Enhanced Bee Colony Approach for reducing the energy consumption during VM migration in cloud computing environment

Suruchi Talwani ¹, Jimmy Singla ²
Lovely Professional University, Punjab, India

Suruchitalwani14@gmail.com, jimmy.21733@lpu.co.in

Abstract. To achieve virtualization in a cloud environment, resource utilization and energy need to be handled carefully. For this one should have to manage the workload, by distributing the load equally among the node. So that, the resources should be distributed equally among the cloud user and access data anytime from anywhere with minimum energy. In this paper, an enhanced Artificial Bee Colony (E-ABC) approach is presented to minimize overall energy consumption with minimum number of migrations. E-ABC approach migrates the VM from the overloaded host to underloaded hosts and hence save energy. The enhancement of the proposed work is exhibited by showing comparison with the Enhanced Cuckoo Search (E-CS) approach and Ant Colony Optimization technique using MATLAB simulator. Enhancement in the reduction of energy consumption of about 15.45 %, and 17.03 % is observed against E-CS, and existing work.

Keywords: Cloud Computing, Energy Saving, Virtualization, VM Migrations, Artificial Bee Colony, Cuckoo Search Algorithm.

1. Introduction

Cloud computing delivers on demand services in a pay-per-use basic. The application of cloud computing is increases day by day. This might be due to the extensive use of smart devices such as laptop, smart phone, tab and many more. Anyone can access data from anywhere through internet connection. The data is stored in the cloud and later on accessed by the user. With the increase in the number of digital devices the size of the cloud center increases rapidly (Shabeera et al. 2017) [1]. The main motive of cloud service providers (CSPs) is to take full advantage of the available resources and hence increase the profit at minimum lost (Lee 2014) [2]. Although, to deliver more cloud services to their users require more energy. So, energy is also considered as one of the main factors that influence the performance of the cloud system. From the previous study (Zhang et al. 2019) it was analyzed that the average resource utilization is estimated around 30%, whereas, the energy consumed by the idle component is about 70 % of the total available power. Therefore, it is observed that most of the energy is wasted on doing nothing. Consequently, it is not wise to leave the server with small workload [3]. The researchers ((Arianyan et al. 2016; (Gai et al. 2015)) have worked on to detect ideal server using Consolidation techniques [4], Task Mapping Algorithm (TMA) and Efficiency-Aware Task Assignment (EATA) Algorithm [5]. Therefore, for data centers, to minimize energy consumption...
becomes a major concern. The goal is not only to minimize energy consumption, but also considered the environmental regulation along with the agreements between the cloud users and CSPs (Soltanshahi, 2019) [6]. In [6], it is stated that the virtualization technology, helps to save energy and the cloud servers that were designed using such technique attracts the cloud users. An energy aware method has been proposed by (Beloglazov, Anton, and Rajkumar, 2012) to allocate Virtual Machines (VMs) by considering when, which, and where should the VMs need to be migrated [7]. VMs are the virtual resources that works in the same fashion as that of PhysicalMachines (PMs). For cloud services such as Infrastructure-as-a-Service (IaaS), Software-as-a-Service (SaaS) or Platform-as-a Service (PaaS), VMs are important virtual resources to cover the provided cloud applications. With the enormous growth of cloud computing, the distribution of VMs is important in Cloud center that is how to serve multiple users requests with available VMs. In short, how an optimal VM allocation can be achieved to satisfy both cloud users and CSPs (Zhang et al. 2020) [8]. This is one of the important issues that is discussed in this paper. Virtualization plays an important role in cloud computing.

Using this technique, multiple Operating Systems (OSs) can run simultaneously on single PM. The isolation between OSs and the Physical environment is performed using VM as shown in Figure 1. The software that is used to manage the operation of these VMs on PMs is known as hypervisor/VM kernel (VMkernel). The role of VM kernel is to map VM to appropriate PM (Vaezi and Ying 2017; Hashem et al. 2017) [9-10]. Resource allocation is mainly performed into two phases: physical and virtual resource allocation performed in the infrastructure and the application level. Initially, the VMs are placed on the selected PMs and use its physical resources. In the second phase, the VMs are placed from one PM to another through migration technique (Gamal et al. 2017; López-Pires and Benjamín 2017) [12-13].

In this paper, a hybrid approach is presented to save energy in cloud system and provide services to cloud users with high efficiency. Two optimization algorithms, Artificial Bee Colony (ABC), and Cuckoo Search (CS) are used to decrease the false migration by selecting appropriate PM to migrate VM and hence utilized the resources efficiently. Here, MBFD algorithm is used to sort the VMs as per their resources and then optimized resources using ABC or CS approach to find the best PM.

The rest of the paper is organised as follows: section 2, highlights the related work on VM allocation with different techniques. Section 3, presents the proposed work associated with algorithm used. Section 4, present the results and discussion of the implemented work. Finally, in section 5, the conclusion is summarized.
Figure 1 Virtualization Architecture (Nashaat, et al. 2019) [11]
2. Related Work

For the very first time, (Pinheiro et al. 2001) have applied power management scheme on the Cloud Data Center (CDC). The power minimization has been performed on a heterogeneous cluster formed by a number of computing nodes served for distinct application. The aim of the technique is to worked on CDC’s workload and manage load by switching off the ideal node. The decision was taken on the basis of utilization of resources such as CPU, storage and network interface that whether the node is switched ON/OFF. The designed algorithm has worked on only on the master node that decrease the performance. This process of finding overloaded node or ideal node is performed one by one node and hence consumes time, and cannot be suitable for large size network [14]. Chase et al. 2001 have worked on to minimize energy consumption by managing resources in homogeneous computing environment. The main issue that is faced by the CSPs is to provide resources to each cloud users as per their demand with an efficient way. This problem has been resolved in [15] by designing a model named as ‘Bid’. The model has considered the SLAs as per the user’s budget as well as satisfying the QoS requirements. Also, flip flops have been used to remove the noise generated by the server during overloaded condition. Elnozahy et al. 2002 have resolve the issues of power efficient resource management with fixed response time and load balancing in a single web application [16]. A number of researchers (Beloglazov et al. 2012; Goudarzi and Massoud, 2015; Pay and Dan, 2015) have worked on to minimize the energy consumption in CDCs [17-19]. Lee and Zomaya 2012 have proposed task consolidation approach to improve the utilization of cloud resources in the presence of both active and idle energy consumption [20]. Hwang and Pedram 2013 have proposed an accessible approach to save energy with increased system’s efficiency by considering the types of VMs along with their correlation [21]. Corradi et al. 2014 have worked to design a cloud framework that can optimize the VM consolidation by considering the factors such as energy consumption, resources and networking [22]. Theja et al. 2015 have proposed a nature inspired approach named as A-GA “Adaptive Genetic Approach” for the detection of underload and overload detection. The A-GA technique has been used to optimized the VM placement with minimum energy consumption and SLA violation [23]. Dabbagh et al. 2015 have presented a hybrid energy efficient approach for CDC. The work has been processed in three phases (i) detect number of VM request from users along with their requirements (CPU and memory), (ii) calculate number of PMs that have been needed by the CDCs to satisfy the user’s requirement, (iii) minimize energy consumption by putting inactive PMs to sleep mode. But, not considered the factor of loading [24]. Sharma and Ram (2016) have presented VM consolidation using GA approach with cat swarm optimization. The work has been performed to put ideal PMs on sleep mode and hence contributed to save energy and save energy upto 30000 wsec has been attained [25]. Kaaouache et al. 2018 have designed a framework to minimize energy consumption in cloud data centre using best-fit decreasing algorithm in addition to GA approach. The work has minimized energy consumption for both PMs and networking. With the increase in the number of VMs, the computation time increases with a constant rate [26]. Sharma et al. 2019 have focused on failure of VMs along with the migration of VMs from overloaded to underloaded PMs. Two checkpoints are inserted to identify fault named as “VM migration”, and “VM checkpoint”. Using these checkpoints, the energy has been reduced upto 34 % [27]. Li et al. (2019) have presented a host aware workload detection technique using a robust linear regression prediction model. The work is divided into four sections: (i) detection of overloaded host, (ii) detection of underloaded host, (iii) selection of few VMs from overgraded and underloaded host, (iv) plan has been made to place the selected VMs from the overgraded and underloaded hosts. Using this technique, the energy saving upto 25.43 % has been achieved with 99.16 % of SLA violation rate [28]. Karthikeyan et al. (2020) have used Naïve Bayes technique in combination to integrated optimization approach ABC with Bat Algorithm (BA) to save energy in VM migration. The energy consumption upto 1200 KWh with a failure rate of 0.2 has been achieved. The classifier is trained based on the time and memory utilization of VMs and hence a threshold value is set for resource uses. If the required resource is greater than the defined threshold then failure apar otherwise considered as non-failure machine [29]. Songara, and
Manoj (2020) have presented a VM consolidation approach in CDC by considering the issues such as (i) to take decision while the system is overloaded when users demanded for multiple resources. VM selection has been performed by considering the policies of cloudsim. After appropriated selection of VMs, Particle Swarm Optimization (PSO) has been used for efficient VM placement with better energy saving [30].

3. Proposed Work

The aim of this research is to maximize the profit of CDCs in terms of energy efficiency by reducing the total energy consumed by ideal and active nodes with reserved VM requests from users. This section presents the components of CDC including PMs and user requests along with the defined problem that we want to resolve. The problem of VM consolidation is mainly categorized into four types (i) detection of underloaded host, (ii) detection of overloaded host, (iii) selection of VM, and (iii) allocation of VM. Here, VM allocation has been performed using ABC and CS algorithm separately, and then their output response in terms of performance parameter is computed. The detail of used technique in the present work is described below.

3.1 Mapping of VM to PM

Let us consider a CDC composed of ‘N’ PMs using M number of servers (S=S₁, S₂, S₃, … … … S₉). Each server (Sᵢ) is characterized by its clock frequency (fᵢ), CPU utilization (cᵢ), memory, and power consumption (Pᵢ). The processing capability of cloud system is calculated by determining the number of instructions executed per second and also have a direct relation to the CPU clock frequency. Let (T=T₁, T₂, T₃, … … … Tₙ) N number of cloud users submitted their request to the CDC. The incoming request is characterized on the basis of three factor (i) arrival time of request, (ii) execution time, and (iii) processing time taken by Tᵢ. After receiving the VM requests from all users, the broker of CDC needs to figure out the availability of PM as per the user’s requirement (Zhang et al. 2019; Calheiros et al. 2011). If the resources as per the user’s requirement are not available then the request is rejected by the system.

Mathematically, the VM to PM mapping N (i,j) can be denoted by equation (1).

\[ N(i,j) = \begin{cases} 1 & T_j \text{ is allocated to } S_i \\ 0 & \text{Otherwise} \end{cases} \]

(1)

The total VM requests that are accepted by the CDC \( \{R_a(N)\} \) is given by equation (2)

\[ R_a(N) = RO - \sum_{i=1}^{itr} P_R_N \]

(2) where Ra is resource available where RO is original resource PR_N is the remaining Physical Resources of Nth Physical Machine and itr is the total number of allocations to nth PM.

As mentioned before, that if the CDC does not meet the users VM requests then the request is rejected. If however, CDC accepts to provide the service to the user, CDC ranks its own PMs on the base of “Request Completion Ratio(RAR)” which includes the total number of services made by the PM maintaining the SLA to the total number of services provided to the PM

\[ RAR(N) = \frac{R_m}{Ras} \]

(3) where Rm is requests completed by PM maintaining the SLA and Ras is the total number of allocated requests.

3.2 Energy Consumption Calculation
VM server and PM server are the two most component of the CDC that consume major portion of power. Among the various hardware components, the CPU is the largest power consumer of the cloud server ($S_i$). Thus, $cp_i$, is a key factor to calculate the consumed power of the CDC. Thus, the power consumption of any system has a direct relation with $cp_i$. The server in its ideal state consumes 70% of its maximum power and is known as “static power”. When the server is in active mode, then the power consumed is known as “dynamic power”. Mathematically, the power model of a server can be represented by equation (4).

$$T_p = \text{static} + \text{CPU utilization} \times \text{dynamic}$$

or

$$T_p = \text{ideal} + cp_i \times (max_p - idle_p)$$

(4)

The total energy consumed by ($S_i$) in a time interval of [0, t] is given by equation (5).

$$T_E = \int_0^t T_p \cdot dt$$

(5)

After considering the above-mentioned models, work has been progressed into three phases (i) VM deployments, (ii) VM selection, and (iii) VM allocation (Dayarathna, et al. 2015 [33]).

### 3.3 VM deployment

After collecting requests from users, CDC foremost duty is to allocate to those VMs to PMs by using VM deployment methods. In this research, we apply Modified Best Fit Decreasing (MBFD) scheme for VM deployment. Using this approach, the VMs are sorting decreasing order according to their CPU utilization capabilities and allocate VM to the least power consumed host (Kumar, et al. 2014) [34]. This algorithm allows to select the most power efficient node first. The algorithm for MBFD is written below.
Algorithm: MBFD

1. **Input:** host list and VM list
2. Sort VM in decreasing order as per CPU utilization
3. **For each** VM request
4. {Calculate, Minpower ← max
5. Allocated host ← [null]
6. For each host
7. **If** host has sufficient resource for VM then
8. {Power, P ← Estimated Power (VMs, host)
9. **If** P < min(P) then
10. {Allocate host ← host
11. Min(P) ← P
12. **if** Allocated host ≠ null then
13. {assign VM to the allocated host}
14. **Return:** Sorted VMs based on power utilization
15. **End**

A CPU threshold has been set based on that the process of VM migration has been taken as shown in Figure 2.

![Flowchart](image)

**Figure 2: Energy saving strategy**
3.4 VM selection (using threshold technique)

The performance of designed cloud system is improved using threshold-based approach as discussed in Figure 2. Using threshold strategy, the energy consumption by servers can be reduced by migrating VMs from overutilized host to underutilized host. The relationship between the $cp_i$, $Th_a$, and $Th_b$ categorized the load as (i) normal, (ii) overload, and (iii) underload (Clark et al. 2005) [35].

3.5 VM placement (ABC/CS)

After VM selection, the process of VM placement using Artificial Bee Colony (ABC), and Cuckoo Search (CS) algorithm has been carried out.

3.5.1 Artificial Bee Colony (ABC) Approach

ABC algorithm is a meta-heuristic-based computational technique that mimics the common behavior of honey bees. In ABC, nectar or pollen was collected by employee bees and updated after each iteration in the hive. Based on the quality of food (fitness function) decision has been taken whether the pollen is of higher quality or not. In the present work, the number of user’s requests are bees. The nectar of bees is to map VM to PM in the cloud environment. In the general ABC scheme, the hive is updated each time, the employee bee collected food and the onlooker bee is a decision maker to select whether the collected food is of higher quality or not. In the proposed Enhanced- ABC (E-ABC) algorithm, a loyal employee bee is allocated to the CDC that update the VM status i.e. (workload capacity, CPU utilization, memory) at a constant time interval and saved in a fitness table. So, each time, when the honey bee search for a solution in the CDC, does not search for best fitness function. Therefore, the searching time has been saved during the allocation of VM to the appropriate host i.e. if the host is underutilized, then place VM from overutilized host to underutilized based. By doing so, the load in CDC is balanced with minimum energy consumption (Thanka, et al. 2019) [36]. The designed algorithm for ABC is written below.
Algorithm: Enhanced-Artificial Bee Colony

1. **Input:** Number of user requests, Number of VMs, VM properties ($N_{PROP}$), Number of servers

2. **Defined ABC Parameters:** Population of Bee (B) //Number of user’s requests
   - i. Employee bee // used to update servers features (memory, CPU utilization)
   - ii. Fitness Function: $F(f) = \begin{cases} 
   1; & \text{if } VM_{PROP} < T_{PROP} \\
   0; & \text{Otherwise} 
   \end{cases}$

3. Determine size of VMs in terms of row and columns (R, C)
4. Start feature selection
5. **For** $i=1$ → $R$
6. **For** $j = 1$ to $C$
7. $Bee_{E} =$ Current Bee (i, j) //update server features
8. $Bee_{On} =$ Respective Bee (i, j) // select PM with higher quality
9. $F(f) =$Call $F(f)$ ($Bee_{E}$, $Bee_{On}$)
10. Fitdata=$ABC (F(f), \text{Feature of VMs})$
11. **End**
12. **End**
13. **Returns:** Create an optimized VM list

3.5.2 **Cuckoo Search (CS) Approach**

CS is a nature inspired approach that consists eggs represents as server. For an example, for four number of server’s ‘A’, ‘B’, ‘C’, and ‘D’, the number of VMs are represented by number (1-10). A= {1,2,3}, B= {4,5,6}, c= {7,8,0}, and D= {9,10}. Therefore, the nest can be denoted as {1,2,3} {4,5,6} {7,8,0} {9,10} (Sait et al. 2016) [37]. CS is a bio-inspired algorithm that is designed from the lifestyle behaviour of the Cuckoo bird. Cuckoo birds lay their eggs in other birds’ nests with amazing abilities, such as choosing nests of other domestic birds that are already late-laying eggs and calling others eggs as their own eggs [4]. Parasitic cuckoos’ sports nests where eggs are just laid, and the laying time of egg is very accurate. They usually lay an egg in a nest that will grow faster than other eggs. When this happened, the outer cuckoo would push the eggs out of the nest. This behaviour of cuckoo bird reduces the probability of hatching legitimate eggs (Mareli et al. 2018) [38]. In addition, the external cuckoo chick can get more food by imitating the host chicks call. Sometimes the host finds out that one of the eggs is foreign. In this case, the cuckoo either gets rid of the egg, or completely abandons its nest and goes somewhere else to build a new nest. The evaluation process of CS algorithm followed by considering three operators (i) Levy flight, (ii) replacement of existing nests with a new solution and (iii) elitist selection methodology. Levy flight operator is used to produced new solution. In replacement process, a new value of each individual solution is selected randomly and find probability. Based on the probability a new value is selected. In elitist selection, the new value is compared to the old one value, if the new value is found with higher quality than past one. Then, it is considered as final solution otherwise, ignore the value (Joshi et al. 2015) [39]. The algorithm for the enhanced Cuckoo Search (E-CS) is written below.
Algorithm Enhanced-Cuckoo Search

1. **Input:** Selected VMs, number of servers
2. **Initialize CS operators**
   - Iterations (ITR)
   - Number of Egg (E\textsubscript{N})// number of VMs
   - Number of Variables (Nvar)// Number of servers
3. Calculate VM size \( P_i \leftarrow \text{Size (VM)} \)
4. Fitness function, \( f_{fit} \)
   \[
   f_{fit} = \begin{cases} 
   \text{True;} & \text{if attained superior quality egg } (f_e > f_{th}) \\
   \text{False;} & \text{otherwise}
   \end{cases}
   \]
   Where \( f_e \) is the random change in egg and \( f_{th} \) is the threshold which should be achieved by host birds
5. For each ITR & \( P_i \) in respect of j
6. \( f_e = \sum_{j=1}^{P} VM \text{ energy} = P_N \)
7. \( f_{th} = \sum_{j=1}^{P} power (VMs) \\
   \sum_{j=1}^{P} P_j \)
8. \( VM_{allocation} = CSO(P_N,Nvar,f_{fit}) \)
9. End
10. **Return:** Allocated VMs
11. End

4. Result and discussion

This section is divided into two section. In the first half, the performance of proposed work E-ABC algorithm has been presented, and compared with the E-CS, basic ABC, and CS algorithms respectively. In the second part, comparative analysis has been performed with the existing work performed by [37,38]. The work has been performed using MATLAB toolkit and experiments were performed using optimization tool. The performance of the system is evaluated on the basis of energy consumption, a number of migrations, and SLA violation. To perform experiment a few assumptions has been taken as listed in table 1.

| Table 1 Required Parameters for Simulation |
|------------------------------------------|
| Number of VMs | 800 |
| Number of Hosts | 300 |
| VMs CPU utilization | 500,750,1000,1250 |
| Hosts CPU Utilization | 2000,3000,4000 |
| Number of CDC | 1 |
| Hosts memory capacity (MIPS) | 2000 |
| VMs memory capacity (MIPS) | 200 |
Taking into account the above input parameters listed in Table 1, i.e 300 hosts and 800 virtual machines, the test is performed with respect to number of VMs. The obtained results for respective parameters are listed in Table2

| Number of VMs | Energy Consumption (KW-h) | SLA Violation | Number of Migrations |
|---------------|---------------------------|---------------|----------------------|
|               | ABC | CS | E-ABC | E-CS | ABC | CS | E-ABC | E-CS | ABC | CS | E-ABC | E-CS |
| 50            | 4.8 | 5.2 | 0.2   | 0.7   | 1.4 | 1.9 | 0.5   | 0.6   | 30  | 32 | 21    | 27    |
| 100           | 6.2 | 6.6 | 0.8   | 0.8   | 1.8 | 2.2 | 0.8   | 1.0   | 42  | 46 | 30    | 35    |
| 150           | 7.6 | 7.9 | 1.6   | 1.9   | 2.4 | 3.6 | 1.0   | 1.3   | 63  | 66 | 52    | 61    |
| 200           | 8.8 | 8.2 | 1.8   | 2.1   | 3.7 | 4.8 | 2.2   | 2.9   | 74  | 82 | 60    | 72    |
| 250           | 9.6 | 8.8 | 2.1   | 2.5   | 8.5 | 5.6 | 3.4   | 4.7   | 90  | 93 | 77    | 83    |
| 300           | 9.8 | 9.2 | 2.4   | 2.8   | 8.6 | 7.8 | 4.6   | 5.1   | 96  | 100| 80    | 88    |
| 350           | 10.2| 10.5| 2.9   | 3.5   | 9.2 | 8.9 | 6.9   | 7.2   | 98  | 109| 92    | 99    |
| 400           | 10.6| 10.8| 3.5   | 4.6   | 10.3| 9.2 | 8.2   | 9.9   | 118 | 124| 100   | 112   |
| 450           | 11.5| 11.8| 4.2   | 4.9   | 11.5| 10.6| 10.6  | 12.3  | 130 | 136| 116   | 127   |
| 500           | 11.9| 12.2| 4.4   | 5.6   | 12.7| 12.5| 10.7  | 13.7  | 140 | 148| 120   | 134   |
| 550           | 12.2| 12.9| 5.9   | 6.2   | 13.8| 14.6| 11.9  | 14.2  | 150 | 152| 129   | 139   |
| 600           | 12.6| 13.2| 6.2   | 6.9   | 14.6| 16.3| 12.3  | 15.6  | 151 | 152| 132   | 141   |
| 650           | 12.8| 13.6| 6.7   | 7.2   | 15.2| 18.2| 14.5  | 16.7  | 158 | 160| 137   | 146   |
| 700           | 13.3| 13.7| 7.2   | 8.9   | 16.3| 20.2| 15.2  | 17.2  | 160 | 162| 143   | 153   |
| 750           | 13.6| 14.1| 7.4   | 9.5   | 19.6| 22.7| 17.8  | 18.6  | 185 | 200| 188   | 192   |
| 800           | 13.8| 14.8| 8.9   | 10.2  | 21.3| 26.15| 18.6  | 19.3  | 225 | 249| 207   | 236   |
| 850           | 13.5| 15.1| 9.5   | 10.8  | 24.6| 27.3| 19.2  | 20.5  | 259 | 262| 246   | 257   |
| 900           | 13.9| 15.9| 9.9   | 11.2  | 27.3| 30.5| 20.4  | 22.4  | 270 | 272| 251   | 269   |
| 950           | 14.2| 16.4| 10.3  | 12.4  | 28.6| 32.9| 22.1  | 24.3  | 290 | 295| 273   | 286   |
| 1000          | 14.7| 16.8| 10.8  | 13.5  | 30.2| 34.5| 22.8  | 25.2  | 300 | 312| 289   | 295   |

The tests have been performed upto 1000 VMs and the simulation has been repeated 10 times. The computed parameters for four different techniques simple ABC, Simple CS, E-ABC, and E-CS, have been listed in table 2.
The comparative graph for consumed energy with four different techniques ABC, CS, E-ABC, E-CS with respect to number of VMs is shown in Figure 3. During the experimental process, it must be noted that how much hosts are required by the VM to complete the user request with minimum energy. To prevent most hosts from sitting idle, it is important to monitor the number of hosts working in the system. Since, the number of hosts sitting idle consume unnecessary energy, which will violate the criteria of minimum energy demand. After the detection of idle hosts, it is switched to sleep mode. From the experiments it has been observed that E-ABC technique consumes less energy compared to E-CS, ABC and CS approach. This is because, E-ABC runs using artificial bee colony optimization algorithm that helps for the selection of appropriate node for VM allocation with less searching time. The enhancement in the ABC approach (E-ABC) utilized resources in a well-managed manner and hence reduced the number of VM migrations, which minimized the energy consumption and hence save energy. The energy saving can be calculated by first determining the average energy consumed by E-ABC and E-CS approach, which was observed as 5.335 KW-h, and 6.31 KW-h respectively. Therefore, the energy saved around 15.31 % has been achieved using E-ABC approach against E-CS.
The examined values for SLA violation with respect to number of VMs is shown in Figure 4. From the graph it is concluded that E-ABC approach is performed better with minimum violation followed by E-CS, ABC, and CS i.e. maximum violation has been observed using simple CS approach. Also, with the increase in the number of VMs, the SLA violation also increases as depicted by the bar lines. The minimum and maximum SLA violation observed for 50 and 1000 number of VMs using proposed E-ABC approach is 0.5, and 22.8 respectively. Also, the average SLA violation analysed for E-ABC, E-CS approach is 11.185%, 14.69 % respectively. Therefore, the percentage reduction in SLA violation using E-ABC approach is 23.86 % against the E-CS approach. Therefore, E-ABC has performed well among all the three approaches as mentioned in Figure 4. This is because, bees search for an appropriate host with minimum searching time.
Number of migrations observed by four different approaches in relation with number of VMs is shown in Figure 5. From the graph it is observed that proposed E-ABC approach utilized a smaller number of hosts and hence results in lesser VM migrations compared to existing E-CS, ABC, and CS approach. The VM migrations has been affected by the ability of E-ABC approach to actively find out the best host without compromising on energy consumption. When a new workload arrives, the overall energy consumption in the CDC is the optimal, because nodes work on threshold-based energy consumption. This is because, all the incoming requests from the users has been performed or allocated to the hosts by considering the threshold value of energy consumed by each request. Therefore, while performing workload distribution based on energy constraints, the demand for causing more and more VM migrations is low. From experiment the average number of migrations examined using E-ABC, E-CS approach are 137.15, and 153.94 respectively. The percentage reduction in the VM migration using E-ABC approach is of 10.91% has been observed against E-CS approach.

4.1 Comparative Analysis

In the last couple of years, a number of researchers have used biologically-inspired optimization approach to deal with heterogeneity and the growing energy crisis in CDC. In this research, we have also used swarm inspired approach to minimize the energy consumption in CDC with lesser number of migrations and SLA violations. To show the enhancement of our work against the existing work comparative analysis has been performed. The analyzed parametric values are listed in table 3.

| Energy Consumption (KW-h) | SLA Violation (%) | Number of Migrations |
|---------------------------|-------------------|----------------------|
| E-ABC                     | E-CS              | ACO Karda et al. (2019) [38] |
| 5.335                     | 6.31              | 6.43                  |
| E-ABC                     | E-CS              | ACO Karda et al. (2019) [38] |
| 11.185                    | 14.69             | 14.5                  |
| E-ABC                     | E-CS              | ACO Karda et al. (2019) [38] |
| 137.15                    | 153.94            | 154.54                |

Figure 6: Comparison of Energy Consumption
The comparison for energy consumption using E-ABC (proposed approach) has been performed with E-CS and ACO approach performed by [38]. The values of energy consumption correspond to E-ABC, E-CS, and ACO has been represented by the blue, the orange and the gray color respectively. The average values of energy consumption observed for E-ABC, E-CS, and ACO are 5.335 KW-h, 6.31 KW-h, and 6.43 KW-h respectively. Therefore, energy saving of 15.45 %, and 17.03 % has been obtained using ABC approach compared to E-CS and [38] approach. This is because, in CS, due to randomness initialization of cuckoo egg population, the fluctuation in population is significantly more than ABC and it creates a poor convergence.

![Figure 7: Comparison of SLA Violation](image)

In the same way, comparison for SLA violation and number of migrations with the existing technique ACO [38], and E-CS is shown in Figure 7, and Figure 8 respectively. From Figure 7, the percentage minimization in SLA violation against E-CS and [38] is obtained about 23.86%, and 22.86 % respectively. Also, the reduction in number of migrations using E-ABC approach against E-CS, and ACO approach has been obtained as 10.91%, and 10.67 % respectively.

![Figure 8: Comparison of Number of Migrations](image)
5. Conclusion

In recent years, energy efficiency has become the utmost requirement of current computing systems. It expands from a single server to large data centres known as clouds as it requires massive energy to operate servers and hence perform operation as the cloud user and cloud providers. Therefore, effective energy management is important for CDC. In this paper, an enhanced ABC (E-ABC) approach along with enhanced Cuckoo Search (E-CS) algorithm has been presented to save energy by resolving the problem of workload. The use of cloud resources has been performed in a well-managed way by using the concept of thresholding. A threshold limit has been defined to find out the workload condition (i) normal, (ii) overutilized, and (iii) Under Utilized. The presented work has used swarm-based ABC approach to determine the overloaded host and hence migrate VMs from overloaded to underloaded host to obtained maximum utilization of energy in CDC. The result shows an improvement in results compared to the E-CS and ACO approach. The E-ABC has performed better with higher scalability and minimum number of VM migration with effective host usages. An improved energy saving of 15.45 %, and 17.03 % has been observed against E-CS and [38] approach with an average reduction in number of migrations of 10.91%, and 10.67 % compared to E-CS and [38] approach and hence contributed towards the green cloud computing.

References

[1]. Shabeera, T. P., SD Madhu Kumar, Sameera M. Salam, and K. Murali Krishnan 2017 "Optimizing VM allocation and data placement for data-intensive applications in cloud using ACO metaheuristic algorithm " Engineering Science and Technology, an International Journal 20, no. 2 : 616-628.
[2]. Lee, Gary 2014 Cloud networking: Understanding cloud-based data center networks. Morgan Kaufmann.
[3]. Zhang, Xinqian, Tingming Wu, Mingsong Chen, Tongquan Wei, Junlong Zhou, Shiyan Hu, and Rajkumar Buyya 2019 Energy-aware virtual machine allocation for cloud with resource reservation. Journal of Systems and Software : 147-161.
[4]. Arianyan, Ehsan, Hassan Taheri, and Saeed Sharifian 2016 Novel heuristics for consolidation of virtual machines in cloud data centers using multi-criteria resource management solutions The Journal of Supercomputing 72 no. 2 pp 688-717.
[5]. Gai, Keke, Zhihua Du, Meikang Qiu, and Hui Zhao. 2015 Efficiency-aware workload optimizations of heterogeneous cloud computing for capacity planning in financial industry IEEE 2nd International Conference on Cyber Security and Cloud Computing, pp. 1-6. IEEE.
[6]. Soltanshahi, Minoo, Reza Asemi, and Nazi Shafiei 2019 Energy-aware virtual machines allocation by krill herd algorithm in cloud data centers Heliyon 5, no. 7: e02066.
[7]. Beloglazov, Anton, and Rajkumar Buyya 2012 Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers Concurrency and Computation Practice and Experience 24, no. 13 pp 1397-1420.
[8]. Zhang, Peiyun, MengChu Zhou, and Xuelei Wang 2020 An Intelligent Optimization Method for Optimal Virtual Machine Allocation in Cloud Data Centers IEEE Transactions on Automation Science and Engineering.
[9]. Vaezi, Mojtaba, and Ying Zhang 2017 Virtualization and cloud computing Cloud Mobile Networks, pp. 11-31. Springer, Cham.
[10]. Hashem, Walaa, Heba Nashaat, and Rawya Rizk 2017 Honey bee based load balancing in cloud computing KSII Transactions on Internet & Information Systems 11, no. 12.
[11]. Nashaat, Heba, Nesma Ashry, and Rawya Rizk 2019 Smart elastic scheduling algorithm for virtual machine migration in cloud computing The Journal of Supercomputing 75, no. 7 pp 3842-3865.
[12]. Gamal, Marwa, Rawya Rizk, Hani Mahdi, and Basem Elhady. 2017 Bio-inspired load balancing algorithm in cloud computing International Conference on Advanced Intelligent Systems and Informatics, pp. 579-589. Springer, Cham.
[13]. López-Pires, Fabio, and Benjamín Barán 2017 Many-objective virtual machine placement Journal of Grid Computing 15, no. 2 pp 161-176.

[14]. Pinheiro, Eduardo, Ricardo Bianchini, Enrique V. Carrera, and Taliver Heath 2001 Load balancing and unbalancing for power and performance in cluster-based systems.

[15]. Chase, Jeffrey S., Darrell C. Anderson, Prachi N. Thakar, Amin M. Vahdat, and Ronald P. Doyle. 2001 Managing energy and server resources in hosting centers. ACM SIGOPS operating systems review 35, no. 5 pp 103-116.

[16]. Elnozahy, EN Mootaz, Michael Kistler, and Ramakrishnan Rajamony. 2002 Energy-efficient server clusters International Workshop on Power-Aware Computer Systems, pp. 179-197. Springer, Berlin, Heidelberg.

[17]. Beloglazov, Anton, Jemal Abawajy, and Rajkumar Buyya 2012 Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing Future generation computer systems 28, no. 5 pp 755-768.

[18]. Goudarzi, Hadi, and Massoud Pedram 2015 Hierarchical SLA-driven resource management for peak power-aware and energy-efficient operation of a cloud datacentre IEEE Transactions on Cloud Computing 4, no. 2 pp 222-236.

[19]. Paya, Ashkan, and Dan C. Marinescu 2015 Energy-aware load balancing and application scaling for the cloud ecosystem IEEE Transactions on Cloud Computing 5, no. 1 pp 15-27.

[20]. Lee, Young Choon, and Albert Y. Zomaya. 2012 Energy efficient utilization of resources in cloud computing systems The Journal of Supercomputing 60, no. 2 pp 268-280.

[21]. Hwang, Inkwon, and Massoud Pedram 2013 Hierarchical virtual machine consolidation in a cloud computing system IEEE Sixth International Conference on Cloud Computing, pp. 196-203.

[22]. Corradi, Antonio, Mario Fanelli, and Luca Foschini 2014 VM consolidation: A real case based on OpenStack Cloud Future Generation Computer Systems pp 118-127.

[23]. Theja, Perla Ravi, and SK Khadar Babu 2015 An evolutionary computing based energy efficient VM consolidation scheme for optimal resource utilization and QoS assurance Indian Journal of Science and Technology 77179 pp1-11.

[24]. Dabbagh, Mehiar, Bechir Hamdaoui, Mohsen Guizani, and Ammar Rayes 2015 Energy-efficient resource allocation and provisioning framework for cloud data centers IEEE Transactions on Network and Service Management 12, no. 3 pp 377-391.

[25]. Sharma, Neeraj Kumar, and Ram Mohana Reddy Guddeti. 2016 On demand virtual machine allocation and migration at cloud data center using hybrid of cat swarm optimization and genetic algorithm Fifth International Conference on Eco-friendly Computing and Communication Systems (ICECCS), pp. 27-32.

[26]. Kaouache, Mohamed Amine, and Sadok Bouamama. 2018 An energy-efficient VM placement method for cloud data centers using a hybrid genetic algorithm Journal of Systems and Information Technology.

[27]. Sharma, Yogesh, Weisheng Si, Daniel Sun, and Bahan Javadi 2019 Failure-aware energy-efficient VM consolidation in cloud computing systems Future Generation Computer Systems 94 pp 620-633.

[28]. Li, Lianpeng, Jian Dong, Decheng Zuo, and Jin Wu. 2019 "SLA-aware and energy-efficient VM consolidation in cloud data centers using robust linear regression prediction model. IEEE Access 7 pp 9490-9500.

[29]. Karthikeyan, K., R. Sunder, K. Shankar, S. K. Lakshmanaprabhu, V. Vijayakumar, Mohamed Elhoseny, and Gunasekaran Manogaran. 2020 Energy consumption analysis of Virtual Machine migration in cloud using hybrid swarm optimization (ABC–BA). The Journal of Supercomputing 76, no. 5 pp 3374-3390.

[30]. Songara, Neha, and Manoj Kumar Jain 2020 Design of QoS and Energy Efficient VM Consolidation Framework for Cloud Data Centers Advances in Cybernetics, Cognition, and Machine Learning for Communication Technologies, pp. 171-179.

[31]. Zhang, Xinqian, Tiongming Wu, Mingsong Chen, Tongquan Wei, Junlong Zhou, Shiyan Hu, and Rajkumar Buyya 2019 Energy-aware virtual machine allocation for cloud with resource reservation. Journal of Systems and Information Technology 147 pp 147-161.
[32]. Calheiros, Rodrigo N., Rajiv Ranjan, Anton Beloglazov, César AF De Rose, and Rajkumar Buyya. 2011 CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms Software: Practice and experience 41, no. 1 pp 23-50.

[33]. Dayarathna, Miyuru, Yonggang Wen, and Rui Fan 2015 Data center energy consumption modeling: A survey. IEEE Communications Surveys & Tutorials 18, no. 1 pp 732-794.

[34]. Kumar, Pradeep, Dilbag Singh, and Ankur Kaushik 2014 Power and data aware best fit algorithm for energy saving in cloud computing. International Journal of Computer Science and Information Technologies 5, no. 5 pp 6712-6715.

[35]. Clark, Christopher, Keir Fraser, Steven Hand, Jacob Gorm Hansen, Eric Jul, Christian Limpach, Ian Pratt, and Andrew Warfield 2005 Live migration of virtual machines Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2, pp. 273-286. 2005.

[36]. Thanka, M. Roshni, P. Uma Maheswari, and E. Bijolin Edwin 2019 An improved efficient: Artificial Bee Colony algorithm for security and QoS aware scheduling in cloud computing environment Cluster Computing 22, no. 5 pp 10905-10913.

[37]. Sait, Sadiq M., Abubakar Bala, and Aiman H. El-Maleh. 2016 Cuckoo search-based resource optimization of datacenters. Applied Intelligence 44, no. 3 pp 489-506.

[38]. Karda, Raksha Kiran, and Mala Kalra 2019 Bio-inspired Threshold Based VM Migration for Green Cloud Advances in Data and Information Sciences, pp. 15-30. Springer, Singapore.