

# A Genetic Algorithm based task scheduling procedure for Cost-Efficient Cloud Data Centers

\*Prabha B<sup>1</sup>, Ramesh K<sup>2</sup> and Angelina Geetha<sup>3</sup>

<sup>1</sup> Assistant Professor/IT, Loyola-ICAM College of Engineering and Technology

<sup>1</sup>\*prabha.bala32@gmail.com

<sup>2</sup> Professor/CSE, Hindustan Institute of Technology and Science

<sup>2</sup>rameshk.me@gmail.com

<sup>3</sup> Professor & Head/CSE, Hindustan Institute of Technology and Science

<sup>3</sup>angelinageetha@gmail.com

**Abstract.** In the current situation, Cloud computing cut itself as a developing innovation which empowers the association to use equipment, programming and applications with no forthright expense over the internet space. The main provocation for the cloud provider is the means of providing productively and adequately the hidden computing assets like virtual machines, arrange, capacity units, and transmission capacity and so forth ought to be overseen with the goal that no computing gadget is in under-usage or over-use state in a unique domain. A decent task scheduling method is constantly required for the dynamic assignment of the task to evade such a circumstance. Through this paper we are going to introduce the Genetic Algorithm based task scheduling procedure, which will disperse the heap adequately among the virtual machine so the general reaction time (QoS) ought to be insignificant. An examination of this Genetic Algorithm based task scheduling procedure is performed on CloudSim test system which shows that, this will beat the current strategies like Greedy based, First - Come first - Serve (FCFS) methods.

**Keywords:** Virtual Machine, genetic algorithm, greedy based, FCFS

## 1 Introduction

Machine Learning is the study of computer algorithms that improve automatically through experience. A Computer task is observed to learn from experience E with respect to some class of tasks T and performance parameter P, if its performance at tasks in T, as measured by P, improves with experience E. It is seen as a subset of artificial intelligence [1]. The traditional machine learning can be sub divided into three parts. 1) Data input – raw information which is used for future decision mak-

ing, 2) Abstraction – using the underlying abstraction the input data are represented, 3) Generalization – the abstracted representation is templated to form as a framework for making smart decisions. There are three types of machine learning approaches, namely

- *Supervised Learning* - It is the process by which the system learns from past information. It is considered to be task driven and has the ability to predict the next value so also called as predictive leaning. The Classification Algorithm (eg: Fraud detection, Email Spam detection, Image Classification, Diagnostics), Regression Algorithm (eg: Risk Assessment, Score prediction) comes under supervised learning and contributes much towards experienceable learning
- *Unsupervised Learning*–It is the process of taking the labelled dataset as input and try to find natural grouping or patterns within the data records. It is considered to be data driven and identify Clusters. It is also called as descriptive learning. The Clustering Algorithm ( eg: Biology, City planning, Targeted Marketing), Dimensionality reduction ( eg: Text Mining, Face/Image recognition, Big data visualization, compression techniques is contributed in unsupervised learning
- *Reinforcement Learning*–Like babies learning to walk without any prior knowledge the system also Learn from Mistakes and react to the environment - eg: Gaming, Financial sectors, Inventory management, Manufacturing, Robotic automation/navigation, Real time AI-like Alexa, driverless cars

The modern techniques of machine learning method demonstrate its appropriate importance to overcome various issues by possessing some proactive features of,

- Simplifies the detection of Trends and Patterns
- Automation (without Human involvement)
- Constant Enhancement
- Manipulating Multi-dimensional and Wide range of data
- Extensive variety of applications

Based on the intuition behind Genetic Algorithms from the very famous naturalist Dr.Charles Darwin who said "*It is not the strongest of the species that survives, not the most intelligent that survives. It is the one that is most adaptable to change*", heuristic search was performed very efficiently in genetic algorithm. In this paper we have discussed about classifier system in section 2, genetic algorithm process for determining the cost efficiency system in cloud center in section 3, proposed work of implementation of genetic algorithm for task scheduling using Cloudsim toolkit for identification of strongest data center node in section 4 and section 5 elaborates about the conclusion and future work.

## 2 Background Study- CLASSIFER SYSTEM

The beginning stage for this way to deal with machine learning is a lot of rule-based systems fit to control revelation algorithms. The standards must loan themselves to forms that separate and recombine "building obstructs" from as of now valuable rules to frame new guidelines, and the standards must associate essentially and in an exceptionally equal style. Classifier systems are greatly equal, message-passing, decide based systems that learn through credit task (the container unit algorithm) and rule disclosure (the genetic algorithm) [2]. They commonly work in conditions that display at least one of the accompanying qualities: (1) never-endingly novel occasions joined by a lot of boisterous or immaterial information; (2) constant, regularly continuous, prerequisites for activity; (3) verifiably or inaccurately characterized objectives; and (4) meager result or fortification possible just through long activity successions. Classifier systems are intended to assimilate new data persistently from such conditions, concocting sets of contending theories (communicated as rules) without upsetting essentially capacities previously gained [3].

## 3 Genetic Algorithm For Determining The Cost Efficiency In Cloud Data Center

From the biological inspiration, we know that cells are the basic building block of all living things. Therefore, in each cell, there is the same set of chromosomes. Chromosome are basically the strings of DNA. Traditionally, these chromosomes are represented in binary as strings of 0's and 1's. A chromosome consists of genes, commonly referred as blocks of DNA, where each gene encodes a specific trait, for example hair color or eye color. A genetic algorithm is a search heuristic that mimics the process of natural evolution used to generate useful solutions to optimization and search problems [4].

Genetic algorithms are a subset of evolutionary algorithms which solves optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover [5]. Crossover and mutation are two basic operators of GA. Performance of GA very depend on them. Type and implementation of operators depends on encoding and also on a problem. There are five phases involved in genetic algorithm as follows

### *Phase 1: Initial population*

The process begins with a set of individuals which is called a Population. Each individual is a solution to the problem you want to solve. An individual is characterized by a set of parameters (variables) known as Genes. Genes are joined into a string to form a Chromosome (solution). In a genetic algorithm, the set of genes of an individual is represented using a string, in terms of an alphabet. Usually, binary

values are used (string of 1s and 0s). We say that we encode the genes in a chromosome

#### *Phase 2: Fitness function*

The fitness function determines how fit an individual is (the ability of an individual to compete with other individuals). It gives a fitness score to each individual. The probability that an individual will be selected for reproduction is based on its fitness score.

#### *Phase 3: Selection*

The idea of selection phase is to select the fittest individuals and let them pass their genes to the next generation. Two pairs of individuals (parents) are selected based on their fitness scores. Individuals with high fitness have more chance to be selected for reproduction. Chromosomes are selected from the population to be parents to crossover. As indicated by Darwin's evolution hypothesis the best ones should survive and make new offspring. There are numerous methods how to select the best chromosomes, for instance roulette wheel selection, parents are selected by their fitness [6].

#### *Phase 4: Crossover*

Crossover is the most significant phase in a hereditary based genetic calculation. For each pair of parents to be mated, a crossover point is chosen aimlessly from inside the genes. There are three types of crossover to be specific

- 1) **Single point crossover**
- 2) **Two-point crossover**
- 3) **Uniform crossover**

#### *Phase 5: Mutation*

Mutation is a genetic administrator used to keep up genetic diversity starting with one age of a populace then onto the next. It is like organic mutation. In straightforward terms, mutation might be characterized as a little arbitrary change in the chromosome, to get another arrangement. A case of a mutation administrator includes a likelihood that a discretionary piece in a genetic grouping will be transformed from its unique state [7]. A typical strategy for actualizing the mutation administrator includes producing an irregular variable for each piece in an arrangement. The reason for mutation in GAs is safeguarding and presenting diversity. Mutation ought to permit the calculation to maintain a strategic distance from nearby minima by keeping the number of inhabitants in chromosomes from getting excessively like one another. The genetic algorithm phases are depicted in the below Fig 1.

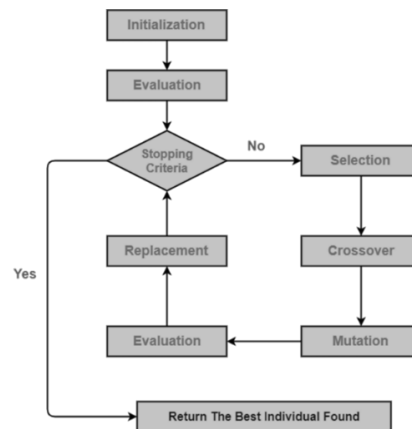


Fig. 1. Phases of Genetic Algorithm

#### 4 PROPOSED WORK

The proposed work will uncover the optimal standard value for the fitness with the assisted help rendered by strengthened learning calculation. Reinforcement learning techniques encourages an operator to self-improve its exhibition by utilizing the input it gets from the parent location [8]. In reinforcement learning the framework gets criticism which binds it together with managed learning. The Cloud supplier should ensure ideal scheduling of client's undertakings in the Cloud processing condition as per SLA. Simultaneously, he should ensure the best throughput and great use of the Cloud resources [9].

---

##### *Pseudocode for GA*

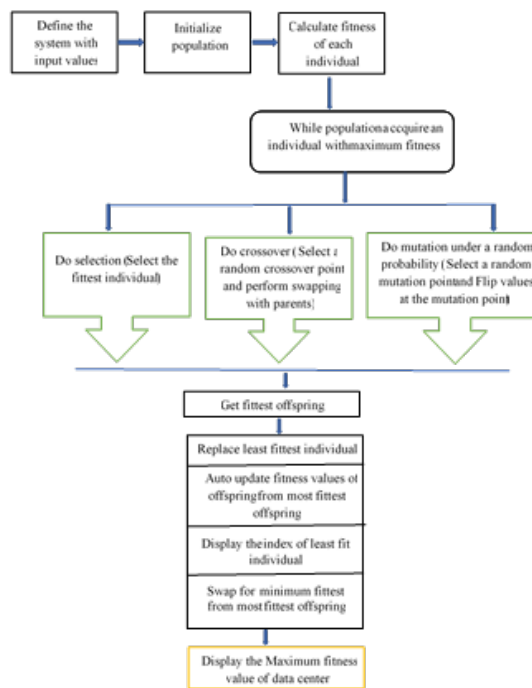
```

START
DEFINE the initial population
Compute fitness
REPEAT
Do Selection
Do Crossover
CALL Mutation
Compute fitness
UNTIL population has converged
STOP
  
```

---

Let us assume the fitness value resultant from the genetic algorithm will have the cost-efficient data center and consumes less amount of energy. Given a lot of 5 genes, every gene can hold one of the paired qualities 0 and 1. The wellness esteem is determined as

the quantity of 1s present in the genome. On the off chance that there are five 1s, at that point it is having greatest wellness. On the off chance that there are no 1s, at that point it has the base wellness. This genetic algorithm attempts to augment the wellness capacity to give a populace comprising of the fittest individual, for example people with five 1s. In this model, after hybrid and mutation, the least fit individual is supplanted from the new fittest offspring. The maximum fitness function will return the best ever data center with minimized energy [10]. The result is tested in cloudsim tool kit and shown as a screen shot in fig 3. The proposed flow of work is represented in flowchart in fig 2.



**Fig. 2.** Process of identifying the maximum fit value

```

1 import java.util.Random;
2 //Main class
3 public class SimpleDemoGA {
4
5     Population population = new Population();
6     Individual fittest;
7     Individual secondFittest;
8     int generationCount = 0;
9
10    public static void main(String[] args) {
11
12        Random rn = new Random();
13
14        SimpleDemoGA demo = new SimpleDemoGA();
15
16        //Initialize population
17        demo.population.initializePopulation(10);
18
19        //Calculate fitness of each individual
20        demo.population.calculateFitness();
21
22        System.out.println("Generation: " + demo.generationCount + " Fittest
23        : " + demo.population.fittest);
24
25        //While population gets an individual with maximum fitness
26        while (demo.population.fittest < 5) {
27            ++demo.generationCount;
28
29            //Do selection
30            demo.selection();
31
32            //Do crossover
33            demo.crossover();

```

```

$javac SimpleDemoGA.java
$java -Xmx128M -Xms16M SimpleDemoGA
Generation: 0 Fittest: 4
Generation: 1 Fittest: 4
Generation: 2 Fittest: 3
Generation: 3 Fittest: 3
Generation: 4 Fittest: 3
Generation: 5 Fittest: 3
Generation: 6 Fittest: 3
Generation: 7 Fittest: 5
Solution found in generation 7
Fitness: 5
Genes: 11111

```

Fig.3. Screen shot of executed GA and identified maximum fitness

## 5. RESULTS AND DISCUSSION

In this segment, the SoA methods have compared in various aspects. The assessment of MAGNETIC has attained by differentiating it with other current methods, such as PABFD and E-eco that ranges from incognizant energy solutions to the Service oriented architecture area, as regards the energy utilization, SLA intrusion, runtime and countable units of migrations. The upgraded version of CloudSim simulator aided to determine the proficiency of the solution [11]. Assume that 1600 hosts and a public repository applications are there in a data center. Subsequent to dynamic consolidation, the value of employed hosts has been marked extremely lesser than 1600, whereas the earlier designation of same VM can be utilized in this environment (with various quantity of VMs). By keep tracking the server and VM, the labeled data can be procured, which helps to determine the performance deviation that coined from several inconsistencies [12].

**PABFD:** The power-aware best fit decreasing algorithm enhances the allotment of VM through shifting the VMs from underused hosts. The enhancement of VM allocation has iterated with 5 minutes interval as this algorithm consecutively identifies the over-used and underused hosts [13]. With the help of static threshold, this algorithm identifies over-utilized hosts and chooses a VM to shift from it with limited use of the CPU and transfers VMs until their over-utilization is settled. Then, PABFD explores the host with the limited usage of the CPU to iterate this process for entire VMs from underutilized hosts and moves them to power turned-off mode. PABFD assigns every transferring VMs' destination host with considerable increment of power, owing to the re-allotment of VM. All active host that is underutilized can be a candidate destination to host the VM [14].

**E-eco:** The allocation of VM and power modes of host tend to reduce the utilization of energy and performance enhancement. E-eco is conscious about the PMT expenditures of host, and at each time slot every host has the capacity to be in the

modes of alert, inactive, and power-off [15]. A few hosts have been preserved in Sleep mode by E-eco to turn those hosts on immediately, when required. E-eco utilizes the estimation results of various energy preserving methods (e.g. DVFS/ dynamic consolidation) for several sizes and allocations of VMs instead of to minimize the conflict between energy utilization& performance [16]. Many sub methods has applied by E-eco for identifying the power modes, energy-preserving method, and frequency of CPU for each host.

MAGNETIC possess the optimum scalability than others. The VMs highest volume of 1463 in PlanetLab results the highest real traces. Then, the available traces have united together in order to generate bigger one, so that the inspection of the scalability eases. The acquired synthetic traces have depicted in Table 1 [17].

**TABLE 1: SYNTHETIC WORKLOAD TRACES**

	<i>Number of VMs</i>	<i>Number of hosts</i>
Synthetic trace1	3870	4020
Synthetic trace2	5100	5240
Synthetic trace3	7520	7650

Table 2 demonstrates the values of energy utilization and the average SLA intrusion of the processing center, evaluated by various algorithms, in which MAGNETIC reduces the utilization of power than other methods of PABFD and E-eco.

**TABLE 2: ENERGY UTILIZATION (kWh) AND AVERAGE SLA INTRUSION (%) OF DIFFERENT ALGORITHMS FOR SYNTHETIC TRACES**

<i>Synthetic Workload Traces/Method</i>	<i>Energy Consumption</i>			<i>SLA Violation</i>		
	<b>PABFD</b>	<b>E-eco</b>	<b>MAGNETIC</b>	<b>PABFD</b>	<b>E-eco</b>	<b>MAGNETIC</b>
Synthetic trace1	1805.21	1921.48	1640.23	12.07	12.61	12.78
Synthetic trace2	2485.41	2651.06	2257.85	11.81	12.73	12.84
Synthetic trace3	3507.20	3735.35	3190	12.10	12.88	12.5

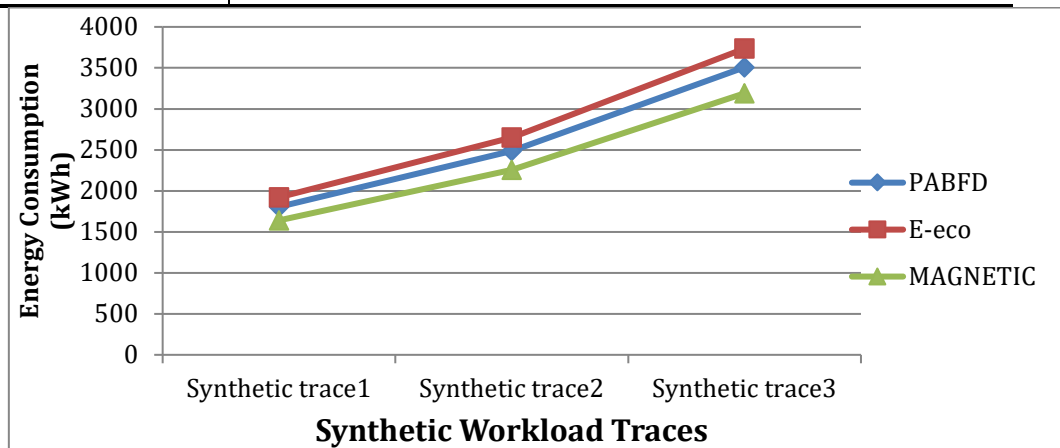
In Table 3, the estimated runtime and total migration count using various algorithms have presented, where the migration volume is zero, besides it procured the much shorter runtime for wide-ranging data centers that is much lesser than the runtime



of small-scale data center. Moreover, the proposed algorithm achieves the lesser migrations, as well as keeps enhancing the efficacy of energy [18].

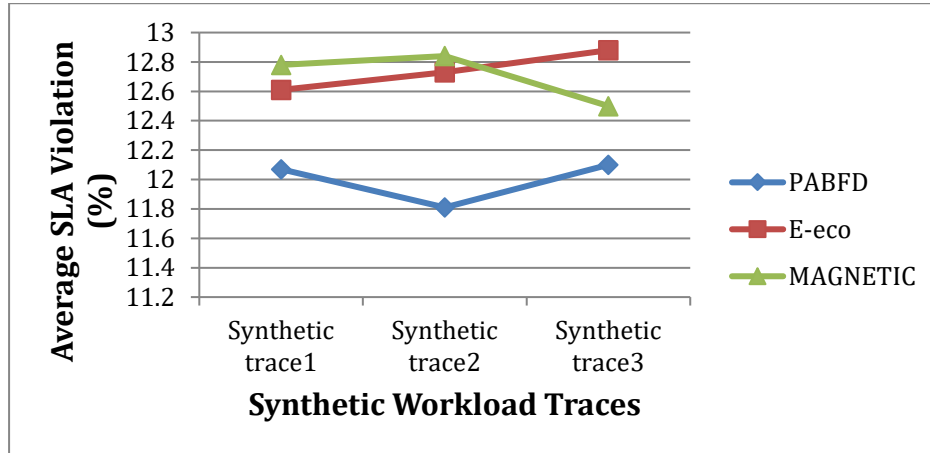
**TABLE 3: DYNAMIC PROCESSING TIME (seconds) AND COUNT OF MIGRATIONS (#) OF DIFFERENT ALGORITHMS FOR SYNTHETIC TRACES**

<i>Synthetic Workload Traces/Method</i>	<i>RunTime</i>			<i>Migrations</i>		
	<b>PABFD</b>	<b>E-eco</b>	<b>MAGNETIC</b>	<b>PABFD</b>	<b>E-eco</b>	<b>MAGNETIC</b>
Synthetic trace1	4.03	3.11	8.71	98464	98428	16605
Synthetic trace2	6.25	4.77	18.75	134981	135031	20984
Synthetic trace3	12.15	9.09	25.2	192078	192223	21202



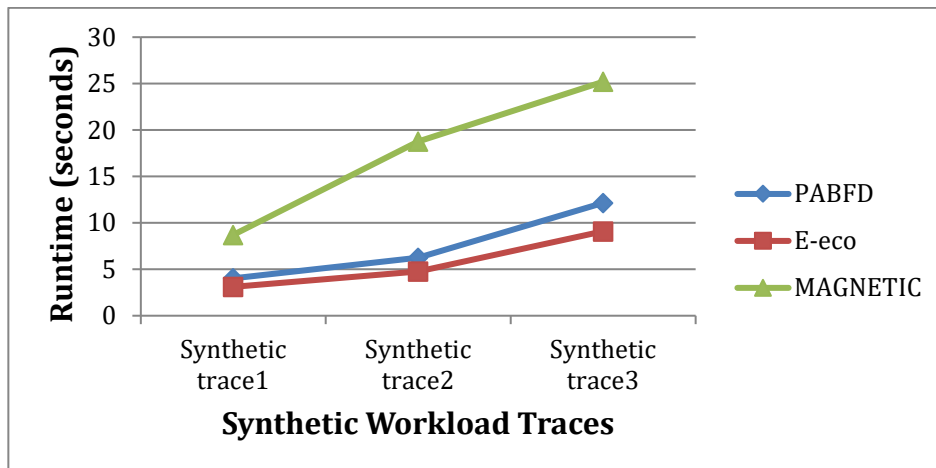
**Fig.4.**Energy consumption (kwh) of different algorithms for synthetic traces

Fig 4 compares the energy consumption results for synthetic traces estimated by the algorithms of MAGNETIC, PABFD, and E-eco. Here, the comparison graph proves that the suggested method of MAGNETIC efficiently reduces the energy utilization than other current methods of PABFD and E-eco [19].



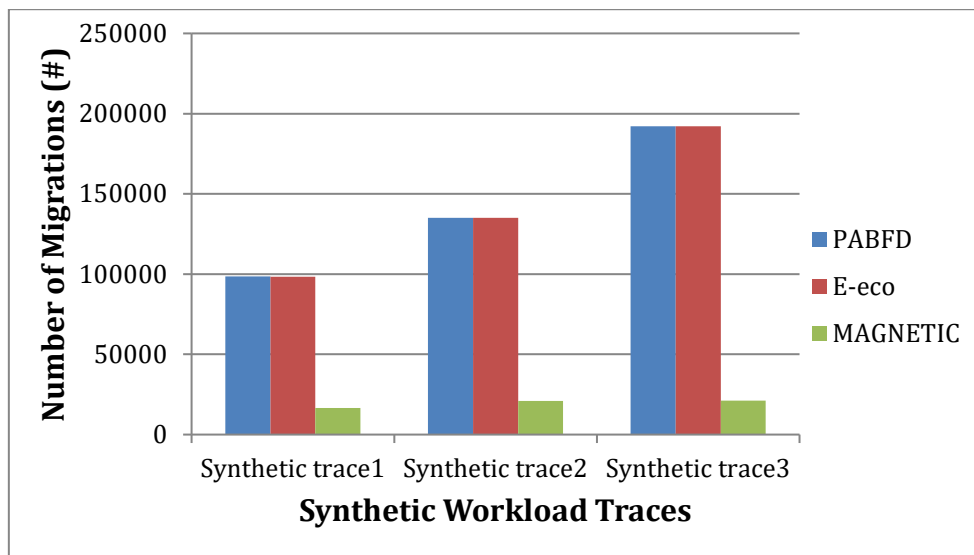
**Fig.5.** Average SLA violation (%) of various algorithms for synthetic traces

In Fig5, the graph illustrates the comparison of overall SLA violation results for synthetic traces using the algorithms of MAGNETIC, PABFD, and E-eco. Here, the proposed MAGNETIC approach enlarges the SLA violation in trace 1, and trace 2 of synthetic traces when compared to other current models of PABFD and E-eco. On the other hand, the proposed method efficiently reduces the SLA Violation than other methods in trace 3 [20,21].



**Fig.6.** Runtime (seconds) of different algorithms for synthetic traces

In Fig 6, the graph demonstrates the comparison of Runtime results for synthetic traces obtained by the algorithms of MAGNETIC, PABFD, and E-eco[26][27]. Now, the suggested method of MAGNETIC extends the Runtime for all synthetic traces compared to other present strategies of PABFD and E-eco [22,23].



**Fig.7.** Number of migrations (#) of different algorithms for synthetic traces

Fig 7 compares the number of Migration for synthetic traces acquired by the algorithms of MAGNETIC, PABFD, and E-eco. Here, the comparison chart represents that the suggested method of MAGNETIC efficiently reduces the number of migrations for entire three synthetic traces compared to other current strategies of PABFD and E-eco [24,25].

## 6. CONCLUSION AND FUTURE WORK

Based on execution work we arrive at a point that the gene based algorithm is capable of producing energy efficient cloud data center with high standards of load balancing mechanism. Genetic algorithm can the ability of incorporating machine learning reinforcement techniques to capture the maximum fitness values which helps us to allocate workload on data centers. The difficulties of computing resource allocation in existing energy consumption methods of contemporary cloud data center has to get over by; 1) The energy and network-conscious Virtual Machine (VM) allocation issue has to handle by the approaches of two-phase greedy heuristic and a Machine Learning (ML) method, and estimating them by considering the factors like power , Quality of Service (QoS), reallocations , dynamic measurability, and network congestion for several states of DC; 2) Executing the Multi-AGent machine learning-based approach for Energy efficient dynamic Consolidation (MAGNETIC) method which uses CloudSim simulator's upgraded version by considering the expenditures of energy and delay accompanied to power mode transition of host and migration of VM, and assessing them through the power traces gathered from several workloads of physical servers, and resource consumption logs from computing system; 3) Obtaining the proportion of performance and energy by implementing dynamic VM consolidation

through the Performance-to-power Ratio based Reinforcement Learning (PPR-RL) of heterogeneous hosts; 4) Forecasting the energy usage of VMs and hosts on the basis of their previous performance detail through regression-based method, concerning the exploitation of those forecasts; 5) Obtain the energy-efficient cloud computing environment by machine learning strategy as it reduces the redundant computation that helps preserving energy; 6) Apply the method of Multiple Linear regression and Wavelet Neural Network (MLWNN), since it widely diminishes the vitality use of servers, system, and cooling frameworks of cloud server farm.

Nevertheless, the machine learning strategy also includes some constraints within. Hence, the exploitation of updated deep learning strategies can aid to overcome the complicated issues of engineering domain. Apparently, the optimum choice to procure the necessary solution for consolidation of Energy-Efficient Cloud Data Centers can be the modern deep learning algorithm.

## References

- [1] Y. Xia et al., "A stochastic approach to analysis of energy-aware DVS-enabled cloud datacenters" *IEEE Trans. Syst. Man Cybern. Syst.* vol. 45 no. 1 pp. 73-83 Jan. 2015.
- [2] Y. Yin et al., "Collaborative service selection via ensemble learning in mixed mobile network environments" *Entropy* vol. 19 no. 7 pp. 358 2017.
- [3] J. Yu et al., "Leveraging content sensitiveness and user trustworthiness to recommend fine-grained privacy settings for social image sharing" *IEEE Trans. Inf. Forensics Security* vol. 13 no. 5 pp. 1317-1332 May 2018.
- [4] J. Muz et al., "Network location-aware service recommendation with random walk in cyber-physical systems" *Sensors* vol. 17 no. 9 pp. 2059 2017.
- [5] J. Yu B. Zhang et al., "iprivacy: Image privacy protection by identifying sensitive objects via deep multi-task learning" *IEEE Trans. Inf. Forensics Security* vol. 12 no. 5 pp. 1005-1016 2017.
- [6] A. Choudhary et al., "A GSA based hybrid algorithm for bi-objective workflow scheduling in cloud computing" *Future Gener. Comput. Syst.* vol. 83 pp. 14-26 2018.
- [7] Q. Peng M. et al., "Multi-objective optimization for location prediction of mobile devices in sensor-based applications" *IEEE Access* vol. 6 pp. 77123-77132 2018.
- [8] W. Li et al., "Fluctuation-aware and predictive workflow scheduling in cost-effective infrastructure-as-a-service clouds" *IEEE Access* vol. 6 pp. 61488-61502 2018.
- [9] R. Xu et al. "Asufficient and necessary temporal violation handling point selection strategy in cloud workflow" *Future Gener. Comput. Syst.* vol. 86 pp. 464-479 2018.
- [10] J. Yu et al., "Learning to rank using user clicks and visual features for image retrieval" *IEEE Trans. Cybern.* vol. 45 no. 4 pp. 767-779 2015.
- [11] D. Nasonov et al., "Hybrid evolutionary workflow scheduling algorithm for dynamic heterogeneous distributed computational environment" *J. Appl. Logic* vol. 24 pp. 50-61 2017.
- [12] Y. Wang et al., "A multi-stage dynamic game-theoretic approach for multi-workflow scheduling on heterogeneous virtual machines from multiple infrastructure-as-a-service clouds" in *Services Computing Springer* vol. 10969 pp. 137-152 2018.
- [13] E. Iranpour et al., "A distributed load balancing and admission control algorithm based on Fuzzy type-2 and Game theory for large-scale SaaS cloud architectures" *Future Gener. Comput. Syst.* vol. 86 pp. 81-98 Sep. 2018.
- [14] L. Wu et al., "Scheduling multi-workflows over heterogeneous virtual machines with a multi-stage dynamic game-theoretic approach" *Int. J. Web Services Res.* vol. 15 pp.

82-96 Oct. 2018.

- [15] W. Jiahao Pet.al., "A multi-object optimization cloud workflow scheduling algorithm based on reinforcement learning" in *Intelligent Computing Theories and Application Cham Switzerland:Springer* pp. 550-559 Aug. 2018.
- [16] Y. Wei Det.al., "A reinforcement learning based workflow application scheduling approach in dynamic cloud environment" in *Collaborative Computing: Networking Applications and Worksharing Cham Switzerland:Springer* pp. 120-131 Dec. 2018.
- [17] D. Cui Wet.al., "Multiple DAGs workflow scheduling algorithm based on reinforcement learning in cloud computing" in *Computational Intelligence and Intelligent Systems Singapore:Springer* pp. 305-311 Nov. 2016.
- [18] B. Waschneck Aet.al., "Optimization of global production scheduling with deep reinforcement learning" *Procedia CIRP* vol. 72 pp. 1264-1269 Jan. 2018.
- [19] M. Kaur et.al., "A novel multi-objective bacteria foraging optimization algorithm (MOBFOA) for multi-objective scheduling" *Appl. Soft Comput. J.* vol. 66 pp. 183-195 2018.
- [20] L. Zhang Ket.al., "Bi-objective workflow scheduling of the energy consumption and reliability in heterogeneous computing systems" *Inf. Sci.* vol. 379 pp. 241-256 Feb. 2017.
- [21] I. Casas Jet.al., "GA-ETI: An enhanced genetic algorithm for the scheduling of scientific workflows in cloud environments" *J. Comput. Sci.* vol. 26 pp. 318-331 May 2018.
- [22] A. Verma et.al., "A hybrid multi-objective particle swarm optimization for scientific workflow scheduling" *Parallel Comput.* vol. 62 pp. 1-19 Feb. 2017.
- [23] X. Zhou Get.al., "Minimizing cost and makespan for workflow scheduling in cloud using fuzzy dominance sort based HEFT" *Future Gener. Comput. Syst.* vol. 93 pp. 278-289 Apr. 2019.
- [24] D. P. Bertsekas "Feature-based aggregation and deep reinforcement learning: A survey and some new implementations" *IEEE/ACM Trans. Audio Speech Language Process.* pp. 1-31 2018.
- [25] H. Mao Met.al., "Resource management with deep reinforcement learning" *Proc. 15th ACM Workshop Hot Topics Netw. (HotNets)* pp. 50-56 2016.
- [26] J. Dafni Rose, K. Vijayakumar and S. Sakthivel, "Students' performance analysis system using cumulative predictor algorithm", *Int. J. Reasoning-based Intelligent Systems*, Vol. 11, No. 2, 2019.
- [27] Vijayakumar. K, Nawaz Sherif. T, Gokulnath.S, "Automated Risk Identification using Glove algorithm in Cloud Based Development Environments", *International Journal of Pure and Applied Mathematics* Volume 117 No. 16 2017