International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-6, August, 2019

Eeadoselfcloud: Energy Efficient Adaptive Depth Optimized Self Cloud Mechanism for VM Migration in Data Centers

Sebagenzi Jason, Suchithra R.,

***Abstract:** Cloud computing is a paradigm where all resources like software, hardware and information are accessed over internet by using highly sophisticated virtual data centres. The cloud has a data center with a host of many features. Each machine is shared by many users, and virtual machines are used to use these machines. With a large number of data centers and data centers with a large number of physical hosts. Two important issues in cloud environment are Load balancing and power consumption which solved by virtual machine migration. In earlier learnings, Artificial Bee Colony (ABC)'s policy could lead to a compromise between productivity and energy consumption. There are, however, two ways in the ABC-based Abstract based approach: (1) How to find effective solutions across the globe. (2) how to reduce the time to decide to distribute BM.To overcome this issue, this project develop one novel VM migration scheme called eeadoSelfCloud. This proposed method introduces Bee Lion Optimization (BLO) for VM allocation. Data Center Utilization, Average Node Utilization, Request Rejection Ration, Number of Hop Count and Power Consumption are employed as parameters for the proposed algorithm analysis. The experimental results indicate that the proposed algorithm does better than the other available methods.*

***Index Terms:** Cloud Computing; Data centre; VM Placement; VM Migration; Energy Efficient Self Organization Cloud; Optimization.*

I. INTRODUCTION

Cloud computing has gained popularity because of its many features, such as cost-effective services and payment services [1], which are independent times and geographical location. This is a general term used to describe the collection / team of hardware, hardware, and Internet infrastructure. These forums hide the complexity and detail of core user and program infrastructure by providing the simplest graphical interface or API (programming interface). Cloud computing technology allows developers and IT specialists to focus on key issues and release them from activities such as maintenance, procurement and capacity planning. With increasing popularity of cloud applications, prototypes and different deployments have emerged to meet the specific needs of different users [2]. Types of cloud services and deployment methods provide different levels of control, flexibility and management. There are three core service models in desktop computers: Software (SaaS), PaaS, and Infrastructure Service (IaaS) [3]. These three services can be deployed in four different ways: public clouds and hybrid [25].

Cloud computing has become a disruptive technology transforming businesses by providing infrastructure and resources as services. Energy efficiency is a crucial concern in cloud data centers. The quantity of power obsessived by the infrastructure, including servers, network devices, cooling equipment and others, highly affects the cloud provider's profit margin. It also has a major environmental impact due to the carbon emissions contributing to global warming. Virtual machine consolidation is one of the fundamental approaches to accomplish power competence in cloud data centers. It involves packing all available Virtual Machines (VMs) on an optimal number of Physical Machines (PMs). The goal is to free underutilized servers, turn them into sleep mode and hence, save their operational power. Servers consume 80% of their peak power even at 20% utilization [26]. In this way, turning the physical server into a low power mode can significantly reduce the total power consumption of the data center. Cloud usage uses the old virtualization technology, which makes the necessary resources used in the virtual machine's form, and then allocates the resources to the server parts to perform the relevant tasks, so that the process of resource management can be done. Converts the process of virtual machine search [27]. Since cloud storage has begun to use extensive virtualization technology, exploring the Cloud Virtual Data Center's Virtual Machine Center offers new research guidelines to enhance energy efficiency in data centers. The initial placement of the VM according to the optimum target can be used to reduce power consumption and reduce the number of active servers after the completion of the cloud platform to a certain level and have a positive effect on functionality and Continuous optimization for the BM Platform integration for cloud computing. In existing works [28, 29], artificial bee colonies [ABC] [28] were accepted to achieve energy efficiency [29]. However, the ABC-based securities distribution policy is scarce in the search for a global solution over time. To address these issues, the Lion has been improved to find as many global solutions as possible in this article.

Revised Manuscript Received on October 30, 2019.
* Correspondence Author
Mr.Jason Sebagenzi*, research scholar in Jain University, Bangalore .
R.Suchithra, director of MCA department in Jain Deemed to be University, Bangalore .

Based on the principle of Lion Optimization, it is likely to jump from a local minimum to find the best places in the world. Considering this concept, the article highlights a policy for distributing the energy efficiency of the electric power efficiently with the installation of the lions.

The most important principles of this report are summarized as follows: (i) The concept of Artificial Bee Colony is employed to overcome the problem of how to meet the best acceleration of the best solution in the database. (ii) The idea of increasing the effectiveness of the Lion Optimization has been approved to achieve the most m estimated global optimums. (iii) After achieving the best global optimums, VM's migration rate has been significantly reduced to achieve high power effectiveness.

The article remains are prepared as follows. Section 2 reviews your work. The rules for initial deployment and time for placing a virtual machine in the data center are proposed in Part III. Section IV illustrates the experimental results using a model simulation model. The conclusion in addition to the future work of this study are presented in Section V.

II. RELATED WORK

With the rapid growth of cloud computing, the data center is expanding as well as the use of energy generated by the data center, resulting in high operating costs. So, reducing energy consumption in the data center is an issue that needs urgent solutions. The use of energy in the data center provided is closely related to the use of resources. Resource utilization to reduce energy consumption should be improved to reduce operating costs of data centers. Currently, the most effective way to achieve this goal is to use virtualization technologies that can better utilize server resources. The improvement of the use of resources through virtual technology is divided into two categories: the initial placement [21, 22] and the consolidation of BM [23, 24]. Most of the VM site's research has been done by domestic and foreign scientists. These techniques are discussed in this segment. Feller et al. [4] propose the Ant Colony Optimization (ACO) approach to overcome the Multidimensional bin-packing difficulty (MDBP) in the context of dynamic workload consolidation. Authors used their own java-based toolkit EnaCloud as a cloud simulation environment. ACO algorithm is compared to classical greedy FFD algorithm. Authors suggests that the workload placement is to be done initially using FFD algorithm because it has low computation time. And then, on periodic-basis, daily or weekly for example, the proposed ACO algorithm is used to optimize the placements. Kansal et al [8] suggest using a Firefly based optimization approach. They attempt to locate the superlative VM-Host pair where through the proposed FireFly Optimization Energy-aware Virtual Machine Migration (FFO-EVMM) approach to maximize energy efficiency through the optimum number of migrations. Farahnakian et al. [9] Creates problems of unification as a multi-purpose issue that addresses three conflicting goals. Reducing energy consumption, maintaining the minimum BM BM exchange and avoiding SLA. Ant Colony System (ACS) is used as a meta-heuristic algorithm, where the amount of pheromone produced by the ants accumulate on the preferred migration plans. Wu et al. [5] regards the consolidation problem as a grouping problem, whose objective is to divide a set of VMs into a number of disjoint subsets, representing PMs. They propose a score function for to evaluate both; a method for estimating the cost of migration and a method to estimate the upper limit for maximum energy savings. An improved version for the classical grouping genetic algorithm is proposed and applied to the consolidation problem. Xu Gaochao et al. [6] has introduced a new cloud-based migration policy based on cloud computing for cloud environments, incorporating the idea of an artificial bee colony (ABC) with the same coincidence idea of binary search and selective policy Boltzmann to improve. ABC-based approaches have better abilities for world-class navigation and the ability to exploit the region. They also use the Bayesian theory to further improve the ABC-based process for the best solution. As a result, the whole approach achieves lasting efficiency for energy savings. The experimental results show that PS-ABC clearly reduces the use of energy and increases the likelihood of VHI work and migration better than the existing research. This makes the results of VM changes live efficient and meaningful. A. Mevada et al. In [7], a good policy for deploying energy efficient virtual machines for cloud balancing systems. The authors requested a modification of the VM usage algorithm to reduce the use of energy, the balance of storage, and enhance the introduction of the VM. The proposed algorithm has not been implemented and tested under different conditions in real time or experimental. Adhianto et al. [10] Creates a DVMC problem as a NP-hard unclear optimization problem. They requested an algorithm based on the Ant Colony Optimization (ACO) method to overcome the issues. The mathematical model for estimating energy prices due to migration, migration costs and SLA abuse because migration is advised. The proposed algorithm is compared to other DVMC algorithms with migration and is said to reduce the total power consumption by about 47%. Marzolla et al. [11] propose V-Man, a DVMC algorithm based on a simple gossiping protocol. The algorithm maintains an overlay network in which all the active hosts are organized. Every host, i.e. node, on the network can only interact with some of the network hosts, its neighbors. Each node periodically broadcasts messages about the number of hosted VMs on that node to its neighbors and receives messages about the number of VMs hosed by its neighbors. The node with the greater number of VMs migrates them to its neighboring node till the destination node reaches a predefined threshold for the maximum number of hosted VMs. However, this work assumes that all the hosted VMs are identical. A host utilization can be determined given only the number of hosted VMs. This is not valid in practice since VMs vary in terms of their sizes and resource demands. Moreover, the migration overhead is ignored. Wang et al. [12] Created VM location algorithm based on optimal swarm optimization (PSO). The algorithm has been improved for in-depth data usage in the National Information Center (NCDC). Wang et al. Do not accept the same host. The proposed algorithm based on the PSO is compared to the Best Limitations Resolution (MBFD), the first (FF) and Best Rule (BF) algorithm.

It has been shown that the algorithm is the power efficiency at the data processing center that looks like wood. One of the definitions of this work is that it assumes that each VM performs service. Although energy savings are the primary goal of BM consolidation, most literary efforts ignore the expense of migration. Rybina et al. [13] Provides mathematical models for estimating energy prices for migration. This is achieved by seeking additional energy from two homeowners involved in the immigration process as well as the length of the process. Researchers have shown that the BM size, the available bandwidth and the type of storage are the parameters that contribute to the energy cost of living in the habitat. During migration, energy increases for both the outside and the target engine as the speed of the network increases. The type of workload applied to the VM changes affects the energy usage of the switch. The VM emulator size and available bandwidth are parameters that contribute to the time of change generally. Large VMs last longer than smaller virtual machines and provide faster speeds to ensure faster exchanges. Strunk [14] also states that the energy regime, due to changes in the switch, differs by two parameters: the number of VM's RAM will be changed and the speed of the network. Many experiments were conducted and the data received were used to create patterns using a linear regression. This job offers a mathematical model for estimating the energy value of a live immigration operation on a virtual machine. This model has been tested and has been proven to be more than 90% accurate. The limit of this job is that the template established is only valid for inactive virtual machines. Rybina [15] et al. There are five resource usage parameters: the number of missed failures in the cache, the number of downloaded instructions, the CPU usage proportion of the active memory used by the resulting server to the data rate and the number of "dirty pages" Found on foreign servers during migration. This work is different. Rybina et al. Use MLR to model the power usage dependencies during migration on the tested resource parameters. At first, it was shown that energy consumption was dependent on linear parameters for each of the parameters considered. Then the regression method of all sub-groups is used to find the best MLR model. The requested experiment shows that the CPU instruction has been retired, the latest level caching has been missed, and the page of the polluted pages that the server viewed during the immigration process is the most important parameter for use. Energy for intensive charge.

Jia Zhao. et al. [16] proposed PSO (Particle Swarm Optimisation) based PS-ES (Placement Selection Energy Saving) algortihm where SA (Simulated Annealing) idea with probability theory was introduced for long term optimization on energy saving Its incremental energy consumption parameter was compared with DAPSO (Dynamic Adaptive PSO) , PS-ABC (PSArtificial Bee Colony) and shows better results than the two. Thiruvenkadam T. et al. [17] compared several VM placement algorithms like Round-robin, Greedy, Energy save, Dynamic priority based, Genetic which are popularly used in cloud environment. The survey shows that Genetic Dynamic priority based and Genetic algorithms consumes least energy. Arnab et al. [18] discussed dynamic consolidation of VMs which is one of the advantages of Live Migration by which idle hosts are switched to sleep mode in cloud computing. They used online based approach like ALG for single VM migration and dynamic consolidation of VMs. Depending on past data they proposed novel heuristics technique by which the threshold utilization is automatically adjusted and it is energy and performance efficient.

Jeyarani et al. [19] SAPSO (Particular Swarm Optimization Optimization) was proposed to provide a virtual cloud virtual machine efficiently targeted when plotting the VM object set to a set of dynamic energy resource resources extracted The mapping is minimal and does not affect the functionality. The benefits of the proposed solution are obvious. It not only focuses on improving the cloud user coordination burden, but it also focuses on creating efficient data center data centers to make it easier for cloud providers. However, this approach may still be ineffective and cause further events and expenditures from a long-term perspective, as it does not matter the future burden. The proposed PS-ES algorithm is a heuristic approach based on PSO, an intellectual algorithm and advances SA (Simulated Annealing). Gaochao Xu et al. [20] Shows a new method called PS-ABC. Its algorithm has two parts. In part, it combines the concept of artificial bee colonies (ABCs), which has the same concept as the beginning of binary search concept and the Boltzmann recruitment policy to achieve ABC-based approaches that are capable of researching Better global. For local operations. The other part is that it uses Bayes theoretical theory to optimize more ABC-based processes to get the best faster solutions. The whole approach achieves long-term optimization for energy savings. However, by achieving a more precise solution to today's problems, the long-term impact of energy efficiency increases has been achieved. It does not take into account the whole issue of long-term outlook for Cloud Data Center. Even though it has achieved an inter-generational global optimization in the algorithm cycle, it can result in better results from the baseline data operations of the Cloud Data Center. As a result, the PSES proposed in this article has an energy saving effect rather than PS-ABC over long-term cloud data centers.

III. PROPOSED WORK

If you are using *Word*, use either the Microsoft Equation Editor or the *MathType* add-on (http://www.mathtype.com) for equations in your paper (Insert | Object | Create New | Microsoft Equation *or* MathType Equation). "Float over text" should *not* be selected.

A. VM Placement

Virtual machines can be located in the data center in addition to during execution. During the launch of the virtual computer software, it is created and placed on a machine or machine appropriate to the data center, called the early assignment of the virtual machine. Nevertheless, during the implementation of the virtual machine application, the machine can be switched from machine to machine to reduce the number of servers called operating areas.

The intelligent and efficient eeadoSelfCloud system seeks to find research issues related to the virtual and virtual premises of the virtual machine. The sequence of execution or flow of eeadoSelfCloud is shown in Figure 1. An important part of the energy used by the server is reported for the CPU go behind by the memory according to the data provided. So, only two basic dimensions of CPU and memory resources are measured for test unit.

For the early assignment of VM machines, physical machines or host servers in data centre are classified into four different groups Critical, Active, Selfish and Overflow group by using Bee Lion Optimization (BLO) algorithm. This BLO algorithm is used to select the best VMs based on their current CPU utilizations, Residual Power and Previous Allotted Count. The four groups are described below.

1) Critical Group: The server uses about 70% of its highest power in the state of the Internet when it is entirely or very low CPU usage. So this work can remain SMs inactive in the power saving situation using the power provided by virtual monitors (VMMs) or the installer. Windows KVM servers support the S4 (hibernation) and S3 (sleep / standby) power levels. VMware has a sub-system called VMware Distributed Power Management (DPM) to decrease the power utilization of the server pool by shutting down the server. Critical groups include AFs that are in an energy saving situation.

2) Active Group: Virtual machines need a lot of resources throughout preliminary unit to establish the guest operating system and the host software. In the active group, this work remain the amount of inactive servers in the online state of which t is a low value, straight relative to the frequency of deploying the virtual machine's client. Each virtual machine is initially deployed to the appropriate officer for the active group. When the virtual machine is deployed at the SM, it turns it into a virtual machine and sends it to the corresponding SM of the selfish group.

3) Selfish Group: Any VM can be placed in SM if the status is approved. This means that after the SM resource allocation, it should not exceed previous allotted count . The selfish group has such primers whose resources are not allocated to helmets. If the resource allocation to the SM resource tree is automatically transferred to the overflow group.

4) OverFlow Group: This class contains SMs whose previous allotted count value is exceed the target allocation count. The machine of this class is deemed complete or complete, and no more virtual machines will be placed on such SMs.

Algorithm I Initial Placement or Virtual Machines

Input: S - Total Amount of SM on Data Center
V - Total Amount of VM to be installed
t — threshold value
TAC -Target Allotted Count
PAC - Previous Allotted Count
Output: m VMS will be placed in am'ropriate SMS
for 1 to S **do**
while 1 to V **do**
Seek suitable SM IBFD using in target class
if SM not found **then**
Turn on next SM from critical group
end if
Install VM on SM
if VM is available **then**
Seek suitable SM IBFD using in selfish group
if PAC < TAC **then**
Install VM to SM
else
Shift SM to selfish group
end if
if PAC < TAC **then**
Shift SM from critical group to active group
end if
end if
end while
end for

Algorithm 1 illustrates the process for restarting the virtual machine from eeadoSelfCloud. Each VM was first created in the appropriate SM of the active group. VM delivers the necessary resources from SM Pi and adjusts the operating system and applications. If there is not enough resources in the SM for each of them, then it will find the out-going request who meets the status of the acceptance and sends it to the state of the VM by sending it to the active group. Then VM is inserted into the new SM offset. When VM completes the work of creating the operating system and applications, it changes into a selfish group. However, it is impossible that none of the upper-level cabinet greed faces a welcome state. In such a condition, the target group itself will change into greedy people. A greedy machine that has previous allocated count exceeds the target allocation count has been promoted to overflow group. These fitness of physical machines have been declared complete because the remaining of resources allocation have been saved to increase resources quickly and efficiently in the future. The number of SMs in the active group is always maintained until a certain value to keep time for an critical VM group. Therefore, if the number of SMs in the active groups falls below the low-cost value, one of the critical SMs will be entered and promoted in the active group.

BLO algorithm

The requested operators in the previous section can define the optimization rules for the BLO. The BLO algorithm is defined as a triangle function that best resembles the optimal solution for the following optimist problems:

BLO(E,F,G) (1)

where E is a function that generates the random initial solutions, F manipulates the initial population provided by the function E, and G gives accurate when the finish condition is fulfilled. The functions E, F, and G are defined as follows:

$$\partial \vec{A} \{P_{Bee}, P_{OB}, P_{Beelion}, P_{OBL}\} \quad (2)$$

$$\{P_{Bee}, P_{Beelion}\} \vec{A} \{P_{Bee}, P_{Beelion}\} \quad (3)$$

$$\{P_{Bee}, P_{Beelion}\} \vec{A} \{true, false\} \quad (4)$$

where P_{Bee} is the matrix of the position of bees, $P_{Beelion}$ includes the location of beelions, P_{OB} includes the consequent fitness of bees, as well as P_{OBL} has the fitness of beelions. The pseudo codes the BLO algorithm are described as follows:

Initialize the first population of ants and beelions randomly
Calculate the fitness of ants and beelions
Discover the finest beelions and assume it as the elite (determined optimum)
while the finish condition is not fulfilled
for each bee
Choose an beelion with Roulette wheel
Renew f and g
Produce a random walk and normalize it
Renew the location of bee
end for
Compute the fitness of every bees
Change an beelion by its equivalent lion it if grow to be fitter
Renew elite if an beelion grow to be fitter than the elite
end while
Return elite

In BLO algorithms, beelion and bees were started randomly by using E-function. In referring to the function F, update the position of the relationship to the ear selected by the wheel and the elite operator. The Location of bees is pre-determined in proportion to the current count. The location for the update was made by two randomly around the white and selected dignitaries. When the bees fly accidentally, they are evaluated by physical function. If a bee is better than anything, their position is considered a new position for beelions in the next talk. The best beelions are compared to the best beelions detected during Optimization and, if necessary, as needed. These steps are times, while G functions return false.

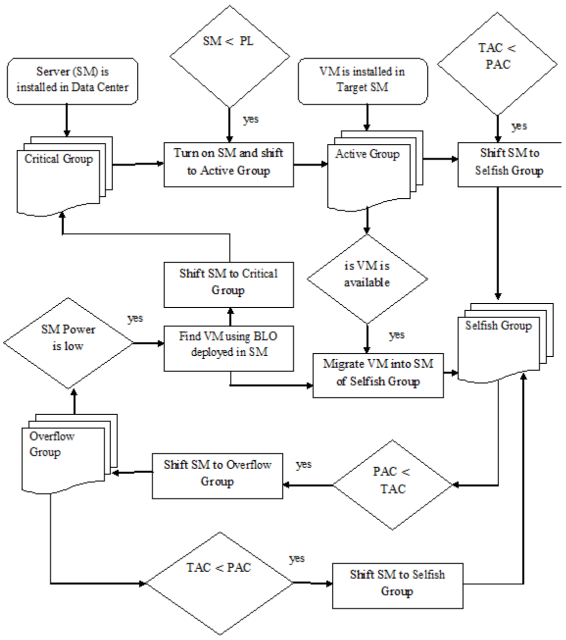


Fig.1Architecture Diagram of Proposed eeadoSelfCloud VM Allocation

The virtual machine allocation process is divided into two steps: In the first step virtual machines are selected to move in the second stage, the selected virtual machine is placed on the machine using Algorithm 1. To determine if the virtual machine needs changes, we can enter two selective levitation rules. The basic idea is to set high and low for machine usage and keep the CPU from all VMs distributed between hosts between these levels. If CPU usage is below the threshold, all virtual machines must be switched from this machine and the machine must be inserted into sleep mode to eliminate battery consumption. If using too high, some virtual machines are switched off from the machine to reduce the usage. The goal is to keep free resources to prevent violations of the SLA as a result of merging when the virtual machine is growing. The difference between old and new position creates a set of virtual machines that need redistribution. This new location has been achieved through the new Energy Efficient Dynamic Resource Allocation (EEDRA) based on local and global research scheme.

This algorithm is the main detection algorithm that searches nodes in a set of rules and certifications. The request for a virtual host throttle across a search network of a node that meets its resource needs. We use the abbreviation query terms for requesting a VM placeholder in the remaining files. The nodeAgent query gets selected the potentially best node with enough capacity to host the VM based on the local saved information. If the selected node is a neighbor rather than the node that receives the request, nodeAgent will send a request to that neighborhood to search for the desired resource. NodeAgents sends a query from a potential solution to better when nodeAgent is not able to find a better solution than it or the query expires. There are six policies that can be used by nodeAgents, and best-of-breed viruses are selected by a Bee Lion Optimization (BLO) algorithm. Six initial VM principles, including.

1) **Most- Exploitation:** Select the most used node with enough capacity from its neighbor (including itself). Using nodes is a proportion of resources used to their total capacity.
2) **Least-Exploitation**: Select the node that is used at least sufficiently from its neighbor (including itself).
3) **First-Fit:** Select the first node that is capable enough from its neighbor (including itself).
4) **Minimization of Migrations:** Immigration Reduction Policy (MM) selects the minimum number of virtual machines needed to switch from the machine to reduce the CPU usage below the high bandwidth level if the upper level is violated. Allow Mj to be a set of virtual machines that are currently distributed to host j. Then N (Mj) is the energy of Mj. The MM policy found the set ∈ N (Mj) set in (3).

$$MM = \begin{cases} \left\{ \frac{T}{T} \in N(M_j),\ d_j - \sum_{m \in T} d_a(v) < L_d \right. & \\ \left. |T| \rightarrow min \right\}, & if\ d_j > L_d; \\ M_j, & if\ d_j > L_o; \\ \emptyset, & otherwise \end{cases} \quad (3)$$

Where Ld is the higher limit. Lo is lower than dj, using the current CPU. And $d_a(v)$ are part of the CPU utilization used for the VM.

5) **Highest Potential Growth :** When the higher level is violated, the HPG policy changes the virtual machine that has the lowest CPU usage to the CPU capability defined by the VM parameters to reduce the potential increase in host use and prevention official SLA (4).

HCP =

$$\begin{cases} \Big\{ \frac{T}{T} \in N(M_j),\ d_j - \sum_{m \in T} d_a(v) < L_d \\ \sum_{m \in T} \frac{d_{a(m)}}{d_{r(m)}} \rightarrow min \Big\}, & if\ d_j > L_d; \\ M_j, & if\ d_j > L_o; \\ \emptyset, & otherwise \end{cases} \quad (4)$$

where $d_{r(m)}$ is the portion of the CPU capacity at first requested for the VM m in addition to describes as the VM's parameter.

Table1: VM Allocation Policies based on Different Entry and Heuristics

Policies	Request Entry	Schemes
Dist-Min	Distributed	Min Exploitation
Dist-FF	Distributed	First-Fit
Dist-Max	Distributed	Max Exploitation
Dist-MM	Distributed	Minimization of Migrations
Dist-HPG	Distributed	Highest Potential Growth
Dist-RC	Distributed	Random Choice

6) **Random choice:** The Random Choice Policy (RC) is based on a random selection of the number of virtual machines needed to reduce the CPU usage of the host under the boot threshold. According to the uniformly divided randomness (Y), which has the index of the value of Mj, the set of policy choices RCP ∈ N (Mj), as shown in (5).

$$RCP = \begin{cases} \Big\{ \frac{T}{T} \in N(M_j),\ d_j - \sum_{m \in T} d_a(v) < L_d \\ Y = D\big(0, \big|N(M_j) - 1\big|\big)\Big\}, & if\ d_j > L_d; \\ M_j, & if\ d_j > L_o; \\ \emptyset, & otherwise \end{cases} \quad (5)$$

Where Y is the randomly divided random variable used to select a subset of M_j.

If nodeAgent and one of its neighbors offer the same benchmark value, the algorithm will select Neighborhoods through NodeAgent to increase the chances of finding better solutions in future visits, and ultimately Skip basic optima. If no neighbors or NodeAgent itself has enough capacity, nodeAgent will send a request to a neighbor. So if the query is stored in the neighborhood without the available resources, random selection may be able to skip requests between the nodes and the exit from the neighborhood, so that the search can continue. Every time a program is redirected to a nodeAgent, it is considered to have taken a one hop. The total number of changes needed to find successful nodes is the time it takes to process the request. We schedule the processing of requests using the highest miss (HMR) ratio. This request was rejected if the no of hops above HMR. Limiting the number of hops decreases the quality of the resulting solution and increases the number of rejections. However, it also protects the unlimited number of questions. We also assume that the queries can reach the system from multiple input points ("distribution records"). In these cases, they are distributed from their entry points according to the same law. Therefore, there are six types of validation and validation policy, as shown in Table I.

Algorithm 2 VM Allocation

Input: Allocation Request [Insist, HMR, ResultantIndex]

Output: The place to put the allocation request, ResultantIndex

When receiving the request:

if ResultantIndex = false **then**
  if HTL > E then
    f = detect the VM with IBFD
    if f != Self **then**
    if f== -1 **then**
    f = arbitrary VM from the local of VM
    end if
    HMR=HMR- 1
    send the request to f
    place = f
    ResultantIndex = true
    end if
    place = - l
    ResultantIndex = true
  end if
end if
return place and ResultantIndex

IV. RESULT AND DISCUSSION

A. Experimental Setup

This section describes an appraisal of the proposed method implementation by simulating the data center with 2500 physical nodes. We have faked our framework in the middle of CloudSim and built our eeadSelfCloud. Each node is connected to nodeAgent and is supposed to use a virtual machine of different types. The maximum capacity of each node is limited to 10 PCs, and the carrier provides 10 VMs with 1 to 10 PCs. We've received a total of 20,000 requests for virtual machines that come into the system after Poisson's arrival. Each request has a capacity requirement that is accidentally selected from [1, ..., 10] set and is matched to the VM type.

Virtual machines are deployed and stopped during each experiment. Cloud-based services are of lesser duration, so we have taken the VM's life pattern by using randomly assigned variants to eliminate the potential gap potential associated with a particular type of program and to show a variety of applications, Which can be deployed in a cloud database. To avoid relaying requests to the environment, we limit the number of hops for HTL = 20 hops. The level of merger again. The burden that resources are supposed to be reunified is determined at 40% of the total node capacity. Experiments are approved to execute for 20000 units, each time is 0.1 seconds of testing and 1 hour of resource utilization.

B. Performance Parameters

In this section the proposed approach is evaluated by using the following evaluation metrics:

*1) **Data Center Utilization:** This variable is to use analysis how much data center is utilized. It is determined to be the total capacity used by the VM to divide the time based on the total capacity contained in the data center.*

Data center usage is found by equation (1),

$$DCU(t) = \frac{\sum_{k=1}^{N} Req_j(t)}{T \ X \ P_{server}} \quad (1)$$

Where N is the amount of placement requests, $Req_j(t)$ is the owed ability for $Request_j$ at time t, T is the full amount of servers, in addition to P_{server} is the capacity of each server.

2) Average Node Utilization: This metric is evaluated by using Equation (2),

$$P_{nodes}(t) = \frac{\sum_{k=1}^{N} Req_j(t)}{T_{active} X \ P_{server}} \quad (2)$$

where T_{active} is the amount of active nodes in the surroundings. This meter gives the concept of distribution, distribution, and the way an active server is currently being used. It is straight relative to the use of energy in the system and related values.

3) Amount of hops: This is the amount of hops needed to find the available nodes needed from the VM request. It calculates how fast the technique is correct and responsive to questions and be able to be evaluated to the calculation period in the collection method.

4) Request Rejection ratio: This is metric is used to find the total no of request is rejected in cloud.

$$RRR(t) = \frac{\sum_{k=1}^{N} Demand_j(t) - \sum_{k=1}^{N} Req_j(t)}{\sum_{k=1}^{N} Demand_j(t)} \quad (3)$$

Rejections can happen for a variety of reasons, including: • Failure to find appropriate resources within HTML limits. Lack of sufficient resources.

5) Power Consumption:

The metric is calculated by using the equation (4)

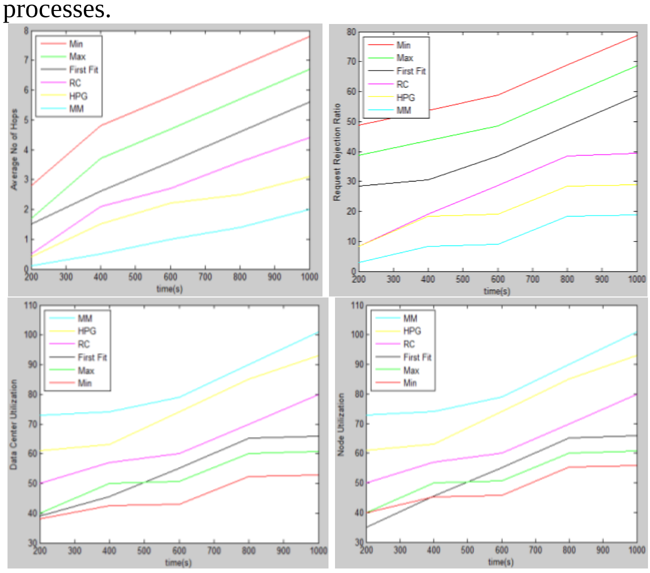
Power Consumption = Total Power - Residual Power (4)

C. Result and Discussion

To analysis the performance of the classifier system, it is compared with various techniques by using the performance metrics which are mentioned above. This is shown in the below tables and graphs.

i) Experiment No #1 : Performance Analysis on Impact of Request Propagation

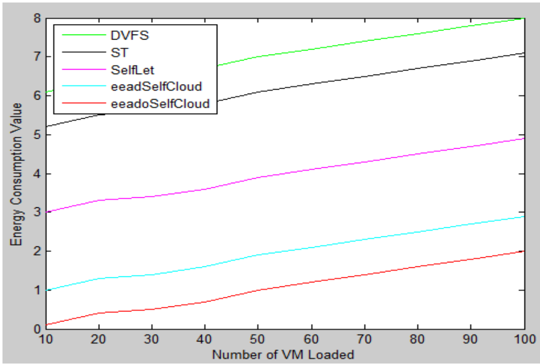
The initial experiment examines the impact of the distribution on the distribution of resources. In order to decide how the Input Policy influences system performance, we compare the effects of implementing a central policy for integrating and distributing virtual machine queries to system processes.



From the graph, it is shown that the average no of hop count value is lesser in Minimum Migration Policy. As well as the request rejection value is also lesser in Minimum Migration Policy. Data Center Utilization value is higher for Minimum Migration Policy than the other policies. Minimum Migration Policy utilizes the nodes in higher level. So Minimum Migration Policy is considered as the best VM migration policy than other policies.

ii) Experiment No #2 : Performance Analysis of Power Consumption based VM Load

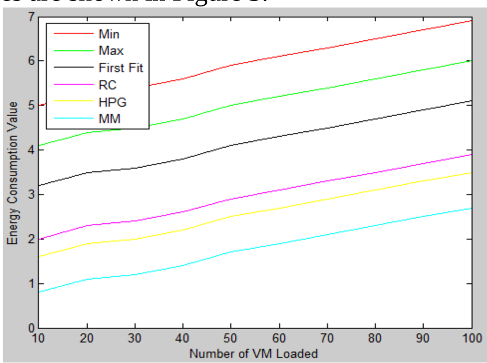
In the second experiment, the energy usage is analysed based on the VM. The use of energy at the data center and the failure rate of the VM deployment has been taken into account as a parameter for the study. Energy consumption is the total power utilization in all server data centers after the deployment of the desired virtual machine. The level of failure in the VM deployment is a number that illustrates the number of Prime Minister's unsuccessful choices for deploying the VM. The experimental results obtained from algorithms for using VM for medium power consumption after the deployment of 100 virtual machines are shown in Figure 3.



From the graph, it is shown that the eeadoSelfCloud consumes less energy than the other VM allocation mechanism. So it is considered as the best VM allocation mechanism for data center in self organization cloud.

iii) Experiment No #2 : Performance Analysis of Power Consumption based Policy

In the third experiment, power consumption based on the various VM engine policies is examined. The experimental results obtained from algorithms for using VM for medium power consumption after the deployment of 100 virtual machines are shown in Figure 3.



From the graph, it is shown that the energy consumption value is less than the other policies. So Minimum Migration Policy is considered as the best VM migration policy for data center in self organized cloud.

V. CONCLUSION

Among other cloud computing issues, virtual machine performance is a major problem. This is because different users require different resources at different times from different locations. Their request is based on specific virtual machines at a particular time. Therefore, an effective method of deploying virtual machines is a requirement for a cloud environment. It should be so that it not only improves resource efficiency, but also reduces the time allocation of resources and reduces energy. In this work, the proposed eeadoSelfCloud method presents an autonomous and energetic resource management solution by deploying virtual machines in a cloud database. The experimental results are obtained using the simulator model for the virtual machine initiative. It is suggested that proposed approaches appear to be more effective than complementary measures. As a future job, experimental models can be expanded to measure the dynamics of the proposed Dynamic Resource Allocation algorithms for virtual machines at the data center. Issues of resource allocation may also be taken into consideration for the further development of the proposed algorithm for better resource utilization..

REFERENCES

1. GRamesh, C., Shalini, S., Suganya, S. and Suvitha, N. "An Approach for Efficient Dynamic Memory Allocation Using Skewness Algorithm in Cloud Environment", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 4, No 2, Pp. 1648-1653, 2016.
2. Buyya, R., Yeo, C.S., Venugopal, S., Broberg, J. and Brandic, I. "Cloud computing and emerging IT platforms : Vision. Hype, and Reality for Delivering Computing as the 5th Utility", Future Generation Computer Systems, Vol. 25, No. 6, Pp. 599-616, 2009.
3. Rani, B.K., Rani, B.P. and Babu, A.V, "Cloud computing and inter-clouds - Types, topologies and research issues", Procedia Computer Science, Vol. 50, Pp. 24-29, 2015.
4. E. Feller, L. Rilling, and C. Morin, "Energy-Aware Ant Colony Based Workload Placement in Clouds," 2011 IEEE/ACM 12th International Conference on Grid Computing, 2011. [Online]. Available: http://ieeexplore.ieee.org/document/6076495/
5. Q. Wu, F. Ishikawa, Q. Zhu, and Y. Xia, "Energy and migration cost-aware dynamic virtual machine consolidation in heterogeneous cloud datacenters," IEEE Transactions on Services Computing, vol. 1374, no. c, pp. 1–1, 2016. [Online]. Available: http://ieeexplore.ieee.org/document/7589069/.
6. G. Xu, Y. Ding, J. Zhao, L. Hu and X. Fu, "A novel artificial bee colony approach of live virtual machine migration policy using bayes theorem," The Scientific World Journal. 2013.
7. A. Mevada, H. Patel and N. Patel, "Enhanced Energy Efficient Virtual Machine Placement Policy for Load Balancing in Cloud Environment," Int J Cur Res Rev, vol. 9(6), 2017.
8. N. J. Kansal and I. Chana, "Energy-aware Virtual Machine Migration for Cloud Computing - A Firefly Optimization Approach," Journal of Grid Computing, vol. 14, no. 2, pp. 327–345, 2016. [Online]. Available: http://dx.doi.org/10.1007/s10723-016- 9364-0
9. F. Farahnakian, A. Ashraf, T. Pahikkala, P. Liljeberg, J. Plosila, I. Porres, and H. Tenhunen, "Using Ant Colony System to Consolidate VMs for Green Cloud Computing," IEEE Transactions on Services Computing, vol. 8, no. 2, pp. 187–198, 2015.
10. L. Adhianto, S. Banerjee, M. Fagan, M. Krentel, G. Marin, J. Mellor-Crummey, and N. R. Tallent, "Multi-objective, Decentralized Dynamic Virtual Machine Consolidation using ACO Metaheuristic in Computing Clouds," Concurrency Computation Practice and Experience, vol. 22, no. 6, pp. 685–701, 2017.
11. M. Marzolla, O. Babaoglu, and F. Panzieri, "Server consolidation in clouds through gossiping," 2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM 2011 - Digital Proceedings, 2011.
12. S. Wang, A. Zhou, C. H. Hsu, X. Xiao, and F. Yang, "Provision of Data-Intensive Services Through Energy-and QoS-Aware Virtual Machine Placement in National Cloud Data Centers," IEEE Transactions on Emerging Topics in Computing, vol. 4, no. 2, pp. 290–300, 2016.
13. K. Rybina, W. Dargie, A. Strunk, and A. Schill, "Investigation into the energy cost of live migration of virtual machines," The Third IFIP Conference on Sustainable Internet and ICT for Sustainability (SustainIT 2013), pp. 1–8, 2013.
14. A. Strunk, "A lightweight model for estimating energy cost of live migration of virtual machines," IEEE International Conference on Cloud Computing, CLOUD, pp. 510– 517, 2013.
15. K. Rybina and A. Schill, "Estimating Energy Consumption during Live Migration of Virtual Machines," pp. 1–5, 2016.
16. Jia Zhao., Liang Hu., Yan Ding., Gaochao Xu., Ming Hu.(2014) 'A Heuristic Placement Selection of Live Virtual Machine Migration for Energy-Saving in Cloud Computing Environment', PLOS ONE Vol.9 No.9, pp.1-12.
17. Thiruvenkadam, T.,Karthikeyani, V. (2014) 'A Comparative study of VM Placement Algorithms in Cloud Computing Environment' inSARC-IRF: Proceedings of 7 th SARC-IRF International Conference, New Delhi, India, pp.11-14.
18. Arnab Kumar Paul., Sourav Kanti Addya., Bibhudatta Sahoo., and Ashok Kumar Turuk. (2014). 'Application of Greedy Algorithms to Virtual Machine Distribution across Data Centers.' in India Conference (INDICON) 2014 Annual IEEE, Pune.
19. Jeyarani R, Nagaveni N, Vasanth RR (2011) Self Adaptive Particle Swarm Optimization for Efficient Virtual Machine Provisioning in Cloud. International Journal of Intelligent Information Technology 7(2): 25–44.
20. Gaochao X, Yan D, Jia Z, Liang H, Xiaodong F (2013) A Novel Artificial Bee Colony Approach of Live Virtual Machine Migration Policy Using Bayes Theorem. The Scientific World Journal 2013(2013): 369209.
21. M. Cardosa, A. Singh, H. Pucha, and A. Chandra, "Exploiting spatio-temporal tradeofs for energy-aware MapReduce in the cloud," IEEE Transactions on Computers, vol. 61, no. 12, pp. 1737– 1751, 2012.
22. C. C. T. Mark, D. Niyato, and T. Chen-Khong, "Evolutionary optimal virtual machine placement and demand forecaster for cloud computing," in Proceedings of the 25th IEEE International Conference on Advanced Information Networking and Applications, AINA 2011, pp. 348–355, Singapore, March 2011.
23. T. Cerling, J. Buller, C. Enstall, and R. Ruiz, Mastering Microsof Virtualization, Wiley Publishing, Inc., Indianapolis, IN, USA, 2009.
24. M. Chen, H. Zhang, Y. Y. Su, X. Wang, G. Jiang, and K. Yoshihira, "Efective VM sizing in virtualized data centers," in Proceedings of the 12th IFIP/IEEE International Symposium on Integrated Network Management, pp. 594–601, Dublin, Ireland, May 2011.

AUTHORS PROFILE

Mr.Jason Sebagenzi is a research scholar in Jain University ,Bangalore . He has published more than 8 papers in national and international journals . His area of specialisations are Cloud Resource management and Software engineering . He is a java programmer and give seminar in various institutions on cloud computing

R.Suchithra is the director of MCA department in Jain Deemed to be University,Bangalore . She has 16 years of experience in teaching and research .She has published more than 24 papers in reputed journals and her area of specialization is on cloud computing , machine learning and data science . She is guiding research scholars in the PhD programme