. In this article, the authors propose algorithms which mainly consider live VM migration techniques for power reduction named "Power_reduction" and "VM_migration." Moreover, the authors implement dynamic scheduling of servers based on sequential search, random search, and a maximum fairness search for convenient allocation and higher utilization of resources. The authors perform simulation work using CloudSim and the Cloudera simulator to evaluate the performance of the proposed algorithms. Results show that the proposed approaches achieve around 30% energy savings than the existing algorithms.

## KEYWORDS

Dynamic Scheduling of Servers, Green Cloud Computing, Live Migration of Virtual Machine, Resource Allocation, Search Algorithms, Server Consolidation, Virtualization

## 1. INTRODUCTION

Cloud computing is evolving as a new standard of comprehensive distributed computing. It has moved away the computation from home PCs and small organizations to large-scale data centers and made it advantageous for consumers and IT organizations by chunking huge amount of capital investments. Cloud is offering cost-effective solutions to almost all types of large scale computations by letting users to access scalable remote resources (e.g. servers, storage, networks, applications etc.) at any time, from anywhere, on-demand basis and also on pay-per-use basis.

This ever-proliferating demand of cloud computing has led the cloud data centers to grow rapidly. Consequently, it is now leading to a concerning issue of increasing amount of power consumption of cloud resources (Duy et al., 2010; Beik, 2012) and excess carbon footprint in the environment.

DOI: 10.4018/IJCAC.2019010105

It has been measured that power consumption of worldwide data centers has increased almost ten times over the past decade (Priya et al., 2013). The cost of electricity for keeping the computing resources 24/7/365 alive and continuously cooling them is thus evidently booming. Hence, there is indubitably a need of planning up for energy efficient and environment-friendly cloud architecture which is called "Green cloud computing".

Green computing could be characterized as the method of designing, composition, utilizing, and arrangement of servers, computers and correlated subsystems proficiently and effectively with no or negligible impact on the environment (Buyya et al., 2010; Caydar et al., 2010). Green computing can be accomplished by virtualization and server consolidation technology (Buyya et al., 2010; Jain et al., 2013; Siddiqui, 2013; Wadhwa & Verma, 2014; John, 2014; Tiwari, 2011, Kumar & Kiruthiga, 2014; Wadhwa & Verma, 2014; Gowri & Harikrishnan, 2014), balancing load appropriately among resources (Caydar & Alagoz, 2012; Green Grid Industry Consortium, 2007; Rassmussen, 2007; Jain et al., 2013; Siddiqui, 2013; Wadhwa & Verma, 2014; John, 2014), energy management of storage (Rassmussen, 2007; Kumar et al., 2014),managing the CPU's power consumption (Siddiqui, 2013; Wadhwa & Verma, 2014; John, 2014; Tiwari, 2011; Kumar et al., 2014; Wadhwa & Verma, 2014), power management of networking resources (Arthi & ShahuHamead, 2015; Atrey et al., 2013), etc.

Virtualization is a technique where many Operating Systems (OSs) and software applications can be easily run on just one physical server or "host" by the help of hypervisor (Garg & Buyya, 2012; Caydar & Alagoz, 2012; Green Grid Industry Consortium, 2007; Rassmussen, 2007; Jain et al., 2013; Kumar & Kiruthiga, 2014; Richariya & Motwani, 2014). Live Virtual Machine Migration (VM) facilitated by virtualization can aid in balancing load, allowing extreme responsiveness and evading hot-spots in data centers thus decreasing power ingestion (Caydar & Alagoz, 2012; Siddiqui, 2013; Mastelic et al., 2015). Server Consolidation increases resource exploitation by consolidating several VMs located on various under-utilized servers onto a single server; so, it turns off unused servers and reduces energy consumption (Green Grid Industry Consortium, 2007; Wadhwa & Verma, 2014; Tiwari, 2011). Power consumption can also be reduced by following proper resource allocation and utilization techniques (Wadhwa & Verma, 2014; Hameed et al., 2016). This paper concentrates on using these three technologies for reducing power consumption of resources in cloud data centers that can lead to green IT (Asadi & Dahlan, 2017).

According to the Cisco Global Cloud Index the traffic of global data center might touch 15.3 ZB by the year of 2020, with 92% of all workloads being performed in the cloud (Cisco Global Cloud Index, 2016). It has also projected that, at the end of 2015, the number of hyper-scale data centers would face a growth by 226% from 259%, so facing a huge increase to 485% by the year 2020 (Cisco Global Cloud Index, 2016). In the nick of time, this concerning issue has already caught hold of the consideration of the researchers worldwide. Several efforts have been put for adopting green cloud computing, but the problem of huge power consumption is still towards an upswing due to the lack of proper power management and energy minimization scheme.

In this research work, firstly we represent our system model based on virtualization technology and discuss the working mechanism of a single VM. Accordingly, we propose algorithms named "Power_reduction" and "VM_migration" which targets at lowering down the power consumption in cloud data centers. Next, we introduce a dynamic scheduling approach of resources (e.g. servers) based on sequential search, random search and maximum fairness search for allocating and utilizing resources efficiently. For assessing the performance of the proposed algorithm, we will carry out essential simulation works.

The following of the paper is arranged as follows. In Section 2, we study the related works. In Section 3, we present the system model based on VMs in data center, our proposed algorithms "Power_reduction" and "VM_migration and finally the searching algorithms for dynamic scheduling of servers. In Section 4, we accomplish simulation works and present the results. Section 5 shows the conclusion and our future work.

## 2. RELATED WORKS

In view of the fact that, the concept of implementing Green cloud computing has been in much limelight now-a-days, many researchers have put efficient contributions to achieve this aim by proposing energy saving models, developing energy aware cost models, managing dynamic scheduling of workloads etc. However, the $CO_2$ emission by cloud data centers is still following an upsurge.

For reducing the energy consumption in cloud data centers, Truong et al. (2010), have proposed the implementation of a green scheduling algorithm which is associated with neural network predictor. The predictor calculates the peak load by predicting the load from time t to the time it takes to restart. The number of server state is selected as stated by the peak load. Let's say, $N_n$ and $N_o$ is the number of necessary servers and servers in ON state respectively. They said, the servers in OFF state are chosen and restarted if $N_n > N_o$; the servers in ON state are chosen and shut down if $N_{nselect} < N_o$.

Again, Rasoul Beik et al. (2012), suggested an energy aware layer in software architecture which determines the amount of consumption of energy in cloud data centers and services are delivered to the users maintaining energy efficiently.

Another cloud computing metrics for branding the green cloud computing have been offered by Bhanu Priya et al. (2013), they have discussed varieties of energy models for reducing the $CO_2$ emission for making cloud greener. They said, three prime factors can make any cloud greener, they are – virtualization, distribution of workload and automation of software.

The authors, Kliazovich et al. (2012) have mainly focused on the distribution of workload among all the resources of data center and measured energy consumption in packet level. They performed the simulation at three levels using NS2 simulator and "CloudSim" - "two-tier, three-tier, and three-tier high-speed architectures of data centers".

A model is anticipated by Kaur and Singh et al. (2013), for measuring the wasted amount of energy; this model includes several data fields, analysis and records of data etc. associated with the virtualization technology to initiate green cloud computing leading to a healthy environment.

Fumiko Satoh et al. (2013), have suggested using a sensor management method with an enhanced tool for VM allocation which indicates a saving of almost 30% energy in simulation results.

Yamini et al. (2012), have presented the important methodologies of green cloud computing like - consolidation of servers and utilization of resources, material recycling, using virtualization, implementing power management techniques etc. in an exquisite way. The results shown in this work have shown the success in saving of electricity in gigantic cloud data centers. Also, Beloglazov and Buyya et al. (2010), have mainly put emphasis on using the technique of virtual machine and the dynamic reallocation technique for VMs and closing off the unused servers which consequences into substantial energy saving cloud data centers.

Moreover, Nimje et al. (2013), have also spoken about the virtualization concept included with a hypervisor environment (a hypervisor works as a security tool), since it mitigates the loads of servers and makes a proper distribution of workloads, controlling and provision of resources obtainable in a modest fashion.

Besides, Buyya et al. (2011), have put their excellent efforts in modeling a carbon aware green cloud architecture which entails two categories of directories titled as green offers and carbon emission where they offer and use the Green services from both the providers and users. According to the least amount of $CO_2$ emission, green brokers grasp the services from the directory of green offers.

As discussed above, we see that all existing approaches both have some positive and some negative sides. An architecture for green cloud proposed by Buyya et al. (2010), has shown the implementation of green computing by the use of $CO_2$ emission directory, which chooses services with least amount of carbon emission possible; hence the consumption of energy will also decrease since both are dependent on each other. However, the weakness is that there are some other factors that could have been taken into consideration besides $CO_2$ emission and energy for instance- Security, service provisioning along with Quality maintenance etc. Hulkary et al. (2010), have discussed these factors along with some other factors e.g. - services are first explored on the private cloud, then on public cloud. It has been found that this approach saves time as well as offers better results than the architecture proposed by Buyya et al. (2010). But the major flaw which lies in it is that the manager is the only and also the central communication point, crash of which might destroy the communication path of the entire system. Again, the decision-making process of the manager is completely manual not intelligent. However, in our work the proposed system model shows a Service demand processor which assigns tasks to servers dynamically according to some pre-processed data; there's no need of manager's manual operations. These are some benefits and drawbacks which are found in the existing designs and which could be developed further in future works.

In, Azad et al. (2017) utilized the task scheduling using cultural and ant colony optimization algorithm that provides better energy-efficiency in cloud environment. It is quite important for data centers in order to achieve greener environmental aspects and reduce power consumption Farahnakian et al. (2015) used ant colony optimization as well to reduce energy consumption while performing VM consolidation and maintaining QoS. Same authors discussed VM consolidation in another paper (Farahnakian et al., 2015) that is based on utilization aware prediction and maintains service level agreement in cloud. In another recent approach, Pawlish et al. (2015), designed a flexible decision support system (DSS) that delivers energy efficient solutions for today's data centers. To determine the carbon footprint left by data centers every year, authors examined different parameters and designed the DSS based on data mining. The data mining process is basically based on decision trees and case-based reasoning. The paper basically tried to find out parameters and their real-life values so that it becomes easy to mitigate the power consumption.

Security and availability of data is another important aspect that should be considered in greener cloud approaches. For example, distributed denial of service (DDoS) and cross site scripting (XSS) can affect the performance of virtual machines and networks. As a result, implementing different security measurements need more power to operate in data centers. In order to mitigate the effects, different approaches have been proposed so far. Deepali et al. (2017) proposes a model where by applying a service cloud request algorithm in fog computing can lessen most of the DDoS attack effects and provide efficient utilization of resources. Author also discusses a detailed survey on different related aspects in (Chaudhary et al., 2018).

On the other hand (Gou et al., 2017) discussed several security approaches in cloud and Gupta et al. (2016), addressed XSS attack mitigation approach through identifying the XSS attack vectors. And thinking about both security and optimization in storage, tokenization can perform a better than encryption (Roy et al., 2017).

As a vast amount of data is being generated every single day, managing data is another important aspect to follow towards green computing. Kim et al. (2018) introduces a multilayered information analysis method for crowdsourcing to generate specific topic groups for distributed topics. Gupta et al. (2017) refers smart grid as a better solution that optimizes resources and prevent different cyber-attacks targeting the cloud. The primary potentiality of this new field is intended to optimize smart power management system and to protect data centers from DDoS or other attacks that affects the performance of cloud.

In a recent approach, Long et al. (Long et al., 2016), discussed about load balancing and they proposed a power efficient immune clonal optimization algorithm based on that. It outperforms other clonal section algorithms in terms of solution and cost. Cloudlet (Whaiduzzaman et al., 2014) seems to be an excellent approach as well that is used primarily for mobile cloud computing. Gai et al. (2016) developed a cloudlet-based model to cut off additional waste of energy created by the restrictions in bandwidth and device capacity. Meanwhile Reguri et al. (2016) proposed an energy efficient VM migration scheme that validates the SLA in with significant expectance. Some research also focuses on hardware or device level works. For example, Wu et al. (2014) developed a scheduling algorithm for data centers that is based on a dynamic voltage frequency scaling technique.

Moreover, based on the review discussion and future implications in (Patel et al., 2015; Jing et al. 2013), in this work, we have tried to measure the total power consumption using necessary equations and accordingly put some conditions to migrate the extra workload to the target VM of the next server chosen by our unique dynamic hierarchical server scheduling approach. According to the algorithm and conditions, the unnecessary servers will be put to sleep mode resulting in a saving of energy and implementing green cloud computing in cloud data centers.

## 3. SYSTEM MODEL AND PROPOSED ALGORITHMS

### 3.1. System Model

All types of cloud services are stored in remote servers, which are offered to clients / customers and accessed via a Service demand processor when they need. Submitting requests for services, granting those requests and after that handling those requests successfully are efficiently managed by that processor. The type and complications of the service request is modified via a special type of cloud manager for providing best services to customers. It manages the resources like – storages, servers, applications, hardware, networks etc. and maintains database about those centrally and accordingly takes the decision of modification. Transportation cost between customer and host machine, network bandwidth, expectancy of service etc. statistical data is retained as pre-processed data. There is also an activator engine that reads those pre-processed data and accordingly either initiates live VM migration or keeps alive the current host machine or shuts down it. Figure 1 shows the system model.

In this work, firstly, we are going to propose an algorithm based on the calculation of static and dynamic power consumption; secondly, in keeping with some predefined criteria, the theory of live migration will be applied for migrating the data from one virtual machine to another host machine and lay the source virtual machine in sleep mode resulting in the reduction of power consumption. Finally, a unique dynamic scheduling of servers will also be shown which utilizes each server fully according to its capacity and hierarchically handovers the incoming load to other servers. Therefore, servers aren't unnecessarily kept switched on again resulting in saving of energy.

### 3.2. Proposed Power Reduction Algorithm: "Power Reduction"

The consumed power by any machine can be categorized as static and dynamic power consumption, where the power consumed by it during its idle state and active state are called static power and dynamic power respectively. Static power is that fixed amount of power, which will always be consumed by any machine if it's in ON but idle state; on the other hand, dynamic power consumption is the power drawn when the machine is in ON state and also performing some tasks and this power consumption always varies according to the workloads being assigned to the machine. Accordingly, the net or total energy consumed by the system can be shown by the following equation:

Net energy ( $\mathcal{N}$E ) = Static energy ( $\mathcal{S}$ ) + Dynamic energy ( $\mathcal{D}$ )

$$\mathcal{N}\mathrm{E} = vi\cos_{\varphi} T + \sum_{i=1}^{n} p_i t_i$$

Where,

$vi\cos_{\varphi}$: static power consumed by the system (in a single phase)

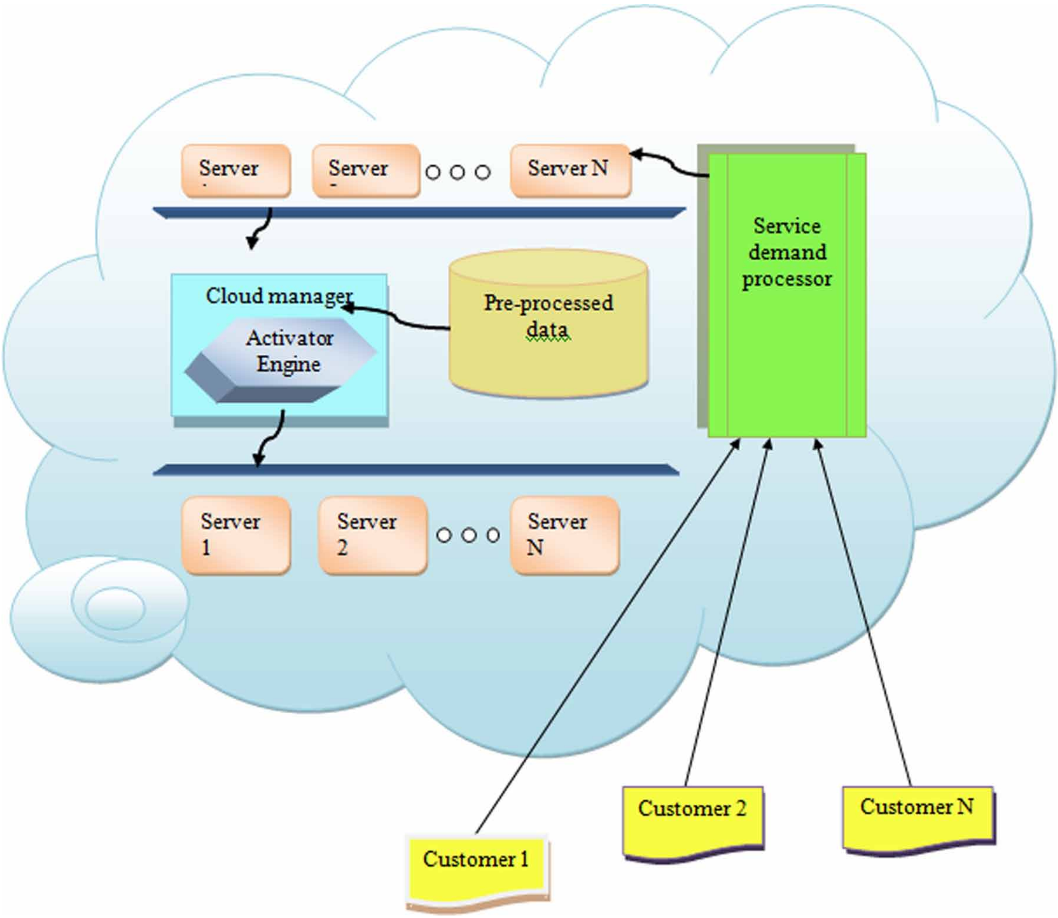
$T$ : the time duration of the system while being ON but idle

$p_i$ : the dynamic power drawn by the system being ON and active

$t$ : the time required for the execution of the task

$i$ : the number of tasks being executed by the server in time $t$

Figure 1. System model




($\mathcal{D}$) - dynamic energy is denoted by $\sum_{i=1}^{n} p_i t_i$

It is not easy to anticipate the dynamic power consumption of any machine since it has to be determined by numerous factors for instance, extent of work load, scheduling of tasks, sum of instructions mandatory for completing any task etc. Therefore, a joule meter software could be used to measure $p_i$.

Let us assume a threshold value, for instance, if the net power being consumed by the machine remains between the range of 115-125 watts, it would be considered energy efficient. Why we have chosen this power as a threshold value will be discussed in the simulation work in the following chapters. If the net power crosses this threshold value, then the live migration of source VM to target VM will be performed in order to turn off the dynamic power consumption of that VM and resulting in a significant amount of saving of energy.

Thus, the proposed "Power_reduction" algorithm for reducing the power consumption of the machine goes as follows:

Statement of variables:

$w_1, w_2, w_3$ : reading of power

$\mathcal{P}_{\mathcal{N}}$ : net power

$\mathcal{P}_{\mathcal{D}}$ : dynamic power consumed by machine

$\mathcal{P}_{\mathcal{S}}$ : static power consumed by machine

$\mathcal{P}_{\mathcal{D}_h}$ : dynamic power consumed by host machine

$\mathcal{P}_{\mathcal{D}_{vm}}$ : dynamic power consumed by VM

$\mathcal{P}_{\mathcal{S}_h}$ : static power consumed by host machine

**Algorithm 1** Power Reduction Algorithm

```
System Initialization
Read the value w1 from watt-meter.
PS ← w1                              ▷ VM and host both are in idle state
Read the value w2 from watt-meter
PN ← w2                              ▷ VM and host both are in active state
if PN > Threshold value then
    Perform Live Migration of data from VM to host machine
    PS = P(Sh)
    PD = P(Dh) + P(Dvm)
    Read the value w3 from watt-meter     ▷ source VM is in sleep
mode leading to the host in active state
    PN ← w3
    PN = PS + PD
else
    No migration
```

Since, the servers having less workload than a particular threshold value thus must be put to sleep mode or shut down in order to save energy, thus, some pre-processed data is required to identify them.

Let us consider that there are α numbers of data centers and β number of servers in each of the $\alpha^{th}$ data centers.

$$\partial \mathrm{C}enter \in \left\{\partial \mathrm{C}enter_1, \partial \mathrm{C}enter_2, \ldots\ldots\ldots, \partial \mathrm{C}enter_\alpha\right\}$$

$$\zeta \in \left\{\zeta_1, \zeta_2, \ldots\ldots\ldots, \zeta_\beta\right\}$$

$$\mho \in \left\{\mho_1, \mho_2, \ldots\ldots\ldots, \mho_m\right\}$$

$$\tau \in \left\{\tau_1, \tau_2, \ldots\ldots\ldots, \tau_n\right\}$$

$$\mu \in \left\{\mu_1, \mu_2, \ldots\ldots\ldots, \mu_l\right\}$$

$$¥ : \mho \rightarrow \mu$$

Where,

$\partial\mathfrak{C}enter$ : is the set of all the datacenters

$\zeta$ : is the set of servers in those datacenters

$\mho$ : is the set of VM instances running in each server

$\tau$ : is the set of time period at which services are requested by customers

$\mu$ : is the set of running services

¥ : is a mapping function such that, each service would be executed by each VM

$\partial\mathfrak{C}enter_{pos} \times \mathfrak{C}ust_{pos} \rightarrow dist$ .

$dist$ function is used to find the distance between the server and customer via measuring their distances. When a customer requests a service, the server which is situated closest will handle it. The nearest server would be identified via some predefined parameters like measuring transportation cost and distance between client and servers.

$$\mathcal{C}on\mathcal{F}act = \frac{number\,of\,customers \times amount\,of\,memory\,used}{\zeta\_\mathcal{U}tilization}$$

The power consumption or utilization factor $\mathcal{C}on\mathcal{F}act$ for any VM instance in a particular server in a particular data center is the product of number of customers demanding services and amount of memory or storage used for a client which is divided by the amount of memory or storage of a particular service of VM in the server, that is server utilization factor $\zeta\_\mathcal{U}tilization$ where,

$$\zeta\_\mathcal{U}tilization = Mem_{size}\left(\partial\mathfrak{C}enter\ ,\zeta\ ,\mho\right)$$

And the number of customers demanding a particular instance of time can be shown as:

$$number\,of\,customers = \sum_{l \in \partial\mathfrak{C}enter_{pos}} \left(customers\left(\tau\ ,\partial\mathfrak{C}enter\ ,\zeta\ ,\mho\ ,l\right)\right)$$

In the stage of actual implementation, live VM migration is made according to the preprocessed data.

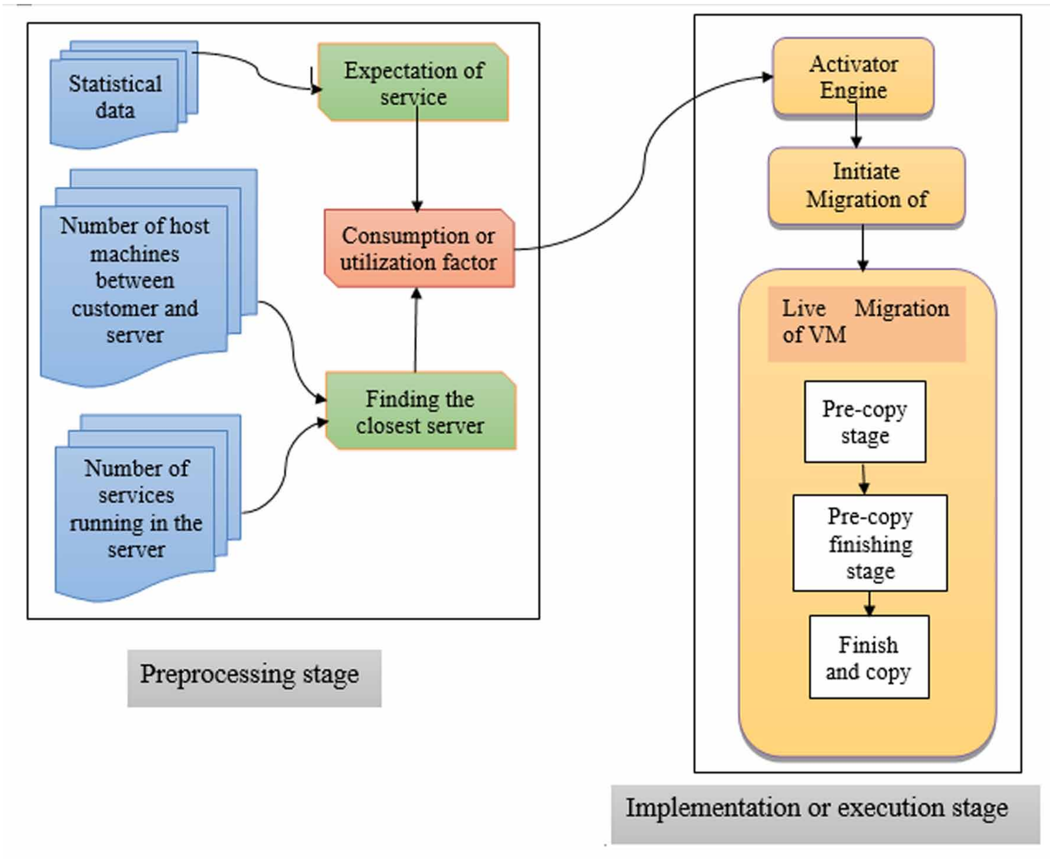
Activator engine states the preprocessed data to either initiate or shut down the server. The number of customers requesting service actually defines the number of VMs to provide service to them. Therefore,, the number of clients requesting VMs at any particular time from any particular globular region $\rho_n$ .where all regions are supposed to be the members of the set $\rho \in \left\{\rho_1, \rho_2, \ldots\ldots\ldots, \rho_n\right\}$, can be described as follows:

$$\sum_{\partial\mathfrak{C}enter \in \left\{\partial\mathfrak{C}enter_1\ ,\partial\mathfrak{C}enter_2\ ,\ldots\ldots\ldots,\partial\mathfrak{C}enter_\alpha\right\}\ ,\ \zeta \in \left\{\zeta_1, \zeta_2, \ldots\ldots\ldots, \zeta_\beta\right\}\ ,\partial\mathfrak{C}enter_{pos} = \rho_n} == customers\left(\tau\ ,\partial\mathfrak{C}enter, \zeta\ ,\mho\right)$$

Now, a diagram of the working mechanism of single VM is shown below in Figure 2:

Now, both the particular location and the number of customers who are retrieving the $\mho_1$ at a particular time $\tau$ in $\partial\mathfrak{C}enter_\alpha$ is given as:

**Figure 2. Diagram of the working mechanism of a single VM**




$$customers_{\mho_1} = \left\{ (n,l) \middle| = \left( customers\left( \tau, \partial Center, \mho, l \right) \right) \& \& l \in \partial Center_{pos} \right\}$$

And only the number of customers accessing the $\mho_1$ at a particular time $\tau$ in $\partial Center_{\alpha}$ is given as:

$$customers\_number_{\mho_1} = \left\{ n \middle| (n,l) \in customers_{\mho_1} \right\}$$

$$\nabla_{max,\mho_1} = \max\left( customers\_number_{\mho_1} \right)$$

$$\varnothing_{max,\mho_1} \leftarrow \partial Center_{pos}$$

Where, $\left( \nabla_{max,\mho_1}, \varnothing_{max,\mho_1} \right) \in customers\_number_{\mho_1}$ ; $\nabla_{max,\mho_1}$ is the maximum or highest count of the customers requesting VM which are running in a particular server in a particular data center in a particular time interval and $\varnothing_{max,\mho_1}$ is the location of VM where maximum or highest number of customers are requesting for retrieving their services.

When the procedure of the activator engine is finished, then the VMs within a particular server can be transferred from the source host machine to another target host machine with higher server utilization factor and least transfer cost.

### 3.3. Proposed Live Virtual Machine (VM) Migration Algorithm: "VM_migration"

The 7th step of the "Power_reduction" algorithm mentions about performing live migration of VM, thus the following algorithm discusses the conditions accordingly:

**Algorithm 2** Live Virtual Machine (VM) migration algorithm: $VM_{migration}$

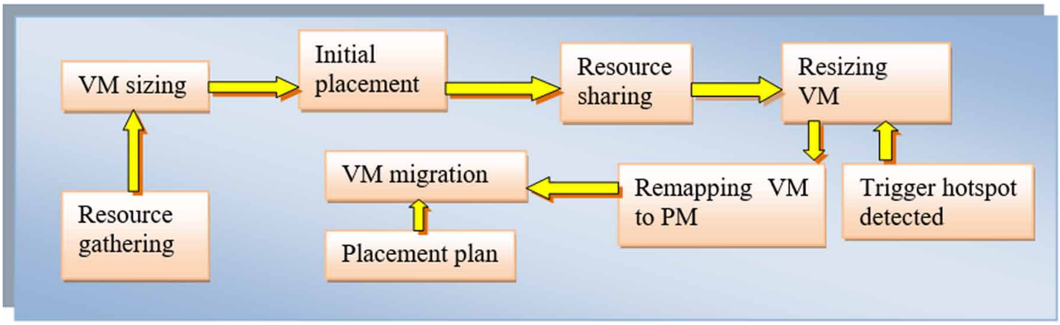
```
Find location = max_customers ℧ at time τ
Find a ℧_target running similar service as ℧_source;
if ℧ ≠ δCenter_pos then
    Calculate ConFact (℧_source) and ConFact (℧_target)
    if ConFact (℧_source) + ConFact (℧_target) ≥ 1 then
        No migration
    else if ConFact (℧_source) ≤ 3/4 (ConFact (℧_target)) then
        Migrate ℧_source to ℧_target
        Put ℧_source into hibernation or sleep mode ▷ saving energy
    else{ ConFact (℧_source) + ConFact (℧_target)} ≤ 0.3
        Migrate to ℧_m
        Put ℧_source and ℧_target into hibernation or sleep mode ▷ saving energy
```

### 3.4. Explanation of "VM_migration" Algorithm

As shown in the diagram of the working mechanism of a single VM, there are three stages called pre-copy stage, pre-copy finishing stage and finish-and-copy stage. Let us suppose that, $\mho_{source}$ and $\mho_{target}$ run the similar services. Let us think that $Con\mathcal{F}act\left(\mho_{source}\right) = 0.15$ and $Con\mathcal{F}act\left(\mho_{target}\right) = 0.70$. Then, the service running at $\mho_{source}$ must be transferred to $\mho_{target}$ because $Con\mathcal{F}act\left(\mho_{source}\right) \leq {}^{3}\!/_{4}\left(Con\mathcal{F}act\left(\mho_{target}\right)\right)$

During pre-copy stage at a particular time period $\tau_n$, all the memory pages of $\mho_{source}$ are copied to $\mho_{target}$. The customers who are retrieving services from $\mho_{source}$ are migrated to $\mho_{target}$ without any loss of the current implementation. Now, the $Con\mathcal{F}act\left(\mho_{target}\right)$ would be 0.85. Thus, the energy consumption of the server running $\mho_{source}$ would be reduced since the VM is migrated to $\mho_{target}$. Simultaneously, $\mho_{source}$ would be getting new requests from more customers. During pre-copy finishing stage, at time period $\tau_{n+1}$, let us suppose that $\mho_{source}$ is busy in getting new requests from more customers for that particular service. But as $\mho_{target}$ is still busy in handling the requests received in time period $\tau_n$, thus, it won't receive new requests from $\mho_{source}$. During finish-and-copy stage, at time period $\tau_{n+2}$, the dirty memory pages of $\mho_{source}$ which were made at time period $\tau_n$, is copied to $\mho_{target}$. So, now the $Con\mathcal{F}act\left(\mho_{source}\right)$ is supposed to be zero because it is not able to provide the required utilization factor to the incoming service requests at time period $\tau_{n+1}$ and thus $\mho_{target}$ won't be accepting new requests as it is in its peak workloads (as it is already handling its current requests and also the requests which came from $\mho_{source}$ at time period $\tau_n$). Thus, $\mho_{source}$ would be put into hibernation or sleep mode in order to reduce unwanted power consumption. In this way, green cloud computing can be implemented, and energy consumption can be minimized in cloud datacenters.

Figure 3. Steps for server consolidation technique




## 3.5. Server Consolidation Technique

Virtualization technology eases the enhanced use of resources using the "server consolidation" technique; it comprises of merging workloads of several physical servers into a single physical server for increasing the utilization of resources (Jain et al., 2013; Siddiqui, 2013; Wadhwa & Verma, 2014; John, 2014; Tiwari, 2011; Kumar & Kiruthiga, 2014). Advantages are -decrease of total physical servers used, minimization of data center space requirements and the server sprawl etc. (Atrey, Jain & Iyengar, 2013; Ahmad et al., 2015). The steps for server consolidation (Figure 3) are shown as follows:

- Firstly, the placing of VMs on PM (Physical Machine) and the capacities of VM (VM sizing) which is essential for running applications, are determined based on their requirements
- Next, the process of monitoring and profiling the utilization of resources for detection of hotspots (overloaded and under loaded PMs) is performed
- Lastly, the VM is resized and remapped to another PM

The placement of VM and its migration is the backbone to server consolidation process. The challenges are to find the proper VM- to-PM mappings for minimizing the standard cost function, detection of dynamic hotspots and performing VM migration with negligible service downtime and resource consumption as possible during migration (Varasteh & Goudarzi, 2017; Reguri et al., 2016; Asadi & Dahlan, 2017; Duan et al., 2017; McWhorter & Delello, 2016).

## 3.6. Proposed Method for Dynamic Scheduling of Servers

The uniqueness of our proposed approach lies in the way the virtual machines are chosen dynamically. We've used a hierarchical selection technique for choosing the servers, which in turn will choose the VMs by sequential or random or maximum fairness selection. The scheduling algorithm is discussed below:

Let's say, there are 5 cluster of servers (i= 1, 2, …, 5 and no. of servers in each cluster is 1, 2, …, N) in a datacenter. According to configurations, speed, time for processing, performance etc. criteria, servers are sorted and clusters are created. When any user request comes, the task is assigned to the servers of the 1st cluster. Now, the distribution of tasks among the servers of any clusters may follow sequential or random or maximum fairness algorithm.

**Algorithm 3** Sequential or random or maximum fairness algorithm

**while** $USER\_DEMAND == True$ **do**
  **for** (i=1; i¡=5; i++) **do** ▷ i is no. of clusters
    **if it's sequential search then**
      For 1,2,...,N servers of a cluster initially $TARGET\_SERVER = S1$;
      Assign the first task to $TARGET\_SERVER$
      $E_{peak}\ (TARGET\_SERVER) = \sum_{a=1}^{k} E_{task_a}$ ▷ suppose, each server can handle (1,2,...,k) no. of tasks
      check When any new task arrives
      **if** $E_{task_a)} \le E_{peak(TARGET\_SERVER)}$ **then**
        Assign it to the current $TARGET\_SERVER$
      **else**
        $TARGET\_SERVER = TARGET\_SERVER + 1$
      $\Delta E_{total} = E_{S1} + E_{S2} + ..... + E_{TARGET\_SERVER}$ ▷ saving energy
    **else if it's random search then**
      any m servers are randomly chosen from for $1, 2, ..., N$ servers of a cluster for performing tasks
      input: NumIterations, TaskSize
      initialization: $SearchSpace = m$ no. of servers among $N$ servers
      output ← $S_{BEST}$
      **for** ($iteration_j$ ∈ NumIterations **do**
        $S_{candidate_b} \leftarrow RandomSolution\ (TaskSize, SearchSpace)$
        **if** $E_{S_{candidate_b}} < E_{task_a}$ **then**
          $S_{BEST} \leftarrow S_{candidate_b}$
        **else**( $iteration_j + +$)
      $\Delta E_{total} = \sum_{b=1}^{m} E_{S_{BEST_b}}$ ▷ saving energy
    **else (it's maximum fairness search)**
      suppose, at time t, a fixed number of m servers from for $1, 2, ..., N$ servers of a cluster are performing tasks, let's set a threshold value of energy $E_{threshold}$ for each server in a cluster
      **for** check every server $S_q$ where $q \in (1, 2, ..., m)$ **do**
        **if** $E_{S_q} \le E_{threshold}$ **then**
          assign $task_a$ to $S_q$
        **else**
          assign $task_a$ to any of the least used server from the group $(N - m)$
      $\Delta E_{total} = \sum_{q=1}^{m} E_{S_k} + \sum_{p=m+1}^{(N-m)} E_{S_p}$ ▷ saving energy

## 4. SIMULATIONS AND RESULTS

Mechanism for reducing the power consumption of a system on cloud has been simulated using CloudSim and Cloudera tool and results of the proposed algorithm is shown in following figures. CloudSim has been used to set up a virtual cloud environment and to implement the proposed algorithms. The measurements of power consumption and offering dynamic loads to the servers have been tested using CloudSim and Cloudera.

The following table 1 shows a simulation scenario of a virtualized datacenter environment. For instance, a datacenter might have 1 server or 5 servers or 10 servers (simulation has been tested for PCs with Windows AMD Duel Core 2.7GHz); it has been found that the servers consume least power when they are in hibernation or sleep mode rather than they are kept running but being idle. For 1 PC, the power consumption increases by 97 Watts (100W – 3W) from being in hibernation mode to in idle mode and increases by 25 Watts (125W – 100W) from in idle mode to being fully loaded up

Table 1. A comparison of power consumption of servers in a datacenter in hibernation, idle and active modes

| PCs with Windows AMD Duel Core 2.7GHz | | | |
|---|---|---|---|
| | 1 PC (Watt) | 5 PC (Watt) | 10 PC (Watt) |
| Hibernate | 3 | 5 | 10 |
| Idle | 100 | 166.6667 | 333.3333 |
| 10% | 104 | 173.3333 | 346.6667 |
| 20% | 110 | 183.3333 | 366.6667 |
| 30% | 114 | 190 | 380 |
| 40% | 116 | 193.3333 | 386.6667 |
| 50% | 118 | 196.6667 | 393.3333 |
| 60% | 120 | 200 | 400 |
| 70% | 121 | 201.6667 | 403.3333 |
| 80% | 122 | 203.3333 | 406.6667 |
| 90% | 123 | 205 | 410 |
| 100% | 125 | 208.3333 | 416.6667 |

to its highest capacity. Also, it is seen that the power consumption is increasing gradually only by some few 3 to 4 Watts when the server is being increasingly loaded with tasks (10%, 20%, …., 100%). Thus, according to the proposed algorithm it has been seen that a PC could be utilized 90%-100% to execute tasks if it is active since the power consumption difference would only increase by 25 Watts. Rather than keeping idle, the optimal solution would be to put it into sleep mode because it'll reduce a huge amount of power consumption and save energy. We've used a threshold value of utilization factor of any VM in our algorithm; we said if the utilization factor of any active source VM is less than 3/4$^{th}$of the utilization factor of the target VM then that particular source VM would be put to sleep mode and the task will be migrated to the target VM. Thus, all the active servers would run at their highest capacity and the remaining less utilized ones will be put to hibernation mode, reduce the power consumption and save energy to implement Green Cloud computing. Figure 4 shows **a** comparison of power consumption of servers in a datacenter in hibernation, idle and active modes

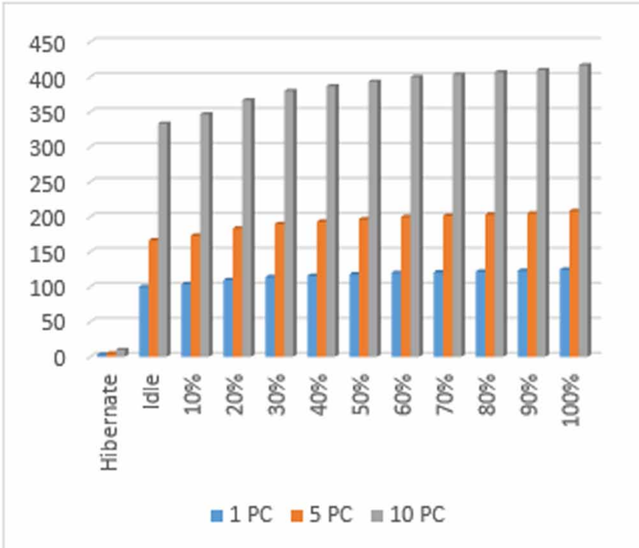
Following table 2, Figure 5 shows the comparison of power consumption of servers in Watts without implementing any algorithm and after applying the proposed algorithm when they are loaded with 50% tasks. It is seen that the amount of total saved power increases with the number of increasing number of servers in total. Thus, the simulation result of the proposed algorithm is satisfactorily reducing power consumption, thus saving energy.

Following table 3, Figure 6 shows the comparison of power consumption of different number of single core and quad core PCs in normal mode (NM) without applying the algorithm and in optimal mode (OP) after applying the proposed algorithm. It is seen that power consumption reduces drastically in the optimum mode which proves the efficiency of the proposed algorithm.

So, the simulation results successfully show that the proposed algorithms reduce around 30% power consumption thus exploiting the aim of Green cloud computing in cloud data centers.

In Sahu et al. (2013), the proposed method is also based on two dynamic (based on application's demand) upper and lower limit of particular threshold values and the algorithm is named as threshold based Dynamic Compare and Balanced Algorithm (DCABA). Based on the up-to-date host load (H_Load), at first the tasks of the overloaded hosts are distributed and secondly the number of active hosts is reduced to lessen their energy consumption. The proposed method is conducted in CloudSim using different upper and lower threshold values of workloads. Since, the live VM migration technique

**Figure 4. A comparison of power consumption of servers in a datacenter in hibernation, idle and active modes**




hasn't been considered there, thus during comparison, our proposed method provided around 20% more energy savings than DCABA. The tabular data is shown in table 4 and the related graphical comparison is shown in figure 7.

Again, in Reguri et al., (2016), four energy and traffic aware placement algorithms have been proposed which are, Least Increased Power with Host Sort and Clustering (LIP-HostSort_VMCL), Best Fit Host with clustering (BFH_VMCL),Best fit VM with Cluster algorithms (BFV_VMCL)and Best Fit Host algorithm(BFH).All these algorithms are based on dynamic migration of VMs considering both energy and communication traffic factors; the clustering of over/under-utilized VMs follow particular equations for energy consumption calculation. The idea of clustering over/under-utilized VMs certainly reduced energy consumption but load balancing of incoming request among all the

**Table 2. (a) The comparison of power consumption of servers without implementing any algorithm and after applying the proposed algorithm when they are loaded with 50% tasks (b) The power consumption statistics in different loads of different no. of servers without implementing any algorithm**

| Load 50 percent Watt | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| No. of servers | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Random | 118W | 158.8W | 198.5W | 238.2W | 277.9W | 317.6W | 357.3W | 397W |
| Proposed | 117W | 156W | 195W | 234W | 273W | 312W | 351W | 390W |
| No. of nodes | | | | | | | | |
| **Load%** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| 70% | 121 W | 161 W | 202 W | 242 W | 282 W | 320 W | 363 W | 403 W |
| 80% | 122 W | 164 W | 205 W | 246 W | 287 W | 328 W | 369 W | 407 W |
| 90% | 123 W | 166 W | 207 W | 249 W | 290 W | 331 W | 373 W | 410 W |
| 100% | 125 W | 168 W | 210 W | 252 W | 294 W | 336 W | 378 W | 417 W |

**Figure 5. The comparison of power consumption of servers without implementing any algorithm and after applying the proposed algorithm when they are loaded with 50% tasks**

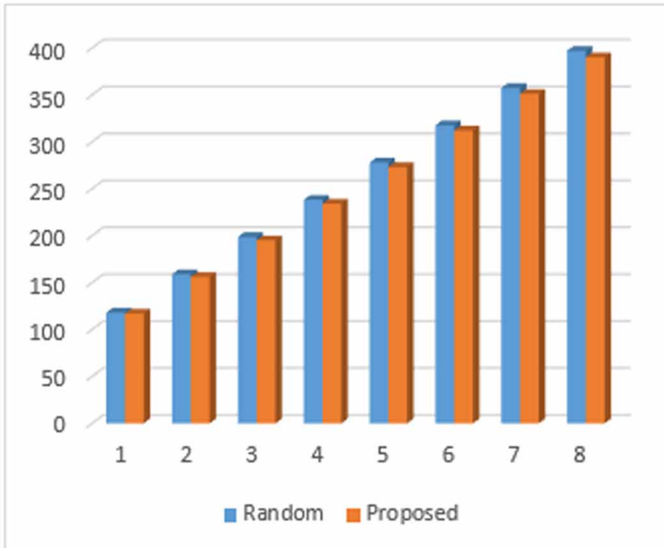
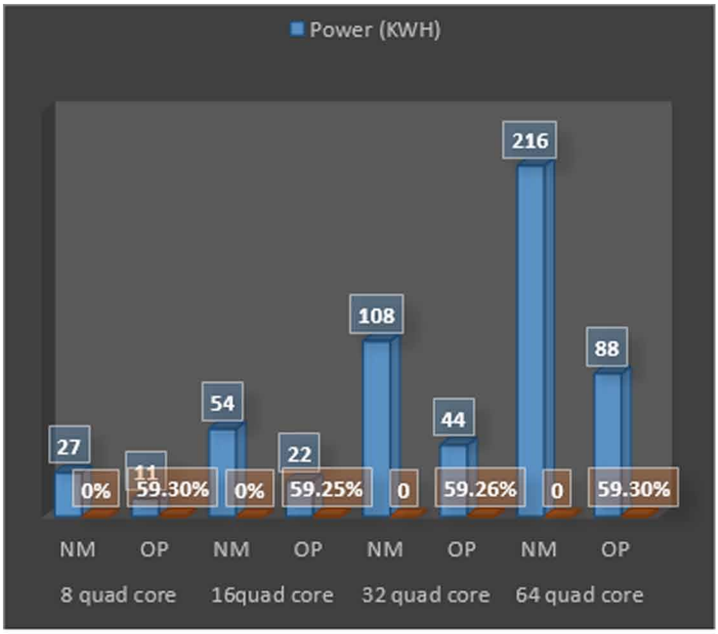


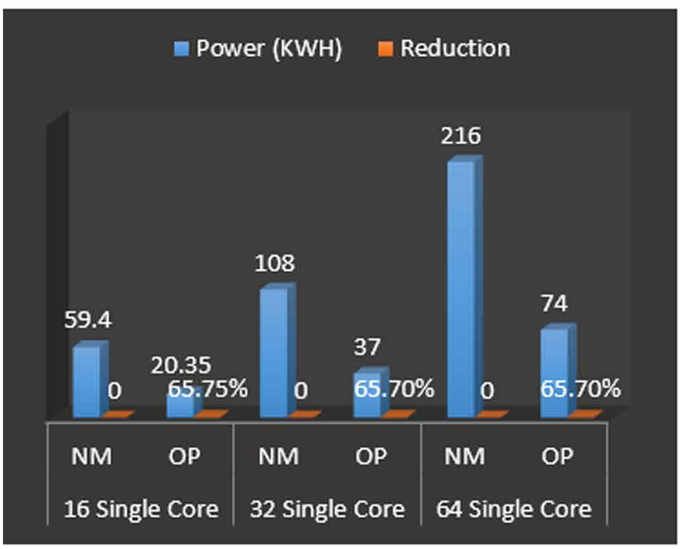

**Table 3. The comparison of power consumption and reduction (%) of single core and quad core PCs in normal mode (NM) and in optimum mode (OP)**

| Datacenter | Mode | Power (KWH) | Reduction (%) |
|---|---|---|---|
| 8 quad core | NM | 27 | 0 |
| | OP (proposed) | 11 | 59.3 |
| 16quad core | NM | 54 | 0 |
| | OP (proposed) | 22 | 59.25 |
| 32 quad core | NM | 108 | 0 |
| | OP (proposed) | 44 | 59.26 |
| 64 quad core | NM | 216 | 0 |
| | OP (proposed) | 88 | 59.3 |
| **Datacenter** | **Mode** | **Power (KWH)** | **Reduction (%)** |
| 16 Single Core | NM | 59.4 | 0 |
| | OP (proposed) | 20.35 | 65.75 |
| 32 Single Core | NM | 108 | 0 |
| | OP (proposed) | 37 | 65.7 |
| 64 Single Core | NM | 216 | 0 |
| | OP (proposed) | 74 | 65.7 |

**Figure 6. The comparison of power consumption and reduction (in %) of (a) quad core and (b) single core PCs in normal mode (NM) and in optimum mode (OP)**




(a)




(b)

VMs in a particular cluster hasn't been properly explored which could've resulted in more savings of energy. Therefore, simulation comparison of these algorithms shows that our proposed method saved around 24.4%, 4.5%, 3.7% and 15.2% more energy than these approaches respectively since it has addressed load balancing among servers. For this simulation experiment, hosts HP ProLiant

**Table 4. Performance comparison of DCBCA and Proposed algorithm in power reduction of host machines while varying number of cloudlets are assigned to different VMs**

| | | Migration of cloudlet (overloaded host) | | | | |
|---|---|---|---|---|---|---|
| **No. of Cloudlet** | **Mode** | **0** | **5** | **10** | **15** | **20** |
| 20 | DCABA | 13.37 | 13.19 | 13.07 | 12.96 | 12.87 |
| | **Proposed** | 14.38 | 14.11 | 13.89 | 13.62 | 13.28 |
| 40 | DCABA | 12.91 | 12.79 | 12.61 | 12.4 | 12.32 |
| | **Proposed** | 13.87 | 13.58 | 13.27 | 12.91 | 12.69 |
| 60 | DCABA | 12.56 | 12.47 | 12.39 | 12.22 | 12.01 |
| | **Proposed** | 13.39 | 13.09 | 12.82 | 12.64 | 12.31 |
| 80 | DCABA | 12.25 | 12.04 | 11.94 | 11.89 | 11.56 |
| | **Proposed** | 12.81 | 12.54 | 12.29 | 12.07 | 11.89 |
| 100 | DCABA | 11.75 | 11.66 | 11.46 | 11.33 | 11.23 |
| | **Proposed** | 12.42 | 12.19 | 11.96 | 11.85 | 11.61 |
| 120 | DCABA | 10.36 | 10.26 | 10.2 | 10.13 | 10.01 |
| | **Proposed** | 11.32 | 11.02 | 10.87 | 10.56 | 10.35 |

Reduction of energy consumption (in percentage)

**Figure 7. Performance comparison of DCBCA and Proposed algorithm in power reduction of host machines while varying number of cloudlets are assigned to different VMs**

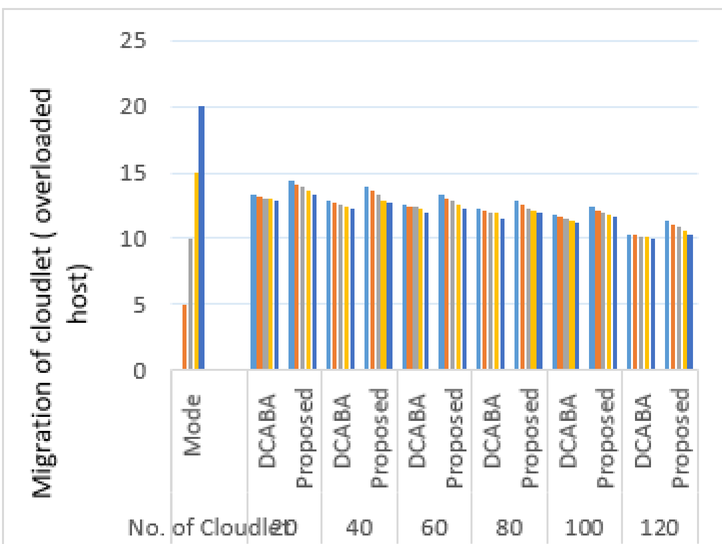


**Table 5. Performance comparison of LIP-HostSort_VMCL, BFH_VMCL, BFV_VMCL, BFH and Proposed algorithm in total power consumption of host machines while different number of VM migration occurs**

| Name of Algorithms | Energy consumption for individual algorithm (KWH) | No. of VM migrations | Total migration energy consumption (in KWH) | Total energy consumption (in KWH) |
|---|---|---|---|---|
| | | | | |
| LIP-HostSort_VMCL | 133.6 | 36 | 25.38 | 158.98 |
| BFH_VMCL | 14.45 | 34 | 39.95 | 54.4 |
| BFV_VMCL | 14.44 | 17 | 8.13 | 22.57 |
| BFH | 96.73 | 17 | 6.45 | 103.18 |
| Proposed | 10.45 | 14 | 5.12 | 15.57 |

ML110 G4 and HP ProLiant ML110 G5 have been used in CloudSim simulator. The tabular data is shown in table 5 and the related graphical comparisons are shown in figure 8.

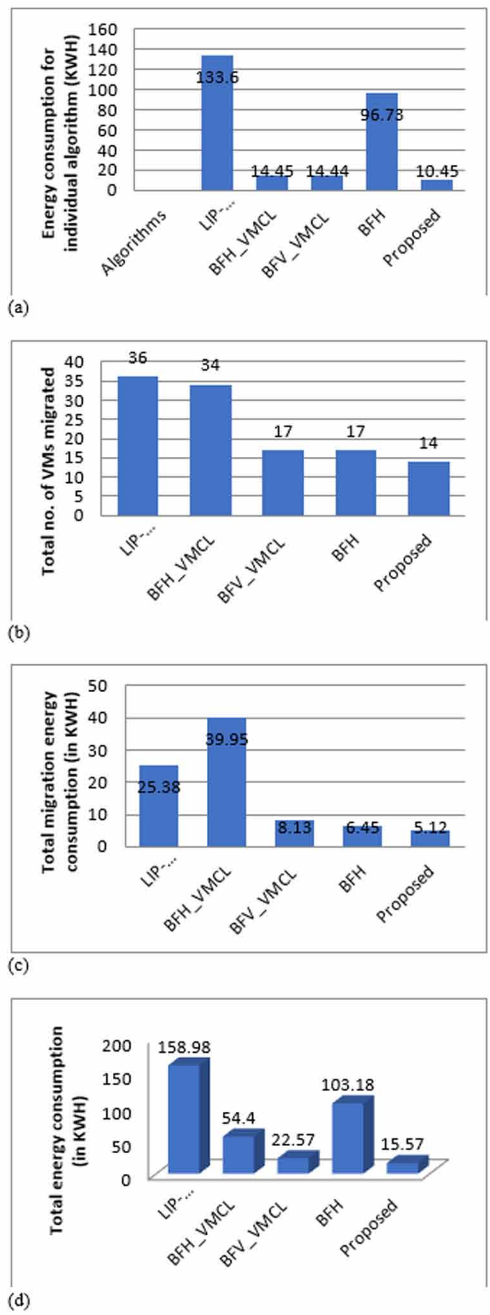
So, comparisons with some existing algorithms show that our proposed algorithm has performed better in terms of saving energy in cloud data centers. It also maintains and restricts the violation of SLA during simulation as lower as possible.

## 5. CONCLUSION AND FUTURE WORKS

In this work, we have proposed live virtual machine migration-based algorithm for decreasing power consumption. Along with this technique, we have adopted the practice of server consolidation for better computing proficiency and lesser power and cooling expenses. Moreover, we have implied dynamic scheduling of resources based on three searching algorithms - sequential search, random search and maximum fairness search, with a view to utilizing the data center resources competently. Then we have performed simulation works and results have shown that our proposed approaches have achieved around 30% energy savings.

In this research, we have considered only software level approaches and algorithms for reducing power consumption. However, there are other hardware level techniques for dropping power. In future, we will work for combining efficient hardware level procedures along with cooling operations with the proposed approach for achieving greener cloud data centers. We intend to utilize fog computing and mobile cloud computing as well that can provide additional service layer and minimize resource allocation in data centers.

**Figure 8. (a) Energy consumption for individual algorithm (KWH) (b) Total no. of VMs migrated (c) Total migration energy consumption (in KWH) (d) Total energy consumption (in KWH)**



## REFERENCES

Duy, T. V. T., Sato, Y., & Inoguchi, Y. (2010, April). Performance evaluation of a green scheduling algorithm for energy savings in cloud computing. In *2010 IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW)* (pp. 1-8). IEEE. doi:10.1109/IPDPSW.2010.5470908

Beik, R. (2012, May). Green cloud computing: An energy-aware layer in software architecture. In 2012 Spring Congress on Engineering and Technology (S-CET) (pp. 1-4). IEEE.

Priya, B., Pilli, E. S., & Joshi, R. C. (2013, February). A survey on energy and power consumption models for greener cloud. In *2013 IEEE 3rd International Advance Computing Conference (IACC)* (pp. 76-82). IEEE. doi:10.1109/IAdCC.2013.6514198

Kliazovich, D., Bouvry, P., & Khan, S. U. (2012). GreenCloud: A packet-level simulator of energy-aware cloud computing data centers. *The Journal of Supercomputing*, *62*(3), 1263–1283. doi:10.1007/s11227-010-0504-1

Kaur, M., & Singh, P. (2013, April). Energy efficient green cloud: Underlying structure. In *2013 International Conference on Energy Efficient Technologies for Sustainability (ICEETS)* (pp. 207-212). IEEE.

Satoh, F., Yanagisawa, H., Takahashi, H., & Kushida, T. (2013, March). Total energy management system for cloud computing. In *2013 IEEE International Conference on Cloud Engineering (IC2E)* (pp. 233-240). IEEE. doi:10.1109/IC2E.2013.46

Yamini, R. (2012, March). Power management in cloud computing using green algorithm. In *2012 International Conference on Advances in Engineering, Science and Management (ICAESM)* (pp. 128-133). IEEE.

Beloglazov, A., & Buyya, R. (2010, May). Energy efficient allocation of virtual machines in cloud data centers. In *2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid)* (pp. 577-578). IEEE. doi:10.1109/CCGRID.2010.45

Nimje, A. R., Gaikwad, V. T., & Datir, H. N. (2013). Green Cloud Computing: A Virtualized Security Framework for Green Cloud Computing. In *Proceeding of the International Journal of Advanced Research in Computer Science and Software Engineering*.

Garg, S. K., Yeo, C. S., & Buyya, R. (2011, August). Green cloud framework for improving carbon efficiency of clouds. In *European Conference on Parallel Processing* (pp. 491-502). Springer. doi:10.1007/978-3-642-23400-2_45

Buyya, R., Beloglazov, A., & Abawajy, J. (2010). Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges. arXiv:1006.0308

Gaea Times. (n.d.). Goals of Green computing. Retrieved from www.tech.gaeatimes.com/index.php/archive/top-10-Cloud-computing-serviceproviders-in-2010

Garg, S. K., & Buyya, R. (2012). Green cloud computing and environmental sustainability. In *Harnessing Green IT* (pp. 315–340). Principles and Practices.

Cavdar, D., & Alagoz, F. (2012, December). A survey of research on greening data centers. In *2012 IEEE Global Communications Conference (GLOBECOM)* (pp. 3237-3242). IEEE. doi:10.1109/GLOCOM.2012.6503613

Green Grid Industry Consortium. (2007). Green Grid Metrics Describing Data Center Power Efficiency. *Technical Committee (White Paper).* Retrieved from http://www.thegreengrid.org/~/media/WhitePapers/Green_Grid_Metrics_WP.pdf

Rassmussen, N. (Ed.). (2007). Electrical Efficiency Modelling of Data Centres (white paper). American Power Conversion.

Jain, A., Mishra, M., Peddoju, S. K., & Jain, N. (2013, April). Energy efficient computing-green cloud computing. In *2013 International Conference on Energy Efficient Technologies for Sustainability (ICEETS)* (pp. 978-982). IEEE. doi:10.1109/ICEETS.2013.6533519

Siddiqui, J. (2013). Green Computing: Protect Our Environment from Computer and its Devices. *Compusoft*, *2*(12), 410.

Wadhwa, B., & Verma, A. (2014, March). Energy saving approaches for Green Cloud Computing: A review. In 2014 Recent Advances in Engineering and Computational Sciences (RAECS) (pp. 1-6). IEEE. doi:10.1109/RAECS.2014.6799608

John, J. (2014). Green computing strategies for improving energy efficiency in IT systems. *International Journal of Scientific Engineering and Technology*, *3*(6), 715–717.

Tiwari, S. (2011). Need of green computing measures for Indian IT industry. *Journal of Energy Technologies and Policy*, *1*(4).

Vinoth Kumar, T., & Kiruthiga, P. (2014). Green Computing-An Eco Friendly Approach for Energy Efficiency and Minimizing E-waste. International journal of Engineering Research, 3(5), 356-359.

Wadhwa, B., & Verma, A. (2014, March). Energy saving approaches for Green Cloud Computing: A review. In 2014 Recent Advances in Engineering and Computational Sciences (RAECS) (pp. 1-6). IEEE. doi:10.1109/RAECS.2014.6799608

Gowri, S., & Harikrishnan, V. (2014). *Green computing: analyzing power consumption using local cooling. International Journal of Engineering Trends and Technology*, 15.

Vinoth Kumar, T., & Kiruthiga, P. (2014). Green computing-an eco friendly approach for energy efficiency and minimizing e-waste. International journal of Engineering Research, 3(5), 356-359.

Arjmandi, M., Balouchzahi, N. M., Raahemifar, K., Fathy, M., & Akbari, A. (2016). Reduction of power consumption in cloud data centers via dynamic migration of virtual machines. *International Journal of Information and Education Technology*, *6*(4), 286–290. doi:10.7763/IJIET.2016.V6.701

Arthi, T., & Hamead, H. S. (2013, April). Energy aware cloud service provisioning approach for green computing environment. In *2013 International Conference on Energy Efficient Technologies for Sustainability (ICEETS)* (pp. 139-144). IEEE.

Richariya, P., & Motwani, A. (2014, August). Energy Efficient computing in wireless sensor network: Application of green computing approach. In *2014 International Conference on Advances in Engineering and Technology Research (ICAETR)* (pp. 1-5). IEEE.

Beloglazov, A., Abawajy, J., & Buyya, R. (2012). Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future Generation Computer Systems*, *28*(5), 755–768. doi:10.1016/j.future.2011.04.017

Dehkordi, Z. G., & Shirazi, H. A. (2014). The Strategic Plan Founded on Efficiency of Using Green Computing Techniques in Data Centers. *Compusoft*, *3*(8), 1038.

Sahu, Y., Pateriya, R. K., & Gupta, R. K. (2013, September). Cloud server optimization with load balancing and green computing techniques using dynamic compare and balance algorithm. In *2013 5th International Conference on Computational Intelligence and Communication Networks (CICN)* (pp. 527-531). IEEE. doi:10.1109/CICN.2013.114

Cisco. (2018). Cisco Global Cloud Index: forecast and methodology (2016-2021) forecast whitepaper (white paper). Retrieved from https://www.cisco.com/c/dam/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.pdf

Atrey, A., Jain, N., & Iyengar, N. (2013). A study on green cloud computing. *International Journal of Grid and Distributed Computing*, *6*(6), 93–10. doi:10.14257/ijgdc.2013.6.6.08

Varasteh, A., & Goudarzi, M. (2017). Server consolidation techniques in virtualized data centers: A survey. *IEEE Systems Journal*, *11*(2), 772–783. doi:10.1109/JSYST.2015.2458273

Wadhwa, B., & Verma, A. (2014, September). Carbon efficient VM placement and migration technique for green federated cloud datacenters. In *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (pp. 2297-2302). IEEE. doi:10.1109/ICACCI.2014.6968597

Ahmad, R. W., Gani, A., Hamid, S. H. A., Shiraz, M., Yousafzai, A., & Xia, F. (2015). A survey on virtual machine migration and server consolidation frameworks for cloud data centers. *Journal of Network and Computer Applications*, *52*, 11–25. doi:10.1016/j.jnca.2015.02.002

Mastelic, T., Oleksiak, A., Claussen, H., Brandic, I., Pierson, J. M., & Vasilakos, A. V. (2015). Cloud computing: Survey on energy efficiency. *ACM Computing Surveys*, *47*(2), 33.

Hameed, A., Khoshkbarforoushha, A., Ranjan, R., Jayaraman, P. P., Kolodziej, J., Balaji, P., & Khan, S. U. et al. (2016). A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems. *Computing*, *98*(7), 751–774. doi:10.1007/s00607-014-0407-8

Murugesan, S. (2010). Making IT Green. *IT Professional*, *12*(2), 4–5. doi:10.1109/MITP.2010.60

Saha, B. (2014). Green computing. *International Journal of Computer Trends and Technology*, *14*(2), 46–50. doi:10.14445/22312803/IJCTT-V14P112

Reguri, V. R., Kogatam, S., & Moh, M. (2016, April). Energy efficient traffic-aware virtual machine migration in green cloud data centers. In *2016 IEEE 2nd International Conference on Big Data Security on Cloud (Big Data Security), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS)* (pp. 268-273). IEEE. doi:10.1109/BigDataSecurity-HPSC-IDS.2016.55

Asadi, S., & Dahlan, H. M. (2017). Organizational research in the field of Green IT: A systematic literature review from 2007 to 2016. *Telematics and Informatics*.

Duan, H., Chen, C., Min, G., & Wu, Y. (2017). Energy-aware scheduling of virtual machines in heterogeneous cloud computing systems. *Future Generation Computer Systems*, *74*, 142–150. doi:10.1016/j.future.2016.02.016

McWhorter, R. R., & Delello, J. A. (2016). Green computing through virtual learning environments. In Professional development and workplace learning: Concepts, methodologies, tools, and applications (pp. 837–864). Hershey, PA: IGI Global. doi:10.4018/978-1-4666-8632-8.ch047

Azad, P., & Navimipour, N. J. (2017). An energy-aware task scheduling in the cloud computing using a hybrid cultural and ant colony optimization algorithm. *International Journal of Cloud Applications and Computing*, *7*(4), 20–40. doi:10.4018/IJCAC.2017100102

Pawlish, M. J., Varde, A. S., & Robila, S. A. (2015). The Greening of Data Centers with Cloud Technology. *International Journal of Cloud Applications and Computing*, *5*(4), 1–23. doi:10.4018/IJCAC.2015100101

Bhushan, K. (2017, August). DDoS attack mitigation and resource provisioning in cloud using fog computing. In *2017 International Conference on Smart Technologies For Smart Nation (SmartTechCon),* (pp. 308-313). IEEE.

Chaudhary, D., Bhushan, K., & Gupta, B. B. (2018). Survey on DDoS Attacks and Defense Mechanisms in Cloud and Fog Computing. *International Journal of E-Services and Mobile Applications*, *10*(3), 61–83. doi:10.4018/IJESMA.2018070104

Gou, Z., Yamaguchi, S., & Gupta, B. B. (2017). Analysis of various security issues and challenges in cloud computing environment: a survey. In Identity Theft: Breakthroughs in Research and Practice (pp. 221–247). Hershey, PA: IGI Global. doi:10.4018/978-1-5225-0808-3.ch011

Gupta, S., & Gupta, B. B. (2016). Enhanced XSS defensive framework for web applications deployed in the virtual machines of cloud computing environment. *Procedia Technology*, *24*, 1595–1602. doi:10.1016/j.protcy.2016.05.152

Kim, M., Gupta, B. B., & Rho, S. (2018). Crowdsourcing based scientific issue tracking with topic analysis. *Applied Soft Computing*, *66*, 506–511. doi:10.1016/j.asoc.2017.09.028

Gupta, B. B., & Akhtar, T. (2017). A survey on smart power grid: Frameworks, tools, security issues, and solutions. *Annales des Télécommunications*, *72*(9-10), 517–549. doi:10.1007/s12243-017-0605-4

Long, Z., & Ji, W. (2016). Power-efficient immune clonal optimization and dynamic load balancing for low energy consumption and high efficiency in green cloud computing. *Journal of Communication*, *11*, 558–563.

Whaiduzzaman, M., Gani, A., & Naveed, A. (2014). *An empirical analysis of finite resource impact on cloudlet performance in mobile cloud computing*. Kuala Lumpur, Malaysia: CEET.

Gai, K., Qiu, M., Zhao, H., Tao, L., & Zong, Z. (2016). Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing. *Journal of Network and Computer Applications*, *59*, 46–54. doi:10.1016/j.jnca.2015.05.016

Farahnakian, F., Ashraf, A., Pahikkala, T., Liljeberg, P., Plosila, J., Porres, I., & Tenhunen, H. (2015). Using ant colony system to consolidate VMs for green cloud computing. *IEEE Transactions on Services Computing*, *8*(2), 187–198. doi:10.1109/TSC.2014.2382555

Farahnakian, F., Pahikkala, T., Liljeberg, P., Plosila, J., & Tenhunen, H. (2015, June). Utilization prediction aware VM consolidation approach for green cloud computing. In *2015 IEEE 8th International Conference on Cloud Computing (CLOUD)* (pp. 381-388). IEEE. doi:10.1109/CLOUD.2015.58

Reguri, V. R., Kogatam, S., & Moh, M. (2016, April). Energy efficient traffic-aware virtual machine migration in green cloud data centers. In *2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS)* (pp. 268-273). IEEE. doi:10.1109/BigDataSecurity-HPSC-IDS.2016.55

Wu, C. M., Chang, R. S., & Chan, H. Y. (2014). A green energy-efficient scheduling algorithm using the DVFS technique for cloud datacenters. *Future Generation Computer Systems*, *37*, 141–147. doi:10.1016/j.future.2013.06.009

Patel, Y. S., Mehrotra, N., & Soner, S. (2015). Green cloud computing: A review on Green IT areas for cloud computing environment. In *2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE)* (pp. 327-332). IEEE. doi:10.1109/ABLAZE.2015.7155006

Jing, S.-Y., Ali, S., She, K., & Zhong, Y. (2013). State-of-the-art research study for green cloud computing. *The Journal of Supercomputing*, *65*(1), 445–468. doi:10.1007/s11227-011-0722-1

Roy, S., Shovon, A. R., & Whaiduzzaman, M. (2017, December). Combined approach of tokenization and mining to secure and optimize big data in cloud storage. In *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)* (pp. 83-88). IEEE. doi:10.1109/R10-HTC.2017.8288912

*Jenia Afrin Jeba has just completed her MSc in Information Technology from IIT, Jahangirnagar University. She is working now in Eastern University, Bangladesh. Her research interests are in Cloud computing, energy minimization, big data, and Fog computing.*

*Shanto Roy received his BSc and MSc in Information Technology, both from the IIT, Jahangirnagar University in 2016 and 2018, respectively. Currently he is working as a lecturer of Dept. of CSE in Green University of Bangladesh where he has been providing academic services since May 2016. The research topics he prefers to study and work on include cloud, fog and edge computing, IoT, data and network security, NV, SDN, and ML, etc. He has been associated with a few projects related to IoT data security in Cloud, E-gov application framework, online patient monitoring, smart education system etc. Now, he is looking forward to study abroad specially to attend in a Ph.D. program.*

*Md.Mahbub-Or-Rashid received his BSc and MSc in Information Technology, both from the IIT, Jahangirnagar University in 2015 and 2016, respectively. Currently, he is working as a lecturer of Dept. of CSE in Daffodil International University of Bangladesh. Previously, he also served as a Research assistant at Access To Information Programme-II, Prime Minister Office's, Bangladesh. His research interest is on Cloud Computing, Big Data and wireless networking. He has already published his first journals on cognitive radio networks.*

*Syeda Tanjila Atik has recently finished her Masters in IT from Jahangirnagar University, Dhaka. She is now working as a lecturer in Daffodil International University of Bangladesh. Her core research interest is cloud computing and telecommunication network. She is also interested in the field of Internet of Things(IoT) and machine learning.*

*M Whaiduzzaman is an Associate Professor in the Institute of Information Technology, Jahangirnagar University. He completed his MSc from London, UK and PhD in Mobile cloud computing from University of Malaya, Malaysia. He has already published in Elsevier, Springer, and IEEE Transactions journals. His research interests include: Cloud, mobile cloud, Fog and vehicular cloud computing. Recently, he was awarded the Elsevier JNCA best paper award in Paris.*