Journal of Green Engineering (JGE)

Volume-10, Issue-3, March 2020

Improving QoS of the Cloud Load Balancer Using Bio-Inspired Optimization Method

[1]P.L Abinaya and [2]A.Suresh

[1]*Student, Department of Information Technology, Kumaraguru College of Technology, Coimbatore, TamilNadu, India. E-mail: abinaya.18mds@kct.ac.in*
[2]*Assistant Professor,Department of Information Technology, Kumaraguru College of Technology, Coimbatore, TamilNadu, India. E-mail: it.suresh.a@gmail.com*

Abstract

Cloud computing has proven to be the emerging model of computing. Nowadays ,the number of clients moving into the cloud are increasing day by day. The cloud's efficiency relies upon the manner in which it handles the load. Cloud computing faces many problems, where the use of bio-inspired algorithms can tackle specific issues such as resource allocation, load balancing, and performance enhancement. Bio-inspired algorithms intend to tackle various types of issues by giving effective solutions. Load balancing is an important issue in the cloud computing network that ensures whether all processors or machines performs the same amount of work in equivalent amount of time. Throughout cloud computing, different models were developed with the goal of making cloud services efficient and beneficial to end-customers. In the proposed study, different types of bio -inspired algorithms are analyzed to improve the cloud scheduler's QoS and make the network load equally divided in order to give quicker availability to all clients that require the service. Bio-inspired algorithms like ant Colony Optimization and honey bee foraging load balancing algorithms are implemented using cloud analyst tool. The response time and processing time of the data centers is obtained.

Keywords: Cloud computing, QoS, Load Balancer, Honeybee foraging, Bio-inspired.

Journal of Green Engineering, Vol. 10_3, 697-712.

Alpha Publishers

1 Introduction

Cloud computing is undoubtedly a developing innovation where clients are able to coordinate applications with improved scalability, versatility and fault tolerance. Cloud computing accumulates all the computing resources and manages them effectively. Today the world relies on cloud infrastructure to store both private and public data. Cloud computing provides the users with the necessary hardware, applications, and services. Cloud computing's primary goal is to deliver optimal, lower cost infrastructure [14]. Nowadays more than hundred million computing devices are associated with the web. Cloud's primary objectives are to decrease costs, improve response times, give enhanced performance, so cloud is frequently known as pool of admininstration. Load has various sorts, such as loading the CPU, network load, memory storage problems, etc. Load balancing within cloud computing means spreading the workload of virtual machines over all nodes so as to maximize resource, service usage and provide a high degree of satisfaction for users. Device information is needed during load distribution if the load balancing method is dynamic in nature [7]. This scheme is more fault-tolerant, and more robust. Load balancing provides quicker response time and increases the throughput. Several algorithms are employed between cloud nodes to optimize cloud data. Cloud service provider manages all customer load to ensure effective delivery of service. This paper corresponds with the load balancing problem in cloud setting. Generally, the load balancing technique is implemented on vast volumes of data traffic and servers to spread the work among the nodes. A successful load balancing method should prevent one node being overloaded or under loaded. This paper compares the various types of bio-inspired algorithms in load balancing using cloud analyst tool. The purpose is to evaluate those algorithms efficiency.

2 Related Works

Amrita Jyoti et.al [1] reported the latest techniques implemented regarding load balancing and service brokering, and they are analysed according to main parameters. They addressed trend solution of load balancing and considered several of the solution and its classifications of cloud computing load balancing algorithms.

Sambit Kumar Mishra et.al [2] expressed a scientific classification for all the cloud load balancing algorithms and spotlights on assessing presentation of heuristic-based calculations and the simulation is directed in the CloudSim. The parameters utilized for assessing the calculations are the makespan and the vitality utilization.

SomulaRamasubbareddy et.al [3] reported the algorithms for load balancing like round robin, ESCE and Throttled. In cloud analyst, the above algorithms are executed, and the outcomes are simulated using various scenarios. By comparing the overall response time, processing time and cost of the data center they conclude that the throttled algorithm appears to be superior to the other algorithm.

Bakul Panchal et.al [4] reported the different throttled calculations utilized inside distributed computing for load balancing. Throttled algorithm process is mind boggling in nature, and can decrease administration demand reaction time.

Violetta N et.al [5] emphasized the cloud analyst platform to concentrate on study of the throttled, round robin and active load balance monitoring algorithms. The analyses have shown that throttled algorithm's overall response time and datacenter time is better than others.

Ahmed M. Manasrah et.al [6] concluded in achieving minimum response time. Analysis is done using cloud analyst tool. The outcome shows that the recommended technique has a huge increment in total average response and processing time and can perform better on the off chance that it embraces throttled algorithm for load balancing.

3 Cloud Analyst

There have been a couple of extraordinary toolkits that are utilized to show a virtual situation to examine the huge, adaptable Internet application. It became evident, however, that providing a useful tool that has a degree of visualization capability is far superior to other toolkits [12]. Cloud analyst is an open source platform with some new extensions, based on cloudsim. Cloud analyst has an excellent facility for GUI. It has two different VM management and service broker functions, that can be accessed in cloudsim together. The outcome of the simulation helps to improve QoS. Fig. 1 represents the cloud analyst architecture. The cloud analyst architecture is based on Cloudsim toolkit with some extensions. Unlike Cloudsim, in cloud analyst the graphical user interface allows developer to concentrate on simulation instead of programming. In cloud analyst an experiment can be repeated any number of times with the help of cloudsim extension framework. The geographical location of the user base and the data center can be selected by the developer and the outcomes are in the form of graphs and charts that helps in easy understanding.

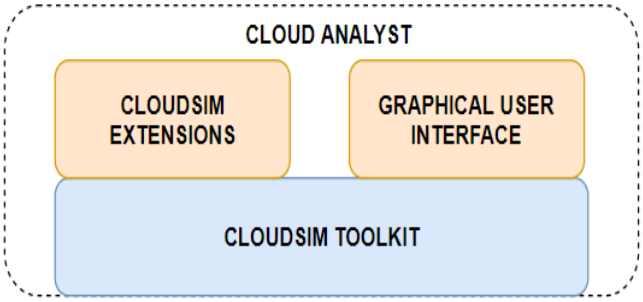

Fig. 1 Cloud analyst architecture

3.1 Main components of Cloud Analyst

3.1.1 Region

The earth is divided as six parts, based on six different landmasses [12]. The six regions and their corresponding region id is mentioned in Table 1.

Table 1 The six regions and their region id

REGION	REGION ID
NORTH AMERICA	0
SOUTH AMERICA	1
EUROPE	2
ASIA	3
AFRICA	4
OCENIA	5

3.1.2 User Base

A User base comprises of set of clients and their primary functions are to send requests to the simulation process. During simulation the user base is considered as a one single unit. A solitary user base may have any number of users and they are considered as one whole unit. The size of the user base can be determined by the amount of traffic they generate during simulation.

Gernerally, for efficiency of the simulation, a user base should contain larger number of users.

3.1.3 Internet

It is the reflection of this present reality Internet, executing just the features that are vital to the simulation. It is utilized to depict the attributes of the Internet applied during simulation, which includes latencies and bandwidth between areas, present level of traffic, and data center execution level data.

3.1.4 Internet Cloudlet

The Internet cloudlet groups the user requests. It contains informations such as application id and user base name. The Internet Cloudlet, for instance, conveys information like the magnitude of the request, and the input and output files. The internet returns the response to the user base with the help of the initiator and target application id from the internet cloudlet.

3.1.5 Data Center Controller

The data center controller is responsible for creating and destructing a VM. The user base routes the request to the data center controller via internet. The data center controller is responsible for allocating the appropriate VM for the request.

3.1.6 VM Load Balancer

The Data Center controller utilizes a VM Load Balancer to make sense of which VM ought to be distributed to request from the user base [10]. As of now three VM Load balancers are available in cloud analyst accompanied with three service broker policies. The VM load balancer policies are as follows

1) Round Robin load balancer
2) Active monitoring load balancer
3) Throttled load balancer

3.1.7 Cloud Application Service Broker

A service broker selects the data center to be allocated for request from the user base [9]. The traffic among user base and data center is managed by

service broker. At present, types of service broker policies available in Cloud analyst are

1) Service proximity-based routing
2) Performance optimized routing
3) Dynamically reconfiguring router

4 Routing of User Requests

Fig. 2 represents the routing of the user request in cloud analyst. It follows nine main steps. They are as follows

1. Initially, the request for the service is sent to the internet cloudlet from the user base which consists of group of users.
2. The internet cloudlet consists of requests application id and user base name.
3. The internet cloudlet with zero network delay send request to internet.
4. Theninternet consults any one of service broker policy to select appropriate data center controller.
5. Now request is forwarded to selected data center controller from the internet.
6. The data center controller selects any one of VM load balancer policies to process request.
7. The simulations result fromselected data center controller is forwarded to internet.
8. Now , response of request is sent to user base via the internet by adding certain netwok delay along with response.

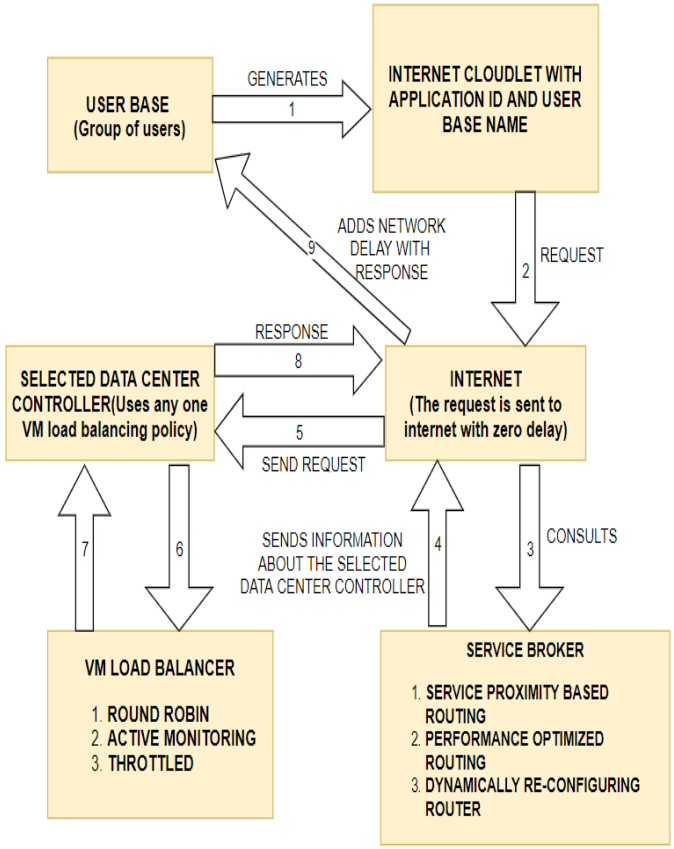

Fig. 2 Routing of user request in Cloud Analyst

5 Proposed Work

The algorithms bearing their natural source of inspiration are called Nature-Inspired Algorithms. The nature-inspired algorithm is of two types namely Swarm Intelligence and Bio-inspired algorithms [11]. In this proposed work, we have performed analysis on load balancing techniques like round robin, throttled then compared it with the bio-inspired algorithms like ant colony optimisation, honey bee foraging technique. Here the input for all the algorithms are the number of tasks from the user base and number of data centers. The obtained output is data centers response and processing time for the requests and the output differs for each algorithm. They are described below:

5.1 Round Robin Algorithm

Round Robin is an algorithm for the static load balancing. For Round Robin the preceeding states are not considered. It is not complicated and utilizes the round robin method in job allotment. It selects first node aimlessly and allocates the job to various nodes uniformly in Round Robin method. Round Robin's primary benefit is that it needn't bother with any inter process communication [10]. The method of the algorithm are follows:

1) Start
2) Make every single virtual machine accessible in the virtual machine state list
3) The value of the current virtual machine is allocated as - 1
4) When the customer sends another request to the server farm controller, the VM will be increased. If the current VM is accessible and if it satisfies the clients request then the VM will be allocated
5) Now after allocation the virtual machine state list should be updated
6) If the VM id is more prominent than list size, set VM id to zero to proceed in clockwise way
7) If data center controller is queued with request, then go to step 3.
8) Stop

5.2 Throttled Load Balancing Algorithm

A virtual machines record table is maintained by the throttled algorithm. The table additionally incorporates the condition of the Virtual Machine as accessible or occupied [4]. In the event that a client or customer sends a request to locate the proper virtual machine to play out the proposed work, the virtual machine list is iterated from start to finish and the virtual machine that is most promptly accessible is distributed. The steps of the algorithm are as per the following:

1) Start
2) Make every virtual machine accessible in the virtual machine state list
3) Throttled allocation is performed when the Datacenter Controller gets a new task. Return the id of the selected VM to the data center controller.
4) The virtual machine state list is refreshed on accepting the assignment or de-allotment occasion from data center controller
5) If data center controller is queued with request, go to step 3.
6) Stop

5.3 Ant Colony Optimization Based Load Balancing Algorithm

This is a bio-inspired algorithm. During traversal, ants deposits a substance which is known as pheromone [8]. Based on those pheromone tracks, remaining ants reach the food source. The ants utilize these trails of pheromones to choose following node. The ants can even change their ways after experiencing any obstacle in their way. The ants upon traversal starting with one node then onto the next update the pheromone trail of that way, so a path turns out to be increasingly feasible if more ants cross upon it. Paths that have the most pheromone strength have the minimum distance between the starting point and the best food source [13]. The steps are as follows:

1) Start
2) Initialize pheromones[n][n]
3) Move ants through Virtual machines to lay pheromone
4) Calculate the probability[n] according to VM visited
5) Update pheromone[n] table
6) Select the underload VM with highest probability from pheromone[n]
7) Allocate cloudlet to the selected underload VM
8) Stop

5.4 Honey Bee Foraging Load Balancing Algorithm

This algorithm goes under bio-inspired algorithms, where they are composed by the idea of the bumble bees. Bumble bees were isolated into two separate structures, in particular finders and reapers [11]. The finder bee is responsible for finding the location of the nectar in the neighbourhood. Then it performs a waggle dance to determine the location and the quality and quanity of the nectar available to the other bees. Then the reaper bee follows the finder bee to the location where the nectar is present and they collect the nectar. The process continues until the amount of nectar in that location decreases. The steps of the algorithm are as folows

1) Start
2) Set the number of virtual machines and calculate the throughput
3) Initially the load of each virtual machine is set to null
4) The first task is sent to the VM which has the high throughput
5) Check whether if the current VM load exceeds the threshold value
6) If yes, the VM with the high throughput is chosen and it is checked whether the heap is beneath the threshold esteem. On the off chance that truly, the task is doled out to that VM
7) If the current VM load is underneath the threshold esteem then the task is

sent to the current VM
8) Go to step 9 if all tasks are allocated to the VM otherwise go to step 4
9) Stop

6 Results and Discussions

Cloud analyst tool is used to execute the algorithms. We assessed four different algorithms and the outcomes from the simulation in different situations are tabulated. The number of data center and the client bases are altered for each situation. All the scenarios were tested over a one-day period to obtain reliable results

Case 1

In this scenario, a single user base at region 0 and six data centers at different regions are considered. The output of the comparison is given in table 2

Table 2 Comparison between the algorithms for case 1

Algorithm	Average response time (ms)	Average data center processing time (ms)	Cost ($)
Ant colony	59.54	9.91	60.99
Honey Bee	55.12	5.84	60.99
Throttled	62.05	12.41	60.99
Round Robin	62.57	12.94	60.99

Case 2

In this scenario, three user bases at regions 2, 4, 5 and six data centers at different regions are considered. The output of the comparison is given in table 3

Table 3 Comparison between the algorithms for case 2

Algorithm	Average response time (ms)	Average data center processing time (ms)	Cost ($)
Ant colony	63.29	13.65	69.20
Honey Bee	57.11	7.47	69.20
Throttled	65.42	15.78	69.20
Round robin	66.94	17.30	69.20

Case 3

In this scenario, four user bases at regions 0, 1, 2, 5 and six data centers are considered. Here two data centers belong to the same region and the remaining 4 data centers are placed at various regions. The output of the comparison is given in table 4

Table 4 Comparison between the algorithms for case 3

Algorithm	Average response time (ms)	Average data center processing time (ms)	Cost ($)
Ant colony	95.68	8.28	78.02
Honey Bee	92.14	4.74	78.02
Throttled	98.49	11.10	78.02
Round robin	100.43	13.04	78.02

The following figures 3,4,5 represents the case 1,2 and 3

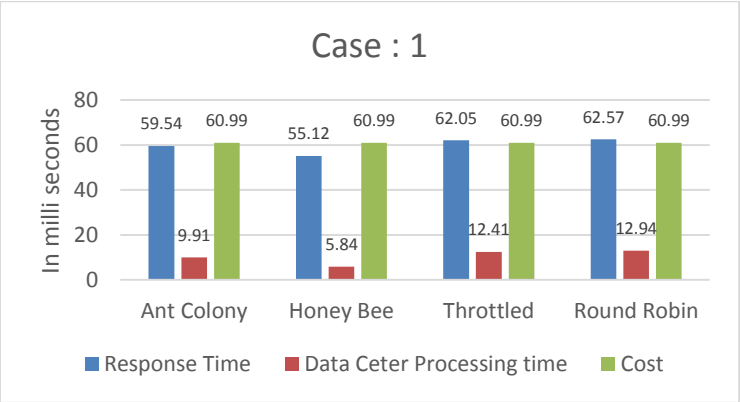

Fig 3 Representation of case 1

Figure 3 shows the outcome of the simulation done in Cloud Analyst where six data centers at different regions were considered. The average response and processing time of the data center are obtained in milli seconds. In this scenario honey bee and ant colony algorithms has minimal response and processing time.

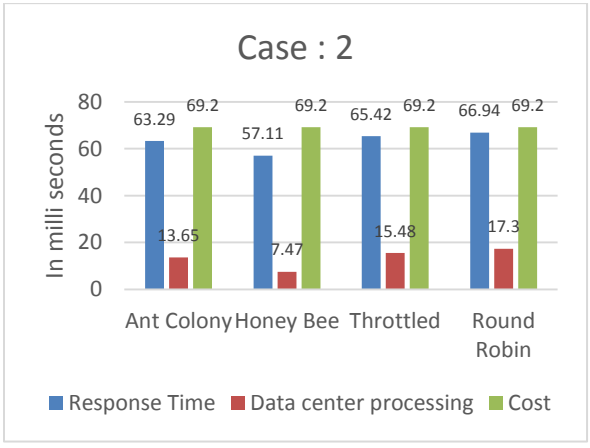

Fig 4 Representation of case 2

Figure 4 shows the outcome of simulation done in Cloud Analyst where three user bases and six data centers at various regions were considered. The average response and processing time of data center are calculated in milli

seconds. In this scenario honey bee and ant colony algorithms has minimal response time and data center processing time when compared to round robin and throttled algorithm.

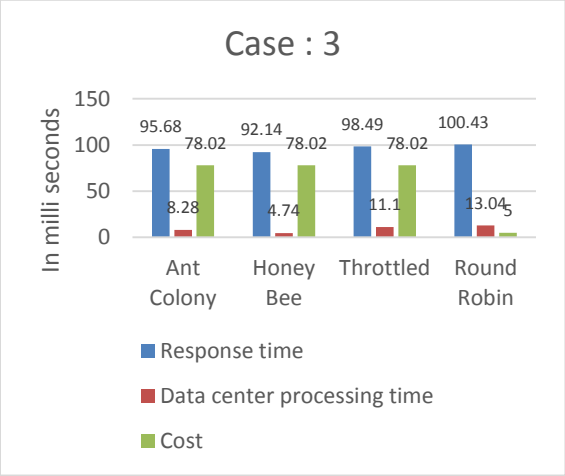

Fig 5: Representation of case 3

Figure 5 shows the outcome of the simulation done in Cloud Analyst. In this case, four user bases are placed at four different regions and six data centers are taken where two data centers are placed in same region and remaining four are placed at various regions. The average response and processing time of the data center are obtained in milli seconds. In this scenario honey bee and ant colony algorithms has minimal response time and data center processing time. From the table 2, 3, 4 it is shown that bio-inspired algorithms namely ant colony and honey bee has lesser response time and data center processing time when compared to round robin and throttled.

7 Conclusion and Future Work

In cloud computing, load balancing acts as major task. Load balancing is needed to efficiently spread the task among all nodes. Deducting an algorithm that substantially decreases the response time and processing costs is a major challenge. The bio-inspired algorithms are performed to enhance performance and thus have a better QoS. The Bio-Inspired Algorithms make up a huge fraction of algorithms inspired by nature. These algorithms are

inspired by some biological phenomenon or by any natural organism. The ant colony algorithm is inspired from the food gathering phenomenon of ants and the honey bee algorithm is inspired from nectar gathering behaviour of bees. Both these algorithms are flexible and are used to solve optimisation problems. From the outcome of the simulation it is evident that bio-inspired algorithms outperform round robin and throttled algorithm. The response time and processing time of data center of the honey bee algorithm are comparatively lesser that other algorithms. Hence if the response time is lesser the Qos will be improved. The analysis can be additionally upgraded by contrasting with other nature-based algorithms like particle swarm optimisation and fire flies algorithms. These Nature Inspired algorithms can thus be combined with other algorithms to improve their QoS.

Reference

[1] Amrita Jyoti, Manish Shrimali, et.al, "Cloud computing using load balancing and service broker policy for IT service: a taxonomy and survey", in Journal of Ambient Intelligence and Humanized Computing, 2020.

[2] Sambit Kumar Mishra, Bibhudatta Sahoo, Priti Paramita Parida, "Load Balancing in Cloud Computing: A big Picture", Journal of King Saud University - Computer and Information Sciences, Vol.32, No.2, pp. 149-158, 2020.

[3] SomulaRamasubbareddy, T. AdityaSaiSrinivas, K. Govinda, S.S. Manivannan& E. Swetha, "Analysis of Load Balancing Algorithms using Cloud Analyst", IJRTE, Vol.7, No.6S2, pp. 684-687, 2019.

[4] Bakul Panchal, Smaranika Parida, "Review Paper on Throttled Load Balancing Algorithm in Cloud Computing Environment", IJSRSET, Vol.4, No.2, pp. 201-204, January-February 2018.

[5] Violetta N, Volkova1, Liudmila V. Chernenkaya, et.al, "Load Balancing in Cloud Computing", Conference of Russian Young Researchers in Electrical and Electronic Engineering, IEEE, pp. 387-390, 2018.

[6] Ahmed M. Manasrah, Tariq Smadi, Ammar ALmomani, "A Variable Service Broker Routing Policy for data center selection in cloud analyst", Journal of King Saud University - Computer and Information Sciences, Vol.29, No.3, pp. 365-377, 2017.

[7] Aayushi Sharma, Anshiya Tabassum, et.al, "A comparative study of load balancing algorithms in cloud computing", International Journal of Innovative Research in Computer and Communication Engineering, Vol.5, No.4, pp. 8322-8331, 2017.

[8] Acharya Mitali Nilesh, et.al, "Load Balancing in Cloud Computing using Ant Colony Optimization", International Journal of Computer

Engineering & Technology, Vol.8, No.6, pp. 54-59, November-December 2017.
[9] Hetal V. Patel, Ritesh Patel, "Cloud Analyst-An Insight of Service Broker Policy", International Journal of Advanced Research in Computer and Communication Engineering, Vol.4, No.1, pp. 122-127, January 2015.
[10] Reena Panwar, Bhawna Mallick, "A Comparative Study of Load Balancing Algorithms in Cloud Computing", International Journal of Computer Applications, Vol.117, No.24, pp. 33-37, 2015.
[11]Ritu Kapur, "Review of Nature Inspired Algorithms in Cloud Computing", International Conference on Computing, Communication and Automation, IEEE, pp. 589-594, July 2015.
[12] Rakesh Kumar Mishra,et.al, "Service broker algorithm for cloud-analyst" , International journal of computer science and information technologies, Vol.5, No.3, pp. 3957-3962, 2014.
[13] Ekta Gupta,et.al, "A Technique Based on Ant Colony Optimization for Load Balancing in Cloud Data Center", International Conference on Information Technology, IEEE, pp.12-17, 2014.
[14] Jayant Adhikari, Sulbha Patil, "Load Balancing the Essential Factor in Cloud Computing", International Journal of Engineering Research & Technology, Vol. 1, No. 10, pp.2-5, 2012.

Biographies

Abinaya P L was born in Coimbatore, India. She received her Bachelor degree in Information Technology from Kumaraguru College of Technology, Coimbatore, in 2018. She is currently pursuing a Master degree in Data Science at Kumaraguru College of Technology, Coimbatore. Her research interest includes Machine Learning, Cloud Computing and Data Mining.

Mr. A. Suresh currently working as Assistant Professor in the department of Information Technology, Kumaraguru College of Technology, Coimbatore, India. He completed his master of Technology (Information technology) in the year 2011 and Bachelor of technology (Information Technology) in the year 2009.Had 8+ year of experience and his research interest was grid scheduling, cloud scheduling, Data analytics, Smart agriculture. He is an active member of IEEE KCT student branch. Delivered Technical many sessions for the IEEE student branch members.