Sir Syed University Research Journal of Engineering & Technology
2023, Vol. 13, No. 1,
https://doi.org/10.33317/ssurj.500

Load Balancing of Cloud Computing Service Model Empowered with Fuzzy Logic

Abdul Hannan Khan[1*], Syed Atir Raza[1], Shahid Rasool[2], Aqsa Anwar[3], and Muhammad Ammar[4]

[1]School of Information Technology, Minhaj University, Lahore, Pakistan
[2]School of Computer Science, Minhaj University Lahore, Lahore, Pakistan
[3]School of Software Engineering, Minhaj University, Lahore, Pakistan
[4]School of Software Engineering, Superior University, Lahore, Pakistan

**Correspondence Author: Abdul Hannan Khan (hod.it@mul.edu.pk)*

Received May 31, 2022; Revised November 14, 2022; Accepted November 27, 2022

Abstract

In today's world, many advancements have been made in artificial intelligence and cloud computing. The computing and artificial intelligence networks have increased as both serve various domains of life. At the same time, multiple challenges and issues like computing performance and security have to be faced. This study aims to improve the service model and measure the cloud computing load with the help of fuzzy logic. Numerous studies have been conducted to learn about integrating cloud computing with artificial intelligence. Various designs, models, and algorithms have been studied and observed from past studies and literature. Cloud computing includes virtualization, internet service, software, and online services. The cloud provider must organize jobs to maximize benefits and meet customer service quality requirements. The cloud is overgrowing; therefore, service providers must handle enormous demands. Service availability and performance under workloads are the primary challenges. Processing many queries concurrently has increased effort. This study uses fuzzy logic and MATLAB to balance the cloud load. Fuzzy logic balances the cloud computing load using the Virtual Machine (VM) processor speed, storage capacity, and allocated load to improve processing time and storage usage.

Index Terms: *Cloud Computing, Fuzzy Logic System, Load Balancing, Service Model, Virtual Machine.*

I. INTRODUCTION

Cloud computing is the number-one computing platform that can provide various services to millions of users worldwide according to their demands [1]. Individuals demand accessibility to various computing services based on their needs without the exact information of where these services locate or how they provide to consumers [2]. These days, many small business sectors, organizations, and commercial enterprise programs manage over the cloud. The primary characteristic of the cloud is remote and on-demand, and the second one is its elasticity. Because of various features, many individuals and organizations moved their private information, toward the cloud [3].

II. CLOUD COMPUTING MODELS

Typical cloud computing is divided into two models:

- Development Model.
- Service Model.

These models are further divided into sub-categories; Development Model consists of, 1) Private Cloud, 2) Public Cloud, and 3) Hybrid Cloud. The Service Model consists of, 1) IAAS, 2) SAAS, and 3) PAAS.

A. Deployment Models

The development models are as follows:

a) Private Cloud:

A private cloud-primarily relies on virtualization, making it possible to decouple IT resources and services from actual physical equipment [2]. Since they do not execute locally on servers or end devices, applications are essentially available in the cloud. It is the perfect option for businesses that deal with stringent data processing and security needs. A more flexible alternative, the Private Cloud allows services in accordance with the requirements of the client [3].

b) Public Cloud:

Public cloud providers often handle administration, upkeep, and enhancements to a shared pool of computer resources used by several customers. High scalability and elasticity are two features that distinguish public cloud solutions. The public cloud model is executed and owned by a third party. All clients share a similar infrastructure path with minimum configuration, security measures, and availability variances administered and validated through the provider.

c) Hybrid Cloud:

The hybrid model, which combines the advantages of the public, private, and community cloud models, is the choice of 69% of companies that use cloud infrastructure. When working with large, complex datasets that include both

Creative Common CC BY: This article is distributed under the terms of the Creative Commons Attributes 4.0 License. It permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

highly confidential user information and more openly accessible data, a hybrid cloud architecture may be the best option [3].

d) Community Cloud:

Community cloud computing enables its customers to more easily define and assess their business requirements. Community cloud can be located either on-premises or in an off-premises data center owned by one of the tenants or by a third-party cloud services provider. But some infrastructure and computing sources are absolute to multiple companies with the same privacy, authorization, and regulatory measures relative to a single company. The community cloud model strives to join the distributed sources supplied by grid computing while more usage of a self-maintenance movement by autonomic computing [4].

B. Service Models

The service models are as follows:

a) Software-As-A-Service (SAAS):

In this model, the user doesn't need to install it on their pc; this model provides a facility for the clients to use and rent this application from the provider without installing it. This service model offers that service where actual application development and testing carry over the platforms supported through the PAAS layer [4]. Software as a service layer is involved chiefly with end-users. For security, the users of this model mostly rely on the provider. It has security challenges like web applications. In this model, the service provider takes data from a user simultaneously, immediately makes multiple copies of data, and distributes them among other cloud data centers worldwide. So the customer needs to find out where the data is actually. It provides flexibility to their users [5]. The customer can control the tools installed with the platform and also be able to create the new platform that they need to fit their environment [6].
Some advantages of SAAS are:

- Slow down the licensing cost.
- It can be handed over to the clients on one to many bases so that one application can be processed by many users simultaneously.
- Infrastructure doesn't need it [7] and [8].

b) Platform-As-A-Service (PAAS):

This provides an appropriate environment for developers to develop the software and application to set out using the internet rather than install and execute the development environment. Hardware and software hosted by the PAAS provider. It provides security and redundancy. It can integrate with other infrastructure parts such as LDAP, web services, and databases. In PAAS, the user cannot rapidly change anything. The applications host by the cloud; Users can need help making their applications secure. Rapid change may affect the whole software development life cycle. The developers need to upgrade their applications to keep their systems changing continuously. There is no permission for developers to access the core layer in this model. So cloud service provider is responsible for security [8]. The SAAS model; can be created with the help of development tools provided by the platform as a service. Developers need to be sure about the security of development tools. Where the developers have control over their applications [3].
Some advantages of PAAS are as follows:

- Increase flexibility in the development process.
- Decrease storage overhead.
- Security also provides backup and recovery.
- Cost-effective.
- Flexibility for the customer to control the development tools [8].

c) Infrastructure As A Service (IAAS):

This model of cloud computing provides virtual computing appliances, for the internet. A third party supports its clients, a supplier, and a host of equipment, software, servers, storage, and other network components within this model. The provider of this application maintains all tasks, including system maintenance, backups, and reliability planning [5]. The developers have more control over security, and there is no security break in the virtualization manager. With the help of virtualization, developers can use hardware appliances to make a new layer like storage devices, network resources, operating systems, or servers to execute their applications. One server can share more than one virtual appliance, such as input/output devices, CPU, memory, and various resources. The typical cloud computing models are described in figure I below.

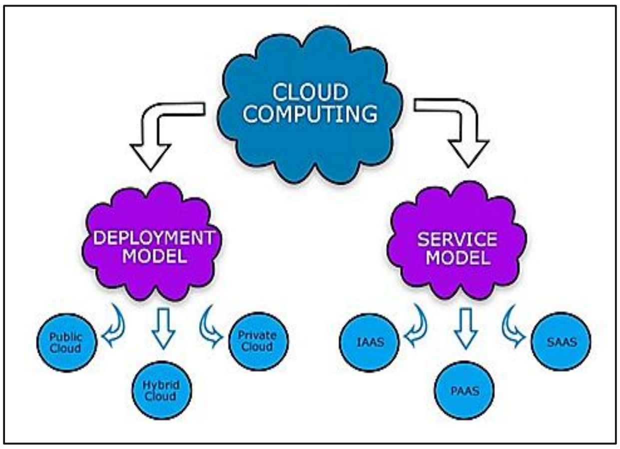


Figure I: Models of Cloud Computing

Some advantages of IAAS are as follows:

- The developers can increase or decrease infrastructure according to their needs.
- Users can execute the virtual machine.
- Therefore, reduce the cost of hardware and HR [8].

III. LOAD BALANCING

Load balancing (LB) is an approach to divide the coming traffic between available servers to handle the requests. As a result, the response is given at a faster rate [9]. Figure II illustrates a standard LB structure utilized inside a cloud environment in which LB balances the load by using common steps, which are the following:

- First, it receives incoming requests from different clients.
- Computes the load size of incoming client requests and constructs a request queue.
- From the server, the pool checks the current load status of servers.
- Different LB techniques make the appropriate selection of servers.

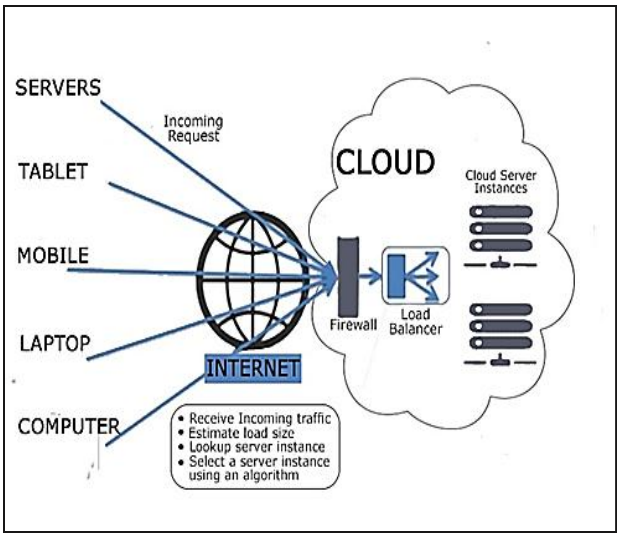


Figure II: Load Balancing in Cloud Computing

Various types of algorithms are present that enable load traffic among accessible servers. Websites may be associated with a simple example of LB in everyday life [9]. Users may face delays, overtime, and perhaps lengthy device responses without LB. Its solutions generally use reliable servers that allow communication traffic to be better distributed so that the website's availability is definitively settled. LB algorithms are available in various forms and categorized into two categories [10].

A. Static Algorithms

Static algorithms break the traffic among servers equally. The traffic up to the servers would be quickly contempt by this method and find the situation more problematic. It is known as the Round-robin algorithm, which splits the traffic uniformly. It includes previous information on the system. The weighted round-robin is described as enhancing the crucial Round-robin challenges. Servers got more connections according to the highest weight [10].

B. Dynamic Algorithms

Dynamic algorithms established appropriate weights onto servers, and the thinnest server tended to match the traffic by scanning the whole network. It does not require the previous information of the system. Instead, it requires the present state of the system. However, it was essential to pick a suitable server to connect with the networks in real-time, increasing the traffic on the system. Consumers can borrow software, computer hardware, infrastructure, and computational recourses in the cloud according to their needs. However, because of the cloud's great future, many severe problems must be reviewed for their most significant realization [11]. Load balancing is simply the most critical challenge in cloud computing. It spreads the dynamic workload across several nodes to ensure that no solitary resource will be overloaded. An effective load balancer must be adapted to modify the environment and jobs [10].

This paper will design newer load balancing criteria based on fuzzy logic in the Virtual Machine (VM) environment of cloud computing to achieve better processing time and storage utilization. We examine the problem related to the optimum setup of VM machines inside the hosted servers within various data centers to achieve optimized objectives such as processor speed, assigned load, and storage capacity.

The technical components of cloud computing service model assessment will be the main emphasis of this study. This article clarifies the cloud computing service model and analyses its evolution by comprehending the advantages of the internet's development. Many individuals have different opinions regarding the terminology ‘Cloud Computing,' and some even claim that it is a brand-new innovation, however, cloud computing has existed online ever since the Internet first appeared. With the help of several input and output factors, the workload of cloud computing will be assessed. This model will help the service provider check the load of that specific service and decide according to the user demand. This model will also be compared with the existing cloud approaches for improving cloud computing infrastructure.

IV. LITERATURE REVIEW

This research prescribed a customized PSO algorithm that LBMPSO prescribed to solve the problem of load balancing and task scheduling. As a result, the LBMPSO task scheduling approach is developed on the PSO algorithm that utilizes an excellent action to assess the ideal adjustment of every object [12]. The authors proposed and developed a dynamic load-balancing algorithm. This algorithm aims to decrease the makespan time and enhance resource utilization in cloud architecture [13]. In an article, a game-theoretical base method has been prescribed for balancing the load task performed best in the long run. Game theory is the mathematical approach that may utilize to make decisions among the firms selling different items [14].

A review-based method is presented based on a modern load-balancing algorithm that is especially suitable for cloud architecture. The author presented a cloud environment to explain the cloud model [15].

A study prescribed a load balancing algorithm, a ‘Vector Dot’. It manages the hierarchical complication of the data center and several dimensionalities of source lading over servers, network transformation, and data storage in alert data centers with embedded server and storage techniques. In Vector Dot algorithm, the dot product differentiates the nodes based on item needs and help erode loads on servers, storage segment, and switches [16].

A research work prescribed the latest algorithm of load balancing called Round-robin in the Virtual Machine (VM) cloud computing platform to get the best response and execution times [17].

An algorithm is designed known as a 'Genetic' algorithm metaheuristic to sort out the problem of choosing the most favorable sorting for jobs. So, that is the method to balance the workload on a network [18].

A new technique for load balancing in a cloud environment: priority queue on the different stages using the binary heap is described. Load balancing in a cloud computing environment manages by moving requests from one server to another using a priority queue [19].

The hybrid algorithm based on fuzzy logic and ant colony optimization to better the load balancing in the cloud computing architecture is described. The author prescribed the most optimum and acceptable version of the 'ACO' algorithm in a cloud environment [20]. It prescribed a technique along with Type-2 fuzzy logic to abide by the unchallenged and effective behavior in assessing the host to load balancing VM appropriation in the cloud architecture [21].

V. PROPOSED MODEL

This section proposed methodology described the flow structure of the offered 'Cloud-Based Load Balancing Performance Measurement Evaluation Fuzzy Expert System'. In this model, input parameters follow the fuzzy inference model, take the decision based on fuzzy rules, and generate the output through a fuzzifier described in figure III to evaluate the proposed approach. The system includes a single fuzzy-based layer, as shown in figure IV.

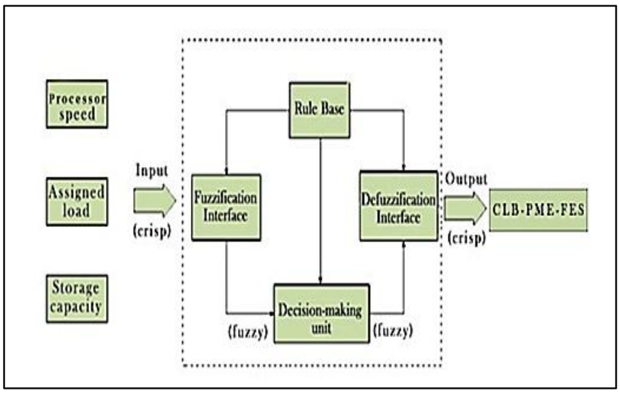


Figure III: Fuzzy-Based Proposed System Model

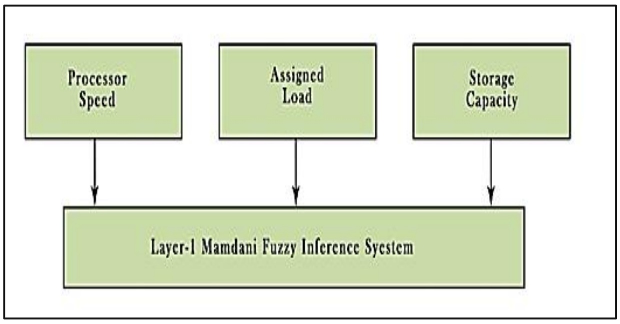


Figure IV: Layer Model of Mamdani Fuzzy Inference System

VI. MATERIAL AND METHOD

It examines the balanced load of the cloud (low/medium/high) by using three input variables; processor speed, assigned load, and storage capacity, as performed in the figure, i.e., figure V using MATLAB R2019a tool.

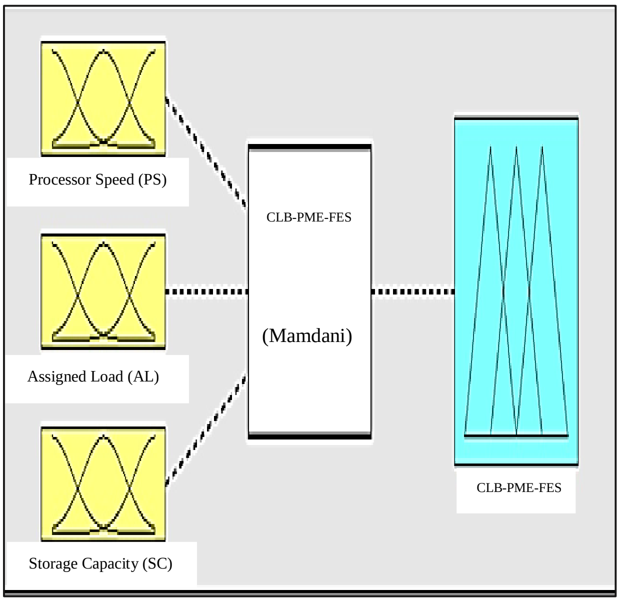


Figure V: Layer Model of Mamdani Fuzzy Inference System

Values of these input parameters are used to construct a lookup table, i.e., table I, for evaluating the balanced load. The proposed; cloud load balancing, and performance measurement evaluation, using Mamdani fuzzy inference-based expert system can be expressed mathematically regarding t-norm and can be written in mathematical form as:

$$\mu Cloud - Load - Balancing\ \ (Load\ Balancing) = t\ [\mu Processor - Speed\ (ps), \mu Assigned - Load\ (al), \mu Storage - capacity\ (sc)]$$

$$\mu\ Cloud - Load - Balancing\ \ (Load\ Balancing) = min\ \ [\mu Processor - Speed\ (ps), \mu Assigned - Load\ (al), \mu Storage - capacity\ (sc)$$

A. Input Variables

Numerical values given to the fuzzy model are known as input variables. In this research, three different input variables are used. The details of these inputs and their ranges are shown in table I.

Table I: Input Variables with Different Ranges of the Proposed System

S. No.	Input Parameters	Ranges	Semantic Sign
1	Processor Speed (PS)	LT < 1 GHz	Low
		B/W 0.8 – 2.5 GHz	Medium
		GT > 2.25 GHz	High
2	Assigned Load (AL)	LT <1.5	No Load
		B/W 1.0 – 4.5	Min Load
		GT > 4.0	Over Load
3	Storage Capacity (SC)	LT < 0.2	Free
		B/W 0.1 – 0.5	Within Capacity
		GT > 0.4	Out of Capacity

B. Output Variables

This research proposes a single-layer mechanism to balance a cloud load. The detailed output and ranges are shown in table II.

Table II: Output Variables of the Proposed System

S. No.	Layers	Output Variables	Range	Semantic Sign
1	Layer-I	Balanced Load	0 – 1.5	Low
			1 – 3.5	Medium
			3 – 5	High

C. Equations

Mathematical fuzzy inference system input variables are shown in table III.

D. Membership Functions

Membership functions of the proposed cloud LB-PME System give the curve values excluding truth values and dispense a mathematical figure of fuzzy logic that offer numerical values of both input and output variables. The membership functions of the system are given in table III and table IV.

Table III: Mathematical Fuzzy Inference System Input Variables

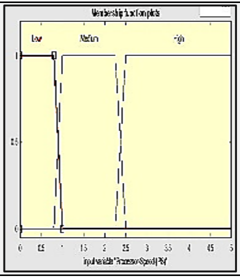
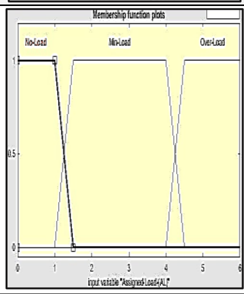
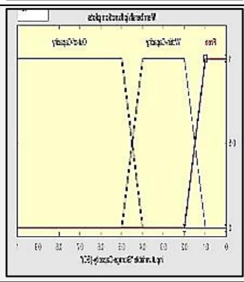
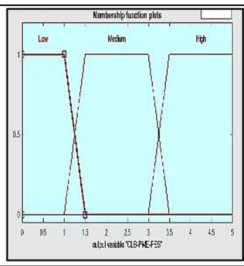
Input	Membership Function	Self MF SS
Processor Speed = ps $\boldsymbol{\mu_{PROCESSOR\ SPEED}(ps)}$	$\mu_{PS,LOW}(ps)=\begin{cases}1 & ,if\ ps\in[0,0.8]\\ \frac{1-ps}{0.2} & ,if\ ps\in[0.8,1]\\ 0 & ,otherwise\end{cases}$ $\mu_{PS,MEDIUM}(ps)=\begin{cases}\frac{ps-0.8}{0.2} & ,if\ ps\in[0.8,1]\\ 1 & ,if\ ps\in[1,2.25]\\ \frac{2.5-ps}{0.2} & ,if\ ps\in[2.25,2.5]\\ 0 & ,otherwise\end{cases}$ $\mu_{PS,HIGH}(ps)=\begin{cases}\frac{ps-2.25}{0.25} & ,if\ ps\in[2.25,2.5]\\ 1 & ,if\ ps\in[2.5,5]\\ 0 & ,otherwise\end{cases}$	
Assigned Load = al $\boldsymbol{\mu_{\mathrm{ASSIGNED\ LOAD}}(al)}$	$\mu_{AL,NO\ LOAD}(al)=\begin{cases}1 & ,if\ al\in[0,1]\\ \frac{1.5-al}{0.5} & ,if\ al\in[1,1.5]\\ 0 & ,otherwise\end{cases}$ $\mu_{AL,MIN\ LOAD}(al)=\begin{cases}\frac{al-1.5}{0.5} & ,if\ al\in[1,1.5]\\ 1 & ,if\ al\in[1.5,4]\\ \frac{4.5-al}{0.5} & ,if\ al\in[4,4.5]\\ 0 & ,otherwise\end{cases}$ $\mu_{AL,OVER\ LOAD}(al)=\begin{cases}\frac{al-4}{0.5} & ,if\ al\in[4,4.5]\\ 1 & ,if\ al\in[4.5,6]\\ 0 & ,otherwise\end{cases}$	
Storage capacity =sc $\boldsymbol{\mu_{STORAGE\ CAPACITY}(sc)}$	$\mu_{SC,FREE}(sc)=\begin{cases}1 & ,if\ sc\in[0,0.1]\\ \frac{0.2-sc}{0.1} & ,if\ sc\in[0.1,0.2]\\ 0 & ,otherwise\end{cases}$ $\mu_{SC,WITHIN\ CAPACITY}(sc)=\begin{cases}\frac{sc-0.1}{0.1} & ,if\ sc\in[0.1,0.2]\\ 1 & ,if\ sc\in[0.2,0.4]\\ \frac{0.5-sc}{0.1} & ,if\ sc\in[0.4,0.5]\\ 0 & ,otherwise\end{cases}$ $\mu_{SC,OUT\ OF\ CAPACITY}(sc)=\begin{cases}\frac{sc-0.4}{0.1} & ,if\ sc\in[0.4,0.5]\\ 1 & ,if\ sc\in[0.5,1]\\ 0 & ,otherwise\end{cases}$	

Table IV: Mathematical and Graphical MF of Fuzzy Inference System Output Variables

Output Variables	Membership Functions	Sample MF SS
Balanced Load=bl $\boldsymbol{\mu_{BL}(bl)}$	$\mu_{BL,LOW}(\mathrm{bl})$ $=\left\{max\left(min\left(1,\frac{1.5-lb}{0.5}\right),0\right)\right\}$ $\mu_{BL,MEDIUM}(\mathrm{bl})$ $=\left\{max\left(min\left(\frac{lb-1}{0.5},1,\frac{3.5-bl}{0.5}\right),0\right)\right\}$ $\mu_{BL,HIGH}(\mathrm{bl})$ $=\left\{max\left(min\left(\frac{bl-3}{0.5},1\right),0\right)\right\}$	

E. Deployment Models

IF-THEN conditions are applied onset of multiple rules that the experts provide for governing the decision-making system based on given information.

The rules of the proposed system is shown in table below, i.e., table V.

Table V: Rules-Based Lookup Table of the Proposed System

Rules	Processing Speed (PS)	Assigned Load (AL)	Storage Capacity (SC)	Results
1	Low	No-load	Free	**Low**
2	Low	No-load	Within Capacity	**Low**
3	Low	No-load	Out of Capacity	**Low**
4	Low	Min load	Free	**Low**
5	Low	Overload	Free	**Low**
6	Low	Min load	Out of Capacity	**Low**
7	Medium	No-load	Free	**Low**
8	Medium	No-load	Within Capacity	**Low**
9	High	No-load	Free	**Low**
10	Low	Min load	Within Capacity	**Medium**
11	Low	Overload	Within Capacity	**Medium**
12	Medium	Min load	Within Capacity	**Medium**
13	Medium	Overload	Free	**Medium**
14	Medium	No-load	Out Of Capacity	**Medium**
15	Medium	Min load	Free	**Medium**
16	High	No-load	Within Capacity	**Medium**
17	High	Min load	Free	**Medium**
18	High	Min load	Within Capacity	**Medium**
19	Low	Overload	Out Of Capacity	**High**
20	Medium	Overload	Out Of Capacity	**High**
21	Medium	Overload	Within Capacity	**High**
22	Medium	Min load	Out of Capacity	**High**
23	High	Overload	Out of Capacity	**High**
24	High	Min load	Out of Capacity	**High**
25	High	Overload	Within Capacity	**High**
26	High	Overload	Free	**High**
27	High	No-load	Out of Capacity	**High**

An Expert system can be constructed based on input and output rules. All rules for the proposed system appear in figure VI.

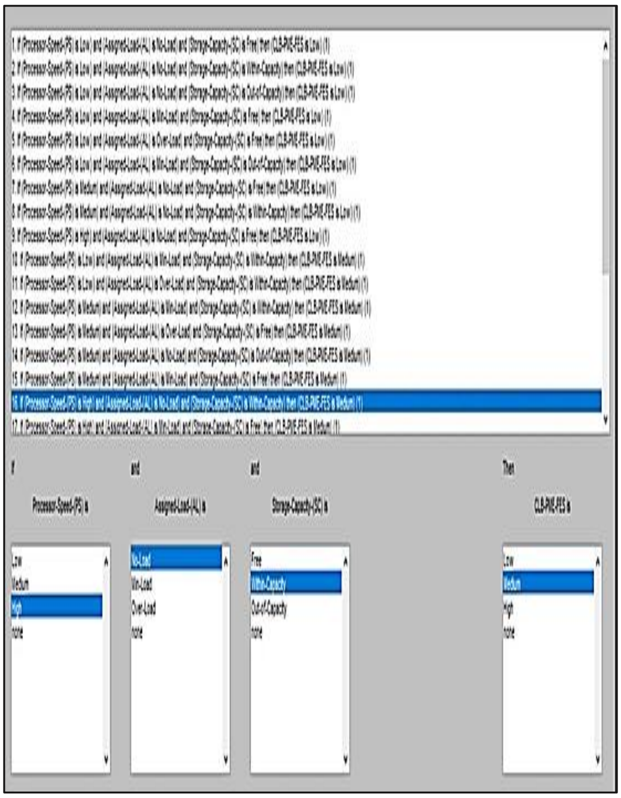


Figure VI: Rules-Based Values of the Proposed System

F. *Interference Engine*

An inference engine is used for decision-making. The inference engine is an essential system element that can apply logical rules to the knowledge base to derive new information. It holds reasoning through which the expert system gets a solution by matching the information in the database with the rules provided by thy rule base.

G. *De-Fuzzifier*

De-fuzzifier is another essential factor of the proposed model that is generally composed of different elements. The proposed model explained the transformation of the fuzzy inference engine's fuzzy output into breakable by utilizing similar membership functionalities compared to fuzzifier output. De-Fuzzifier graphical illustrations of the proposed ‘CLB-MPE-FES’ system is shown in figure VII.

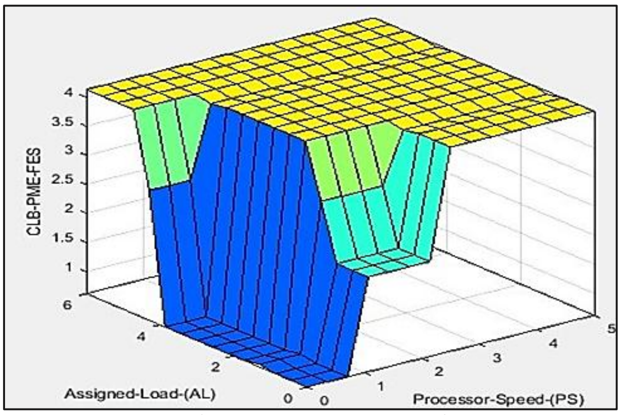


Figure VII: De-Fuzzifier Graphical Illustrations of the Proposed CLB-MPE-FES System

VII. RESULTS AND SIMULATIONS

MATLAB R2019a is used for many fields; it's an efficient tool for designing software, making calculations, simulations, algorithm development, data analysis, and various fields. For the simulation of results, three inputs and one output of cloud load balance are used, shown in figure VIII.

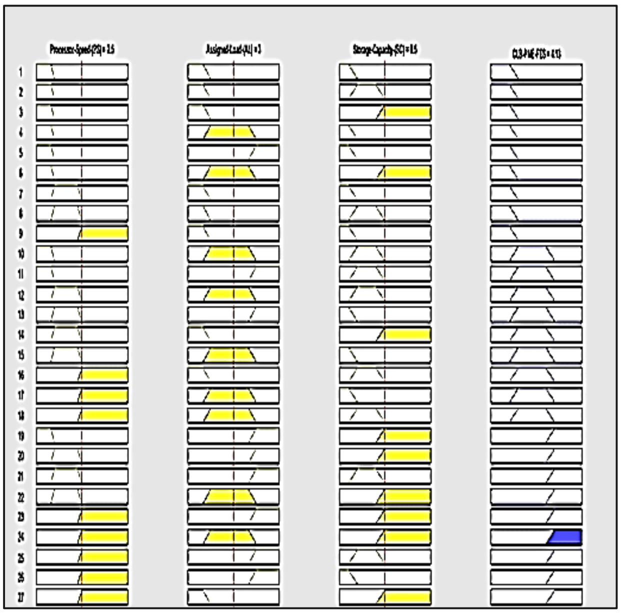


Figure VIII: Lookup Diagram of Proposed CLB-MPE-FES Expert System

VIII. CONCLUSION

Nowadays, cloud computing is overgrowing as an experimental paradigm; many people move towards the cloud and store their information on the cloud because they need quality service. The biggest challenge is the availability of services and maintaining the performance equivalent or more effective whenever workload occurs. In this situation, load-balancing occurs as one of the most critical issues. A load balancer is used to share the dynamic load equally among all the servers. Various methods have been proposed to solve the load-balancing issue in cloud computing. In this article, we have presented a model that balanced the loudness of cloud computing using a fuzzy inference engine. Within the proposed study, the fuzzifier executes the fuzzification procedure that changes three kinds of input, including processor speed, storage capacity, and the assigned load of VM, and generates single output such as a balanced load.

The limitation of this work only targets three input parameters. These constrained parameters reduce system performance time and computational expenses. Initially, these parameter facilities handle data fast, but as the cloud service scale increases, it will be realized that considerably more parameters will be needed to evaluate cloud load. In future work, this work can be extended by adding more input and output parameters and evaluating the infrastructure, platform, and software service of cloud computing through a service model. This model can be improved through the neuro-fuzzy approach. This future work will help the user with quick and smooth cloud service.

Acknowledgment

The authors would like to thank the management of Minhaj University, Lahore, Pakistan, for their support and their assistance throughout this study.

Authors Contributions

The contribution of the authors was as follows: Abdul Hannan Khan's contribution to this study was the concept, technical implementation, and correspondence. The methodology to conduct this research work was proposed by Syed Atir Raza. Data collection and supervision were performed by Shahid Rasool. Aqsa Anwar facilitated the data compilation and validation. Muhammad Ammar's contribution was project administration, and paper writing.

Conflict of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Data Availability Statement

The testing data is available in this paper.

Funding

This research received no external funding.

References

[1] Alam, T. (2020). Cloud Computing and its role in the Information Technology. *IAIC Transactions on Sustainable Digital Innovation (ITSDI), 1*(2), 108-115.

[2] Diaby, T., & Rad, B. B. (2017). Cloud computing: a review of the concepts and deployment models. *International Journal of Information Technology and Computer Science, 9*(6), 50-58.

[3] Wang, J., Yang, Y., Wang, T., Sherratt, R. S., & Zhang, J. (2020). Big data service architecture: a survey. *Journal of Internet Technology, 21*(2), 393-405.

[4] Saadat, A., & Masehian, E. (2019, December). Load balancing in cloud computing using genetic algorithm and fuzzy logic. In *2019 International Conference on Computational Science and Computational Intelligence (CSCI)* (pp. 1435-1440).

[5] Deepa, T., & Cheelu, D. (2017, August). A comparative study of static and dynamic load balancing algorithms in cloud computing. In *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)* (pp. 3375-3378).

[6] Dašić, P., Dašić, J., & Crvenković, B. (2016). Service models for cloud computing: Search as a service (SaaS*). International Journal of Engineering and Technology (IJET), 8*(5), 2366-2373.

[7] Rashid, A., & Chaturvedi, A. (2019). Cloud computing characteristics and services: a brief review. *International Journal of Computer Sciences and Engineering, 7*(2), 421-426.

[8] Yasrab, R. (2018). Platform-as-a-Service (PaaS): The Next Hype of Cloud Computing. *arXiv preprint arXiv:1804.10811*.

[9] Xu, P., He, G., Li, Z., & Zhang, Z. (2018). An efficient load balancing algorithm for virtual machine allocation based on ant colony optimization. *International Journal of Distributed Sensor Networks, 14*(12).

[10] Tong, R., & Zhu, X. (2010, November). A load balancing strategy based on the combination of static and dynamic. In *2010 2nd International Workshop on Database Technology and Applications* (pp. 1-4).

[11] Khiyaita, A., El Bakkali, H., Zbakh, M., & El Kettani, D. (2012). Load balancing cloud computing: state of art. *2012 National Days of Network Security and Systems*, 106-109.

[12] Pradhan, A., & Bisoy, S. K. (2020). A novel load balancing technique for cloud computing platform based on PSO. *Journal of King Saud University-Computer and Information Sciences.*

[13] Kumar, M., & Sharma, S. C. (2020). Dynamic load balancing algorithm to minimize the makespan time and utilize the resources effectively in cloud environment. *International Journal of Computers and Applications, 42*(1), 108-117.

[14] Swathy, R., Vinayagasundaram, B., Rajesh, G., Nayyar, A., Abouhawwash, M., & Abu Elsoud, M. (2020). Game theoretical approach for load balancing using SGMLB model in cloud environment. *PloS one, 15*(4).

[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.

[16] Singh, A., Korupolu, M., & Mohapatra, D. (2008, November). Server-storage virtualization: integration and load balancing in data centers. In *SC'08: Proceedings of the 2008 ACM/IEEE conference on Supercomputing* (pp. 1-12). IEEE

[17] Issawi, S. F., Al Halees, A., & Radi, M. (2015). An efficient adaptive load balancing algorithm for cloud computing under Bursty workloads. *Engineering, Technology & Applied Science Research, 5*(3), 795-800.

[18] Panwar, P., Kaushal, C., Singla, A., & Rattan, V. (2022). Load Balancing in Multiprocessor Systems Using Modified Real-Coded Genetic Algorithm. In *Innovations in Computational Intelligence and Computer Vision* (pp. 201-210). Springer, Singapore.

[19] Rathore, R., SHARMA, V., & GOLA, K. (2014). A new approach for load balancing in cloud computing. *International Journal of Engineering and Computer Science, 2*(5), 1636-1640.

[20] Ragmani, A., Elomri, A., Abghour, N., Moussaid, K., & Rida, M. (2019). An improved hybrid fuzzy-ant colony algorithm applied to load balancing in cloud computing environment. *Procedia Computer Science, 151*, 519-526.

[21] Moura, B., Schneider, G., Yamin, A., Pilla, M., & Reiser, R. (2019, August). Type-2 fuzzy logic approach for overloaded hosts in consolidation of virtual machines in cloud computing. In *11th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT 2019)* (pp. 668-675).