# International Journal of Engineering Sciences & Research Technology

(A Peer Reviewed Online Journal)
Impact Factor: 5.164

IJESRT

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

IJESRT
INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

# DEEP LEARNING BASED LOAD BALANCING USING MULTIDIMENSIONAL QUEUING LOAD OPTIMIZATION ALGORITHM FOR CLOUD ENVIRONMENT

**Deeksha Verma**[*1] **& Er. Shivani Rana**[2]
[*1&2]Himachal Pradesh Technical University, Hamirpur, Himachal Pradesh

**DOI**: https://doi.org/10.29121/ijesrt.v9.i10.2020.15

## ABSTRACT

Cloud computing becoming one of the most advanced and promising technologies in these days for information technology era. It has also helped to reduce the cost of small and medium enterprises based on cloud provider services. Resource scheduling with load balancing is one of the primary and most important goals of the cloud computing scheduling process. Resource scheduling in cloud is a non-deterministic problem and is responsible for assigning tasks to virtual machines (VMs) by a servers or service providers in a way that increases the resource utilization and performance, reduces response time, and keeps the whole system balanced. So in this paper, we presented a model deep learning based resource scheduling and load balancing using multidimensional queuing load optimization (MQLO) algorithm with the concept of for cloud environment Multidimensional Resource Scheduling and Queuing Network (MRSQN) is used to detect the overloaded server and migrate them to VMs. Here, ANN is used as deep learning concept as a classifier that helps to identify the overloaded or under loaded servers or VMs and balanced them based on their basis parameters such as CPU, memory and bandwidth. In particular, the proposed ANN-based MQLO algorithm has improved the response time as well success rate. The simulation results show that the proposed ANN-based MQLO algorithm has improved the response time compared to the existing algorithms in terms of Average Success Rate, Resource Scheduling Efficiency, Energy Consumption and Response Time.

**KEYWORDS:** Cloud computing, Resource Scheduling, Load Balancing, Virtual Machines, MQLO algorithm, MRSQN technique and ANN.

## 1. INTRODUCTION

In recent years, there has been an exceptional development in the measure of information being accumulated around the world and to store the information a large storage required [1]. The term cloud is a network design used by network engineers to locate and connect with different network devices to access or store information [2]. The shape of this network design is similar to cloud and hence named as “Cloud”. The general structure of cloud computing is shown in the below Fig. 1.

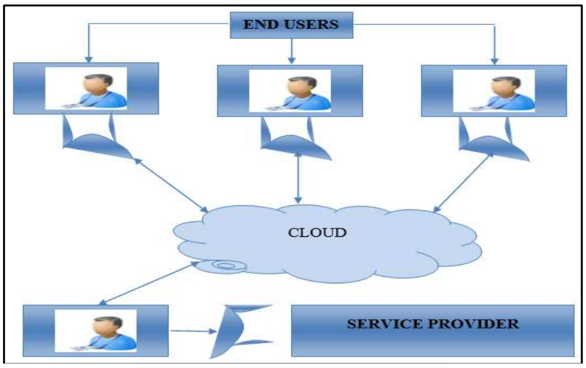


*Fig 1: General Cloud Model*

Cloud computing is a novelty that utilizes Web and focused remote servers to store information and applications. Cloud computing is a way to deliver information technology (IT) services from the Internet to resources through web-based tools and applications, instead of being directly linked to a server. Instead of storing files on a specific hard drive or local storage device, cloud-based storage lets you store them in a remote database. This innovation takes into account a great deal more effective [3] figuring by concentrating information storage, handling and data transfer capacity.

Cloud computing environment is a new, great way to provide IT-related services. In fact, this technique is a dream of many entrepreneurs who want to get all possible IT services together. In the age of information society, such services are even more popular, as entrepreneurs helps to solve IT problems quickly and efficiently in an integrated company [4]. From a simple point of view, the cloud computing environment is about the requirements of IT companies and about IT: various types of services are offered to the cloud users through Internet and on real-time applications. So, in general, let's define the most common types of services based on what we call cloud computing [5].

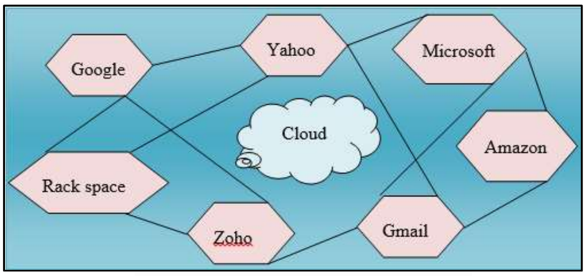


*Fig 2: Cloud Computing Platform*

Cloud computing is a general term for anything that includes conveying facilitated administrations over the Internet. These administrations are extensively separated into three classifications: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS). Cloud computing is separated into three portions: application, storage and network [6]. Each section fills an alternate need and offers distinctive items for organizations and people the world over. Cloud computing consists of two elements namely; front and back end. The front end is the customer part of the cloud computing system [7]. It consists of the interfaces and applications needed to access the cloud computing platform. The back end comprises of the number of resources required for cloud computing services, if the cloud belongs to it. This is a virtual machine, server, data storage, security mechanism, and so on and controlled by cloud service provider [8]. The architecture of cloud computing model is shown in the Fig. 3.

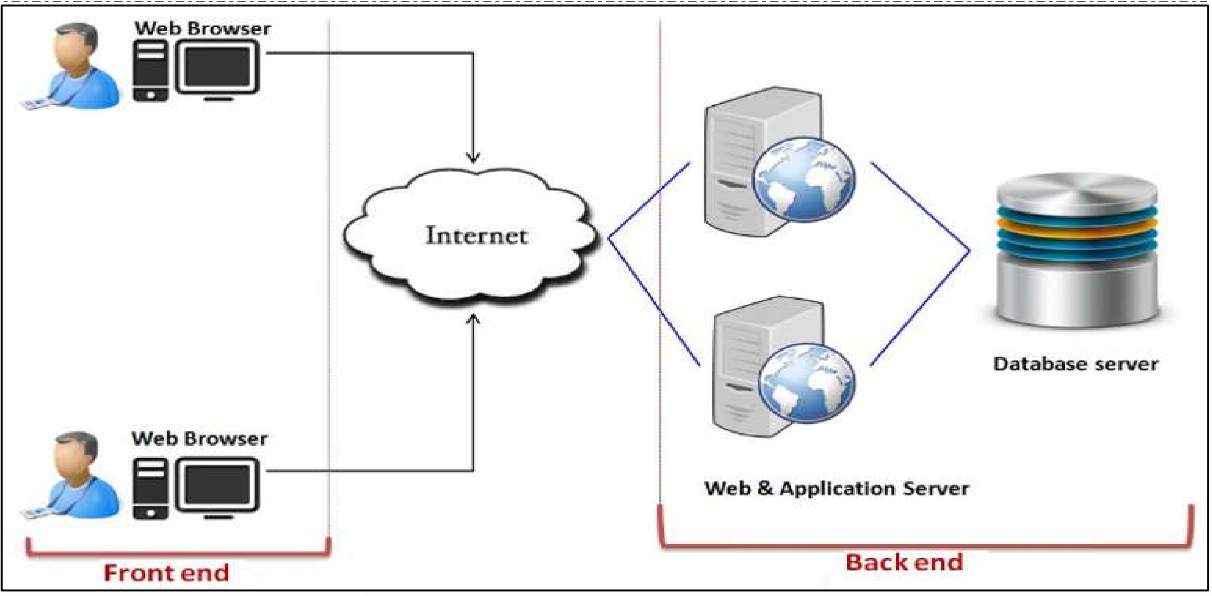


*Fig 3: Architecture of Cloud Computing Model*

Fig 3 represents the architecture of proposed Cloud Computing Model based on the load balancing concept using the Artificial Neural Network (ANN) as deep learning mechanism with Multidimensional Resource Scheduling and Queuing Network (MRSQN) and Multidimensional Queuing Load Optimization (MQLO) algorithm [9]. Exiting cloud computing model faced some crucial issues:

***Load Balancing:*** It is the process of distributing jobs or load among various nodes of the system so that time efficiency can be increased. Also, proper utilization of resources takes place. Load balancing is initializes by collecting the awaiting execution time from each of the created VMs then arranging it in ascending order to classify the number of tasks in each VM and arrange it in increasing order. Load balancing is accomplished by rearranging the classified tasks, based on the latest execution time in each VM.

***Virtual Machine Migration:*** Virtual Machine (VM) migration is a powerful management technique that gives data center operators the ability to adapt the placement of VMs in order to better satisfy performance objectives, improve resource utilization and communication locality, mitigate performance hotspots, achieve fault tolerance, reduce energy consumption. A virtual machine provides interface identical to underlying hardware i.e. all devices, interrupts, memory, page tables etc. Advantages of Virtualization:

- Efficient use of resources
- Superior degree of abstraction
- Replication
- Scalable and flexible infrastructure

***Fault Tolerance:*** Fault tolerance allows the virtual machines to continue its job even any part of system fails. This technique migrates the virtual machine from one physical server to another physical server based upon the prediction of the failure occurred. Fault tolerant migration technique improves the availability of physical server and avoids performance degradation of applications. So, the main motivation behind the development of ANN-based MRSQN with load balancing using MQLO algorithm for cloud computing environment and the major contributions in this research are listed as:

- We present an ANN based load balancing model with MQLO algorithm for cloud computing environment to manage the load on the cloud server.
- The concept of Multidimensional Resource Scheduling and Queuing Network (MRSQN) is used to detect the overloaded server and migrate them with ANN as a classifier.

- To validate the proposed load balancing model in cloud computing, a comparison with the existing state of the art is presented in terms of Average Success Rate, Resource Scheduling Efficiency, Energy Consumption and Response Time.

The remaining research article is organized as follows. In Sect. 2, related works are presents their analysis. Sect. 3 describes the methodology and the simulation results are presented in Sect. 4. Finally, Sect. 5 described the overall conclusion with the future trends of load balancing in the cloud computing environment.

## 2. RELATED WORK

The number of exiting work are reviewed to find out the important points and issues in the cloud computing model. In 2018, ***V. Priya et al.*** presented a mechanism for the resource scheduling algorithm with load balancing for cloud service provisioning. In this research, authors introduced an integrated resource scheduling with the concept of load balancing algorithm for efficient cloud service provisioning. They used a fuzzy-based multi-dimensional resource scheduling model using the concept of MQLO with MRSQN to obtain resource scheduling efficiency in cloud infrastructure. Increasing utilization of Virtual Machines through effective and fair load balancing is then achieved by dynamically selecting a request from a class using MQLO algorithm. A load adjusting calculation is then actualized to dodge underutilization and overutilization of assets, improving idleness time for each class of solicitation. Recreations were directed to assess the adequacy in cloud server farms and results shows that the proposed strategy accomplishes better execution as far as normal achievement rate, asset planning proficiency and reaction time. Recreation investigation shows that the strategy improves the asset booking effectiveness by 7% and furthermore diminishes the reaction time by 35.5 % when contrasted with the best in class works. ***Guo et al.*** 2019 presented an energy efficient dynamic offloading and resource scheduling scheme to minimize the energy consumption as well as decrease the completion time. The main motive of this paper is to solve the problem of energy consumption under hard constraints. For this, the authors have initially divide the problem into energy efficient cost reduction with satisfactory task dependency along with completion time. The author mainly used three algorithms such as (i) Algorithm for selection of computation offloading (ii) clock frequency control as (iii) broadcast power allocation. The selection of computation depends upon the workload of tasks and also enhances the completion time. ***Mukherjee et al.*** in 2019 proposed a two-fold model to solve the problem of (i) mobility of mobile devices as these devices are moving with different speed , (ii) the task allocation become difficult in case if the location of mobile device changed. The primary problem has been resolved by using a technique in which the mobile devices utilized distant cloud server for the execution of task. In case if the connection break between server and mobile device a message (push notification) has been sent by the server to reconnected the device to the server. TO resolve the location problem, the virtual machine migration concept has been used. The current state of the sample is transferred from the previous cloud to the new download platform where the unloading process is resumed. It has been analyzed that the suggested task reduces the power consumption by about 30-63%. Experienced results are achieved using a mobile device at different speeds inside and outside the university building. ***Zhou et al.*** in 2017 recommend a code offloading architecture, named as mobile or m-Cloud. This model mainly comprises of different units such as: mobile devices, nearby cloudlets and public cloud services. The impact of mobile device in terms of network condition for emptying decisions is investigated by offering a discharge decision algorithm that is context-aware to enable decentralized media to decide on the decommissioning of appropriate cloud resources for selecting and unloading. A malfunction detection function and recovery policies has been discovered for the proposed m-cloud system. Experiments have been conducted to evaluate the performance of the algorithm. The results show demonstrated that the designed system and the integrated decision approach are capable to offer decision on selecting wireless medium as well as cloud resources that are based on various perspectives of mobile devices. The performance in terms of makes pan, SLA and energy consumption has been measured.

Based on the analysis of existing research using regarding the resource scheduling or load balancing the optimization techniques were used and resource scheduling become proper but we examine the load balancing face some issues related to the classification of over or under loaded cloud server. So we decided to use an artificial intelligence algorithm as a classifier and we select ANN technique. An ANN-based MRSQN with load balancing using MQLO algorithm for cloud computing environment is proposed.

## 3. METHOD OF RHM MODEL

In this section of research article, we explain the methodology and algorithms of the proposed model using ANN-based MRSQN for load balancing and resource scheduling with the concept of MQLO algorithm for cloud computing environment. The procedural steps of proposed model in cloud computing environment are defined as follows:

***Proposed Simulator:*** Firstly design simulator for the simulation of proposed ANN-based MRSQN with load balancing using MQLO algorithm for cloud computing environment using the concept of graphical user interface in MATLAB 2016a software and it is shown in the Fig. 4.

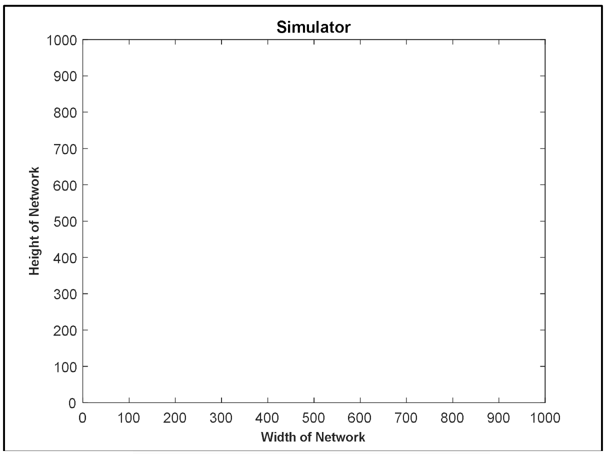


***Fig 4: Simulator with Area=1000m$^2$***

***Deployment of User & Servers in the Simulator:*** After the simulator designing, we deploy the user and servers within the simulation area that is shown in the Fig. 5

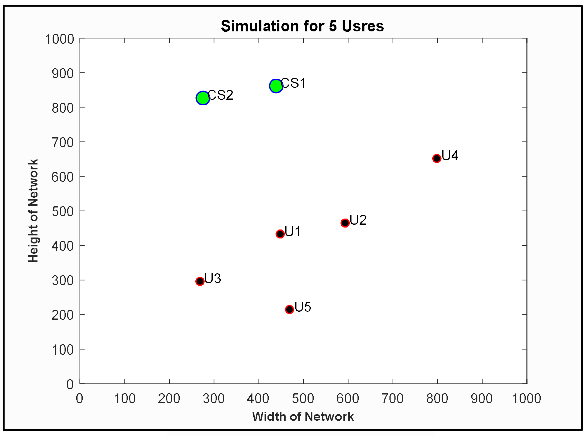


***Fig 5: Deployment Process***

After the deployment, we initial provides some basic parameters to each servers, users and VMs for the simulation purpose, then we sorted the users, servers and VMs parameters in the descending order that helps to identify the most effective that having better CPU, Memory and Bandwidth. The presentation of users, servers and VMs based on their unsorted and sorted value is shown in the Fig. 6.

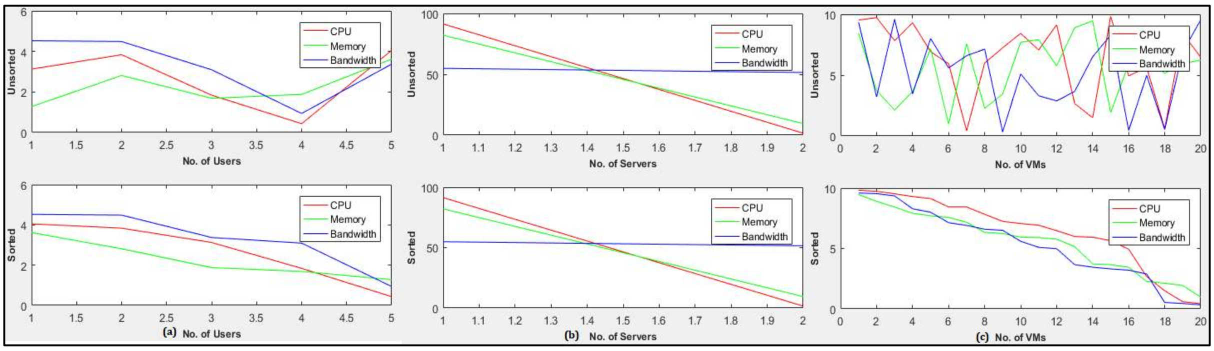


*Fig 6: Parameters of (a) Users (b) Servers and (c) VMs*

Above Fig. 6 represents the basic parameters of users, servers and VMS that is used in the proposed ANN-based MRSQN with load balancing using MQLO algorithm for cloud computing environment. After that, resource allocation and load balancing is simulation using the ANN as deep learning approach and the allocation user and server with load balancing is shown in the Fig. 7.

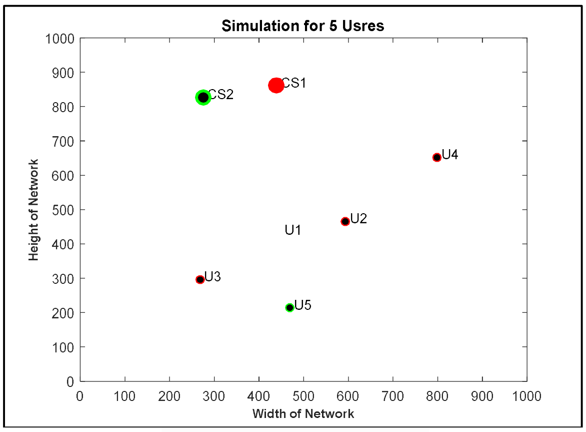


*Fig 7: Resource Allocation*

The develop simulation model for resource scheduling and load balancing is shown in the Fig 7 and then we compute the performance parameters of the model in terms of Average Success Rate, Resource Scheduling Efficiency, Energy Consumption and Response Time using the given algorithm:

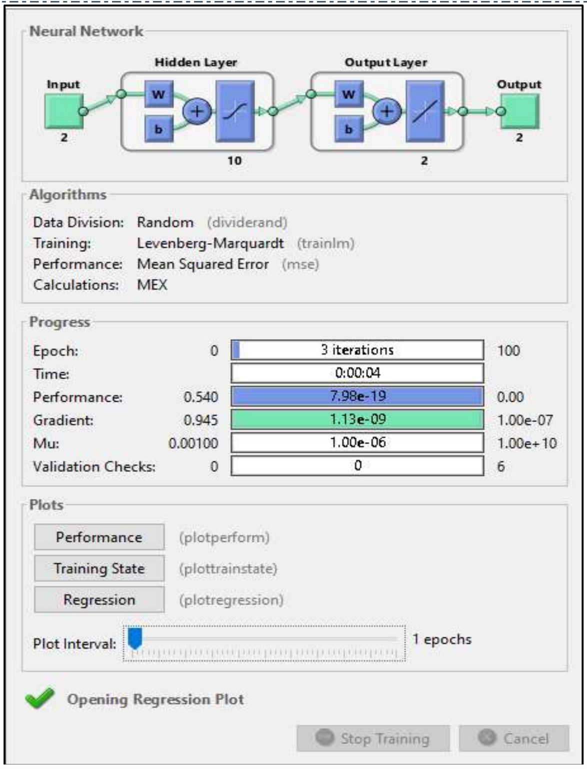
**Algorithm: ANN with MQLO:**

```
Start
Load Parameters, RTF = [sCPU sMem sBW]
For i=1→cUser
TimInt(i)=rand;
DemR(i)=max([uCPU uMem uBW]);
UtilRate(i)=DemR(i)*TimInt(i)
If max (UtilRate(i))<=max(RTF)
AvgPTimeSer(i)=DemR(i)/(1-UtilRate(i))
            AvgPTimeAllSer(i)=(AvgPTimeSer(i)*TimInt(i))/(TimInt(i))
AllocatedServer(i)=ceil(cServer*rand);
End – If
End – For
For i=1→cServer
```

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

```
        ListAlloc(i)=numel(find(AllocatedServer==i))
    End – For
    For i=1→cServer
        If ListAlloc(i)>mean2(ListAlloc)
            Training_data= []
            Training_data= [sCPU sMem sBW]
            Target= []
            For j=1→size (Training_data)
                If sCPU(j)>=mean2(sCPU)
                    Target (j, 1) = 1
                Else if sCPU(j)<min(min(sCPU))
                    Target (j, 2) = 1
                Else
                    Target (j, 3) = 1
                End – If
            End – For
            Call ANN using
            Net = newff(Training_data,Target,Neurons=10)
            Net = train (Net,Training_data,Target)
            Outp = sim (Net,Training_data)
            AllocatedServer(i)=max(max(outp))
        Else
            AllocatedServer=AllocatedServer
        End
    End
    Return: Allocated Server
End - Algorithm
```

After that, we analyze the allocation calculate the performance parameters of proposed ANN-based MRSQN with load balancing using MQLO algorithm for cloud computing environment using the average success rate, resource scheduling efficiency, energy consumption and response time. The architecture of ANN is shown in Fig. 8 and their description is provided below the figure.



Algorithms

Data Division: Random (dividerand)
Training: Levenberg-Marquardt (trainlm)
Performance: Mean Squared Error (mse)
Calculations: MEX

Progress

```
Epoch:               0         3 iterations     100
Time:                          0:00:04
Performance:         0.540     7.98e-19         0.00
Gradient:            0.945     1.13e-09         1.00e-07
Mu:                  0.00100   1.00e-06         1.00e+10
Validation Checks:   0         0                6
```

Plots

Performance (plotperform)
Training State (plottrainstate)
Regression (plotregression)

Plot Interval: 1 epochs

✓ Opening Regression Plot

*Fig 8: ANN Architecture for resource scheduling and load balancing in proposed work*

ANN architecture consist of three basic layers such as input, hidden and output layer. The ANN is trained using users, servers and VMs properties for both scheduling and balancing purpose. The properties are applied as input parameters like as energy consumed by server to execute the task and CPU utilization with bandwidth consumption by servers or VMs. If the output is not as per the desired then the error generated is sent back to the hidden layer for adjusting the properties of nodes accordingly. In this way, network is trained with minimum error. The network composed of N number of interlinked neurons as indicated by an arrow in Fig. 9. These neurons updates the activation function of each neurons independently. During training the error generated by ANN network is known as Mean Square error.

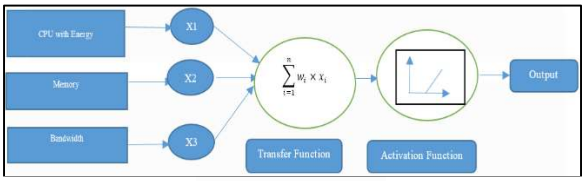


*Fig 9: ANN Layers*

After the development of the model, we presents an experimental section that comprises the simulation results of proposed ANN-based MRSQN with load balancing using MQLO algorithm for cloud computing environment and compare with the existing work.

## 4. RESULTS AND DISCUSSION

In this section, we describe the simulation results of the proposed ANN-based MRSQN with load balancing using MQLO algorithm for cloud computing environment. Firstly, we presents the simulation results of the proposed work.

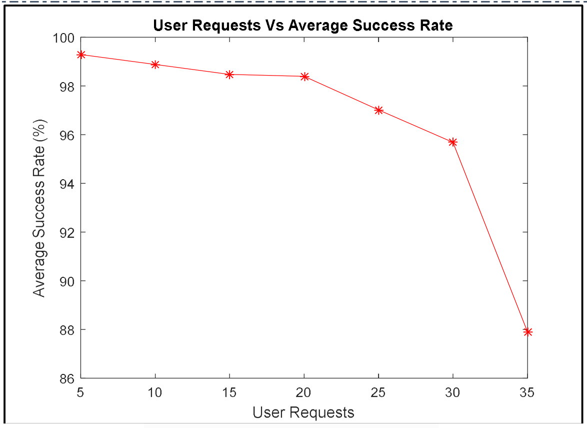


*Fig 10: Average Success Rate*

Above figure represents the average success rate of proposed model with respect to user request and we consider variation in user request from 5 to 35 users. From the observation, we find that the average success rate is decrease with increase in the user request.

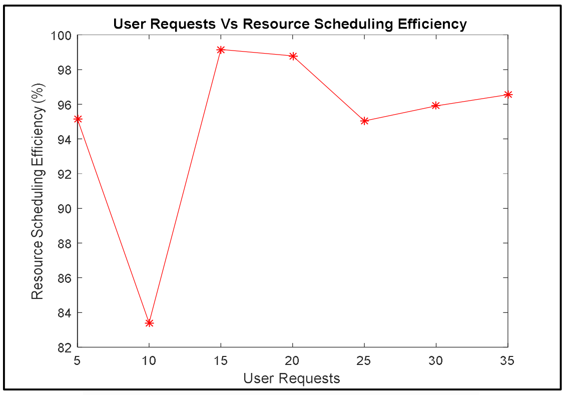


*Fig 11: Resource Scheduling Efficiency*

Above figure represents the Resource scheduling efficiency with respect to user request and we consider variation in user request from 5 to 35 users. From the observation, we find that the efficiency of resource scheduling is varies from 85 to 99 with increase in the user request.

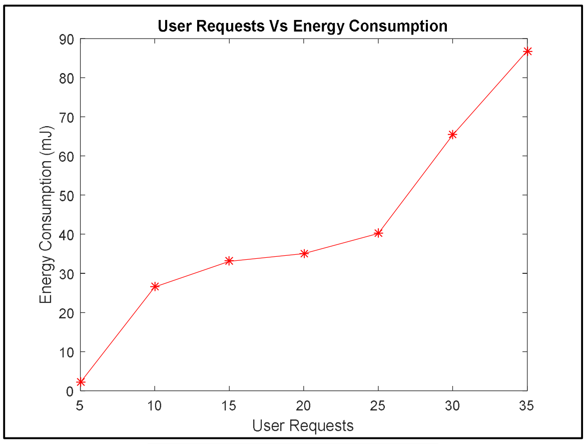


*Fig 12: Energy Consumption*

Above figure represents energy consumption with respect to user request and we consider variation in user request from 5 to 35 users. From the observation, we find that the consumption of energy is increasing as well as increasing in the user request.

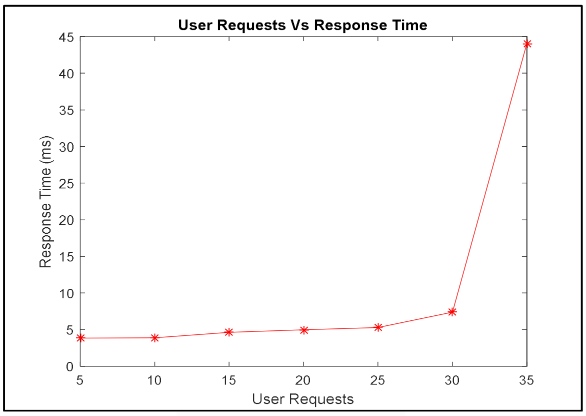


*Fig 13: Response Time*

Above figure represents Response time with respect to user request and we consider variation in user request from 5 to 35 users. From the observation, we find that the response time is increasing as well as increasing in the user request. After the experimental results, we compare the simulation with existing work by ***V. Priya et al.*** [10] are given as:

*Table I: Comparison of Parameters*

| Parameters | Existing | Proposed |
|---|---|---|
| **Average Success Rate** | 93.77 | 97.19 |
| **Resource Scheduling Efficiency (%)** | 92.83 | 96.48 |
| **Response Time (ms)** | 5.59 | 4.86 |

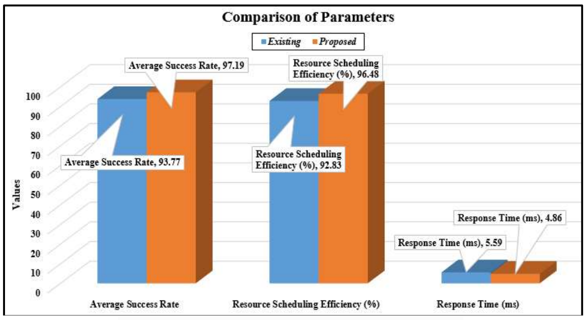


*Fig 14: Comparison of Parameters*

From the above Table I and Fig 14, we observed that the proposed work efficiency is better than the previous work by utilization of deep learning based ANN classifier

## 5. CONCLUSION AND FUTURE WORK

In this paper, an ANN as a deep learning based load balancing and resource scheduling using MRSQN with load balancing using MQLO algorithm for cloud computing environment is proposed. We studied the issue of resource scheduling and load balancing in cloud to maximize the cloud provider's services by reducing the response time of model with reduction in energy consumption and balancing the load on servers. The main challenges include the load balancing on the server is a difficult task nature with the management of new the user's task arrival. So, we introduced the concept of ANN to identify the over or under loaded servers and if possible, then transfer the load to its VMs by introducing migrant properties. The simulation results and their comparisons show that the proposed model is far better than the existing work in terms of average success rate, resource scheduling efficiency, energy consumption and response time, but consumption of energy is still high for cost effective model. So, in future of cloud, the idea of optimized ANN will be utilized as a classifier to prepare a better understandable model using swarm-based Meta heuristic algorithm.

## REFERENCES

[1]. J. Zhao, K. Yang, X. Wei, Y. Ding, L. Hu and G. Xu. 2016. A Heuristic Clustering-Based Task Deployment Approach for Load Balancing Using Bayes Theorem in Cloud Environment. IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 2, pp. 305--316.

[2]. GIBET TANI, H. and C. EL AMRANI. 2016. Smarter Round Robin Scheduling Algorithm for Cloud Computing and Big Data. Journal of Data Mining and Digital Humanities, Special Issue on Scientific and Technological Strategic Intelligence. Y. F. Wen and C. L. Chang. 2014. Load balancing job assignment for cluster-based cloud computing. Sixth International Conference on Ubiquitous and Future Networks (ICUFN), Shanghai, pp. 199--204.Google Scholar

[3]. Bui Thanh Khiet, Nguyen Thi Nguyet Que, Ho Dac Hung, Pham Tran Vu, Tran Cong Hung. 2017. A Fair VM Allocation for Cloud Computing based on Game Theory". Proceedings of the 10th National Conference on Fundamental and Applied Information Technology Research (FAIR'10), Da Nang, Viet Nam. J. Zhang, Q. Liu and J. Chen. 2016. An Advanced Load Balancing Strategy for Cloud Environment. 17th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT), Guangzhou, pp. 240--243.

[4]. Rahmani, Amir M., Tuan Nguyen Gia, BehailuNegash, Arman Anzanpour, ImanAzimi, Mingzhe Jiang, and PasiLiljeberg. "Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: A fog computing approach." Future Generation Computer Systems 78 (2018): 641-658.

[5]. Ebadifard, F., & Babamir, S. M. (2018). A PSO-based task scheduling algorithm improved using a load-balancing technique for the cloud computing environment. Concurrency and Computation: Practice and Experience, 30(12), e4368.

[6]. Saeed, F., Javaid, N., Zubair, M., Ismail, M., Zakria, M., Ashraf, M. H., & Kamal, M. B. (2018, September). Load Balancing on Cloud Analyst Using First Come First Serve Scheduling Algorithm. In International Conference on Intelligent Networking and Collaborative Systems (pp. 463-472). Springer, Cham.

[7]. Xavier, V. A., & Annadurai, S. (2019). Chaotic social spider algorithm for load balance aware task scheduling in cloud computing. Cluster Computing, 22(1), 287-297.

[8]. Ashouraei, M., Khezr, S. N., Benlamri, R., & Navimipour, N. J. (2018, August). A new SLA-aware load balancing method in the cloud using an improved parallel task scheduling algorithm. In 2018 IEEE 6th international conference on future internet of things and cloud (FiCloud) (pp. 71-76). IEEE.

[9]. Belgaum, M. R., Soomro, S., Alansari, Z., & Alam, M. (2018). Cloud service ranking using checkpoint-based load balancing in real-time scheduling of cloud computing. In Progress in Advanced Computing and Intelligent Engineering (pp. 667-676). Springer, Singapore.

[10]. Priya, V., Kumar, C. S., & Kannan, R. (2019). Resource scheduling algorithm with load balancing for cloud service provisioning. Applied Soft Computing, 76, 416-424. Chavva, Subba Reddy, and Ravi Sankar Sangam. "An energy-efficient multi-hop routing protocol for health monitoring in wireless body area networks." Network Modeling Analysis in Health Informatics and Bioinformatics 8, no. 1 (2019): 21.

[11]. Guo, S., Liu, J., Yang, Y., Xiao, B., & Li, Z. (2019). Energy-Efficient Dynamic Computation Offloading and Cooperative Task Scheduling in Mobile Cloud Computing. IEEE Transactions on Mobile Computing, 18(2), 319-333.

[12]. Mukherjee, A., Roy, D. G., & De, D. (2019). Mobility-aware task delegation model in mobile cloud computing. The Journal of Supercomputing, 75(1), 314-339.

[13]. Zhou, B., Dastjerdi, A. V., Calheiros, R. N., Srirama, S. N., & Buyya, R. (2017). mCloud: A context-aware offloading framework for heterogeneous mobile cloud. IEEE Transactions on Services Computing, 10(5), 797-810.

[14]. Hung, T. C., Hieu, L. N., Hy, P. T., & Phi, N. X. (2019, January). MMSIA: improved max-min scheduling algorithm for load balancing on cloud computing. In Proceedings of the 3rd International Conference on Machine Learning and Soft Computing (pp. 60-64).

[15]. Mishra, S. K., Sahoo, B., & Parida, P. P. (2020). Load balancing in cloud computing: a big picture. Journal of King Saud University-Computer and Information Sciences, 32(2), 149-158.