International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

A FRAMEWORK FOR FUZZY Q-LEARNING BASED QOS PARAMETER-DRIVEN SCHEDULING OPTIMIZATION FOR INTERNET OF THINGS APPLICATIONS IN CLOUD ENVIRONMENTS

***Smitha G Prabhu[1], Poongodai. A [2], Dattatray G Takale [3], Naresh Kumar [4], Yashaswini. B. M [5], Geetha D Devanagavi [6], Ankita Gaur [7], Thangamani. M [8]**

[1]Professor, Department of Electronics and Communication Engineering, Nitte Meenakshi Institute of Technology (NMIT), Nitte (Deemed to be University), Bengaluru, Karnataka, India, smitha.prabhu@nmit.ac.in

[2]Assistant Professor, Department of Computer Science and Engineering (AI), Madanapalle Institute of Technology & Science, Madanapalle, Andhra Pradesh, India, a.poongodai@gmail.com

[3]Assistant Professor, Department of Computer Engineering, Vishwakarma Institute of Information Technology, Pune, Maharashtra, India, dattatraygtakale@gmail.com

[4]Assistant Professor, Department of Electrical Engineering, Deen Bandhu Chhotu Ram University of Science and Technology Murthal, Sonipat, Haryana, India, nareshkumar.ee@dcrustm.org

[5]Assistant Professor, Department of Electronics and Communication Engineering, JSS Academy of Technical Education, Bengaluru, Karnataka, India, yashaswinibm96@gmail.com

[6] Professor, School of Computing and Information Technology, Reva University, Bengaluru, Karnataka, India, dgeetha@reva.edu.in

[7]Assistant Professor, Department of Sciences, Manav Rachna University, Faridabad, Haryana, India, ankitagaur50@yahoo.com

[8] Professor, Department of Computer Science and Engineering, Muthayammal Engineering College, Rasipuram, Namakkal, Tamilnadu, India, manithangamani2@gmail.com

Abstract

Effective scheduling in cloud environments is becoming more and more necessary as Connectivity of Things (IoT) applications expand quickly in order to manage the huge data streams produced by IoT devices in real time.Furthermore, adaptive resource allocation solutions are necessary to maintain optimal performance while minimizing energy consumption due to the dynamic nature of IoT workloads.For cloud-based IoT applications, this research suggests a novel framework for fuzzy Q-Learning-based QoS parameter-driven scheduling optimization.The system dynamically evaluates and ranks activities according to several QoS factors by combining fuzzy logic and Q-learning. Fuzzy logic manages task-specific uncertainty,

Received: August 14, 2025

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

while Q-learning facilitates adaptive decision-making via ongoing learning. The simulation was run using the CloudSim toolkit. The outcomes show that it outperforms the baseline approaches in terms of makespan, timing of response, degree of imbalance (DOI), and resource utilization.
Keywords: CloudSim simulation, Fuzzy Q-learning,QoS based Driven scheduling optimization, IoT, Cloud environment

Introduction

Cloud computing is rapidly gaining traction as a technology delivery and service business model due to its capacity to reduce labor, structures and maintenance costs. Additionally, Internet of Things (IoT) applications are made to make it easier for users and developers to access networks of intelligent data, devices, and services. Despite providing almost limitless resources, cloud services have a limited reach.Integrating the cloud with IoT enables the design of cohesive and well-structured applications. Given their Quality of Service (QoS) characteristics, it is NP-hard to determine which accessible atomic services are suitable for building the required composite service in a coordination model with their cooperation[1].Numerous intelligent IoT devices that produce a lot of tasks are found in the IoT layer; these devices are all different in terms of size, computing demand, connection requirements, and latency limits[2].Metrics such as Mean TAT, price, Energies, and the Extremely Critical Task TAT ratio are used to evaluate the effectiveness of the priority-based FCFS scheduling algorithm[3]. To build the proposed model, the CloudSim simulation was executed on the Eclipse platform as indicated [4]. In Internet of Things (IoT) contexts, traffic has increased and load imbalance issues among network entities have arisen due to increasing and diverse service demands. Quality of Service (QoS) metrics may be impacted[5].

Related Work

Thomas Bartzanas et al. [6]have proposed,a creative study of Internet of Things (IoT) devices for optimal agricultural environments. Descriptive and statistical methods were used to analyze the connections between emerging technologies, precision agriculture, farming 4.0, the global web, and advancements in commercial farming.Christos L. Stergiou et al. [7]have suggested, a design that aims to create an ecologically friendly, cloud-user-independent industry IoT-based big data handling system (EEIBDM) by combining the functionalities obtainable from numerous cloud providers.

M.Mohankumar et al. [8] have recommend, a novel method is used to track the position of a motor vehicle, weight measurement, fuel level, tire pressure, engine temperature, controlling speed, and tire pressure. Kamlesh Kumar [9]have described,aexamine cyber security in light of malware, DDOS, and B-IDS attacks.Souvik Palet al. [10] have introduced,a novel IoT-based System Content Optimization Model (IoT-SCOM), an optimization technique for integrating IoT service elements in edge-cloud-hybrid systems, aims to lower transmission latency.

Abdullah Ganiet al. [11]have recommend,acomprehensive analysis to look at the architecture, performance metrics, contributions, applications, state of the art, and assessment metrics in the SC, WSN, and IoT research fields.Jiayin Feng et al. [12]have reported,auses cloud computing, an improved extreme learning machine, and other techniques to thoroughly investigate IoT detection, web node monitoring, and invasion response in cloud computing environments. Ashish Kaushal et al.[13] have proposed,a enhance Optimized-Greedy Nominator Histogram (EO-GNH), a methodology for maximizing Quality of Service (QoS) by strategically placing machine learning (ML) and AI applications in edge environments.

Shan Li et al. [14] have reported, a collection of novel approaches and concepts for studying 5G/6G enabled IoT and emerging IoT technology trends. Ahmed Afif Monratet al. [15] have introduced,a three well-known IoT platforms—ThingsBoard, Eclipse Ditto, and Microsoft Azure IoT—are compared, and their compliance with the main IIoT standards is evaluated.Suzhi Bi et al. [16] have proposed, a using conventional numerical optimization approaches to address such issues is wasteful, especially when the problem size is large.

Wang Ping et al. [17]have described,aemergency conscious distributed adaptive scheduling reduces transmission delay by selecting parents using fuzzy Bayesian learning and channels using multi-objective gravitational search techniques.Ledan Qian et al. [18] have proposed,a increase the accuracy of face recognition, a face capture algorithm based on Internet of Things services is used.

This Work's Contribution

The main elements of this work are explained as follows:

1. A new task scheduling framework that combines Q-learning and fuzzy logic to optimize scheduling choices according to QoS parameters, guaranteeing effective resource use for cloud-based IoT applications.
2. The suggested approach calculates the cumulative trust value, which is the most recent or ongoing cloud transaction for the provider and is dynamically updated at each transaction.
3. Comprehensive simulations utilizing the CloudSim toolbox to confirm the framework's effectiveness.
4. The results demonstrate that it works better than traditional scheduling methods in terms of degree of imbalance (DOI), makespan, reaction time, and resource utilization.

Here is a summary of the rest of this work: The introduction and pertinent literature were presented in Sections 1 and 2, the suggested approach was elucidated in further detail in Section 3, the simulation results were presented in Section 4, and conclusions and future research directions were discussed in Section 5.

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Proposed System

The method for assessing cloud service dependability presented in this study uses fuzzy Q-learning and a scheduling optimization model that is influenced by QoS parameters. The suggested design establishes the total trust value, which is dynamically updated with every transaction. It represents the provider's most recent or ongoing cloud transaction.The CloudSim platform with Eclipse was used to simulate the proposed model, and the resulting QoS settings maximize the degree of imbalance (DOI), makespan, reaction time, and resource consumption. The suggested block schematic is displayed in Figure 1.

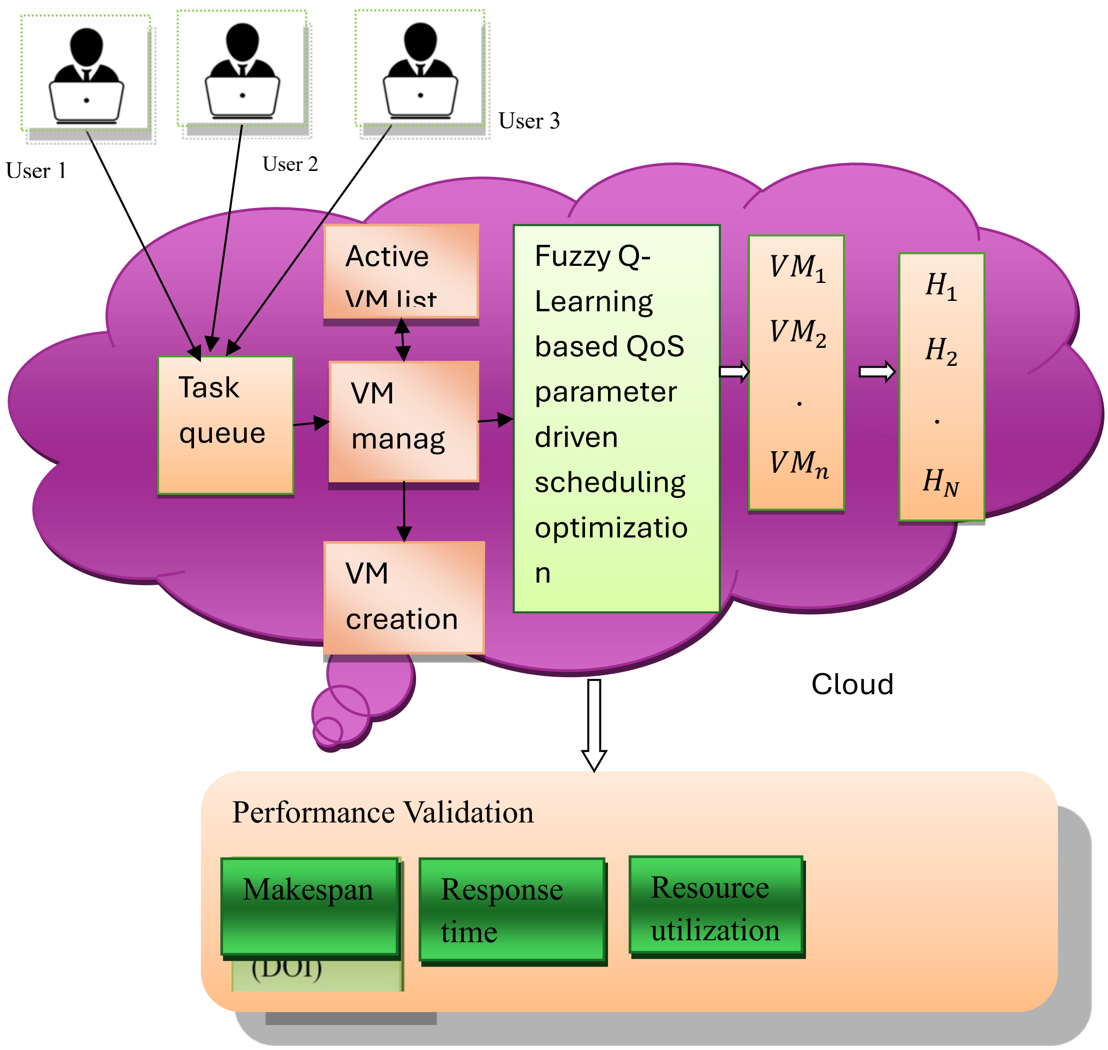

Fig 1: The suggested algorithm for scheduling optimization

3.1 Fuzzy Q-Learning

Received: August 14, 2025

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Fuzzy Q-Learning uses fuzzy rules to create decisions by combining fuzzy logic and standard Q-Learning to deal with uncertainty and imprecise situations. Triangular membership functions are used to divide each input space into fuzzy sets. Architecture for Fuzzy Q-Learning is shown in Figure 2.

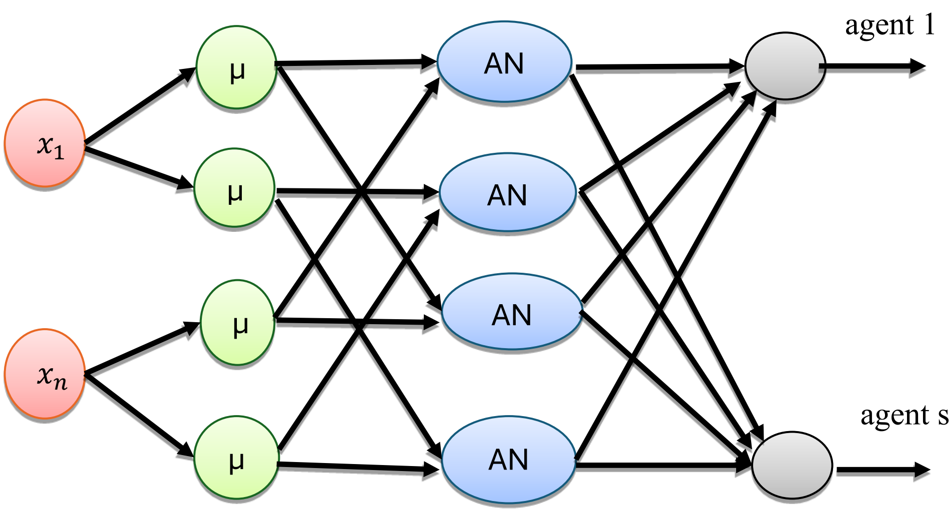

Figure 2: Architecture for Fuzzy Q-Learning

The fuzzy rules are defined as follows: Rule number r, $r = 1$ to N, of agent number s, $s = 1$ to S, calculates the control action for every cell, or hypercube while acting on the entire state space, much like a fuzzy rule base.

$rule\ r: if\ x_1\ is\ A_1^{r(1)}\ and\ x_2\ is\ A_2^{r(2)} \ldots .x_n\ is\ A_n^{r(n)} then\ B\ is\ b_r$(s)   (1)

Every agent should define: 1) A Q-function,$Q[c, s]$,where s stands for the agent and c for the cell.2) A rule quality,$q[k, s]$for agent number S use of rule number k.The connection between Q and q-functions can be found using

$Q[c, s] = \frac{1}{2^n} \sum_{k \in H(c)} q[k, s]$ (2)

The cell activity$\alpha(c_r)$indicates each fuzzy rule's level of significance given the current input conditions. Higher membership values $(\mu k(c_r))$increase cell activity, guaranteeing that the most pertinent rules have an impact on judgment.The activity level $\alpha(c_r)$ of a cell is defined as:

$\alpha(c_r) = \frac{1}{M} \sum_{K=1}^{M} \frac{\mu k(c_r)}{\in_j \mu j(c_s)}$ (3)

$\Delta q[k, s_0] = 2^n act[k] \Delta q$ (4)

The quality is demonstrated by Equation 4,q[k,S_0] is updated.

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$\mathrm{Q}[\mathrm{c}, s_0]_{t=t_1} = \mathrm{Q}[\mathrm{c}, s_0]_{t=t_0} + \Delta\mathrm{Q} = \frac{1}{2^n}\sum_{k\in H(c)} q[k, s_0]_{t=t_1} \qquad (5)$$

Based on the cell activity and reinforcement received, this is the update's central calculation. It illustrates how new encounters enhance the agent's comprehension.

3.2 Proposed Fuzzy Q-Learning algorithm based QOS parameter Driven Scheduling Optimization

Based on the constraint of interval scheduling efficiency, a multi-objective driven scheduling optimization equation is proposed to achieve energy-efficient equipment in the Internet of Things (IoT) environment. In an IoT setting, the first is the proposed fuzzy Q-Learning based QoS driven scheduling optimization, which optimizes scheduling time and minimizes time consumption.The fuzzy approach is then used to reduce multiple target time loss problems to single-objective problems. The foundation was laid using schedule optimization models. Assuming that Ti is the sum of the time spent on all device scheduling, the overall amount of time spent on device scheduling can be explained as follows:

$$T_{total} = \sum_{i=1}^{n} T_i \qquad (6)$$

The previously mentioned method may be used to determine the total resource usage of each device schedule in an Internet of Things (IoT) environment. The overall resource need can be stated as follows when individual usage is combined:

$$R_{total} = \sum_{i=1}^{n} R_i \qquad (7)$$

where Ri represents the resource consumption of the i-th device, and n is the total number of scheduled devices.The total of all equipment scheduling operating times is shown as follows in the response time optimization scheduling modeling method:

$$T = \sum_i (d_{i,end} - d_{i,start}) \qquad (8)$$

Received: August 14, 2025

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

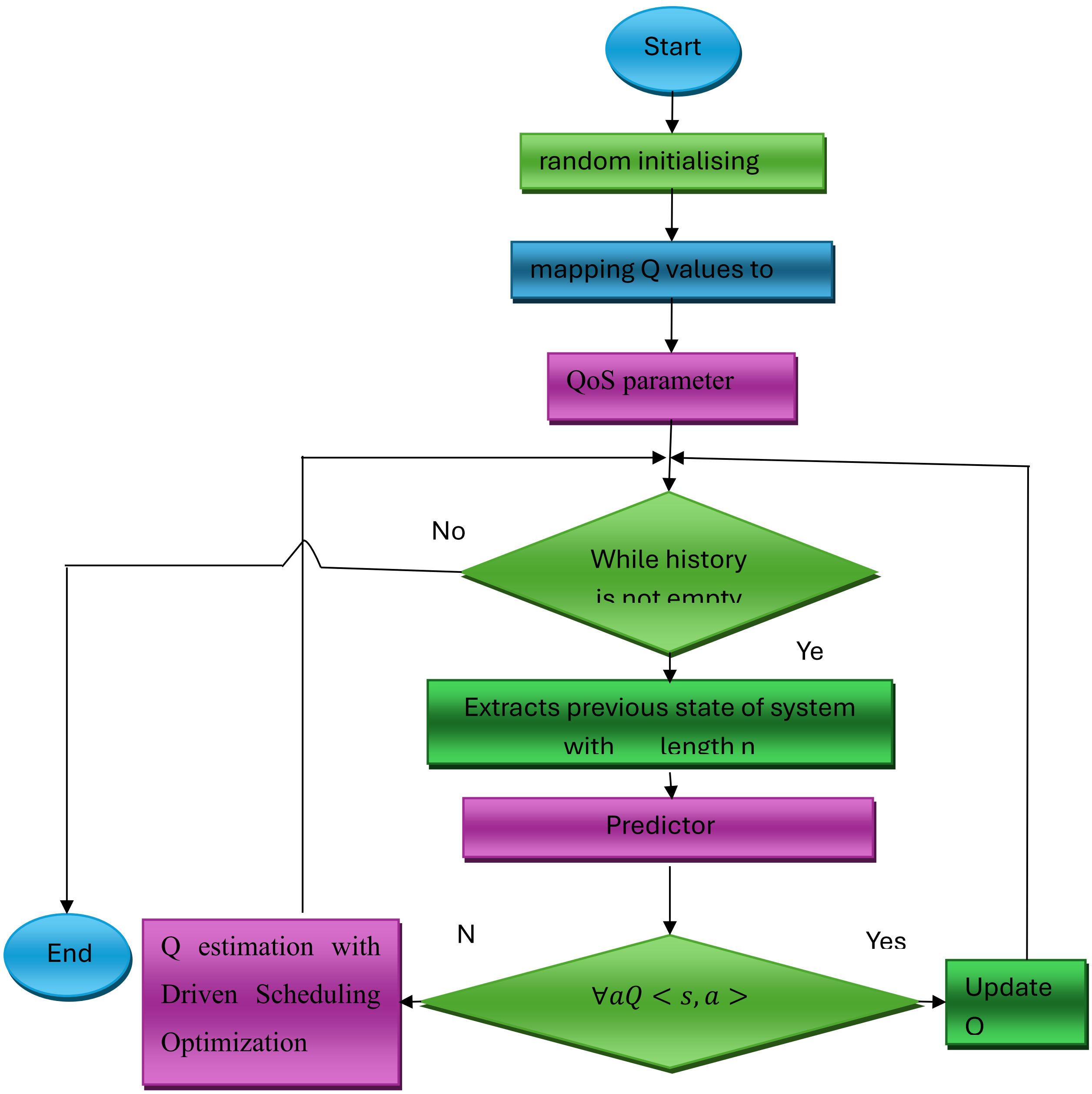

Figure3: Diagram for the Suggested Algorithm

Figure 3 displays the flowchart for the proposed algorithm. The combined certificate planning problem from the previous scheduling optimization equation (9) is represented by the following computation..

$$minF(x) = [G_1(x), G_2(x)]^T \qquad (9)$$

Received: August 14, 2025

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Converting multiple target resource problems into single target problems requires the use of the fuzzy algorithm. As demonstrated by the subsequent formula (10),

$$max\mu + \varepsilon\big(M_1(x) + M_2(x)\big)/2 \quad (10)$$

where $M_1(x)$and $M_2(x)$ represent the objectives under fuzzy constraints.

Thus, the multi-objective in the context of the Internet of Things (IoT) was established under the driven scheduling efficiency constraint, which was based on the resource consumption for scheduling and the equipment scheduling time. The fuzzy approach reduces multiple target energy loss problems to single-objective problems using the driven scheduling optimization equation, which serves as the foundation for the use of QoS-driven scheduling optimization models that take energy loss in an IoT setting into account.

3.3 CloudSim simulation

To model and simulate the cloud environment, the CloudSim toolkit's classes have been expanded. The virtualized environment created by this simulator facilitates on-demand provisioning. Cloud services and their applications can be modeled, simulated, and experimented with with the aid of this simulator. The CloudSim architecture layers offer support for modeling and simulating cloud environments.The CLOUDS Lab team created it, and Java is used throughout. In order to test a hypothesis before creating software and to replicate tests and outcomes, it is used to model and simulate a cloud computing system. QoS has taken into account the following performance metrics: degree of imbalance (DOI), makespan, reaction time, and resource consumption.

3.3.1 Makespan: Refers to a task's maximum time to be completed among the M_s .$T_{CT_{IJ}}$is the accomplishment of a task Ti by VM_j, as shows in Equation 11,

$$Makespan = max\left\{T_{CT_{IJ}} \quad \backslash i = 1,2, \ldots n; j = 1,2, \ldots\ldots m\right\} \quad (11)$$

Where, $T_{CT_{IJ}}$shows how long the virtual machine takes VM_j to complete the task T_i.

3.3.2Response Time: The amount of time needed to reply to a user's request. If the response time is small and expressed in milliseconds, the suggested approach is considered effective. Equation 12, determines the response time, where n is the total number of incoming user requests.

$$RT = n \times T_{CT_{IJ}} \quad (12)$$

3.3.3 ResourceUtilization: The degree of virtual machine usage and load balancing's objective—maximizing resource consumption while reducing makespan—have a reverse linear relationship. With mis being the total number of VMs, Equation 13 is used to get the average utilization of all virtual machines.

$$Average\ Utilization_{VM} = \frac{\sum_{i=1}^{n} T_{CT_{IJ}}}{makespan \times m} \quad (13)$$

3.3.4Degree of Imbalance (DOI): Task imbalances between virtual machines are measured by the degree of VM imbalance. $Tmax$ and $Tmin$, which stand for the maximum and minimum completion times of tasks Ti over all simulated machines, respectively, are used to measure it in Equations 14 and 15. Ti, the mean value of all VM jobs is also called tavg. The ith virtual machine's (VM) processing elements are denoted by PEnumi, the number of instructions per second by PEMIPSi, and the length of all instructions by L.

$$DOI = \frac{T_{\max -} T_{min}}{T_{avg}} (14)$$

Equation 15 is used to define the time required to process each job on a virtual machine (PTj).

$$p_{Tj} = \frac{L(VM_{i,t})}{Average\ system\ load} (15)$$

Result and discussion

Over time, the scheduling relationship between devices becomes more complex. Thus, the space-time data of task scheduling in the Internet of Things (IoT) environment is used to examine real-time device scheduling and associated activities. A thorough description of the scheduling time series and the relationship that produces a particular scheduling time series are developed through this in order to accomplish dynamic modeling of device scheduling in the Internet of Things (IoT) environment.

4.1 **Environment setup**

GoCJ_Dataset_100.txt, one of the text files with 100 tasks, is regarded as a cloudlet in MI due to its length. Google released the GoCJ: Google cloud jobs dataset for clouds and distributed processing architectures in September 2018. It consists of 19 text files with varying numbers of jobs per million of instructions (MI) that are stored in the Mendeley is data repository. In the experimental arrangement, each of the four hosts had a specific number of CPU, RAM, storage, throughput (BW), and processor speed cores; they could all create virtual machine (VM) instances and share resources; The technical details of each host are listed in Table 1, and the technical specifics of the 36 virtual machines (VMs) that operate on each host with distinct features are listed in Table 2.For each virtual machine's execution, Consideration is given to the quantity of cloudlets (20–100) with different instruction sizes (1000–5000).

Table 1: Technical details of the host

Host ID	Cores for Processing	MIPS, speed	RAM, GB	Storage, GB	MIPS, BW

Received: August 14, 2025

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

2	2	3501	51	1023	102300
3	3	4001	101	1023	102300
4	4	4501	151	1023	102300
5	5	5501	201	1023	102300

Table 2: The technical features of VM

CPU	The quantity of cores	MIPS, speed	RAM, GB	Storage, GB	MIPS, BW
Extreme Edition Core i5	1	1000	2	20	1023

Several algorithms, such as (1) Round Robin (RR), (2) FCFS, and (3) SJF, have been taken into consideration for comparison in order to assess the effectiveness of the suggested approach. The findings were acquired by executing each method ten times and recording the mean values for each performance metric. When compared to the other current techniques mentioned above, the suggested method performs better in terms of equitably distributing the load among nodes, as presents in Figure 4.

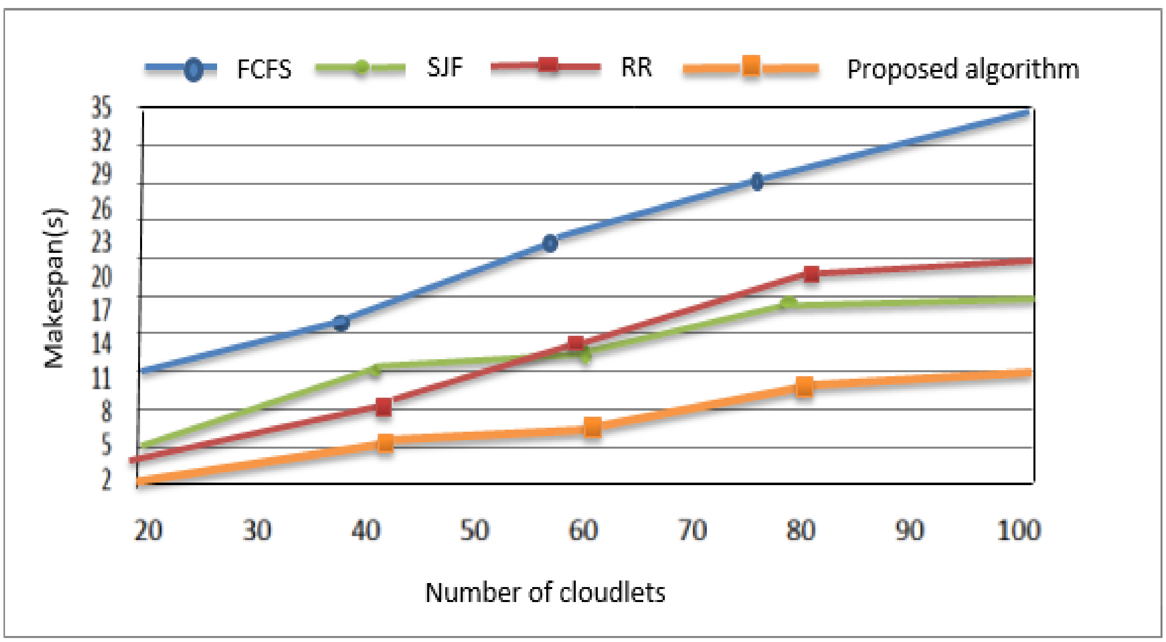

Figure 4: Makespan Compare of different existing algorithms with Fuzzy Q-Learning based QOS parameter Driven Scheduling Optimization

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

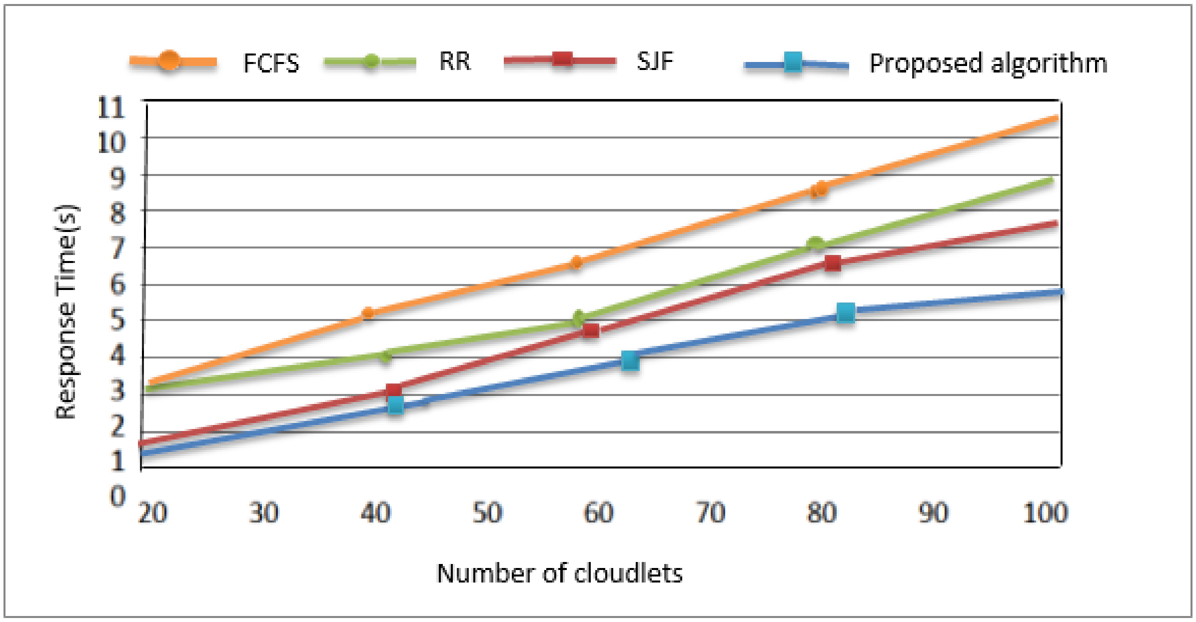

Figure 5: Evaluation of several Response Time methods

Figure 5 shows the reaction time values for many cloudlets using the suggested methodology for the various approaches.The results of the simulation show that the suggested approach performs better than the others and that the effectiveness of other compared techniques on reaction time increases with the number of cloudlets.

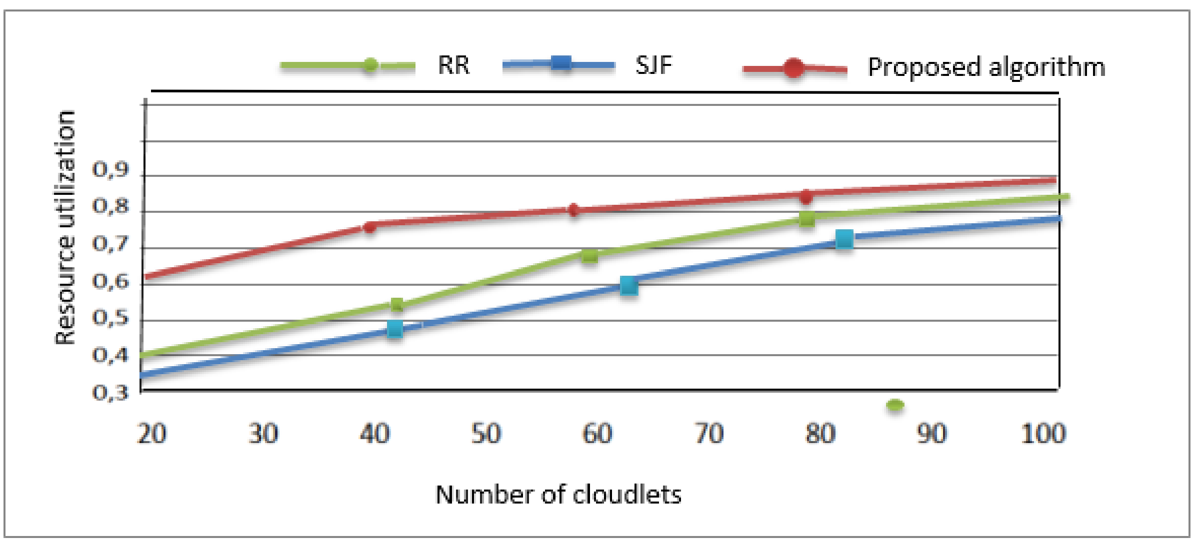

Figure 6: Resource Utilization Comparison

Figure 6 displays the resource usage comparison between the suggested method and the current algorithms. By assigning the loads onto the appropriate virtual machines, the suggested technique is able to make efficient use of the resources, which improves performance.

Received: August 14, 2025

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

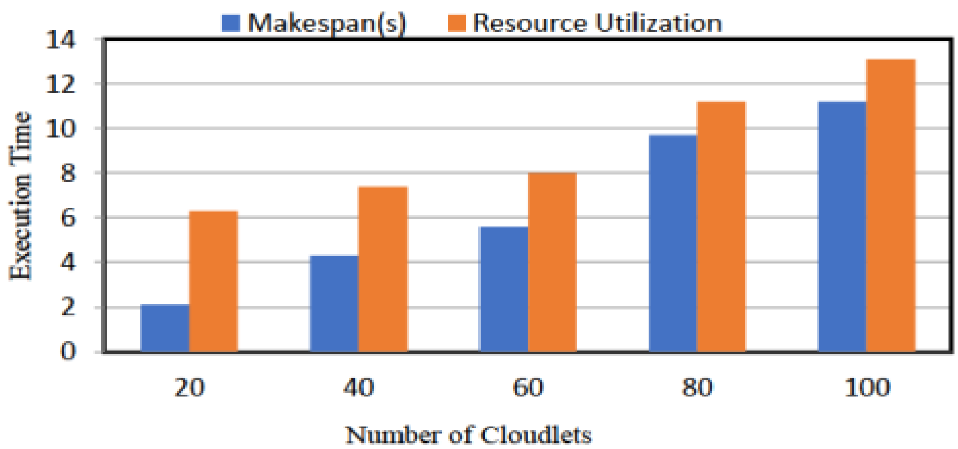

Figure 7: Makespan and Resource Utilization Comparison

The proposed approach maximizes resource consumption while minimizing makespan, as shown by the reverse connection between makespan and resource use in Figure 7.The recommended approach can maximize resource use while drastically lowering makespan..

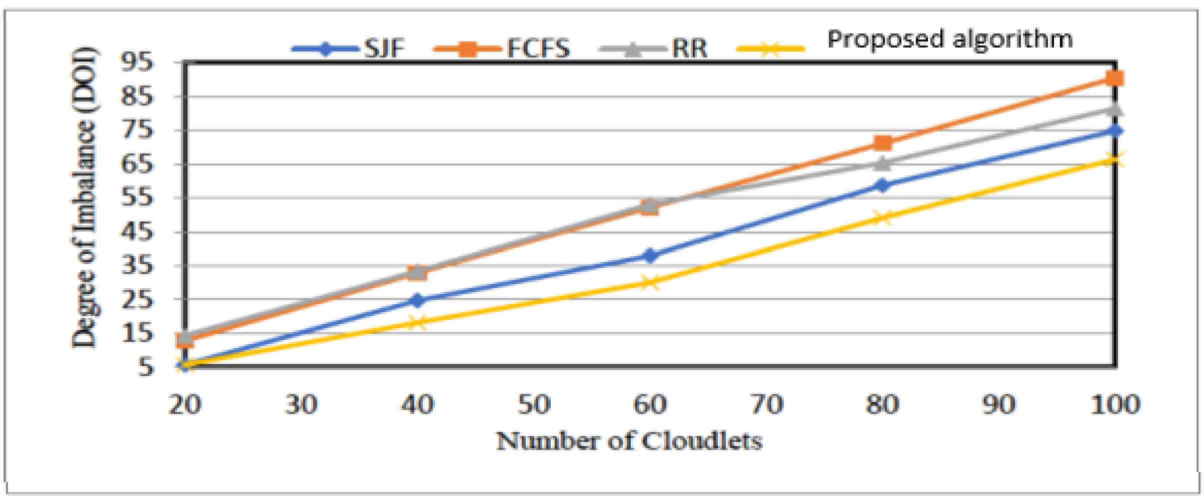

Figure 8: Three additional methods in addition to the suggested methodology were used to acquire the Degree of Imbalance (DOI) values.

The comparison of the Fuzzy Q-Learning based QOS parameter Driven Scheduling Optimization method with the DOI and other algorithms is shown in Figure 8. In comparison to other algorithms, the current algorithm has a higher DOI.

Table 3 displays the present systems' relative performance ratings.

Table 3: A comparison of proposed algorithm

Reference	Utilized Technique	Performance Metric	Evaluation tools

Liang Huang et al. [19]	DQN	Energy consumption, Delay	Simulation (TensorFlow)
Nikougoftaret al. [20]	LSTMDE+ Fuzzy Qlearning	CPU usage, execution time, and delay violation	Simulation (iFogSim)
Proposed	Fuzzy Q-Learning	Makespan, reaction time, use of resources, The extent of the imbalance	Simulation (CloudSim)

Conclusion

In this paper, we suggested a QoS parameter-driven scheduling optimization framework based on fuzzy Q-learning that is specifically designed for cloud-based IoT applications. Fuzzy logic and Q-learning are used in the suggested method to successfully handle the unpredictable and dynamic nature of IoT workloads. Q-learning guarantees adaptive decision-making through continuous learning, while fuzzy logic allows for the intelligent handling of uncertainty in task features.In cloud environments handling massive IoT data streams, the results show how effective it is at optimizing key performance metrics like makespan, degree of imbalance (DOI), response time, and resource usage. These metrics help to improve scheduling efficiency, allocate resources more effectively, and use less energy. Future research can investigate additional optimizations, such as hybrid AI techniques and edge-cloud collaboration, to enhance scalability and efficiency.

Reference

1. Vakili, Asrin, Hamza Mohammed Ridha Al-Khafaji, Mehdi Darbandi, Arash Heidari, Nima Jafari Navimipour, and Mehmet Unal. "A new service composition method in the cloud-based internet of things environment using a grey wolf optimization algorithm and MapReduce framework." Concurrency and Computation: Practice and Experience 36, no. 16 (2024): e8091.
2. Abdulazeez, Dezheen H., and Shavan K. Askar. "A novel offloading mechanism leveraging fuzzy logic and deep reinforcement learning to improve IoT application performance in a three-layer architecture within the fog-cloud environment." IEEE Access (2024).

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

3. Roy, Vandana. "A Context-Aware Internet of Things (IoT) founded Approach to Scheming an Operative Priority-Based Scheduling Algorithms." Journal of Cybersecurity & Information Management 13, no. 1 (2024).
4. Hassan, Hala, Ali Ibrahim El-Desouky, Abdelhameed Ibrahim, El-Sayed M. El-Kenawy, and Reham Arnous. "Enhanced QoS-based model for trust assessment in cloud computing environment." *IEEE Access* 8 (2020): 43752-43763.
5. Rostami, Mohammad, and Salman Goli-Bidgoli. "An overview of QoS-aware load balancing techniques in SDN-based IoT networks." Journal of cloud computing 13, no. 1 (2024): 89.
6. Maraveas, Chrysanthos, and Thomas Bartzanas. "Application of Internet of Things (IoT) for optimized greenhouse environments." AgriEngineering 3, no. 4 (2021): 954-970.
7. Stergiou, Christos, and Konstantinos Psannis. "Digital twin intelligent system for industrial internet of things-based big data management and analysis in cloud environments." Virtual Reality & Intelligent Hardware (2022).
8. Thamaraimanalan, T., M. Mohankumar, S. Dhanasekaran, and H. Anandakumar. "Experimental analysis of intelligent vehicle monitoring system using Internet of Things (IoT)." EAI Endorsed Trans. Energy Web 8, no. 36 (2021): e10.
9. Waqas, Muhammad, Kamlesh Kumar, Asif Ali Laghari, Umair Saeed, Muhammad Malook Rind, Aftab Ahmed Shaikh, Fahad Hussain, Athaul Rai, and Abdul Qayoom Qazi. "Botnet attack detection in Internet of Things devices over cloud environment via machine learning." Concurrency and Computation: Practice and Experience 34, no. 4 (2022): e6662.
10. Pal, Souvik, N. Z. Jhanjhi, Azmi Shawkat Abdulbaqi, D. Akila, Abdulaleem Ali Almazroi, and Faisal S. Alsubaei. "A hybrid edge-cloud system for networking service components optimization using the internet of things." Electronics 12, no. 3 (2023): 649.
11. Ali, Ihsan, Ismail Ahmedy, Abdullah Gani, Muhammad Umair Munir, and Mohammad Hossein Anisi. "Data collection in studies on Internet of things (IoT), wireless sensor networks (WSNs), and sensor cloud (SC): Similarities and differences." IEEE Access 10 (2022): 33909-33931.
12. Lin, Haifeng, Qilin Xue, Jiayin Feng, and Di Bai. "Internet of things intrusion detection model and algorithm based on cloud computing and multi-feature extraction extreme learning machine." Digital Communications and Networks 9, no. 1 (2023): 111-124.
13. Almurshed, Osama, Ashish Kaushal, SouhamMeshoul, Asmail Muftah, Osama Almoghamis, Ioan Petri, Nitin Auluck, and Omer Rana. "Enhancing performance of machine learning tasks on edge-cloud infrastructures: A cross-domain internet of things based framework." Future Generation Computer Systems 166 (2025): 107696.
14. Li, Shan, Muddesar Iqbal, and Neetesh Saxena. "Future industry internet of things with zero-trust security." Information Systems Frontiers 26, no. 5 (2024): 1653-1666.

15. Jamil, Mohammad Newaj, Olov Schelén, Ahmed Afif Monrat, and Karl Andersson. "Enabling Industrial Internet of Things by Leveraging Distributed Edge-to-Cloud Computing: Challenges and Opportunities." IEEE Access (2024).
16. Yan, Jia, Suzhi Bi, and Ying Jun Angela Zhang. "Offloading and resource allocation with general task graph in mobile edge computing: A deep reinforcement learning approach." IEEE Transactions on Wireless Communications 19, no. 8 (2020): 5404-5419.
17. Rafiq, Ahsan, Wang Ping, Wei Min, and Mohammad Saleh Ali Muthanna. "Fog assisted 6TiSCH tri-layer network architecture for adaptive scheduling and energy-efficient offloading using rank-based Q-learning in smart industries." *IEEE Sensors Journal* 21, no. 22 (2021): 25489-25507.
18. Zhou, Xiao, Ledan Qian, Haider Aziz, and Marvin White. "A model study of teaching method reform of computer laboratory course integrating internet of things technology." Plos one 19, no. 4 (2024): e0298534.
19. Huang, Liang, Xu Feng, Luxin Zhang, Liping Qian, and Yuan Wu. "Multi-server multi-user multi-task computation offloading for mobile edge computing networks." Sensors 19, no. 6 (2019): 1446.
20. Nikougoftar, Elaheh, and Mostafa Ghobaei-Arani. "A Fuzzy Q-learning-based Approach for Auto-scaling and Computation Offloading in Edge/Cloud Computing." (2023).