Optimization of Fog Computing Efficiency by Decreasing the Latency Level in the Medical Environment

Mohammed Abdullah Al-Hagery [1], Ameenah Mohammad Alharbi [2]

[1] *Department of Computer Science, College of Computer, Qassim University, Saudi Arabia*
hajry@qu.edu.sa

[2] *Department of Computer Science, College of Computer, Qassim University, Saudi Arabia*
ameenah.a@tvtc.gov.sa

Abstract: Fog computing is the latest technique in today's cloud computing environment, with the advent of this technology. One of the most important parameters affecting fog computing performance is latency. Cloud computing was very effective but needed help in its efficiency according to the values of this parameter, mainly due to its distributed nature. With the advent of the "fog" technology, the big cloud was separated into many small cloudlets, where each cloudlet is a mobility-enhanced small scale. A new architectural element extends today's cloud computing infrastructure. The term "Cloudlets" combined to form fog Computing, which has the main advantages of improving its characteristics listed above.

This paper aims to improve the efficiency of the cloud-computing environment by developing a simulation system to test the best structure before applying it in a real environment. This system will improve the efficiency based on the latency parameter to provide researchers and interested parties with a clear vision to improve the work in a simulation environment and achieve the same methodology in real environments. The simulation system was developed in such a way as to reduce the delay of tasks to increase the performance rate. The results of the simulation system are promising and can be applied in the medical environment.

Keywords: Fog Computing Performance, Performance Measures, Cloud Computing, Latency.

1. INTRODUCTION

Cloud computing benefits both customers and service providers, but it also has certain drawbacks that must be considered. Some drawbacks include the need for internet access and latency, lack of location awareness, and centralized computing systems that use much bandwidth [1]. Cloud computing was further hampered by the billions of queries from Internet of Things (IoT) devices [2]. A new computing paradigm at the intermediary between end devices and cloud computing is suggested to tackle this cloud computing problem.

Next-generation cloud computing is based on the notion of fog computing compared to cloud computing because fog computing has a more robust design. Because fog computing is more dispersed than cloud computing, it receives a higher grade than cloud computing [3]. Fog's processing power is spread out and smaller in number compared to the complicated current centralized cloud computing. Cloud computing can handle many requests, but the fog is far better at managing the little requests from actuators or end devices than cloud computing [4]. Fog computing is not an alternative to cloud computing but rather an extension designed for emergency response, reducing the time it takes to move data to and from the cloud and other latency-sensitive applications. Fog computing refers to a distributed computing architecture in which some application services are managed locally on the network in smart devices while others are managed in the cloud [5]. Fog also gives proximity to its end users. Additionally, it enhances the quality of the service, making it possible to do analytics in real-time and localization. Cloud computing has been adopted as an effective method to process data because of its high computing power and storage capabilities [6]. However, since the cloud computing paradigm is a concentrated computing architecture, most calculations occur in the cloud [7]. Almost all data and inquiries ought to be delivered to a centralized cloud. Even though the data processor has grown significantly, the bandwidth has remained relatively high. So, the network bandwidth is becoming the bottleneck of cloud computing for such a massive amount of data [8]. Different issues still exist in the environment of fog computing.

However, it offers a more significant advantage over the cloud because the whole cloud architecture is geographically separated and connected via a network called nodes or hops. The concept of nodes and clustered approach reduces energy consumption and provides more scaling in terms of energy and efficiency. With the emergence of the fog computing idea, the current complex cloud computing is partitioned into distinct nodes to segregate the paths of the request being handled independently by using each node's processing capacity without heavy reliance on the complicated cloud computing [9].

The rest of the paper is organized as follows: Section 2 introduces cloud computing and fog computing with IoT in healthcare, including basic principles and a comparison of fog computing and cloud and IoT. Section 3 discusses the related works. The research methodology is presented in section 4, which also explains the simulation of the proposed method, while section 5 demonstrates the results and discussion. Section 6 presents the conclusions and future works.

2. CLOUD, FOG COMPUTING AND IOT

There needs to be more consensus in standardizing fog computing, cloudlets, edge computing, and other terms that have been used to describe fog. Different research groups have proposed many different definitions of fog. Since there is a research gap in definitions and standards for fog computing, this paper uses the definitions of Jafari [4] and Atlam et al. [5], in which fog computing is defined as a type of distributed computing architecture that manages a set of application services locally in the network and smart devices. In contrast, others are managed in the cloud [6]. This section highlights some paradigms proposed to bring the cloud closer to the end devices. In addition, it explains the advantages and disadvantages of introducing fog computing as an ideal platform for IoT. IoT has revolutionized the ever-evolving field of information and communication technology. Today's innovative technologies, such as tablets, computers, and smartphones, have changed how machines, sensors, and vehicles are used in various applications [7]. Table 1 compares fog Computing, Cloud Computing, and IoT using various factors [8]. Currently, the new IoT devices are more fixated on using smaller and smaller processors while increasing memory and speed as microprocessor technology continues to grow.

Table 1. Summary of Fog Computing, Cloud Computing, and IoT

Features	Cloud Computing	Fog Computing	IoT
Target User	General Internet user	Mobile users	Stationary mobile
Working environment	Indoors with massive space and ventilation	Outdoors or indoors	Outdoors Indoors
Latency	High	Low	Low
Delay jitter	High	Very low	Low
Service Type	Localized information service is limited to the specific location.	Global information collected worldwide	Information specific to the end device
Distance from Client to Server	Multiple hops	One hop	One hop
Security	Undefined	Can be defined	Can be defined
Attack on Data Enroute	High probability	Very low probability	Low probability
Location awareness	No	Yes	Yes
Geo Distribution	Centralized	Distributed	Dense and Distributed
Number of the Server Node	Few	Very large	Very large
Support for Mobility	Limited	Supported	Supported
Real-time Interaction	Supported	Supported	Supported
Type of Last-mile Connectivity	Leased line / wired	Wireless	
Big Data and Duration of Storage	Months and years as it manages big data	Short duration as it transmits big data	Transient as it is the source of big data
Major Service Provider	Amazon, Microsoft, IBM	Cisco IOx	ARM, Atmel, Bosch

3. RELATED WORKS

Many research works have been introduced to improve the performance of fog computing architecture, resource management, and task scheduling algorithm concepts. Abdulgalil [10] proposed a multitier fog computing architecture based on the analysis of ECG and EEG signals. The proposed architecture compares three tiers. The first tier generates data and transmits it to the second tier. The second tier of the proposed architecture exploits the benefits of emerging fog computing technology for lightweight analysis based on ECG signals. The third tier is responsible for extensive analysis based on EEG signals and data storage in the cloud. Santos [11] introduced a new concept of collaboration architecture for fog computing. His proposed architecture constitutes a referential model for better design and implementation of fog platforms. It powers the freedom of abstraction to make development and deployment at the fog nodes easier and more efficient. Moreover, it introduces expressive mechanisms to define and abstract objects, data analytics, and services.

Yousefpour et al. [12] introduced an IoT-fog-cloud framework for reducing the service delay for IoT applications. The analytical model was also developed to evaluate their policy of the proposed framework, which shows how it helps reduce IoT service delay [13]. Hao et al. [14] proposed a prototype system in the software architecture of fog computing to incorporate the different designs of communication IoT devices with fog nodes. The authors reported their design of WM-FOG, a computing framework prototype for fog environments that embraces this prototype of software architecture and shows the evaluation of their prototype system [15]. Varghese et al. [16] proposed a hybrid fog cloud architecture for low-response time applications, such as firefighting, to provide the solution for IoT service. It supported a growing variety of applications, fifth-generation (5G) wireless systems, including those in the IoT and embedded artificial intelligence.

Gao [17] investigated a task about an offloading strategy for minimizing energy in mobile devices. He also developed a priority-based task-scheduling algorithm with an edge server. The energy consumption, execution time, and execution cost against both the task data sizes and latency requirements were adopted as the performance metric. The performance evaluation results from his research showed that his proposed algorithm reduces task completion time and improves QoS. Mohan et al. [18] also proposed distributing task processing on Edge-Fog-Cloud, which participates in the network. They developed the Least Processing Cost First method to assign the processing tasks to nodes, providing optimal processing time. Peixoto et al. [19] introduced the scheduling algorithms in the hierarchical layer of fog and cloud computing with a three-layer architecture. Their work showed the strategies of scheduling that can be designed in a fog computing environment. In their research, the scheduler prioritizes cloudlet use and optimizes other objectives like reducing network use and cloud costs. Liu et al. [20] proposed a task scheduling algorithm based on fog computing devices' classification data mining technique. They contributed a novel classification mining algorithm and task scheduling model based on the Apriori algorithm. The task with the minimum completion time was selected to be executed at the fog node with the minimum completion time.

Wang et al. [21-23] proposed the task scheduling strategy in the fog computing scenario. A task scheduling strategy based on a hybrid heuristic algorithm was proposed that mainly solves the problem of terminal devices with limited computing resources and high energy consumption and makes the scheme feasible for real-time and efficient processing tasks of terminal devices. Nguyen et al. [9] introduced a new approach to optimizing task scheduling problems for Bagof-tasks applications in a cloud-fog environment regarding execution time and operating costs. Their algorithm could flexibly satisfy users' requirements for high-performance processing or cost efficiency. Their work could have been improved, as they tested the algorithm only on small datasets.

4. THE METHOD

The description of the methodology can be summarized in the following steps and then presented in Figure 1:

  i. Identifying the basic system requirements.

  ii. Initializing both the cloud layer, the fog layer, and the Initializing the edge layer.

  iii. Gathering sensor data from the edge layer and encrypting it.

  iv. Applying an optimization algorithm.

  v. Verifying node availability.
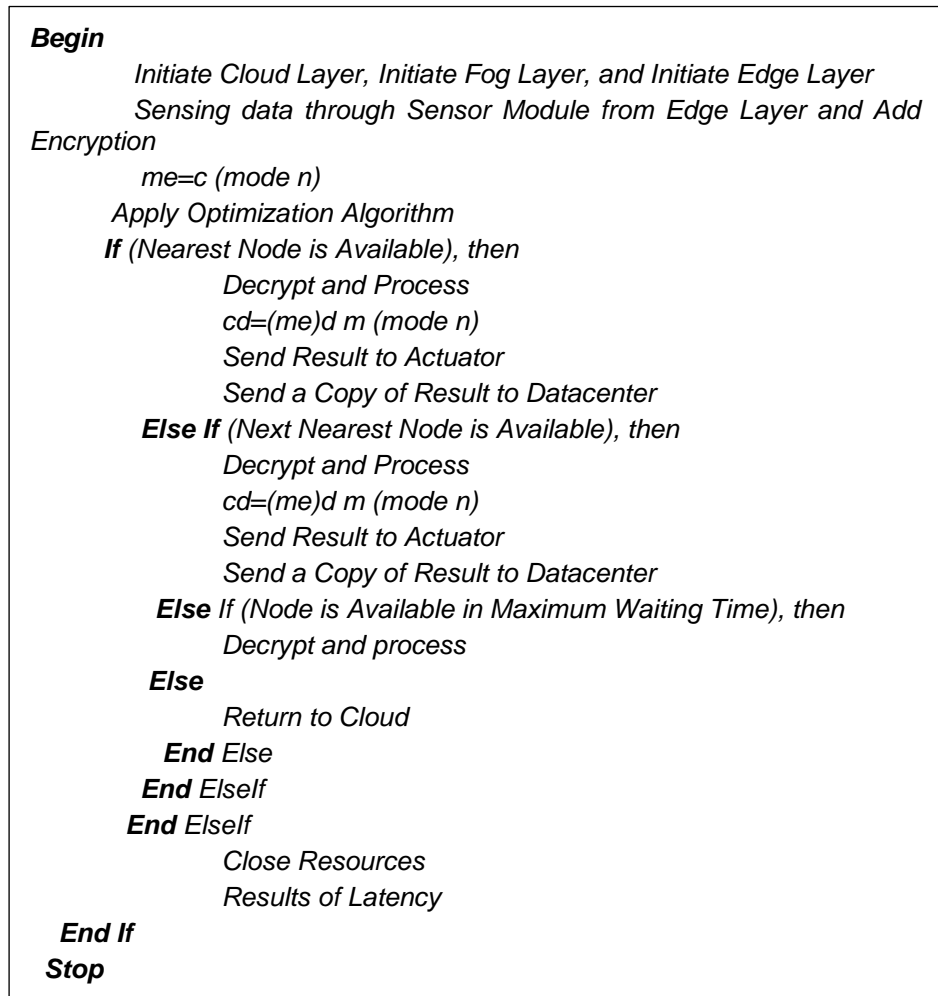
vi.   Calculating and delivering the results.

Begin

    Initiate Cloud Layer, Initiate Fog Layer, and Initiate Edge Layer

    Sensing data through Sensor Module from Edge Layer and Add Encryption

    me=c (mode n)

   Apply Optimization Algorithm

  If *(Nearest Node is Available), then*

    Decrypt and Process

    cd=(me)d m (mode n)

    Send Result to Actuator

    Send a Copy of Result to Datacenter

  Else If *(Next Nearest Node is Available), then*

    Decrypt and Process

    cd=(me)d m (mode n)

    Send Result to Actuator

    Send a Copy of Result to Datacenter

  Else *If (Node is Available in Maximum Waiting Time), then*

    Decrypt and process

  Else

    Return to Cloud

  End *Else*

 End *ElseIf*

 End *ElseIf*

    Close Resources

    Results of Latency

 End If

Stop

Figure 1. Algorithm for improving the fog Computing performance regarding the latency.

4.1. Classes Interaction of the Proposed Application Execution

The Class Interaction Model shows the components and the relationships among the main parts of the proposed method of fog simulation. Fig 2 demonstrates the design of the Class Diagram of the System.

5. LATENCY RESULTS AND DISCUSSION

Latency is defined as the delay in service once requested, possibly due to many factors. The architecture of the service turned out to be not good, and fog Computing jobs are usually offloaded to Edge servers called Cloudlets and sometimes called Edge machines. The main tasks are broken down and assigned to specific nodes or leaps. Our developed application scenario has network latency and service level delay once things are done on the cloud instead of in the fog. Cloud servers fulfill the demands at the cloud level, the application consumes data late, and the information preview comes later than fog.

5.1. Latency Mathematical Models Used in the Application

Two mathematical equations are employed for the latency results calculations, as follows:

For Cloud:

$$L(Cloud) = (pinging + TaskIncreased)/ActiveTime \qquad (1)$$

For Fog:

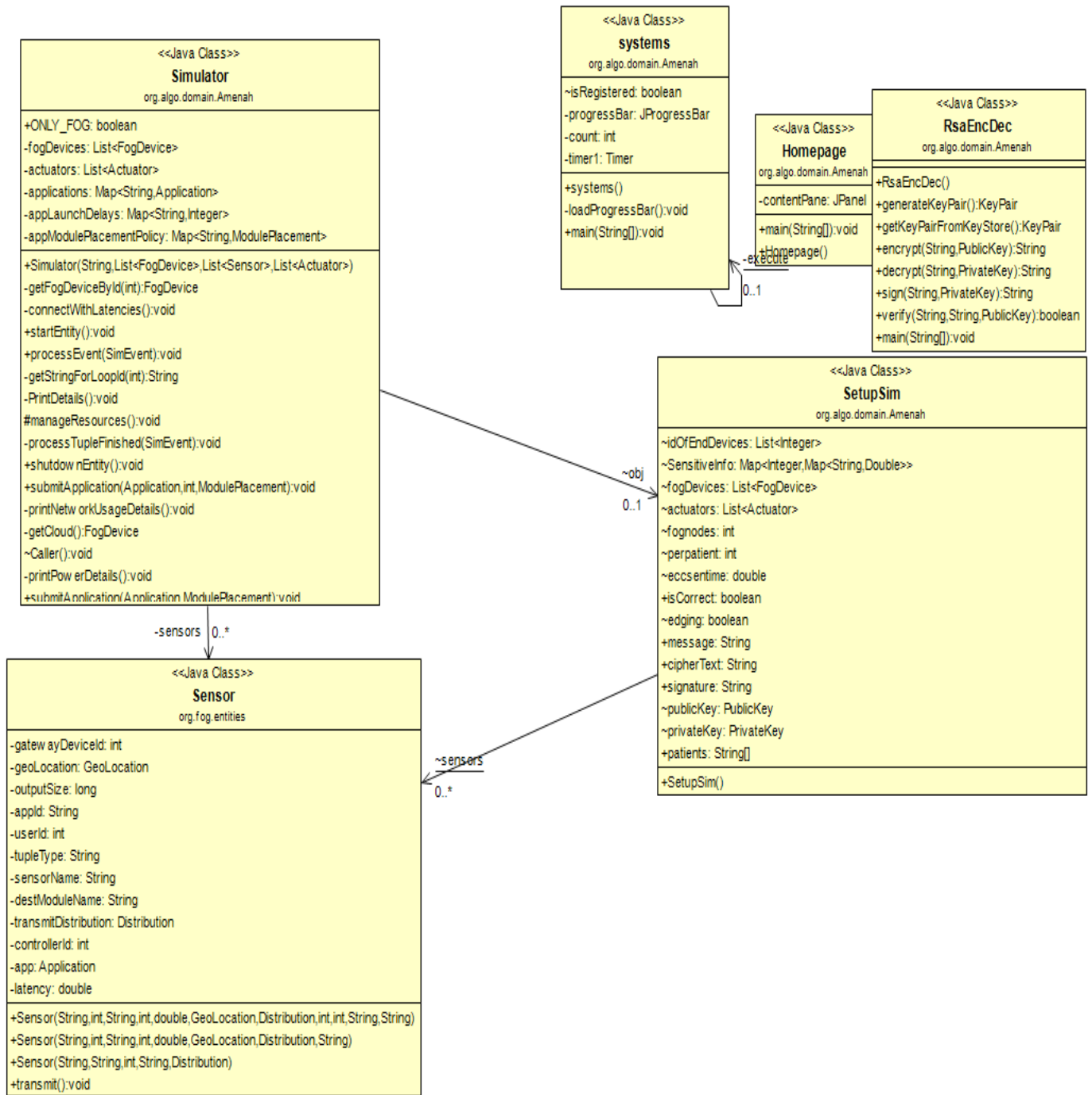$$L(Fog) = (pinging + TaskIncreased)/ActiveTime \qquad (2)$$

Figure 2. Demonstrates the design of the Class Diagram of the proposed system.

Figure 3 and Table 2 show the experimental results of this paper using equations 1 and 2, where the first column in the table represents the first experiment and its active time value = 25 and ends with the fourth experiment with active time value =400 as a doubling of the values. At the same time, the number of tasks is increasing gradually from 500 to 6000 from the first experiment to the last one.

In addition, Table 2 contains the data achieved from the iFogSim simulator to preview the optimized values by introducing fog Cloudlets to the application IoT application. One of the table contents is the active time when the application responds to preview the results back to the application. Here, we see the latency results for both fog and Cloud. The same tasks that need to be done on the Cloud have a latency of 30.8; the same task in fog has a latency of 22.97 when the number of tasks is 500, and the rate of reduction by fog is 10; kindly see Figure 3.
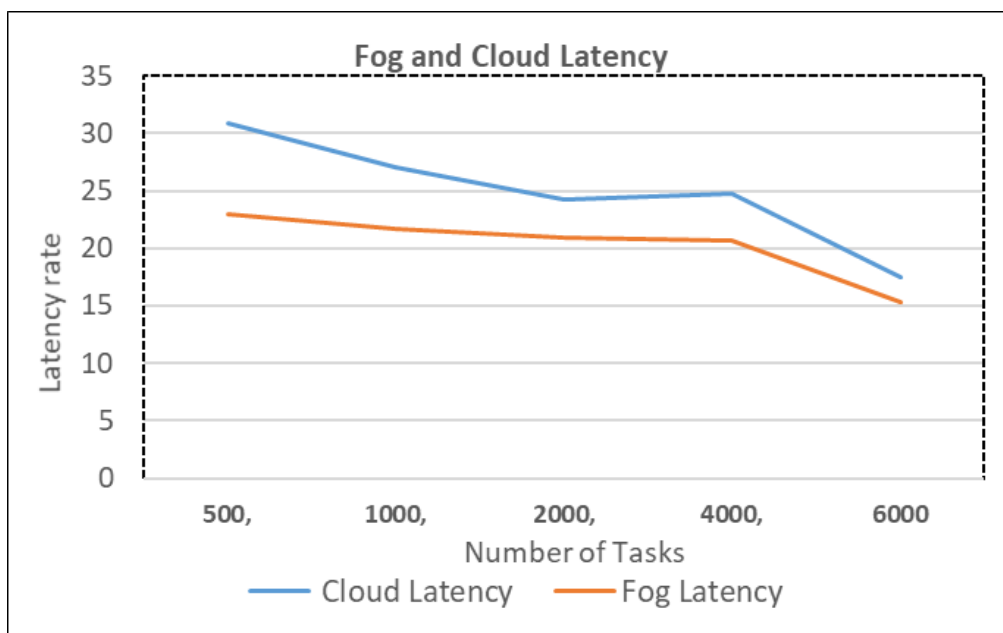
Figure 3. Comparison between Cloud and Proposed fog using Latency values

On the other hand, the cloud is at its peak with a fair increase in task size of 4000 latency Cloud, which grows pinging time to 946.5 and active time to 200. Still, the fog curve does not increase significantly as it increases on cloudlets, the latency is improved and fairly controlled by the fog server. But the latency curve is decreased with the fog and vice versa with the Cloud with all given tasks, where all values of the Cloud latency compared with the fog in each experiment give higher values than the fog. The latency is reduced and reasonably controlled with this fog server instead of putting the entire job request on the server. Hence, the results indicate that the latency is reduced with increased tasks and active time, processed offloading on the computational edge is much faster than in the Cloud, and better-optimized features arrive, positively affecting final performance

Table 2. Latency Results

Active Time	Pinging Time Cloud	Pinging Time Fog	Tasks	L(Cloud-Based)	L(Fog-Based}
25	270.904	74.4300408	500	30.836	22.977
50	353.2144	83.3376611	1000	27.064	21.666
100	419.092	95.3453056	2000	24.190	20.953
200	946.5893	123.369048	4000	24.732	20.616
400	1020.345	123.434544	6000	17.500	15.308

6. CONCLUSIONS AND FUTURE WORKS

The proposed system was implemented using various layers in a simulation environment for the medical environment. The application was packaged and containerized using fog servers connected to the centralized and compact cloud through the fog application network. The objective of this paper was to process the main functions of fog in an optimized form so that the optimization and sharing of the processes are done in a fast and accurate method, which was only possible after the implementation based on cloud computing. The results show that the proposed system achieves very significant results. The latency was optimized to a better level. Based on the results discussed in Section 5. The proposed system can be applied in the real environment and thus provides excellent results and achieves significant benefits in terms of performance optimization according to the Latency parameter. This work can be extended by adding more experiments and improving efficiency depending on other new parameters in different environments.

7. REFERENCES

[1] Tabrizchi, H., Kuchaki Rafsanjani, M. A survey on security challenges in cloud computing: issues, threats, and solutions. J Supercomput 76, 9493–9532 (2020). https://doi.org/10.1007/s11227-020-03213-1

[2] Gurjanov, A. V., Korobeynikov, A. G., Zharinov, I. O., & Zharinov, O. O. (2021). Edge, fog, and cloud computing in the cyber-physical systems networks.

[3] Abdali, T. A. N., Hassan, R., Aman, A. H. M., & Nguyen, Q. N. (2021). Fog Computing Advancement: Concept, Architecture, Applications, Advantages, and Open Issues. IEEE Access, 9, 75961-75980.

[4] Jafari, V., & Rezvani, M. H. (2021). Joint optimization of Energy Consumption and time delay in IoT-fog-cloud computing environments using NSGA-II metaheuristic algorithm. Journal.

[5] H. F. Atlam, R. J. Walters, and G. B. Wills, "Fog computing and the internet of things: A review," Big Data and Cognitive Computing, vol. 2, no. 10, pp. 1-18, 2018.

[6] M. Chiang, S. Ha, F. Risso, T. Zhang, and I. Chih-Lin, "Clarifying fog computing and net-working: 10 questions and answers," IEEE Communications Magazine, vol. 55, no. 4, pp. 18-20, 2017.

[7] S. Chatterjee, "Fog Computing Applications, in Sensors, Cloud, and Fog: The Enabling Technologies for the Internet of Things," CRC Press, pp 167-186, 2019.

[8] N. A. C and R. Lavanya, "Fog computing and its role in the internet of things, in Advancing Consumer-Centric Fog Computing Architectures," IGI Global, pp. 63-71, 2019. [

[9] B. M. Nguyen, H. T. T. Binh, and B. D. Son, "Evolutionary algorithms to optimize task scheduling problem for the IoT based bag-of-tasks application in cloud–fog computing environment," Applied Sciences, vol.9, no.9, p. 1730, 2019.

[10] H. D. Abdulgalil, " A Multi-Tier Distributed fog-based Architecture for Early Prediction of Epileptic Seizures," M.S. thesis, University of Waterloo, 2018.

[11] H. L. M. D. Santos, "A Multi-tier fog architecture for video on demand streaming," M.S. thesis, Federal University of Par´A Institute Of Technology, 2018.

[12] A. Yousefpour, G. Ishigaki and J. P. Jue, "Fog Computing: Towards Minimizing Delay in the Internet of Things," 2017 IEEE International Conference on Edge Computing (EDGE), pp. 17-24, 2017.

[13] T. Salman and R. Jain, "Networking protocols and standards for internet of things," Internet of Things and Data Analytics Handbook, pp. 215-238, 2017.

[14] Z. Hao, E. Novak, S. Yi and Q. Li, "Challenges and Software Architecture for Fog Computing," in IEEE Internet Computing, vol. 21, no. 2, pp. 44-53, 2017.

[15] E. Simmon and R. Bohn, "An overview of the NIST cloud computing program and reference architecture," Concurrent Engineering Approaches for Sustainable Product Development in a Multi-Disciplinary Environment, pp. 1119-1129, 2013.

[16] B. Varghese, C. Reaño and F. Silla, "Accelerator Virtualization in Fog Computing: Moving from the Cloud to the Edge," in IEEE Cloud Computing, vol. 5, no. 6, pp. 28-37, 2018.

[17] L. Gao and M. Moh, "Joint Computation Offloading and Prioritized Scheduling in Mobile Edge Computing," 2018 International Conference on High Performance Computing & Simulation (HPCS), pp. 1000-1007, 2018.

[18] N. Mohan and J. Kangasharju, "Edge-Fog cloud: A distributed cloud for Internet of Things computations," 2016 Cloudification of the Internet of Things (CIoT), pp. 1-6, 2016.

[19] M. L. M. Peixoto, T. A. L. Genez and L. F. Bittencourt, "Hierarchical Scheduling Mechanisms in Multi-Level Fog Computing," in IEEE Transactions on Services Computing, vol. 15, no. 5, pp. 2824-2837, 2022.

[20] L. Liu, D. Qi, N. Zhou, and Y. Wu, "A Task Scheduling Algorithm Based on Classification Mining in Fog Computing Environment," Wireless Communications and Mobile Computing, vol. 9, pp. 1-11, 2018.

[21] J. Wang and D. Li, "Task Scheduling Based on a Hybrid Heuristic Algorithm for Smart Production Line with Fog Computing," Sensors (Basel), vol. 19, no. 5, 2019.

[22] S. Iftikhar, M. M. Ahmad., S, Tuli., D, Chowdhury, M, Xu, S, S, Gill, & S, Uhlig. (2023). HunterPlus: AI based energy-efficient task scheduling for cloud–fog computing environ-ments. Internet of Things, 21, 100667.

[23] A. Bonguet and M. Bellaiche, "A survey of denial-of-service and distributed denial of service attacks and defenses in cloud computing," Future Internet, vol. 9, no. 3, p. 43, 2017.