International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 9 Issue 3 ǁ 2021 ǁ PP. 82-89

Hybrid Fuzzy-Genetic Load Balancing Scheme for Cloud Computing

Birhanu Gardie[1], Kassahun Azezew[2], and Haimanot Bitew[3]
[1]*School of Computing and informatics, Mizan-Tepi University, Ethiopia*
[2]*School of Computing and informatics, Mizan-Tepi University, Ethiopia*
[3]*School of Computing and informatics, Mizan-Tepi University, Ethiopia*

Abstract
Cloud computing is the recently emerging technology trend in the information technology industry which aims to provide utility-based services over the internet. The prevalent access of cloud technology needs attentions to some of the essential metrics to its accessible resource under service and utilization. Among the parameters load balancing, energy consumption, makespan, quality of service, resource allocation and resource utilization are of the primary concern. Efficient load balancing mechanism can realize customer's requirement, and enhance resource allocation and utilization and reduce the make span, thereby improving the overall performance of the cloud-computing environment. In this paper, we proposed a hybrid algorithm based on Fuzzy set theory and Genetic algorithm. The proposed scheme aims to realize optimal load balancing by minimizing execution time, decreasing the makespan, and maximizing resource utilization. The proposed load balancing approach employs in cloud-based environment an enhanced execution time, load imbalance factor, better resource utilization and minimum makespan for improving the performance of the system by making efficient allocation of computational resources based on their capacity to perform the tasks. Simulation results proved that the resourceutilizationratio is 0.6 and the value resides in the high utilization range.

Keywords: *Cloud Computing, Load Balancing, Fuzzy-genetic Algorithm*

Date of Submission: 10-03-2021 Date of acceptance: 25-03-2021

I. INTRODUCTION

Distributed computing leads to a new emerging technology called cloud computing implemented in academia and industry to store and retrieve files and necessary documents[1]. Currently, cloud computing becomes an essential computing model emerged from the rapid development of internet[2].In the advancement and evolution of ondemand services and products, cloud computing would be the next step in the informationtechnology development[3]. Currently, there is a dramatic change in increasing popularity ofcloud computing services which can rent computing resource when needed, billing on a pay asyou go basis of cloud service usage instead of traditional computing in which "own and use"technique is used, and multiplex many cloud service users on the same physical machine.From academia to information technology enterprises, cloud computing is highly required. In cloud based environments, job allocation to a particular resource is a basic problem in which the system performance is in unceasing abruptly of state devoid because of the drastic increment on demand use of the cloud service by the enterprises and users[4].It presents advanced structures to realize an efficient marketing scale across the distributed networking system by providing a mechanism to share a pool of computing resources, which have a basic target of allowing customers to access the services with no support of expert.The cloud system resource should coordinate to provide users request response that needs intercommunication among different parts of the system, this leads to challenge in an imbalanced charge in the diversified networking system where some node get involved in over charge, some get in light charge and others might involve idle[5]. In cloud based environment because of it is economical users would pay based on the service usage or utilization, as a result it is very essential to reduce the processing time and the makespan[6]. As the cloud service user growth up dramatically currently and the service providers needed to address the massive task requests. The first problem facing them is to enhance the performance of the system when an outburst workload occurs[7], [8],[9].For better realization of performance improvement significant issues would be explore, the main bottleneck is load balancing.

Due to the cloud service providers as they present services in an explosive dynamic web content facilities comprising social media network including Facebook, e-commerce are offered across the cyber global. As a result, load balancing should be deliberate in equal distribution to computing resources. To initiate

execution of requests, cloud based system applies plentiful resources. Therefore, proper mechanisms can exploit the effectiveness of the cloud based computing environment to execute tasks with minimum processing time. Minimum execution time and efficient utilization of resources are essential in all cloud service oriented architecture.

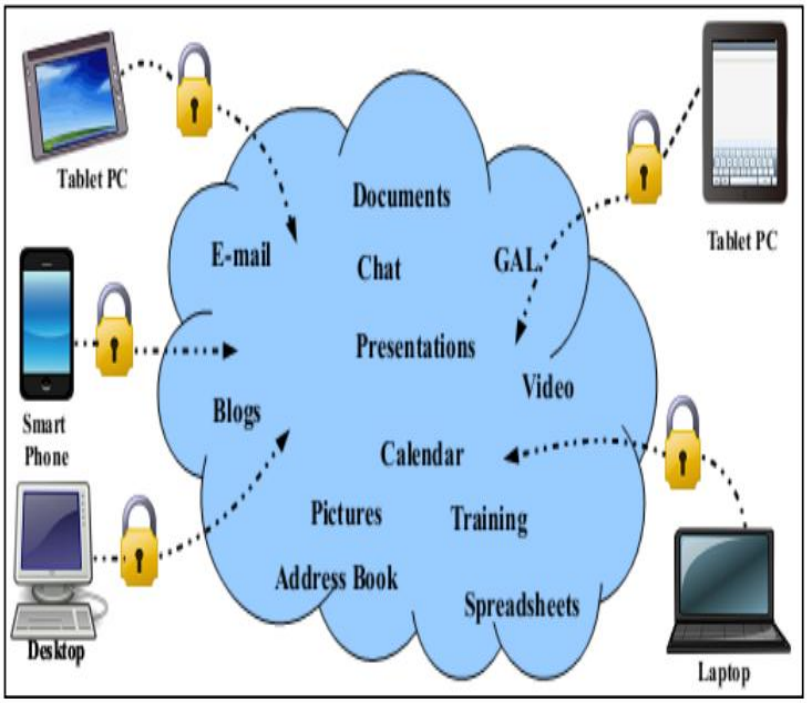

Figure 1: Cloud computing applications

II. LITERATURE AND THEORETICAL FOUNDATIONS

A new time sharing system technology was created in 1961 after McCarthy in the history of computer technology suggested that likewise water, electricity power and telephone are public utility, utility of computing may structure as public in MIT anniversary celebration of centenary[10]. As a computer science professional, he was the first who established timesharing technology would lead to the most powerful and dominant controlling computing paradigm. His state of art ideas was very popular during the time however, slowly faded away in 1990 following the starting of the 20th century that his creative thoughts are winding up within a new approach which is now called as cloud computing. In October 2009, conference is seized authorized as "Effectively and securely using the cloud computing paradigm" by NIST(National Institute of Standards and Technology) is an Information Technology Laboratory, Cloud computing is defined as follows[11]. The cloud computing model consists of five significant characteristics, three service models and four deployment models. These things are highlighted here under.

2.1 Characteristics of cloud computing

The five significant characteristics of cloud computing are as follows:

On-demand self-service: users can assign and release computing resources like server time, network storage and others when they need automatically without necessitating human interactions in cloud service providers.

Ubiquitous network access: cloud computing capabilities are available with reachability over the network and accessed via standard methods that promote use by heterogeneous thin and thick client platforms.

Resource pooling: cloud computing services are pooled in an organized manner to work for several users using multi-tenant model in which various physical and virtual resources are assigned and released dynamically.

Rapid Elasticity: cloud service consumers can scale up or down the cloud resources, they are going too used; in some cases, it can be automatically scale out when is required and rapidly released when users do not need.

Measured service: customers and enterprises rented infrastructure and facilities from cloud service vendor is charged according to the resource they utilize.

2.2 Cloud computing service models

Service means various types of software offered by various servers over the cloud. The three services models in cloud are the following.

2.2.1 Software as a service (SaaS)

In SaaS, a service provider licenses different software application from various servers to the customers for use through the internet as service on demand [12]. The customer uses the application without change and does not to do many modifications or no need to integrate to other systems. The service providers do the changes and maintenance despite the fact that the infrastructure is operating.

2.2.2 Platform as a service (PaaS)

In PaaS, providers offer a computing resource delivery platform that are required to build applications, where users can deploy their application and run on it without monitoring the underlying hardware and software layers, however can maintain the control over the deployed applications.

2.2.3 Infrastructure as a service (IaaS)

IaaS is a way of providing computing resources such as Network, Storage, processing unit and operating system in over the internet. Instead of buying servers, storage and software enterprises can enlarge their resource competences by getting these computing resources on demand service over the internet as IaaS. Infrastructure as a service provider serves enterprises through private and public clouds.

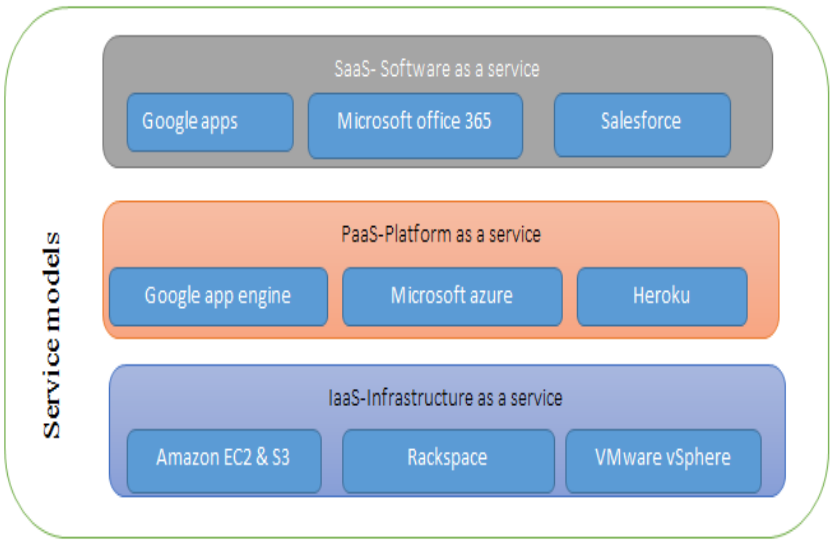

Figure 2: cloud computing services

2.3 Related works

In [13], authors proposed a load balancing algorithm that is upon on a particular behavior of the ants, in which they move toward the place where high amount of nourishing food is available. Antsalways find their nourishment and the food source to deliver back to their home. In ant's colony optimization algorithm, a head node first is selected based on nodes that have high number of neighboring nodes. Ant colony always underway searching nourishment from the head node. Every ant in the colony moves for food in identical way and same time. After they get their food they move back to the direction to their home, similarly the ant colony optimization algorithm proposed that when an ant moves forward to the way that encounters the overloaded, under loaded and free available nodes. During the time, ants get an overloaded node while in the earlier time it searches under loaded nodes then it moves backward and confirmed whether the node is still under loaded or freely available. If it is confirmed that the node is free or under loaded, the task is now distributed throughout all under loaded and freely available nodes in the system. As a result, this ant colony optimization algorithm is an efficient algorithm in utilization of computing resources. Despite the fact that it is limited because of very high extent of ants in a colony, the network traffic might be congested. The status of nodes after ant's observation is also not taken into account.

In [14] proposed a virtual machine mapping policy that depend on multi-computing resource load balancing algorithm, which deployed on private cloud setting in which virtual machines includes central scheduling controller to monitor resource availability for a work and free available resources are allocated for nodes in a system for processing in a cluster of nodes. Resource controller is there to calculate and investigate the detail information of resources, which are available free. This algorithm is based on mapping of virtual machines in private cloud. Load balancing procedure in this algorithm includes many levels such as acceptances of request, gaining detailed available resource information, and controllers of resource and scheduling which

performs scheduling of tasks and investigate resource availability for scheduled tasks. Resources are allocated for processing of tasks in which they have high amount then clients can access through the application. The limitation over this algorithm is that it does not take into account capabilities of node and the load of the network and if single point of failure happened the whole system is getting stop functioning.

Efficient load balancing technique [15] is performed using divisible load scheduling and weighted round robin method. In this approach, requests are divided into sub tasks, which are executed one after the other, and some can run in parallel. Every server is allocated with his or her corresponding weight and his or her previous allocation status. During a task is arrived to a cloud server, it is conceded to the load balancer which can divide requests to many arbitrary sub tasks based on divisible load balancing technique, which can implement a priority-based task allocation to the available servers. The load is assigned to a server having a higher weight to all sub tasks and the status of the server is changed to busy and again changed to ready. Transforms from busy to ready status for a server supports that it is no longer available in the ready server list, in such kind of way overlapping of tasks are prevented. The server again joins the ready server list after it finished its allocated work. In this load balancing scheme, network performance is increased whereas request completion time is minimized and it can eliminate task starvation problems. The limitation of this algorithm is in the distribution of loads to server is poor allocation.

Honey bee behavior inspired load balancing [16] is presented that is based on the behavior of honeybees' colony, which can be categorized under two types. The first honeybee is that it searches honey nourishment whereas the other is the honeybee that reaps for honey. The first category of bees goes in to have honey and to search honey nectar origins. After getting nectar sources these bees turns back to their home and they perform waggle dance, which is executed in the dance floor to where inactive forager bees can attend and following, to indicate the qualitative and quantitative amount of the nectar sources they searched. Likewise, in the honeybee inspired load balancing algorithm of tasks in the cloud environments, several internet servers are grouped together as virtual server. Depending on the nature of contributions of the profit virtual server processes the cloudlets. If after they do calculations and get less profit the server turns to their forage as forager bee and do migrations from one flower patch to the other flower patch. As a result, this conserves that the balance of the load in the system which improves performance. Although the computation in the profit may cause an additional overhead which result in a whole decrease in throughput.

A scheduling strategy [17] suggested a scheduling strategy in load balancing on Virtual machine resources that depends on genetic information to advance scheduling of tasks to be processed to realize load balancing by using the previous historical data and the recent status of the system. This algorithm makes a plotting relationship between the set of physical machines and the set of virtual machines and it finds the least effective solution by computing ahead in effect of the system afterwards the setting out of the needed virtual machine resources. It uses the same formula to find the finest load balancing scheduling using population. The output result of the experiment indicates that an enhancement in resource utilization, whereas the algorithm has high cost to store and retrieve the previous state of the data of the system nodes, and it may rise the response time and the processing cost.

III. PROPOSED HYBRID LOAD BALANCING SCHEME

In the existing fuzzy and genetic hybrid algorithm it allocate jobs to virtual machines to balance the load but it is not efficient in resource utilization in which it fails to allocate. In the existing hybrid, algorithm jobs have allocated to some virtual machines repetitively that leads to overloading and some machines remain idle. This leads to under resource utilization. Even if computational resource were free, they have used power, so resources should have effectively utilized. This problem have addressed in by modifying the existing algorithm in our proposed algorithm. The proposed algorithm can control all the existing virtual machines through utilization model that have employed in order to present a fine grained monitoring over available computing resources used by cloudlets.

In our proposed load-balancing scheme, we use a hybrid algorithm of fuzzy set theory that supports the genetic algorithm in finding fitness of the individual population, which reapplied in the crossover stage. In the first place, initial population that is common variables uses in the solution set for the problem has generated. Then datacenters, broker, virtual machines have created. Virtual machines have submitted to a broker, which specifies which host then provides a service to a particular VM. Required number of user cloudlets or tasks would create in which a response is route back to this requests. The algorithm calculates free available processing elements in the VM allocation that represents the provisioning policy of hosts to virtual machines in a datacenter. Based on the result, fitness value of each chromosome is calculated using fuzzy set theory. For better resource utilization, when a new cloudlet arrives the proposed algorithm confirms available free virtual machines. If it exists, the algorithm allocates the task to the VM based on its processing capacity. If there is no available idle virtual machine, the task has allocated to the VM whose recent task is going to be finished with shortest time as compared to the rest virtual machines. In this regard, virtual machines are efficiently utilized, no

virtual machine leftover free or idle and no over utilization of virtual machines. While it checks the utilization threshold level and if it did not exceed the maximum level it applies the genetic operators and it reiterates until the set conditions is meet. When the threshold level is exceeds the maximum level it submits cloudlets to the broker that determine the datacenter, which provide service and call to start the simulation function that finally produces the expected output

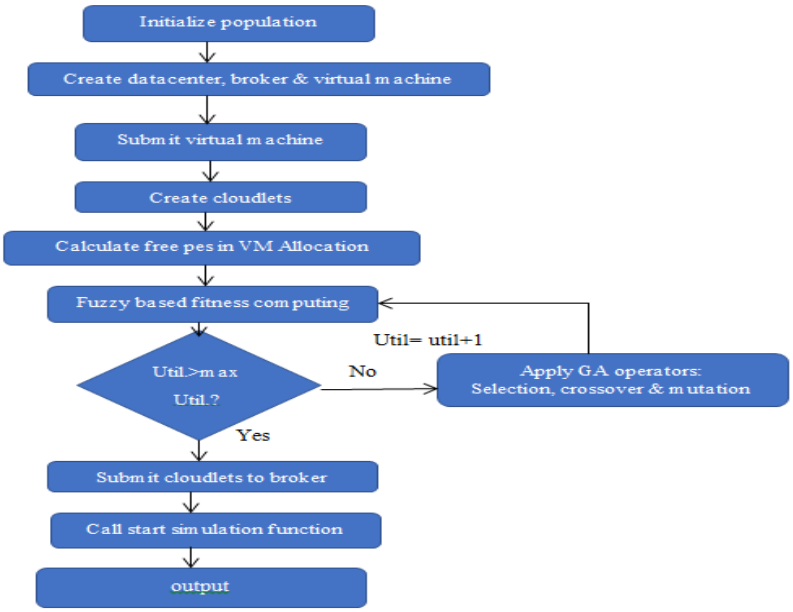

Figure 3: flowchart of the hybrid algorithm

IV. DISCUSSION AND RESULTS

In this chapter, we will present the experimental simulation results by comparing with other load balancing algorithms in the cloud-computing environment based on some evaluation parameters. We proposed an efficient load balancing which are upon effective utilization of computing services that can remove the challenge of resource over provisioning and under provisioning based on Virtual machine time shared provisioning policy. This load efficient load balancing is networked to all hosts and customers. The host controller that provides the service manages virtual machines. Our proposed algorithm is efficient in utilizing the service resources due to the algorithm allocates computational resources to tasks is through by considering the task length (task variations). Task with high length is allocated to a resource, which are powerful to perform the task in minimum execution time.

4.1 Simulation setup

Expression of Load

The whole virtual machine running on a physical machine at a particular virtual server can be defined as the load of the physical machine. Mathematically it can be expressed as follows.

Imagine that there is n number of cloudlets required to be allocated specified as:

$$C = \{C1, C2 \dots \dots \dots \dots Cn\}$$

In addition, there are k numbers of virtual machines in a datacenter.

$$V = \{V1, V2 \dots \dots \dots \dots Vk\}$$

The recent cloudlet load on the datacenter then is:

$$DC = \{VC1, VC2 \dots \dots \dots \dots Vk\}$$

Then we require getting a function f(C), in which the cloudlet C needed to be allocated to virtual machines V, that would make the load VC of every virtual machine V, is necessarily equal, that is:

$$VC1 \approx VC2 \approx \cdots \dots \dots . \approx VCk$$

In this work, we specified 50 virtual machines in datacenters and the size employed to host the application is 1000 MB in the experiment. Virtual machines would have from 256-2048 MB RAM memory with 500-1000 MB of existing bandwidth. Hosts in simulation have x86 system architecture, virtual machine monitor "XEN", and Linux operating system the hosts have 10048 MB RAM memory, 12000 million instructions per second, 100 GB host storage and 100000 available bandwidths. Every virtual machine has processors having various kinds of CPU and speed. The experiment considers the effect of the CPU processing capacity with its

processing speed. The algorithms perform by effectively allocating cloudlets to appropriate resource based on based on high resource capacities and high bandwidth for better utilization of computational resources and it minimizes the available free RAM. Those instructions, which weigh more spans that would be executed, would distribute to resources that have high bandwidth, good storage capacity, processing speed of the CPU. In the simulation experiment, the VM policy is based on timeshared in which, the resources are being shared among the instructions. Every instruction would get a resource for execution to a specific period and after they get finished, it is just released and allocated to other instructions. The following is the CloudSim toolkit-based experiment simulation configuration for the algorithm as shown in table. The result of the simulation of the proposed algorithm is based the resource capacity and instruction weight is compared with other load balancing algorithms

Table: 1 Experiment configuration

Datacenter	#VM	Image size	Memory	bandwidth	Mips	VM policy
10	50	1000	128-2048	50-1000	100-2000	Timeshared

Table: 2 Host configuration

Mips	Host storage	RAM	Bandwidth
12000	100 GB	10048	100000

4.2. Performance evaluation and analysis

The performance of the proposed algorithm FGA is conducted and analyzed based on the simulation result done using the CloudSim via Java programming language in Net Beans IDE. This simulation mainly evaluates the execution time, makespan, imbalance factor and resource utilization in the proposed load balancing algorithm in cloud computing environment. The performance of this work under these evaluation metrics is compared with the standard genetic algorithm. The time units in the performance evaluation parameters are in milliseconds.

Execution time

Various types of resource allocation approaches were presented [18]. In [19], execution time is well-thought-out in resource allocation strategy; it can address the involved issues of contention of computational resource and can maximize resource utilization by applying various kinds of approach of renting computing capacities. Execution time is the elapsed times of the cloudlet from submitting to it have been completed. In the following figure 2, indicates the execution time of the FGA or hybrid algorithm is less or minimized than the standard genetic algorithm, this is because the hybrid algorithm allocates task based on the fitness of the capacity of the computational resource. The maximum execution time in the hybrid algorithm is 2.29 milliseconds and the minimum lies at 0.5. However in the standard genetic algorithm these values are high.

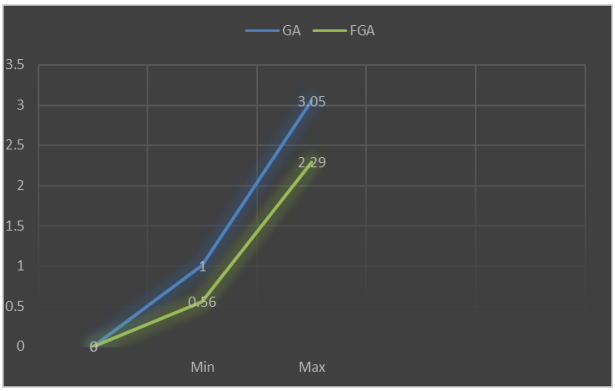

Figure 4: Execution time

Load imbalance factor

This is an aspect, which is related to determine load balancing in virtual machines. It can determine load among the virtual machines. This can be measured using the execution time of cloudlets in virtual machines. A less imbalance factor value shows good load balancing in Vm's. This can be computed through the following formula.

$$IF = \frac{Tmax+Tmin}{Tavg}$$

Where IF is load imbalance factor, $Tmax$ maximum execution time, $Tmin$ minimum execution time and $Tavg$ average execution time of all virtual machine. In the following figure 3, the fuzzy guided genetic algorithm has lower imbalance degree than the standard genetic approach. The FGA with less imbalance value is less loaded of jobs in VMs. Load imbalance degree in needed to identify the allocation of jobs in virtual machines. The following figure 3 indicates that the hybrid algorithm is have less value which is less loaded and balanced than the GA. The hybrid resource allocation is assigned services based on the fitness value due to this it is balanced.

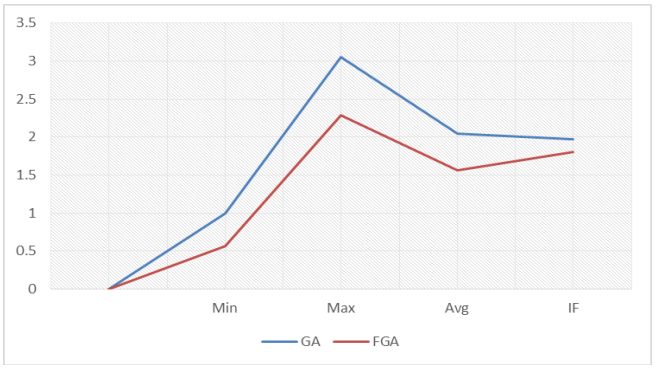

Figure 5: load imbalance factor

Resource utilization

Resource utilization is to use to find the capability of the cloud system to help in the course of utilization of resource parameters such as virtual machine list, task lists executed in the VMs and their corresponding time needed for processing tasks. The FGA offers better resource utilization by instruction size, which has a quick processing time in the distributed cloud system and homogeneous tasks. The algorithms take into account instruction or cloudlet size along with the capacity of processing heterogeneous VMs to assign tasks, so a greater number of cloudlets are assigned to higher processing capacity in terms of CPU, RAM, and bandwidth of virtual machines in homogeneous tasks within distributed based cloud systems environments. This helps to in realizing or completing the cloudlets execution time in shorter time. The load balancing capability of nodes can be determined through resource utilization. In Fig. 4, FGA have better utilization than the GA. the proposed algorithm lies on 0.6 which is moderate utilization of resource because of the resource allocation policy is based on time shared and computational resources are assigned based on the suitability of the nodes.

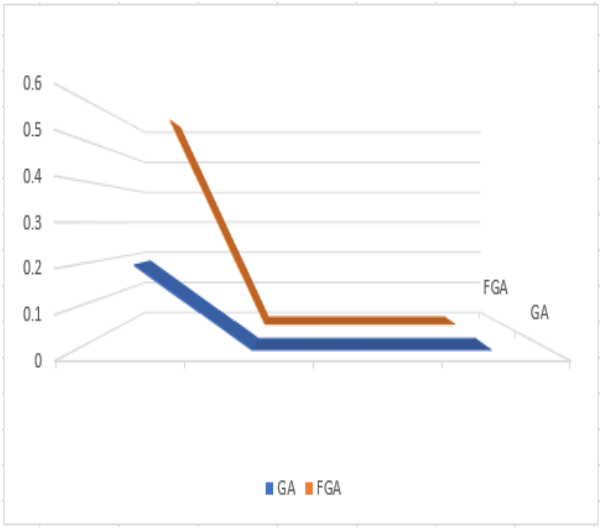

Figure 6: Resource utilization

V. CONCLUSION

In this work, we tried to explore the current experiences of the cloud computing technology. The main concept of load balancing in cloud-based systems is the primary research theme. The main involved problem in cloud-based system is load balancing and resource utilization, in which some nodes become overloaded in that computational resources are over utilized as a result response time increases and nodes may be underutilized or idle in which nodes ere under the threshold value of using computational resource and it increases in using

power consumption. As a result, load balancing of systems should be efficient to improve performances of the cloud-based technology. The recent load balancing approaches in the cloud computing environment have some drawbacks and that would impact on the system performances. Therefore, we tried to present a hybrid genetic algorithm that takes the advantages of the two algorithms through considering the VMs processing speed, memory usage cloudlet variations and bandwidth of the VMs, by having these parameters the algorithms targets to allocate computational resources to cloudlets based on their length variations. This is because of virtual machines have various processing capacity in distributed system. Our scheme tries to decrease the total instruction execution time and maximize utilization ratio. The simulation is experimented using CloudSim simulator and the analysis of the result shows that the execution time of the cloudlets is reduced as compared to the standard genetic algorithm.

REFERENCE

[1]. S. G. Domanal and G. R. M. Reddy, "Optimal load balancing in cloud computing by efficient utilization of virtual machines," in *2014 6th International Conference on Communication Systems and Networks, COMSNETS 2014*, 2014.

[2]. H. J. Younis, A. Al Halees, and M. Radi, "Hybrid Load Balancing Algorithm in Heterogeneous Cloud Environment," no. 3, pp. 61–65, 2015.

[3]. M. D. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis, and A. Vakali, "Cloud Computing: Distributed Internet Computing for IT and Scientific Research," *IEEE Internet Comput.*, vol. 13, no. 5, pp. 10–13, Sep. 2009.

[4]. S. Swarnakar, Z. Raza, S. Bhattacharya, and C. Banerjee, "A novel improved hybrid model for load balancing in cloud environment," *Proc. - 2018 4th IEEE Int. Conf. Res. Comput. Intell. Commun. Networks, ICRCICN 2018*, pp. 18–22, 2018.

[5]. A. Khiyaita, H. El Bakkali, M. Zbakh, and D. El Kettani, "Load balancing cloud computing: State of art," *Proc. 2nd Natl. Days Netw. Secur. Syst. JNS2 2012*, pp. 106–109, 2012.

[6]. S. Singh and I. Chana, "A Survey on Resource Scheduling in Cloud Computing: Issues and Challenges," *J. Grid Comput.*, vol. 14, no. 2, pp. 217–264, Jun. 2016.

[7]. K. Dasgupta, B. Mandal, P. Dutta, and J. Kumar, "A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing," *Procedia Technol.*, vol. 10, pp. 340–347, 2013.

[8]. M. Rana, S. Bilgaiyan, and U. Kar, "A study on load balancing in cloud computing environment using evolutionary and swarm based algorithms," *2014 Int. Conf. Control. Instrumentation, Commun. Comput. Technol. ICCICCT 2014*, pp. 245–250, 2014.

[9]. S. F. Issawi, "Efficient Adaptive Load Balancing Algorithm for Cloud Computing Under Bursty Workloads," vol. 5, no. 3, pp. 795–800, 2015.

[10]. Ari Liberman García, "THE CLOUD The Evolution of The Cloud," p. 9, 2013.

[11]. P. M. Mell and T. Grance, "The NIST definition of cloud computing," Gaithersburg, MD, 2011.

[12]. Y. Balagoni and R. R. Rao, "Importance of Load Balancing in Cloud Computing Environment: A Review," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 3, no. 5, pp. 77–82, 2014.

[13]. K. Nishant *et al.*, "Load Balancing of Nodes in Cloud Using Ant Colony Optimization," in *2012 UKSim 14th International Conference on Computer Modelling and Simulation*, 2012, pp. 3–8.

[14]. J. Ni, Y. Huang, Z. Luan, J. Zhang, and D. Qian, "Virtual machine mapping policy based on load balancing in private cloud environment," in *2011 International Conference on Cloud and Service Computing*, 2011, pp. 292–295.

[15]. S. S. Narayanan, M. Ramakrishnan, and M. Saadique Basha, "Efficient Load Balancing Algorithm For Cloud Computing Using Divisible Load Scheduling And Weighted Round Robin Methods," *Adv. Nat. Appl. Sci.*, vol. 11, no. 1, 2017.

[16]. L. D. Dhinesh Babu and P. Venkata Krishna, "Honey bee behavior inspired load balancing of tasks in cloud computing environments," *Appl. Soft Comput. J.*, vol. 13, no. 5, pp. 2292–2303, 2013.

[17]. J. Hu, J. Gu, G. Sun, and T. Zhao, "A scheduling strategy on load balancing of virtual machine resources in cloud computing environment," in *Proceedings - 3rd International Symposium on Parallel Architectures, Algorithms and Programming, PAAP 2010*, 2010, pp. 89–96.

[18]. V. Vinothina, S. Lecturer, and R. Sridaran, "A Survey on Resource Allocation Strategies in Cloud Computing," vol. 3, no. 6, pp. 97–104, 2012.