# Energy Aware Scheduling using Genetic Algorithm in Cloud Data Centers

Ipsita Kar
School of Computer Engineering
KIIT University
Bhubaneswar, India
ipsitakar0@gmail.com

R.N. Ramakant Parida
School of Computer Engineering
KIIT University
Bhubaneswar, India
rk.parida@gmail.com

Himansu Das
School of Computer Engineering
KIIT University
Bhubaneswar, India
das.himansu2007@gmail.com

*Abstract*— **Cloud computing is a sophisticated technology which provides services to the end users within a fraction of time based on a pay-as-you-go model. However, energy consumption is one of the important concerns in cloud environment as large amount of energy is wasted by the data centers, hosting cloud applications and the carbon dioxide gas is released into the atmosphere polluting the environment. So pollution needs to be reduced by lowering the energy usage. Therefore, we'd prefer to propose a green computing solution which is not only able to minimize makespan and operational costs however in addition to minimize the environmental pollution. In this paper, we tend to stipulate a framework that specifies energy minimization is a generalization of makespan minimized by using the Energy-Aware Task Scheduler using Genetic Algorithm. We tried to conduct a survey on different scheduling methods of energy minimization in cloud data centers and their limitations and at the end of this paper, we render a green optimized energy-aware task scheduling algorithm for Cloud data centers.**

***Keywords— Cloud Computing ; Energy-aware ; QoS ; Genetic Algorithm.***

## I. INTRODUCTION

With the advancement of technology the requirement of deploying basic hardware and software infrastructures in the IT companies is increasing day by day. The Cloud provides substantial advantage to the sophisticated Information Technology companies by relieving them from the requirement of deploying basic hardware and software infrastructures and thus facilitating more concentration on innovative and creative ideas for their services. Furthermore, developers with advanced brains of new Internet services don't need huge capital expenditures in hardware to set up their business or human outlays to run it. In cloud environment, resources and tasks need to be allocated not only to satisfy quality of service (QoS) requirements specified by users via service level agreements (SLAs) but also to reduce makespan and energy consumption. As there are various requests submitted by consumers of cloud, there exists an intensive demand for improved scheduling technique for proper allotment of tasks to assets within minimum expenditure of time and energy [7]. Cloud contributes to the deployment of huge data centers that are responsible for the energy conserved worldwide and consequently the carbon emission and environmental drawbacks. A survey proves that servers in data centers are responsible for an instant growth of energy consumption and in 2010 it was approximated to contribute 1.1 to 1.5 percent of the entire electrical energy usage. In accordance with the trends the power requirement by data centers grows at 18% rate per annum. Energy conservation is recorded as the second largest operating cost for data centers in 2007 [3]. Furthermore, the carbon dioxide emitted by the IT industry contributes to 2 percent of the global emissions [12]. Data centers are not only expensive, but also harmful to the environment.

Cloud computing is a novel intelligence computing technique that provided as a utility, has the potential to transform a large component of the advanced IT industry, developing software more productive as a service. Previously, high performance has been the one and only important job in data center placements, and this prerequisite has been achieved without giving much concentration to the energy usage. According to a report on "Revolutionizing Data Center Energy Efficiency", A typical data center conserves energy as total amount of energy consumed in 25,000 homes. Therefore, we need Energy efficient solutions for Cloud computing that can not only minimize usable costs but also reduce the environmental impact. In this work we have surveyed some previously used algorithms and tried to propose a new scheduling algorithm using Genetic algorithm that can efficiently diminish the energy utilization for operating jobs. We provide a green optimized energy-aware scheduling algorithm using the Genetic algorithm for Cloud computing datacenters.

The remainder of the paper is organized as follows: Section 2 describes the related work, Section 3 explains the problem statement, Section 4 describes the proposed model, Section 5 describes the methodology used in our approach, Section 6 describes the Experimental Analysis done in the research work and finally in Section 7 conclusions and suggestions for future work are provided.

978-1-4673-9939-5/16/$31.00 ©2016 IEEE

## II. RELATED WORKS

Scheduling of tasks in virtual machines with minimum expenditure of time and energy is a major concern in cloud computing environment. However among various scheduling algorithms Genetic Algorithm have been proven to give optimized results than any other algorithms.

In [6] the authors provided a novel approach that demonstrates energy-aware scheduling is an abstraction of the minimum makespan scheduling problem. By considering the system graph and program graph as inputs they provided three different energy-aware scheduling algorithms. They verified the algorithms on well-known program graphs and compared their answers with other existing algorithms, which confirms the efficiency of their approach. Their method can be extended to a monetary value-aware framework with the involvements of cost elements.

In [7] as heterogeneous requests submitted by consumers of cloud, which are set up by the available assets, there exists a requirement for improved scheduling technique for proper allotment of tasks to resources within minimum expenditure of time and energy. The authors proposed a genetic algorithm based scheduling technique where the population initialization is produced with an advanced form of Max-Min by which they obtained optimized values in relations with makespan. They evaluated the execution of the MGA (Modified Genetic Algorithm) and other existing algorithm and compared with the existing data and the output showed that the proposed algorithm outperforms the existing algorithms. However the proposed algorithm can also be applied to several QoS parameters like cost of service, energy, security etc.

In [8] the authors proposed an energy-aware task consolidation (ETC) method that diminishes energy utilization by refraining CPU use below a particular threshold value. This is done by consolidating tasks amongst virtual clusters. Ching-Hsien et.al compared ETC with MaxUtil, the results show that ETC reduces power consumption with 17% improvement over MaxUtil. The data center for Virtual Machines in ETC is designed to exist in the same rack where network bandwidth is approximately constant. So it would be better for the VMs to consider fluctuations in network bandwidth.

In [9] the authors proposed a genetic algorithm to efficiently allocate tasks to virtual machines, which allocates resources based on available resources and the energy consumption of each virtual machine. The evaluation results show that the proposed algorithm gives better results than first-fit decreasing (FFD) and best-fit decreasing (BFD) algorithms. This method needs to be optimized with other algorithms, as well as considering QoS parameters in the cloud environment.

In [10] the authors model their problem as a parameter optimization problem. The objective is to reduce the carbon footprint produced in the data centers by employing sustainable energy while fulfilling: (1) the request processing time constraint; (2) the total electricity budget in each time slot; (3) the intermittent supply of the renewable resources; (4) the maximal number of servers in each data center. The solution to the above stated problem is creatively converting it into an ILP model, and computing the decision variables using existing method. Computations indicate that the scheduler can diminish carbon emissions using sustainable assets, while fulfilling the conditions stated above.

The predictable utilization of energy by not only personal computers but also business domain computers in the United States was about 2% out of the total electricity utilization in the year 2010, which causes the Information Technology companies the second major carbon footprint provider after aeronautics [11]. In this research paper, the authors have proposed a novel approach for scheduling, sharing and migrating Virtual Machines (VMs) for a bag of cloud tasks which is built to diminish energy ingestion within definite period of execution time and high system output. This methodology is withdrawn from an Enhanced First Fit Decreasing (EFFD) algorithm combined with our VM reuse strategy. Moreover, a virtual machine migration technique is developed to dynamically control the cloud position for required migration. The simulation results obtained using the easily accessible Cloud Report tool demonstrate that EFFD with the VM reutilization scheme could achieve a higher resource utilization rate and lower energy conservation than regular Round Robin (RR), Greedy, and FDD without Virtual Machines reuse.

Finally from the related works we conclude that Genetic Algorithm can be used to find proper schedule in which tasks are executed with minimum time so we can utilize Genetic Algorithm inorder to find schedule in which tasks are deployed in virtual machines with lesser energy consumption.

## III. PROBLEM STATEMENT

In accordance with the technological advancement in cloud computing environment, service requests have numerous resource requirements. Cloud resources required to be apportioned inorder to fulfill Quality of Service (QoS) demands stated by customers via Service Level Agreements, as well as to lessen the utilization of energy and enhance the income of the providers. In related works many researchers [6-7][9] have used a Genetic Algorithm for minimizing makespan however we are focused not only minimizing makespan but also minimizing energy conservation in cloud data centers.

The main purpose of our proposed model is to allocate each task to a specified machine as a result , all other tasks are finished with approximately minimum expenditure of energy, this is evaluated by the fitness function which is a result of mapping from each schedule to the energy conserved by the servers in data centers. This can be done by calculating the energy consumption of each virtual machine after the task is allocated to a virtual machine.

The energy consumption can be measured by using the following formula $E_{ij}=p_i t_i$ where $p_i$ is the power consumption of each task allocated to virtual machines and $t_i$ is the execution time of each job allocated to respective virtual machines.

Assuming the total energy consumed when a task is allocated to a Virtual Machine to be $\tau_{ij} = 70kWh$ such that if the energy consumed by allocation of task to a virtual machine is less than $\tau_{ij}$ then that particular genome in the chromosome is represented by 1 and if the energy consumed is more than $\tau_{ij}$

then that particular genome in the chromosome is represented by 0.

*A. Objective Function*

Consider job $a_i$ is organized to be executed on a cloud $C_j$ which consists of several virtual machines $VM_N$, where $p_i$ represents the power of each VM in Hosts and $t_i$ represents the execution time of the job for virtual machines then the energy consumption for execution of $a_i$ is given by:

$$E_{ij}=p_i t_i \quad (1)$$

We need to minimize energy consumption in virtual machines when a task is allocated to them. Minimize $E(x) = min\ ( \sum^{M}_{i=1}\sum^{N}_{i=1}E_{ij})$ (2)

$$s.t. \quad E_i(x) <= \tau_{ij,\ 1}$$

$$E_i(x) >= \tau_{ij,\ 0}$$

*B. Necessary Conditions:*

The necessary conditions are stated as follows:

1) The job $a_i$ should be completed within the deadline $d_i$, otherwise, it is concluded that the schedule is failed.

2) Assume $\tau_{ij} = 70\ kWh$
where $\tau_{ij}$ is the minimum energy to be consumed when a task is assigned to a virtual machine.

## IV. PROPOSED MODEL OVERVIEW

In the suggested proposed model, when a business is processed as an accumulation of tasks which takes a complicated mathematical process by utilizing useful resources available in cloud data centers, and the set $A= (a1)$ $(a2)$………… $(a_n)$ is a collection of tasks submitted in a specified time period. During this particular scheduling process, the user submits a service request for job, $a_i$ (1<=i<=M) with the resource requirements represented as a triplet $(t_i,n_i,d_i)$, where, $t_i$ represents the execution time of the job for virtual machines (VMs), $n_i$ represents the number of VMs needed for $a_i$ and $d_i$ for the deadline after which the tasks in the schedule will be concluded to be failed. The problem needed to be solved is how to arrange these M tasks to the given N machines under the necessary conditions with optimized objective function.

When different cloud users submit their respective requests they are submitted to the cloud service provider and then to the cloud service manager and finally scheduled to the Virtual machines with the help of energy-aware task scheduler which is implemented in cloud service manager. In this proposed model various components are described as follows:

*A. Cloud Service Provider*

It is responsible for provisioning the requests submitted by the users with several resource choices. In the subsequent step, the Cloud Service Manager (CSM) will interpret the resource requirements of each granularity and maps it on to optimized virtual machines to achieve an efficient solution.

*B. Cloud Task Manager*

It is responsible for managing the state of the available tasks (start, stop, cancel), defining the appropriate sequence of scheduling and allocating each job to suitable virtual machines under the help of the scheduler.

*C. Energy-aware Task Scheduler*

It is responsible for performing the process of scheduling algorithm.

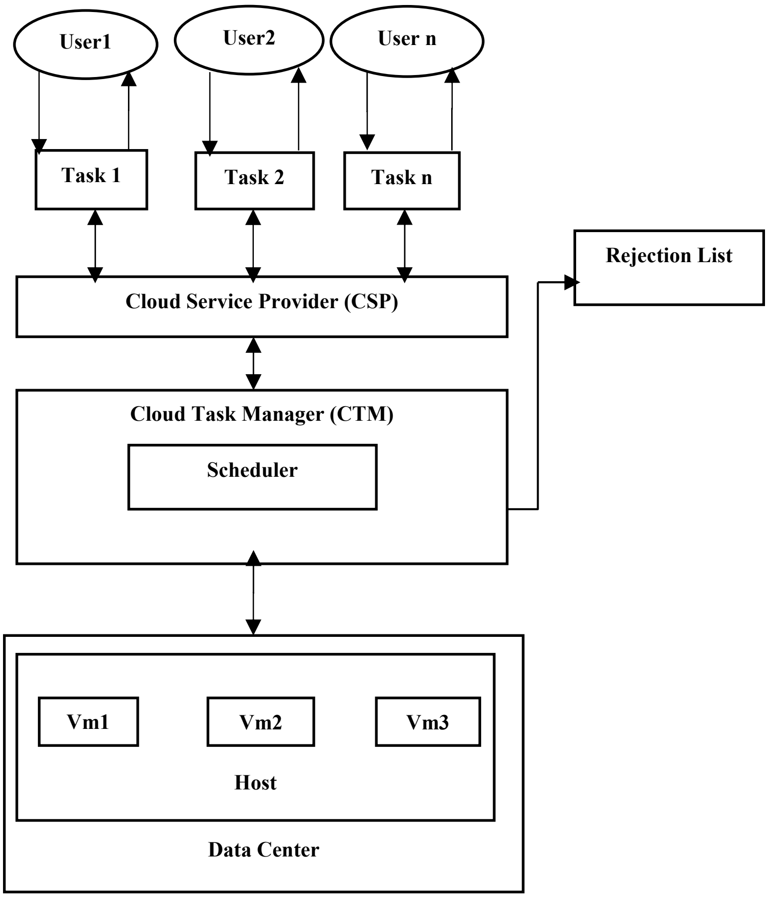


Fig. 1. Proposed Model for Energy-Aware Scheduling

## V. METHODOLOGY

To find an optimal solution for the aforesaid problem, we use genetic algorithm. The algorithms used for the problem are as follows:

*A. Genetic Algorithm*

The algorithm has the following steps:

*1) Initialization*

The procedure of the algorithm starts with the of the population initialization in which the collection of individuals is defined which denotes the approximate solution to the scheduling problem. These individuals are known as chromosomes.

In the procedure of scheduling, each chromosome is denoted as an arrangement of individual schedules for which it practices particular form of encoding. In the encoding scheme a chromosome in this Genetic Algorithm consists of genes, each of which stands for energy consumption of a task. The value of a gene is either 0 or 1 where 0 represents when the energy consumed by task allocation is greater than $\tau_{ij}$ and 1 represents when the energy consumed by task allocation is less than $\tau_{ij}$.

*2) Evaluation and Selection*

The fitness value of each chromosome is computed by considering fitness function (scheduling, goal) and all the genes in the chromosome are categorised by using these values. The appropriateness of the solution for the problem can be determined by the fitness function and then the best result will be selected for generating the subsequent generation.

*3) Crossover*

It is the genetic operator which can be used for selecting two parent chromosomes and exchanging their genetic features by swapping one or more genomes, so that improved quality offsprings can be developed in the subsequent generation, to discover the novel solution space which owns both their features.

*4) Mutation*

It is the genetic operator which can be used for randomly selecting a chromosome to change the structure of the genes in the chromosome by changing single bit of the parent chromosome with another bit thus it creates a new offspring.

*5)* Termination

After getting optimal solution by using crossover and mutation operators the algorithm stops further iteration when the algorithm reaches the stopping criteria.

### *B. Max-min Algorithm*

The minimum completion time is calculated for each task and that task with overall maximum completion time is selected and assigned to a corresponding machine. This technique finds the task with minimum execution time and assigns them to machine with maximum completion time. Then removes the task from the task set and updates the selected machines and this process continues until the task set is empty and all the tasks are scheduled. Though max-min has only been implemented in minimizing execution time we extend it to implement in minimizing energy as energy is directly proportional to time.

### *C. Process of Proposed Scheduling Algorithm*

Both the algorithms can also be used to find out which task to be assigned to the virtual machines in the host of cloud data centres with reference to minimization of the energy consumed in data centers deployed in cloud environment as energy is directly proportional to time. However, among these algorithms which one gives better result is very essential to be found out. So, these two algorithms are applied in our proposed scheduling algorithm where the Scheduler decides the better among the two algorithms by comparing the experimental results calculated manually.

Fig. 2. shows the process of ETSG(Energy-Aware Task Scheduling using Genetic Algorithm).

1. Begin
2. Receives a job and its resource requirements ($t_i, n_i, d_i$) from the user.
3. All tasks or requests are deposited into CSP.
4. Calculate energy value of each task by using $E_{ij}=p_i t_i$.
5. If energy value of task is greater than $\tau_{ij}$ it goes to CTM.
6. CTM monitors the tasks and sends all information to scheduler.
7. Scheduler schedules the tasks by checking the conditions and arrange them.
8. Scheduler applies Genetic Algorithm to allocate tasks to Virtual machines in the host.
9. If the task cannot be completed within its deadline it puts the task into rejection list.
10. End.

Fig.2. Process of Proposed ETSG Scheduling Algorithm

## VI. EXPERIMENTAL ANALYSIS

The aim of the proposed model is to find a solution to minimize the energy consumption while assigning tasks to virtual machines by using Genetic Algorithm. So we implemented the proposed objective function in this example using Genetic Algorithm.

### *A. An Example using Genetic Algorithm:*

The Energy Consumption matrix is calculated by using energy calculation formula mentioned in 3.3 equation (1).

TABLE I. ENERGY CONSUMPTION MATRIX FOR TASKS IN VMS

| | T1 | T2 | T3 | T4 | T5 | T6 | T7 |
|---|---|---|---|---|---|---|---|
| VM1 | 10 | 20 | 30 | 40 | 100 | 60 | 120 |
| VM2 | 12 | 80 | 20 | 80 | 50 | 61 | 120 |
| VM3 | 15 | 17 | 30 | 35 | 45 | 68 | 125 |
| VM4 | 16 | 20 | 35 | 94 | 23 | 123 | 48 |
| VM5 | 18 | 25 | 40 | 56 | 30 | 112 | 58 |
| VM6 | 20 | 37 | 45 | 45 | 86 | 110 | 88 |
| VM7 | 25 | 93 | 46 | 78 | 98 | 117 | 98 |
| VM8 | 28 | 40 | 84 | 90 | 120 | 119 | 108 |
| VM9 | 30 | 40 | 50 | 140 | 110 | 160 | 102 |
| VM10 | 40 | 58 | 80 | 76 | 109 | 115 | 119 |

*1) Chromosome Generation:*

As we have already mentioned about the constraints, so considering the constraints if the energy consumed by the task in a VM is more than $\tau_{ij}$ then it is represented as 0 otherwise represented as 1. Considering the minimum energy consumption $\tau_{ij}$ = 70 kWh.

*2) Objective Function:*

In this case Objective Function can be defined as $E=min(\sum_{i=1}^{M}\sum_{i=1}^{N}E_{ij})$ where M is the total number of tasks and N is the total number of Virtual machines.

*3) Fitness Function:*

To calculate the total number of ones which indicates the energy consumption value is less than the minimum energy consumed when a task is assigned to a Virtual Machines in the cloud and then map the binary digits into decimal numbers and calculate the energy consumption percentage by each virtual machine. So the initial population can be represented with considering the Fitness Function as shown in Table II .

TABLE II. BINARY ENCODING REPRESENTATION OF THE ENERGY CONSUMPTION MATRIX.

| Initial Population | Fitness |
|---|---|
| 1 1 1 1 0 1 0 | 5 |
| 1 0 1 0 1 1 0 | 4 |
| 1 1 1 1 1 1 0 | 6 |
| 1 1 1 0 1 0 1 | 5 |
| 1 1 1 1 1 0 1 | 6 |
| 1 1 1 1 0 1 1 | 6 |
| 1 0 1 0 0 0 0 | 2 |
| 1 1 0 0 0 0 0 | 2 |
| 1 1 1 0 0 0 0 | 3 |
| 1 1 0 0 0 0 0 | 2 |

To get the optimal chromosome, some sequences of operation using some special operators like crossover, mutation operator on the whole population, which helps to obtain best individual with minimum energy consumption of each task when assigned to a Virtual Machines.

*4) Selection*

Each chromosome comprises a fitness value which is computed by a fitness function and all other genes in the chromosomes are arranged by considering these values. The appropriateness of the solution for the problem can be determined by the fitness function and then the best result will be selected for generating the subsequent generation.

*5) Crossover*

Let's take first and second chromosome as parent so the operation is done as follows:

First parent 1 1 1 1 0 1 0 → 1 1 1 1 1 1 0 = Offspring 1
Second parent 1 0 1 0 1 1 0 → 1 0 1 0 0 1 0 = Offspring 2

Before Crossover After Crossover

In the above presentation as per the crossover rule, we select one random position on each parent, then swap the remaining part with each other. The above procedure will be continued for each random pair of parent in the initial population.

TABLE III. NEW POPULATION AFTER CROSSOVER

| Initial Population | Fitness | Population in Decimal |
|---|---|---|
| 1 1 1 1 1 1 0 | 6 | 127 |
| 1 0 1 0 0 1 0 | 3 | 82 |
| 1 1 1 1 1 1 0 | 6 | 126 |
| 1 1 1 0 1 0 1 | 5 | 117 |
| 1 1 1 1 1 0 1 | 6 | 125 |
| 1 1 1 1 0 1 1 | 6 | 123 |
| 1 0 1 0 0 0 0 | 2 | 80 |
| 1 1 0 0 0 0 0 | 2 | 96 |
| 1 1 1 0 0 0 0 | 3 | 112 |
| 1 1 0 0 0 0 0 | 2 | 96 |

*5) Mutation*

After the above crossover operation the mutation operation is performed. In this operation randomly a chromosome is selected and a bit position on it, then randomly change the bit with another bit thus it creates a new offspring.

Parent : 1 1 1 1 1 1 0 → Before Mutation
Offspring : 1 1 1 1 1 1 1 → After Mutation

Now fitness = 7. As we have considered the tasks as binary values by using binary encoding so here the string 1 1 1 1 1 1 0 is modified to 1 1 1 1 1 1 1 which indicates that the string is optimized and the tasks which are allocated to virtual machine consume energy less than the assumed energy i.e. $\tau_{ij}$.

TABLE IV. NEW POPULATION AFTER MUTATION AND ENERGY CONSUMPTION PERCENTAGE IN VIRTUAL MACHINE.

| New Initial Population | Fitness | Energy in kWh | Energy % For 7 tasks | Energy % For 1 task |
|---|---|---|---|---|
| 1 1 1 1 1 1 1 | 7 | 380 | 77.55% | 11.07% |
| 1 0 1 0 0 1 0 | 3 | 423 | 86.32% | 12.33% |
| 1 1 1 1 1 1 0 | 6 | 335 | 68.36% | 9.76% |
| 1 1 1 0 1 0 1 | 5 | 359 | 73.26% | 10.46% |
| 1 1 1 1 1 0 1 | 6 | 339 | 69.18% | 9.88% |
| 1 1 1 1 0 1 1 | 6 | 431 | 87.95% | 12.56% |
| 1 0 1 0 0 0 0 | 2 | 555 | 113.26% | 16.18% |
| 1 1 0 0 0 0 0 | 2 | 589 | 120.20% | 17.17% |
| 1 1 1 0 0 0 0 | 3 | 632 | 128.97% | 18.42% |
| 1 1 0 0 0 0 0 | 2 | 597 | 121.83% | 17.40% |

*6) Termination*

The above procedure continues 10 numbers of iterations thus we get optimal chromosome set with better fitness, means strings with minimal number of zeros as it is considered that strings with minimum or no zeros are with best fitness. Finally the assignment of tasks to a Virtual Machine is done thus to get the optimal solution.

### B. An Example using Max-min algorithm

The minimum completion times set is calculated for each task and that with overall maximum completion time is selected and assigned to a corresponding machine.

By using a max-min algorithm the energy consumption by allocation of a single task to the machine is calculated in Table V.

TABLE V. ENERGY CONSUMPTION FOR A TASK ALLOCATED TO A VIRTUAL MACHINE USING MAX-MIN.

| Stages | Tasks | Machines | Energy in kwh | Energy% for1 task |
|---|---|---|---|---|
| 1 | T6 | M1 | 60 | 23.25% |
| 2 | T7 | M5 | 58 | 7.75% |
| 3 | T4 | M3 | 35 | 13.56% |
| 4 | T5 | M4 | 23 | 8.91% |
| 5 | T2 | M6 | 37 | 22.98% |
| 6 | T3 | M2 | 20 | 14.34% |
| 7 | T1 | M7 | 25 | 9.68% |

So by comparing energy % for a task allocation in a virtual machine in Table IV and Table V it is concluded that Genetic Algorithm gives better result than Max-min algorithm. However, practically it has not been implemented in any tool so further work would be done to implement the algorithm in cloudsim toolkit.

## VII. CONCLUSION AND FUTURE WORK

In this research work, two different approaches such as Genetic Algorithm and Max-min algorithm have been discussed and a green optimized energy-aware scheduling using genetic algorithm have been provided. Furthermore we aim to compare the algorithms manually for successfully accomplishing each and every business individually so that the VMs are selected according to the SLA level given by the user. Our method can meet the minimum resource requirement of a business and refrain the extra consumption of energy hence, we can decrease the energy utilization in data centers. The Genetic Algorithm technique is better than Max-min algorithm to be used to assign those tasks which consume less energy to the Virtual machines.

In future we aim to implement proposed ETSG algorithm in simulation tool inorder to produce practical results and Genetic Algorithm can be applied to find out the optimum results for other QoS requirements by the user such as priority, deadline, privacy, security, etc. In addition we want to generate the initial population using some other algorithms as in traditional genetic algorithm population initialization is obtained randomly which gives inappropriate values and compare it with some other approaches like PSO, SA, ACO, etc. to produce a statistical result.

## REFERENCES


[1] Armbrust, Michael, et al. "A view of cloud computing." Communications of the ACM vol 53, pp. 50-58, 2010.

[2] Sohrabi, S., and I. Moser. "A survey on Energy-Aware Cloud." European Journal of Advances in Engineering and Technology 2, vol. 2 pp. 80-91, 2015.

[3] Buyya, Rajkumar, Christian Vecchiola, and S. Thamarai Selvi. Mastering cloud computing: foundations and applications programming. Newnes, 2013.

[4] Koomey, Jonathan, "Growth in data center electricity use 2005 to 2010." A report by Analytical Press, completed at the request of The New York Times pp. 9, 2011.

[5] Report to Congress on Server and Data Center Energy Efficiency: Public Law 109-431. Energy Star Program, US Environmental Protection Agency, 2007.

[6] Agrawal, Pulin, and Smitha Rao. "Energy-aware scheduling of distributed systems." Automation Science and Engineering, IEEE Transactions on 11, no. 4, pp. 1163-1175, 2014.

[7] Singh, Shekhar, and Mala Kalra. "Scheduling of Independent Tasks in Cloud Computing Using Modified Genetic Algorithm." In Computational Intelligence and Communication Networks (CICN), 2014 International Conference, pp. 565-569. IEEE, 2014.

[8] Hsu, Ching-Hsien, et al. "Optimizing energy consumption with task consolidation in clouds." Information Sciences, vol. 258, pp. 452-462, 2014.

[9] Sahar hosseinzadeh, et.al. “Optimizing Energy Consumption in Clouds by using Genetic Algorithm” Journal of Multidisciplinary Engineering Science and Technology (JMEST) ISSN: 3159-0040 vol. 2, 2015

[10] Gu, Chonglin, et al. "Green scheduling for cloud data centers using renewable resources." In Computer Communications Workshops (INFOCOM WKSHPS), 2015 IEEE Conference, pp. 354-359. IEEE, 2015.

[11] Alahmadi, Ahmed, et al. "Enhanced first-fit decreasing algorithm for energy-aware job scheduling in cloud." In Computational Science and Computational Intelligence (CSCI), 2014 International Conference, vol. 2, pp. 69-74. IEEE, 2014.

[12] Chen, Huangke, et al. "Towards energy-efficient scheduling for real-time tasks under uncertain cloud computing environment." Journal of Systems and Software, vol. 99, pp. 20-35, 2015.