Improved Honey Bee Scheduling Algorithm For Load Balancing In Cloud Computing

Krushal T. Mohta[1], Krunal J. Panchal[2]
[1]P.G. Student, Department of Computer Engineering, L.J Institute of Engineering and Technology, Ahmedabad, Gujarat, India
[2]Assistant Professor, Department of Computer Engineering, L.J Institute of Engineering and Technology, Ahmedabad, Gujarat, India

ABSTRACT

Cloud computing a new paradigm in the field of distributed computing after Grid computing. Cloud computing seems to me more promising in term of request failure, security, flexibility and resource availability. Its main feature is to maintain the Quality of service (QoS) provided to the end user in term of processing power, failure rate and many more .So Resource management and request scheduling are important and complex problems in cloud computing, Since maintaining resources and at the same time scheduling the request becomes a complex problem due to distributed nature of cloud. Many algorithms are been proposed to solve this problem like Ant colony based, cost based, priority based algorithms but all these algorithm consider cloud environment as non fault, which leads to degrade in performance of existing algorithms. So a load and fault aware Honey Bee scheduling algorithm is proposed for cloud infrastructure as a service(IaaS). This algorithm takes into consideration fault rate and load on a datacenter to improve the performance and QoS in cloud IaaS environment

Keyword: *Cloud, QoS, Cloud IaaS, Fault, System load, Network load, Datacenters.*

1. INTRODUCTION

With the evolution of cloud and grid computing distributed system and distributed computing has shown a great benefit to the industry and society by providing the services which are cheaper and more reliable. With grid computing we only have the benefit of computing the bigger task in smaller span of time, but with cloud computing any task can be computed easily, much faster by using the computation power of server and in more secure way. Cloud computing is best example of Distributed computing. It have all the properties of distributed systems namely atomicity, consistency, isolation and durability. It provides all type of services that may be software or platform based or simply hosting services. Services provided by cloud can be classified into three categories SaaS (Software as a service), PaaS (Platform as a service), IaaS[1]. SaaS provides services over cloud in terms of applications which are used to process data.
Cloud IaaS has the four basic features which are as follows [1]:

1. **Pay per Use:** Traditionally organizations procure their own computing resources to build IT infrastructure to archive their business goals. Cloud computing will change this trend in building IT infrastructure through providing infrastructure as a service. Cloud computing allows quick implementation of computing resources without huge amount of upfront capital investment [1].

2. **Elastic Computing:** In the model of cloud computing, computing resources form a shared pool, in which a resource can join and depart from the pool dynamically. Users request the resources on demand, and return the resources to the pool after competition of use. The resources can be re-allocated to other users. Elastic computing feature supports high scalability of computing resources in cloud computing environment.

3. **Virtualization:** All types of computing resources can be virtualized with the development of cloud computing technology. VLAN is an example of network virtualization. Server virtualization is the most common virtualization type in computing resources and supported by most all cloud product vendors. Distributed file system often use storage virtualization to tackle with the heterogeneous storage devices.

4. **Cost:** It plays an important role in cloud computing, because as the definition of cloud says that it will provide the service at cheaper rates.

But the main problem over cloud IaaS is resource management and request scheduling because in this the VM once allocate and running runs for long span of time. So we need to allocate resources in such a way that the request is fulfilled at least cost and highest quality of service which can be provided to a user.There are many cloud IaaS frameworks that provide cloud computing services and virtualization services to the user like OpenNode [11], CloudStack [9], Eucalyptus [8], CloudSigma [10], EMOTIVE(Elastic Management of Tasks in Virtualized Environments) and Archipel.Many solution have been proposed over the time based on priority, cost, rank based which is used in OpenNebula and round robin and power aware scheduling algorithm used in Eucalyptus and many more. But they do not take in to consideration the QoS parameters of the datacenters like fault rate, initialization time, MIPS and many more. So here a QoS aware honey bee algorithm is proposed to provide higher QoS to the user and at the same time taking into consideration least cost to be provided to the user.

2. RELATED WORK

Load balancing is an approach to reducing energy consumption and improving utilization of hosts. Babu L.D et al. [2] have proposed a load balancing technique to balance the load and priorities of tasks that removed from heavily loaded VMs. This technique is based on behavior of honey bee foraging strategy and improves the overall throughput of processing and reduces the response of time of VMs. However authors have not investigated the power consumption.

Dalapati et al. [3] have proposed a Green scheduling algorithm that optimizing power consumption in cloud computing. It uses bee colony algorithm for service rescheduling and ant colony algorithm for power consumption management. In contrast, in this paper we use bee colony algorithm for detection of over utilized hosts and for the VM selection we use MMT.

Beloglazov et al. [9] have presented an architectural principles for energy aware management of clouds. Moreover, they proposed energy-efficient resource allocation policies and scheduling algorithms. However, because of the fact that they used fixed utilization thresholds, this approach may not be efficient for the cloud computing environments. In their morerecent work [4], Beloglazov et al. have proposed an adaptive heuristics for dynamic consolidation of VMs based on analysis of historical data from the resource usage by VMs. This algorithm reduce the energy consumption. Authors propose algorithms like Median Absolute Deviation, Interquartile Range, Local Regression (LR) and Robust Local Regression for host overloading detection. Moreover, for the VM selection they use The Minimum Migration Time policy, The Random Choice Policy and Maximum correlation policy.

Based on their algorithm after the detection of overloaded hosts and select VMs to migrate from these hosts, system finds the host with the minimum utilization and if it is possible tries to place the VMs from this host on the other hosts while keep them not overloaded and when all the migration have been complete switch host to the sleep mode. If this cannot be accomplished, the host kept active. This process is iteratively repeated for all host except the overloaded hosts. Whereas we have a same approach for underutilized hosts and VM selection policy (MMT), our host overloading detection methods are different and we are using artificial bee colony algorithm (ABC) to detect over utilized hosts. Yao et al. [5] have presented a load balancing mechanism based on artificial bee colony algorithm (ABC). Authors propose an improved artificial bee colony algorithm to increase the system throughput. However, they did not investigate the energy consumption or SLA violation.

3. HONEY BEE FORGING BEHAVIOUR

A colony of honey bees can extend itself over long distances (over 14 km) and in multiple directions simultaneously to harvest nectar or pollen from multiple food sources (flower patches). A small fraction of the colony constantly searches the environment looking for new flower patches. These scout bees move randomly in the area surrounding the hive, evaluating the profitability (net energy yield) of the food sources encountered. When they return to the hive, the scouts deposit the food harvested. Those individuals that found a highly profitable food source go to an area in the hive called the "dance floor", and perform a ritual known as the waggle dance. Through the waggle dance a scout bee communicates the location of its discovery to idle onlookers, which join in the exploitation of the flower patch. Since the length of the dance is proportional to the scout's rating of the food source, more foragers get recruited to harvest the best rated flower patches. After dancing, the scout returns to the food source it discovered to collect more food. As long as they are evaluated as profitable, rich food sources will be advertised by the scouts when they return to the hive. Recruited foragers may waggle dance as well, increasing the recruitment for highly rewarding flower patches. [7]

Bee system consists of two essential components:

- Food Sources

The value of a food source depends on different parameters such as its proximity to the nest, richness of energy and ease of extracting this energy.

- Foragers

Unemployed foragers: If it is assumed that a bee have no knowledge about the food sources in the search field, bee initializes its search as an unemployed forager. There are two possibilities for an unemployed forager:

Scout Bee: If the bee starts searching spontaneously without any knowledge, it will be a scout bee. The percentage of scout bees varies from 5% to 30% according to the information into the nest. The mean number of scouts averaged over conditions is about 10% (Seeley, 1995).

Recruit: If the unemployed forager attends to a waggle dance done by some other bee, the bee will start searching by using the knowledge from waggle dance.

Employed foragers: When the recruit bee finds and exploits the food source, it will raise to be an employed forager who memorizes the location of the food source. After the employed foraging bee loads a portion of nectar from the food source, it returns to the hive and unloads the nectar to the food area in the hive. There are three possible options related to residual amount of nectar for the foraging bee.

If the nectar amount decreased to a low level or exhausted, foraging bee abandons the food source and become an unemployed bee.

If there are still sufficient amount of nectar in the food source, it can continue to forage without sharing the food source information with the nest mates

Or it can go to the dance area to perform waggle dance for informing the nest mates about the same food source. The probability values for these options highly related to the quality of the food source.

4. PROPOSED WORK

As from above scheduling algorithms they only take into consideration either the power or simple resource allocation algorithm, to overcome that and map them to real paradigm the cost based algorithm was proposed and rank based algorithm were proposed[1][2].

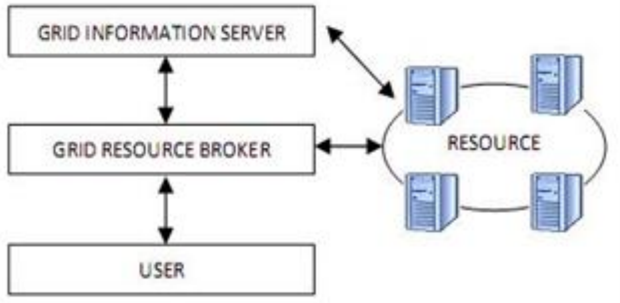
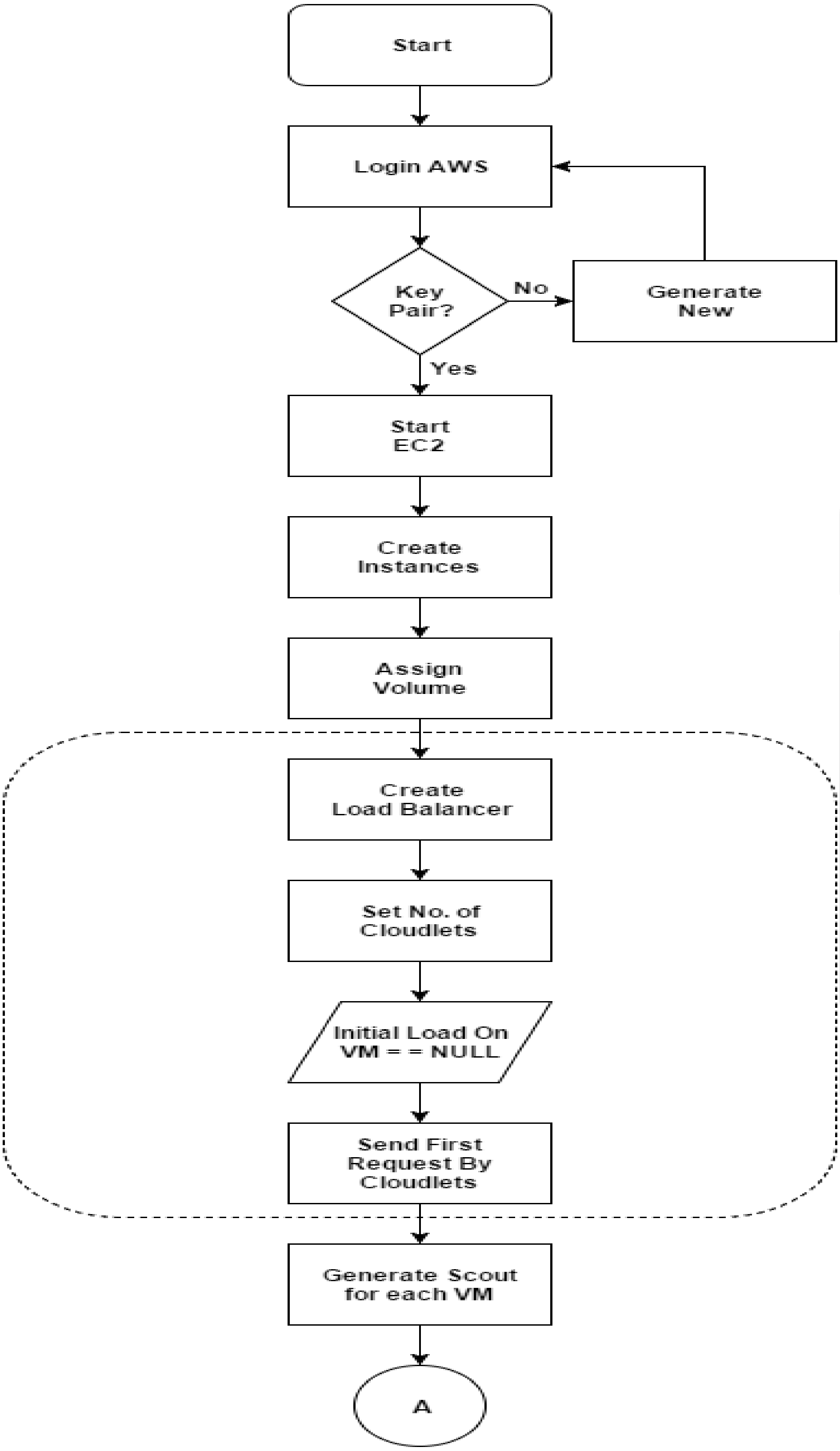


Fig. 1. Ant Colony System Architecture

The proposed algorithm is as followed


Start
Login AWS
Key Pair?
No
Generate New
Yes
Start EC2
Create Instances
Assign Volume
Create Load Balancer
Set No. of Cloudlets
Initial Load On VM = = NULL
Send First Request By Cloudlets
Generate Scout for each VM
A

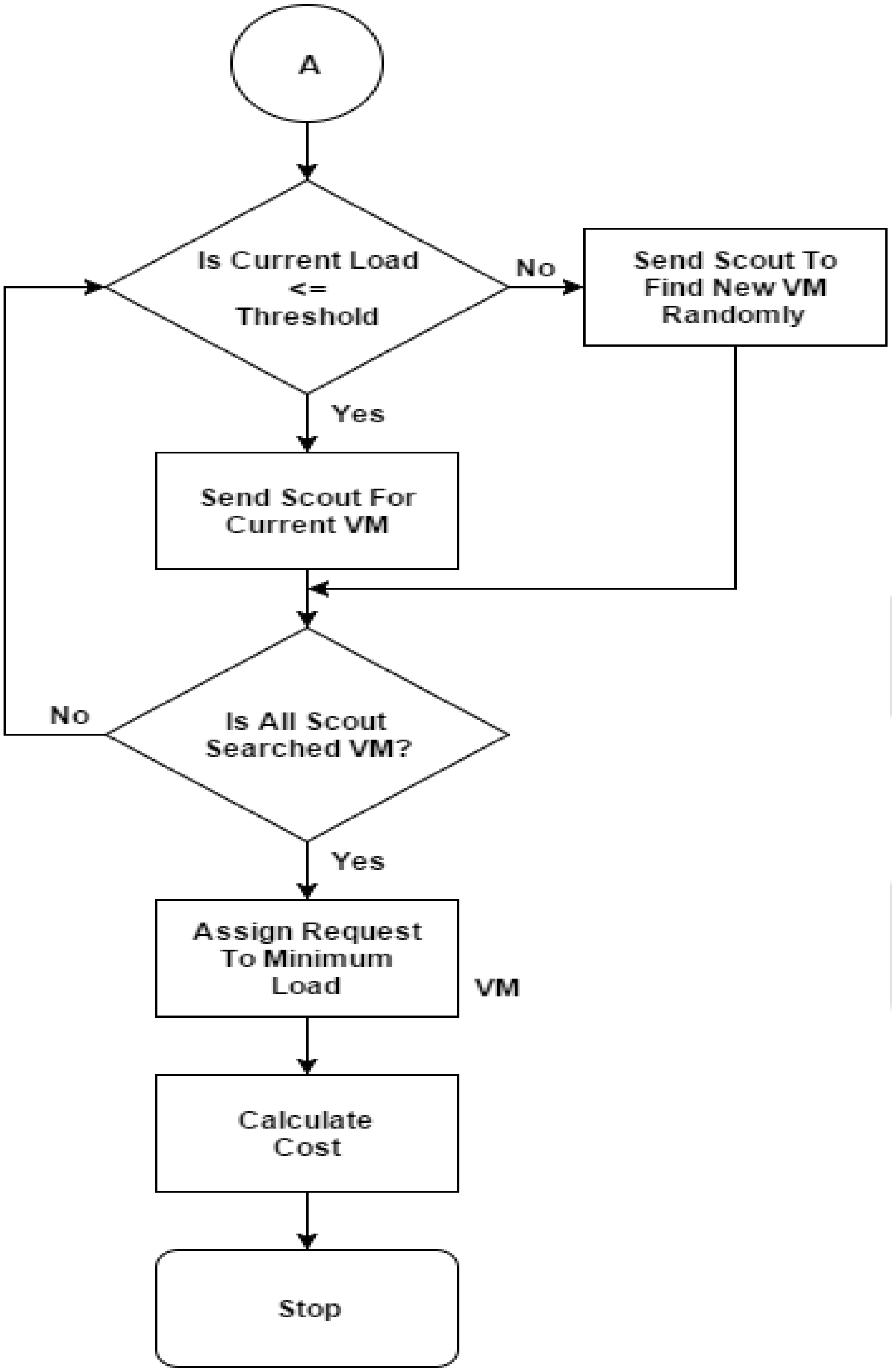

A
Is Current Load <= Threshold
No
Send Scout To Find New VM Randomly
Yes
Send Scout For Current VM
No
Is All Scout Searched VM?
Yes
Assign Request To Minimum Load
VM
Calculate Cost
Stop

Fig 2. Proposed Flow Chart

The steps in the proposed algorithm are as below

1. Start the AWS EC2 and login in and after that pair the key. If the key is correct then create instances and if key is wrong then generate new key and again login

2. After instances are created assign volume to them and then create load balancer and set number of cloudlets
3. Assign initial load on VM=null and then send first request by cloudlet
4. Generate scout for each VM
5. Send the request. It will check whether the current load is <= to threshold value or not
6. If it is less than or equal to threshold value then it will asssign task to current Vm or it send scout to find new VM randomly
7. After that it will check whether all scout have searched all VM
8. If yes then it will assign task to VM which has min. load or If no then again it will check whether current load is <= threshold or not
9. After that it will calculate cost

Formulas For Calculating

1**. Capacity of a VM**, ***m*** is,

$$Ci = pe_{ni} \times pempi + VMbwi \quad (1)$$

Where pe*ni* is the number of processors in VMi, pempi is millions of instructions per second of all processors in Vmi and VMbwi are the communication bandwidth ability of Vmi

2. **Capacity of all VMs,**

$$C = C1+C2+........Cn \quad (2)$$

3. **Load on a VM *m* is,**

Total length of tasks that are assigned to a VM is called load,

$$Lvmi,t=N(T,t)/S(Vmi,t) \quad (3)$$

Load of a VM at time *t* can be calculated as the Number of tasks at time *t* on service queue of *VMi* divided by the service rate of *VMi* at time *t*.

4. **Load of all VMs,**

$$L = Lvm1+Lvm2+.........Lvmn \quad (4)$$

5. **Processing time** of a VM ***m*** is,

$$Pti=LVMi/Ci \quad (5)$$

6. **Processing time of all VMs**:

$$PT=L/C \qquad (6)$$

7. **Cost**

$$C(ti,mj)=\sigma * PT(ti, mj) * Vco(\frac{Cmj}{Ccap}) \quad (7)$$

Ccap denotes the capacity for the VM having less capacity and Vco is the cost of that VM.

8. **Fault rate**

$$FR(t)=f(MP,N\_L)$$

Where MP denotes machine instruction per second and N_L is network load

5. EXPERIMENTAL RESULT

Proposed algorithm is simulated using cloudsim simulator [18] is used. Cloudsim basically support cost estimation, and FIFO algorithm for scheduling the resource sequentially. Firstly the Cloudsim API does not support fault rate at datacenter. So firstly fault is added as a parameter of datacenter which responds to fault occurring at the datacenter. This cloudsim API is used for simulation of cloud IaaS. So it includes all the cloud IaaS request parameters. Honey bee scheduling algorithm is implemented in cloudsim replacing basic FCFS (first come first serve) scheduling algorithm and Round Robin Algorithm is used. Comparative study is done between basic load aware honey bee (BLHB) which allocates request on datacenter with least system load and proposed algorithm

Table 1.

Data Centre Request Serving Time		
DataCenter	Existing	Proposed
DC1	0.875	0.871
DC2	0.885	0.88
DC3	0.886	0.885

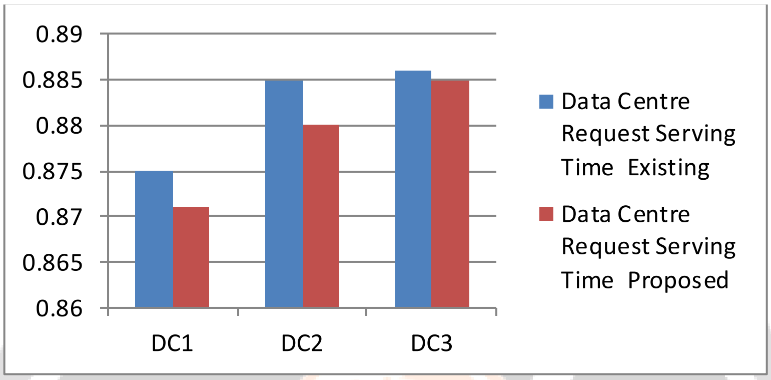
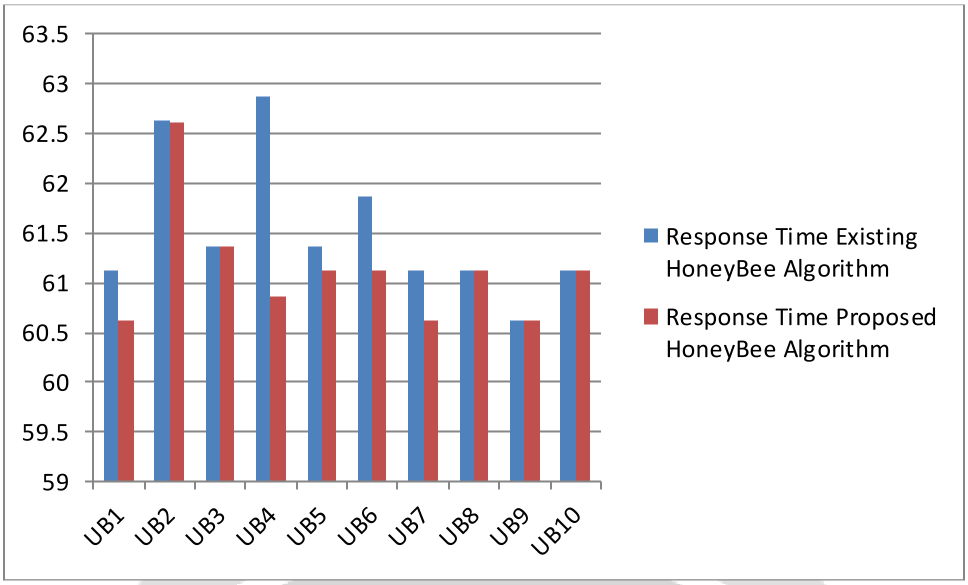


Table 2.

Response Time		
User Base	Existing HoneyBee Algorithm	Proposed HoneyBee Algorithm
UB1	61.127	60.611
UB2	62.623	62.62
UB3	61.364	61.364
UB4	62.872	60.87
UB5	61.355	61.12
UB6	61.872	61.12
UB7	61.122	60.619
UB8	61.124	61.127
UB9	60.623	60.619
UB10	61.127	61.127



6. CONCLUSION

In this paper, Honey bee behavior algorithm for load balancing using better energy consumption is Proposed. Energy efficient cloud computing technology is very recent technology, and there been many areas which is still not explore fully as compared to other technologies, and thus there is a huge prospective for growth and new innovations. The implementation results show that with the VM allocation Policy BEE-MMT this algorithm will give better results. In Energy efficient cloud computing our first aim is to reduce the energy usage. By this the power consumption will be less which will help us to achieve green cloud computing. According to results of simulations proposed algorithm can decline the power consumption; hence, it can be a green solution and reducing production of carbon dioxide and operational cost. , In this model we mainly focus on energy but due to that we have increased no of SLA violation, this will consider that we allow to compromise with QoS which is not good when large scale application upload on cloud like social networking site's data, Search engine data, so as a future enhancement we try to build this algorithm with QoS Parameters.

7. REFERENCES

[1] Seyed Mohssen Ghafari, Mahdi Fazeli, Ahmad Patooghy, Leila Rikhtechi, "Bee-MMT: A Load Balancing Method for Power Consumption Management in Cloud Computing", 978-1-4799-0192-0/13, 2013 IEEE, pp 76-80

[2] Sheeja Y.S. Jayalekshmi S, "Cost Effective Load Balancing Based on Honey Bee Behaviour in Cloud Environment", 978-1-4799-6013-2, 2014 IEEE, pp 214-219

[3] L.D. Dhinesh Babua, P. Venkata Krishna, "Honey Bee Behaviour Inspired Load Balancing of Tasks in Cloud Computing Environments", 1568-4946, 2013 ELSEIVER, pp 120-131

[4] Punit Gupta, Satya Prakash Ghrera, "Load and Fault Aware Honey Bee Scheduling Algorithm for Cloud Infrastructure", 978-3-319-12012, 2015 Springer, pp 135-143

[5] Ms. Kunjal Garala, Ms. Namrata Goswami, "A Performance Analysis of Load Balancing Algorithm in Cloud Environments", 978-1-4799-6805-3, 2015 IEEE, pp 45-50

[6] Shaw, S.B., Singh A.K., "A Survey on Scheduling and Load Balancing Techniques in Cloud Computing Environment" Computer and Communication Technology (ICCCT), 2014, pp 87-95

[7] Ekta Gupta, Vidya Deshpande, "A Technique Based on Ant Colony Optimization for Load Balancing in Cloud Data Center", ICIT 2014, 978-1-4799-8084-0/14, pp 12-17

[8] Mayanka Katyal, Atul Mishra, "A Comparative Study of Load Balancing Algorithms in Cloud Computing Environment", International Journal of Distributed and Cloud Computing 2013, pp 28-37

[9] D. B. L.D and P. V. Krishna, "Honey bee behavior inspired load balancing of tasks in cloud computing environments," *Applied Soft Computing*, vol. 13, no. 5, pp. 2292–2303, May 2013.

[10] P. Dalapati and G. sahoo, "Green Solution for Cloud Computing with Load Balancing and Power Consumption Management," *International Jornal of Emerging Technology and Advanced Engineering*, vol. 3, no. 3, pp. 353–359, March 2013

[11] L. Maghali S.de Oliveira, R. Schirru and J. A.C.C. de Medeirose, "On The Performance of An Artificial BEE Colony Optimization Algorithm Applied to The Accident Diagnosis In a PWR Nuclear Power Plant," in *International Nuclear Atlantic conference-INAC*, vol. 1, Rio de Janeiro Brazil, September 27 to October 2, 2009.