Computer Systems
Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2023.036316
Article

Two-Way Approach for Improved Real-Time Transmission in Fog-IoT-Based Health Monitoring System for Critical Patients

Abeera Ilyas[1,*], Saeed Mahfooz[1], Zahid Mehmood[2,3], Gauhar Ali[4] and Muhammad ElAffendi[4]

[1]Department of Computer Science, University of Peshawar, Peshawar, 25120, Pakistan
[2]Department of Computer Engineering, University of Engineering and Technology, Taxila, 47050, Pakistan
[3]The FAMLIR Group, The University of Lahore, Lahore, 54000, Pakistan
[4]EIAS Data Science and Blockchain Lab, College of Computer and Information Sciences, Prince Sultan University, Riyadh, 11586, Saudi Arabia
*Corresponding Author: Abeera Ilyas. Email: abeerailyas@uop.edu.pk

Received: 25 September 2022; Accepted: 06 January 2023

Abstract: Health monitoring systems are now required, particularly for essential patients, following the COVID-19 pandemic, which was followed by its variants and other epidemics of a similar nature. Effective procedures and strategies are required, though, to react promptly to the enormous volume of real-time data offered by monitoring equipment. Although fog-based designs for IoT health systems typically result in enhanced services, they also give rise to issues that need to be resolved. In this paper, we propose a two-way strategy to reduce network latency and use while increasing real-time data transmission of device gateways used for sensors by making educated judgments for connection setup with BS and task assignment. For this, a simulation using iFogSim in the Eclipse IDE showed how effective the suggested strategy for massive IoT health monitoring systems is. The algorithm is analyzed for network usage and latency, and the results reveal 20%–25% improvements compared to the existing methods regarding network usage and latency.

Keywords: IoT; fog computing; delay; healthcare; network usage; quality of service

1 Introduction

E-health is a state-of-the-art system that combines Internet health technologies, healthcare, and medical informatics. Combining these factors drives technological development to address enduring problems like healthcare quality and cost containment. Parallel to this, the growth of several innovative projects that provide consumers access to shared/configurable technologies like the network-connected hardware, cloud computing or storage, Internet, software, databases, and other resources is made possible by the Internet of Things (IoT). IoT and cloud computing have become ground-breaking innovations that enhance one another's capabilities when combined with flexible, accessible, and effective patient health care. The link has advantages over traditional networks, including easier implementation, improved information security during communication, increased access to data,

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

and increased productivity. IoT and cloud-based e-Health systems, which also promote ongoing, methodical improvement, can significantly improve healthcare services. Medical data is preserved in the cloud by the underlying IoT networks utilized by cloud-based e-Health systems to communicate with users, providers, and servers [1].

On the other hand, cloud computing has several significant drawbacks, such as traffic congestion, massive data processing requirements, and sluggish data transmission. The primary reason for these issues is because IoT devices and cloud servers are physically separated from one another [2]. Because healthcare apps are critical, they cannot tolerate delays. As a result of high bandwidth utilization and huge transmission latency, cloud computing services are not suitable for collecting and analyzing medical data from patients dispersed across a wide geographical area [3]. Fog computing has come into being as a new paradigm in response to the fundamental problems with cloud computing [4]. A range of different types of equipment are connected to the network by the architecture of the geographically scattered fog computing site, providing computational and storage capacity. Fog computing offers an infrastructure and a more secure approach to manage data because it uses less bandwidth. The idea of healthcare systems draws attention to the reality that most countries' healthcare systems have problems that get worse as their populations age [5].

The Internet of Things and wireless and cellular network advancements in health monitoring systems greatly increase performance and lower healthcare expenses. IoT promotes monitoring of critical patients by providing affordable home monitoring solutions that identify early signs of deterioration and enable quicker reaction and treatment [6]. To provide IoT-based health monitoring, a fog computing-based infrastructure is suggested. A fog-based health monitoring system analyzes information acquired and transmitted by sensors to determine the patient's health status. The proposed structure's three storeys. The initial tier of the architecture consists of sensors connected to patients for monitoring and transmitting critical signals including core temperature, heart rate, and pulse rate as well as other key studies. All IoT devices are connected to the base stations (BSs) through the intermediate fog layer. In the first stage, IoT devices provide sensor-generated data to the top layer proxy server. The fog nodes use this server to analyse the digital input and assess whether the patient is in a critical condition [7]. Thanks to the suggested health monitoring system architecture and fog computing's success in time-sensitive applications, patients can receive ongoing real-time medical help. To manage the massive amounts of data generated by end-user devices, fog computing places resources close to end users. A real-time efficient data analysis is required in the example situation, so fog computing should be used.

Fog servers can only store a finite amount of data, though. As there are more available queries in large systems, the demand on the fog server climbs [8]. More patients' requests for its installation globally will put pressure on the proposed fog-based health monitoring system. The delay in this instance is due to fog nodes becoming overloaded when the other fog nodes are quiescent. Response times are lengthened as a result of this situation. It is also important to select the fog nodes for the IoT devices properly to reduce network latency.

There are three levels in the proposed architecture. First-tier fog nodes monitor and communicate vital information, such as heart rate, body temperature, and pulse rate, from the sensors on the patients. Base stations (BS) are located next to the fog servers that form the intermediate fog layer. As the first tier's IoT devices send monitoring data to the fog nodes in real time, they evaluate it to determine whether the patient is in a critical state and transmit that information to the third tier's proxy server. As well as transmitting the patient's health status findings to their cellphones, the fog nodes also display them on their phones [8]. By putting resources near to end-users, fog computing offers a mechanism to manage a significant amount of data generated by end-user devices. As a result, fog computing is ideal for the suggested method since it requires effective real-time data processing. The IoT gateway must search for suitable fog nodes to transmit monitoring data because fog nodes are often overloaded and the process is prone to delays. IoT sensor gateways are not suitable for real-time monitoring due to the time necessary to connect to the appropriate fog node. Furthermore, the stress on the fog nodes lengthens the time needed for Dp processing, delaying transmission. It is important to address both network delays in order to improve the quality of service (QoS).

The total delay is,

$$\text{Total delay}, D_t = D_c + D_p \tag{1}$$

The research makes a ground-breaking two-way suggestion for controlling the overall network latency. In order to control the incoming monitoring traffic at fog nodes with real-time monitoring data, a network-assisted load balancing technique is originally advised. Due to the time-sensitive nature of the health monitoring system, the strategy distributes the load among nearby fog nodes to decrease latency and network consumption. Second, a scanning algorithm for the IoT gateway proposes a technique to reduce the time between communications by connecting to the optimal fog node. According to [9], IoT data transfers incur processing and traffic delays. Fog-based computing is used in the system architecture to minimize the frequency of data transfer to the cloud server, minimizing the application's bandwidth requirements.

The following are the contributions of this paper:

- Consideration is given to a three-tier architecture with fog nodes consolidated at cellular BS. IoTD uses body sensors and telephones to broadcast the patient's physiological data streams to the fog node at BS. Processing and patient health status checks are performed on the incoming data streams. After that, the results are delivered to the patient's smartphone or sent to the cloud for storage.
- The load is balanced using the Load Balancing of Real-Time data technique (LBRT), which equally distributes incoming jobs among fog nodes without introducing any additional delay.
- It is advised that the IoT gateway employ the Efficient Scanning Mechanism (ESM) to quickly and efficiently select an appropriate fog node for transmission.
- The performance of suggested approach is evaluated against latency during connection establishment and overall delay until response reaches back to PDAs.
- Extensive simulations are done using the iFogSim toolbox to compare the performance of the LBRT with the traditional fog node placement algorithm (FNPA) [5] and the load balancing scheme (LBS) given in [9].

The format of the paper is as follows: In Section 2, the background and inspiration for the composition are described. The most recent research on load balancing in fog-based systems and the development of health monitoring systems is described in Section 3. It is outlined in Section 4 how health monitoring systems can be designed, and in Section 5 how the LBRT algorithm can be implemented. Conclusions and suggestions for further improvement are given in Sections 6 and 7.

2 Background and Motivation

Covid-19 has highlighted the need to adopt more innovative digital health technology, particularly remote health monitoring. Digital wearable technologies have enabled patients to self-analyze issues before they become severe cases requiring immediate attention. Medical professionals should review sensor data and establish a diagnosis prior to visiting a clinic or hospital, since self-diagnosis could be dangerous. For remote patient monitoring to meet the needs of patient trials or strict criteria, real-time processing of data streams from patient health sensors is required [10,11].

The COVID-19 pandemic's indirect consequences on healthcare services are particularly dangerous for infants. An extensive amount of network resources are needed for a cloud-based health monitoring system since user IoT devices provide a lot of sensor-generated data to a cloud server. As technology becomes more widely used, the delay becomes more significant in time-sensitive contexts, such as health monitoring systems. The recommended fog aided IoT reduces the frequency of data transfer to the cloud server, reducing the latency of the connection establishment phase. Fog computing is ideal for IoT-based health monitoring deployments because it lowers communication traffic over the network and improves scalability. Deploying fog-enabled architectures greatly reduces network utilization, which is still a key factor in real-time applications [12]. With more people using IoT devices, real-time data transfer in huge applications puts more strain on fog servers. To properly sustain applications, it is crucial to balance the demand on fog nodes. The distribution of application traffic among many servers helps load balancing techniques boost the capability and stability of the applications. More client requests may cause one server to become overloaded, in which case the overloaded fog node may shift some of the load to the subsequent server [13]. The system can therefore optimize the network and resources and reduce processing time as a result of this workload allocation [14]. The motivation for this work is:

- Reduced communication delay for IoTD by deploying a BS-assisted approach.
- Introducing threshold checks to alert the BS about the overburdened fog nodes.
- Utilizing a BS-generated list for reduced network usage by eliminating excessive handshake protocols.
- Providing a fault-tolerant method to reduce the unnecessary connection establishment with a faulty or overburdened fog node.

3 Related Work

As they transmit data to the fog node, the authors of [9] offer a new Load Balancing Scheme (LBS) for balancing traffic by lowering communication and compute latency. The authors used the iFogSim tools to run numerous simulations, comparing the outcomes to the cloud-only approach, the Fog Node Placement Algorithm (FNPA), and the LoAd Balancing (LAB) scheme in terms of latency and network utilisation, in order to assess the effectiveness of the suggested strategy. The suggested method, however, makes use of IoT sensors, which demand intricate calculation. In order to keep them small and light, gadgets made for sensitive and critical patients typically have minimal hardware needs. The weighted majority game theory is used by the suggested approach in [10] to choose a fog device for both indoor and outdoor locations. The approach offers an energy-efficient option, but real-time transmission—which is crucial in a health monitoring system—must take delay and traffic into account. The authors of [15] have created a system architecture with integrated artificial intelligence for the purpose of carrying out health monitoring activities. This system architecture consists of Edge and Fog computing, low-powered wide area network (LPWAN) technologies, IoT, and deep learning algorithms. To demonstrate the practicality and effectiveness of this approach, they selected a use case of a fall detection system utilizing recurrent neural networks. But for real-time data transmission, the system must consider latency and network congestion. The authors suggested an intelligent health monitoring system design in a different work [16]. Their research has been put into practise on a test platform, and the system is an implementation of the Internet of Things that integrates fog computing and LoRa wireless communication. The offered solution only provides a broad overview of the configuration, making no mention of the data type, latency, network traffic, or the availability of fog nodes. Another piece of literature cited in [17] presents a privacy-focused e-healthcare framework for electronic medical records (EMRs). Additionally, the writers tested the suggested response time and delay work and contrasted it with current findings. The network use and workload of the fog node must, however, be considered in the solution. The calculation done to achieve privacy is also complex for sensing devices. A Fog-assisted health monitoring system is created in [18], and its effectiveness is assessed and demonstrated. Because data is reviewed locally, this demonstrates how this strategy may be enhanced to decrease data traffic in the network's core. Additionally, it strengthens the security of locally stored health data, enhancing data security and delivering greater understanding of a patient's health status. For data transport, the framework must concentrate on network use and fog node availability. Additionally, the approach is not fault-tolerant. A tiny, inexpensive IoT node, a smartphone application (app), and fog-based Machine Learning (ML) tools for data analysis and diagnosis make up the framework suggested in [19,20]. Health indicators including body temperature, frequency of coughing, and blood oxygen saturation are monitored by an Internet of Things (IoT) node. The mobile app then receives an update about the user's health. The suggested system's data analysis requires complex calculations, which can cause data transfer to lag. The system does not take into consideration the network utilization or load balancing of the fog nodes. In an IoT-enabled healthcare system, the authors suggested that [21] utilizes a number of fog nodes to coordinate patient requests from far-off cities. The fog node keeps track of the patient's condition; if it becomes critical, the request is forwarded right away to a cloud server; otherwise, the fog node handles it. The results of this study were not affected by the fact that the study participants were from different countries or that the study participants were from different generations.

For health monitoring systems, several researchers put forth fog-based architectures that performed better than cloud-based ones. However, none of the researchers compared their proposed strategy with the previously published fog-based architecture for health monitoring. The development of a fog-based strategy that is more successful than previously proven fog-based strategies is preferred to achieve improved QoS. Even though latency and network utilisation are crucial factors for health monitoring systems, the majority of research have only examined their suggested solutions against delay at the fog layer. However, a technique that can reduce the overall system latency is desired. As a result, measuring the entire system latency using our suggested model is also necessary.

4 Proposed System Architecture

This section demonstrates a fog-based health monitoring system's three-tier design, which is like the strategy outlined in [9]. The critical patient wears sensors in the first layer of the proposed architecture to track vital indicators including body temperature, pulse rate, heart rate, and other vitals, which are subsequently communicated to the fog nodes through telephones. The second tier of the architecture is made up of the fog nodes. It is the fog nodes, located adjacent to the BSs, that analyze the data collected from IoT devices. It is then sent to the patient's smartphone with the findings of their health status. A fog network's edge nodes are placed closer to IoT devices so they can react quickly in a real-time environment. Fig. 1 depicts the fog-based Internet of Things health monitoring system.

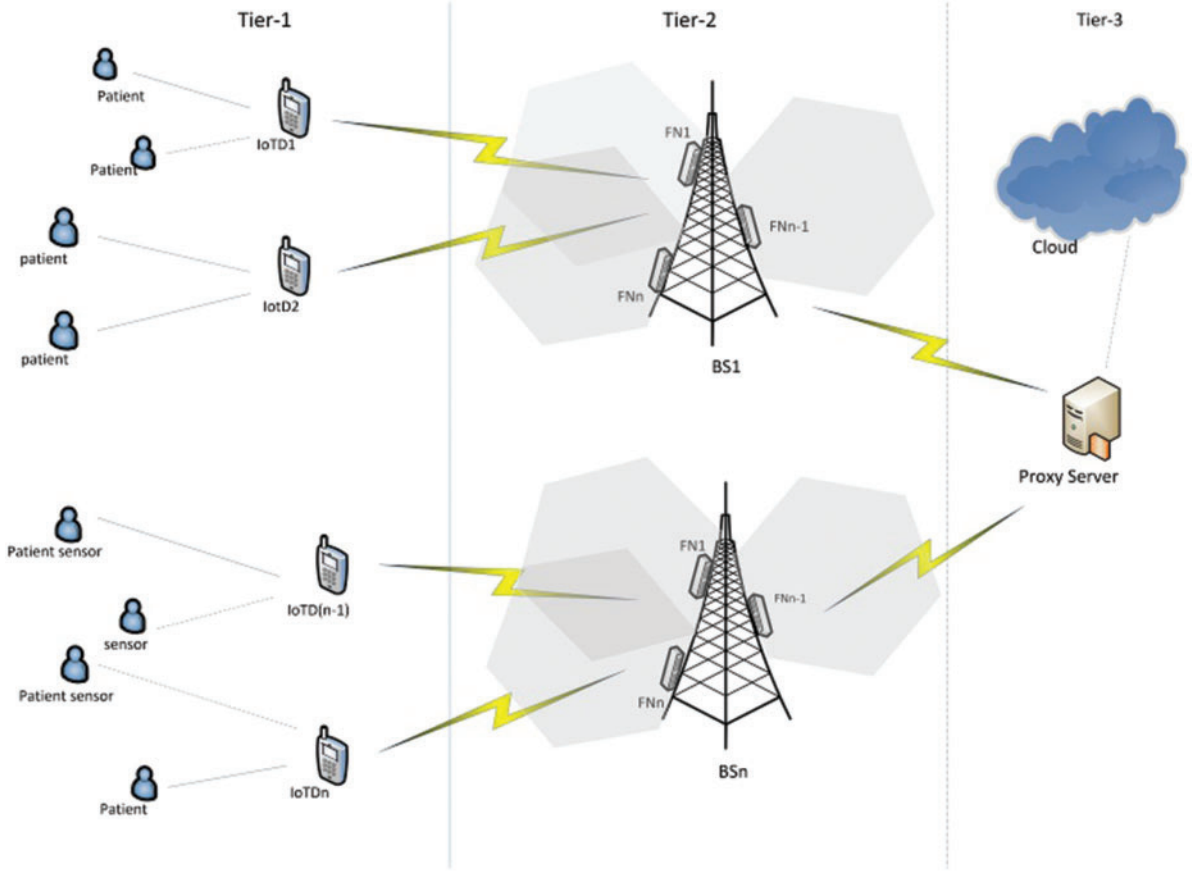


Figure 1: Framework of the proposed model

4.1 Data Generation Layer

The data generation layer is made up of IoT sensing devices. The patients' sensors record the vital signs of the critically ill patient and transmit this information to higher-level IoT device gateways (IoTDs).

4.2 Middle Layer

There is a layer that sits in between the data generating and data analysis layers. Sent to the fog layer for processing is the data gathered by sensors. There are BSs where the fog nodes are co-located, as shown in Fig. 2. Fog node and BS coverage zones may overlap [22]. Every IoTD falls under the coverage of several BS. In terms of networking, processing speed, and storage capacity, fog nodes are less powerful than cloud servers. The fog layer in the suggested fog-based architecture functions as a helpful intermediate layer for processing real-time data close to end users for faster response times.

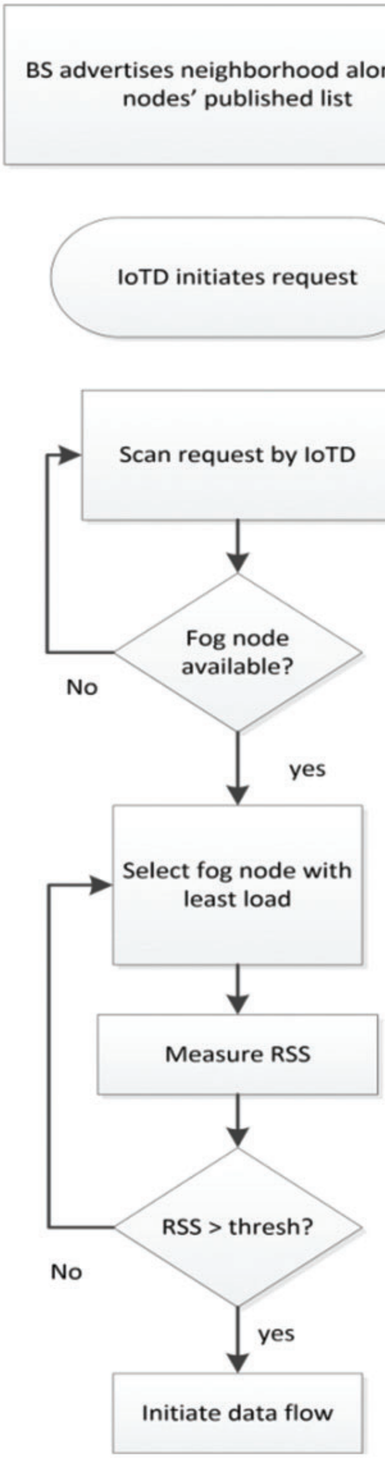


Figure 2: ESM for IoTD

4.3 Data Analysis Layer

The top layer of the health-IoT architecture is made up of the cloud server and the proxy server. Adding extra on-demand processing and storage resources is the cloud's main purpose. A proxy server connected to the fog layer makes it easier for data to be sent to and received from the cloud server. The patient health status results are sent from fog nodes to the cloud server, which contains a permanent database to keep the results. Additionally, healthcare professionals are always able to get data from the cloud [23].

5 Proposed Two-Way Delay Management Scheme

Real-time communication in health monitoring cannot afford any delay. In fog networks, a delay is caused primarily by communication delay D_c and processing delay D_p at fog nodes. So, the total delay

$$D_L = D_c + D_p \tag{2}$$

5.1 Communication Delay

Usually, IoTD performs complex handshaking protocols to establish a connection with a suitable fog node for sending data.

Efficient Scanning Mechanism (ESM)

The proposed efficient scanning mechanism (ESM) lessens the burden of selecting a suitable node from the IoT device gateways. In this method, a list (BS_{SL}) containing the basic stats of each fog node is published and broadcasted by the BS periodically. The IoTD selects a suitable fog node based on BS_{SL}. If the selected node has reached its limit, the fog node will handle it further without requiring the IoTD to scan and choose a suitable node again. The flowchart for SSM is described in Fig. 2.

5.2 Processing Delay

Mainly, a processing delay occurs till the job waits in a queue for its turn to be processed by the fog node. Traditionally, the incoming tasks are handled as mentioned in Fig. 3.

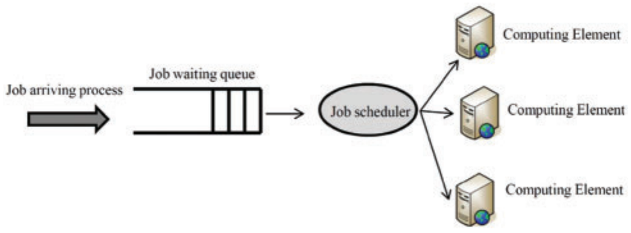


Figure 3: Job scheduling

However, in this scheduling method, the priorities of jobs are analyzed after they are received. Also, there is a demand for a more suitable technique for handling critical data. Therefore, with the introduction of the fog layer, servers can schedule tasks with minimum response time. But, another issue is the overloading of fog nodes which substantially affects the processing delay. LBRT can reduce it by efficiently managing the workload among fog nodes. A simple and less computational method is devised to support the real-time monitoring tasks. The real-time data sent by the IoTD transmit data with high priority. Therefore, the approach suggested focuses more on handling the incoming jobs at fog nodes rather than prioritizing the data streams.

Load Balancing for Real-time Data (LBRT)

In *LBRT,* the fog node (FN_n) receives a request for real-time data transmission from the IoT device gateways ($IoTD_1$, $IoTD_2 \ldots$ $IoTD_n$), which are present under its coverage area. Initially, the fog nodes cater to the request as a first come, first serve method (*FCFS*). However, when $\mathrm{FN_n}$ reaches its threshold, it will update BS_{SL} and publish it, informing all the neighboring fog nodes and IoTDs. Afterward, FN_n will only serve the incoming tasks close to it. When FN_n receives further connection requests, it assesses it according to BS_{SL} i-e., FN_n cell load, location of IoTD and RSS. In case of a mismatch, the data stream is passed on to the nearest fog node, FN_{n+1}, after checking the updated BS_{SL}.

The novelty of this approach is dynamic request selection by the fog node when the threshold is reached. The incoming request is assigned to an FN_n using LBRT, as described in Fig. 4.

$$\mathrm{FN_n}\ \text{load} < \text{AND} > \text{RSS IoT device} < \text{AND} > \text{proximity of IoT device} \tag{3}$$

There are three primary parts to the model.

- Discovery manager: By adopting ESM for task thread transmission, this agent attempts to serve the IoTD. The published BS_{SL} supports the device manager in identifying a suitable BS for establishing the connection.
- Fog service pool: A database called the fog service pool continually supports fog nodes integrated with LBRT. This database provides information on all fog node specs, including CPU speed and core count, storage size, current capacity, and current cell load.
- Cloud Proxy server: The proxy server is available in the data analysis layer to execute incoming threads from the middle fog layer. In this case, the platform as a service (PaaS) approach is adapted for the cloud server. The threads may be stored on the medical servers or sent to the practitioner for further evaluation or response in a highly critical situation.

6 Discussions

The effectiveness of the suggested solution is contrasted with that of FNPA [8] and the conventional method for fog-based IoT [9]. The model mentioned works on load balancing at fog nodes—however, the proposed scheme is in this method, as shown in Fig. 5, the IoTD uses ESM to select an appropriate fog node.

The BS-generated list, $\mathrm{BS_{SL}}$, is advertised periodically and received by all the IoTD within range. At the time of connection establishment, the IoTD uses ESM to select a suitable BS and fog node subsequently. However, at the receiving end, as there is a time-lapse, so the chosen candidate fog node might have reached thresh. In such a scenario, the fog node consults BSSL to select the most appropriate node based on the proximity of the IoTD and the load of the fog node. This approach substantially reduces the amount of data sent between IoTD and the fog node, thus, saving bandwidth and reducing network costs.

On the other hand, balancing the workload on fog nodes [9] suggests a complex computation that is not appropriate for real-time data or critical situations. Thus, by implementing LBRT at fog nodes, the incoming threads can be handled by the most suitable and available fog node. The suggested model differs from the existing mechanisms due to its dynamic load-balancing mode. Each fog node can adjust its selection method dynamically as soon as it reaches its threshold.

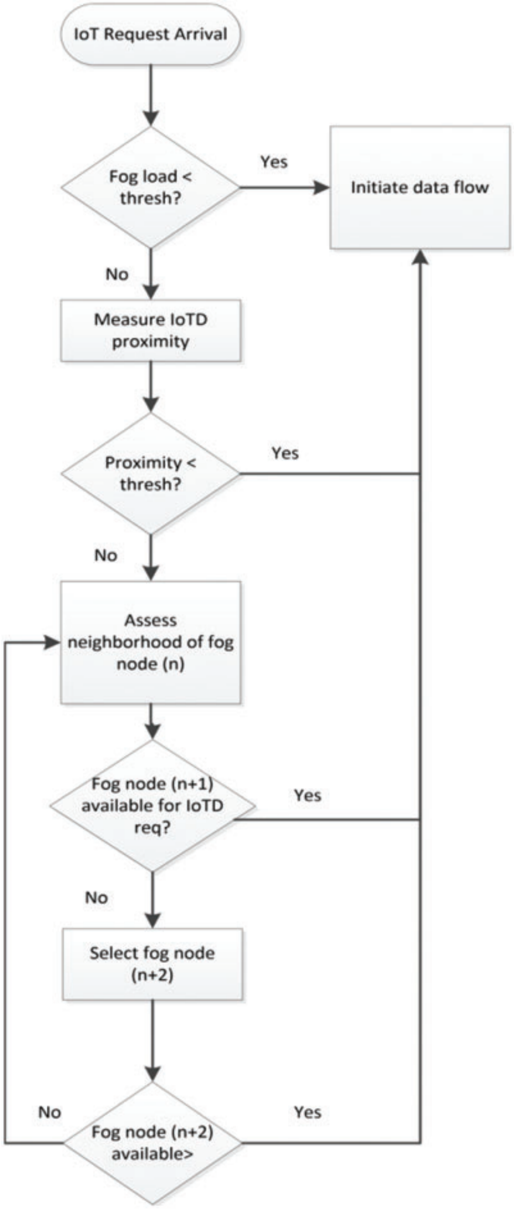


Figure 4: LBRT algorithm

The following points can demonstrate the strength of this architecture:

- Dynamic: Depending on the location of the IoTD, the architecture facilitates cooperation between fog nodes and IoTDs to scale dynamic changes in the BSSL produced by BS periodically.
- Energy Efficient: The architecture focuses on utilizing minimum bandwidth using *ESM* for the scanning phase and opting for the best possible fog node for task assignment using *LBRT*. Also, it would reduce the incoming threads sent to the cloud.

- Response Time: To reduce latency, the design chooses the best fog node between the IoTD and the fog node. Also, early detection of the target at the scanning phase, i.e., whether the fog node FNn will execute the thread or FN_{n+1}, FN_{n+2} . . . , reduces the response time substantially. The existing techniques are not sufficient for transmitting critical data promptly. There can be energy waste and delays in responding to demands when threads are shifted to already overloaded fog nodes. Therefore, ESM and LBRT can efficiently handle these issues.

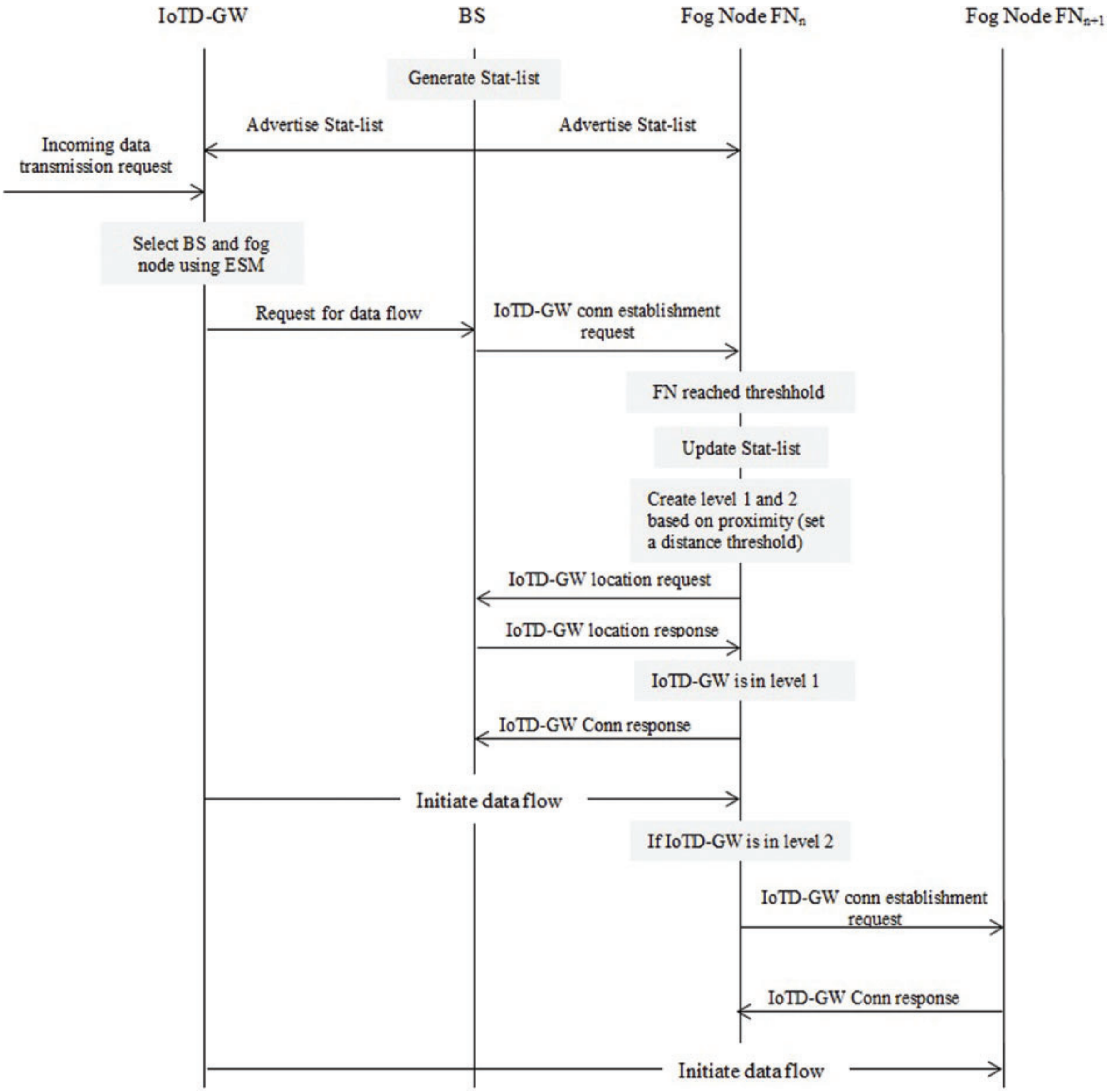


Figure 5: Proposed framework

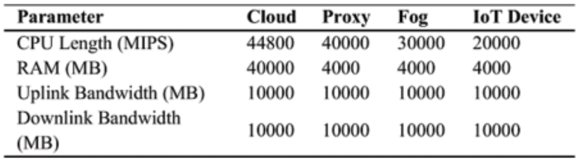
7 Simulation and Results

To achieve the simulation outcomes, the suggested algorithms i.e., ESM and LBRT is programmed, executed and simulated in Eclipse IDE with the iFogSim toolkit [24]. The schemes are analyzed with existing strategies of FNPA and LBS. The parameters for the BS used in the simulation are mentioned in Table 1.

Table 1: BS parameters

Channel frequency	2.5 GHz
BS transmission power	10/17 dbm
BS antenna gain	3 db
IoTD-GWantenna gain	2 db
Signal thresh level	−80 dbm
Bandwidth	10 MHz

Topology parameters used to analyze the suggested scheme is similar to the article [9] as shown in Fig. 6.



Parameter	Cloud	Proxy	Fog	IoT Device
CPU Length (MIPS)	44800	40000	30000	20000
RAM (MB)	40000	4000	4000	4000
Uplink Bandwidth (MB)	10000	10000	10000	10000
Downlink Bandwidth (MB)	10000	10000	10000	10000

Figure 6: Topology parameters

The scanning phase computes the total time required to scan the candidate list of fog nodes and evaluates their RSS and CINR values to account for communication delay. The processing delay is the time the incoming task takes to start transmission. Eq. (4) can calculate the delay as

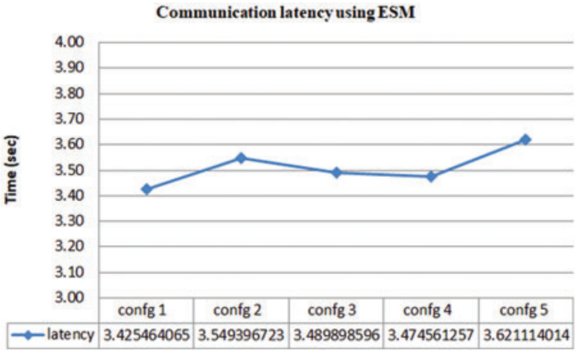
$$\oint_{\mathrm{pro}} = \oint_{\mathrm{F}} + \oint_{\mathrm{G}} \tag{4}$$

where $\oint_{\mathrm{F}}$ is the time a thread had to wait in its initial association selected by IoTD-GW, the node might have reached its threshold by the time the incoming task associates itself to the fog node. Therefore, the fog node connects it with the neighboring most suitable fog node based on proximity; thus, $\oint_{\mathrm{G}}$ is the time taken to identify the incoming thread's location to pass it on to the closest option. The simulation evaluated ESM for various configurations i.e., various number of available fog nodes and BS. It was seen that communication delay was almost stable, as shown in Fig. 7.

The total latency comprises communication and processing delay, which play a vital role in real-time monitoring data transmission. On running the simulation, it was seen that the suggested methods perform notably well as compared to previous methods as shown in Fig. 8.

Another critical parameter affecting the real-time flow of data is network usage. The simulation outcome reveals a substantial improvement in reduced bandwidth usage, thus, decreased network traffic, as shown in Fig. 9.

It is seen in Figs. 8 and 9 that the overall performance for handling delays has significantly improved. The proposed method avoids the needless trial-and-error connection setup with the fog node; thus, performance is enhanced. Secondly, it can reduce processing delay by taking load into account during the handshaking procedure between the IoTD-GW and fog node. It is clear that for various configurations of cell load, the delay and network usage of the suggested approach is better than the existing methods, increasing by an average of 20% to 25%.



	confg 1	confg 2	confg 3	confg 4	confg 5
latency	3.425464065	3.549396723	3.489898596	3.474561257	3.621114014

Figure 7: Communication delay using ESM

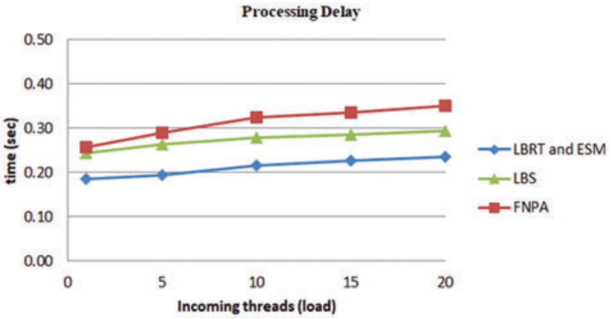
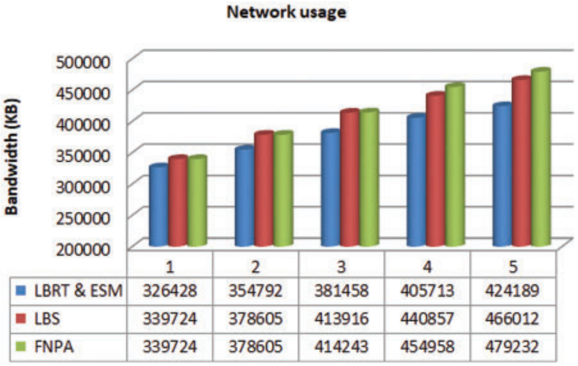


Figure 8: Latency analysis with other schemes



	1	2	3	4	5
🟦 LBRT & ESM	326428	354792	381458	405713	424189
🟥 LBS	339724	378605	413916	440857	466012
🟩 FNPA	339724	378605	414243	454958	479232

Figure 9: Network usage comparison

8 Conclusion and Future Work

IoT applications yield substantial tasks that must be dealt accurately and promptly, as mentioned in [25]. Fog computing is suggested as a solution to accommodate the cloud server by offering services close to IoT devices. When handling urgent real-time health data, many fog computing architectures are ineffective. As a result, a method is suggested to address the problem by developing an effective connection establishment mechanism for IoTDW and a timely and efficient task assignment at the fog node [26]. The proposed model is simulated using the iFogSim toolkit in Eclipse IDE. The results reveal a notable improvement devouring reduced network usage and improved latency compared to the previously popular approaches used in fog-based IoT for health monitoring.

Acknowledgement: This work was supported by EIAS Data Science and Blockchain Lab, College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia.

Funding Statement: The authors would like to acknowledge the support of Prince Sultan University for paying the Article Processing Charges (APC) of this publication.

Author Contributions: Conceptualization, methodology, software and formal analysis by Abeera Ilyas; Investigation by Dr. Saeed Mahfooz; Resources and data collection by Gauhar Ali and Muhammad ElAffendi; Validation by Dr. Zahid Mehmood.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] C. Butpheng, K. H. Yeh and H. Xiong, “Security and privacy in IoT-cloud-based e-health systems—A comprehensive review,” *Symmetry*, MDPI, vol. 12, no. 7, pp. 1191, 2020.

[2] G. Fersi, “Fog computing and internet of things in one building block: A survey and an overview of interacting technologies,” *Cluster Computing*, Springer, vol. 24, pp. 2757–2787, 2021.

[3] G. L. Santos, D. Gomes, J. Kelner, D. Sadok, F. A. Silva *et al.,* “The internet of things for healthcare: Optimizing e-health system availability in the fog and cloud,” *International Journal of Computational Science and Engineering*, vol. 2, no. 4, pp. 615–628, 2020.

[4] Z. Safdar, S. Farid, M. Qadir, K. Asghar, J. Iqbal *et al.,* “A novel architecture for internet of things based e-health systems,” *Journal of Medical Imaging and Health Informatics*, ASP, vol. 10, no. 10, pp. 2378–2388, 2020.

[5] E. Andrade, B. Nogueira, I. D. Farias Júnior and D. Araújo, “Performance and availability trade-offs in fog–cloud IoT environments,” *Journal of Network and Systems Management*, Springer, vol. 29, no. 1, pp. 1–27, 2021.

[6] Q. Fan and N. Ansari, “Towards workload balancing in fog computing empowered IoT,” *Transactions on Network Science and Engineering*, IEEE, vol. 7, no. 1, pp. 253–262, 2020.

[7] A. A. Alli and M. M. Alam, “The fog cloud of things: A survey on concepts, architecture, standards, tools, and applications,” *Internet of Things*, Elsevier, vol. 9, 2020.

[8] A. Asghar, A. Abbas, H. A. Khattak and S. U. Khan, “Fog based architecture and load balancing methodology for health monitoring systems,” *IEEE Access*, vol. 9, pp. 96189–96200, 2021.

[9] T. R. Gadekallu, and Q. Gao, “Design of telemedicine information query system based on wireless sensor network,” *EAI Endorsed Transactions on Pervasive Health and Technology*, SJR, vol. 8, no. 4, 2022.

[10] S. P. Rao, N. Minckas, M. M. Medvedev, D. Gathara, Y. N. Prashantha *et al.,* “Small and sick newborn care during the COVID-19 pandemic: A global survey and thematic analysis of healthcare providers’ voices and experiences,” *BMJ Global Health*, BMJ Global Health, vol. 6, no. 3, 2021.

[11] S. Shukla, S. Thakur, S. Hussain, J. G. Breslin and S. M. Jameel, “Identification and authentication in healthcare internet-of-things using integrated fog computing based blockchain model,” *Internet of Things*, Elsevier, vol. 15, 2021.

[12] P. Singh, R. Kaur, J. Rashid and S. Juneja, "A fog-cluster based load-balancing technique," *Sustainability*, MDPI, vol. 14, no. 13, 2022.

[13] B. Mamta, B. B. Gupta, and M. D. Lytras, "Fog-enabled secure and efficient fine-grained searchable data sharing and management scheme for IoT-based healthcare systems," *IEEE Transactions on Engineering Management*, Springer, vol. 10, pp. 59906–59918, 2022.

[14] A. Mukherjee, D. De and S. K. Ghosh, "FogIoHT: A weighted majority game theory based energy-efficient delay-sensitive fog network for internet of health things," *Internet of Things*, Elsevier, vol. 11, 2020.

[15] J. P. Queralta, T. N. Gia, H. Tenhunen and T. Westerlund, "Edge-AI in LoRa-based health monitoring: Fall detection system with fog computing and LSTM recurrent neural networks," in *Proc. of TSP*, IEEE, Budapest, Hungary, pp. 601–604, 2019.

[16] J. Kharel, H. T. Reda and S. Y. Shin, "Fog computing-based smart health monitoring system deploying LoRa wireless communication," *IETE Technical Review*, vol. 36, no. 1, pp. 69–82, 2019.

[17] R. Saha, G. Kumar, M. K. Rai, R. Thomas and S. J. Lim, "Privacy ensured e-healthcare for fog-enhanced IoT based applications," *IEEE Access*, vol. 7, pp. 44536–44543, 2019.

[18] P. H. Vilela, J. J. Rodrigues, P. Solic, K. Saleem and V. Furtado, "Performance evaluation of a fog-assisted IoT solution for e-health applications," *Future Generation Computer Systems*, Elsevier, vol. 97, pp. 379–386, 2019.

[19] S. S. Vedaei, A. Fotovvat, M. R. Mohebbian, G. M. Rahman, K. A. Wahid *et al.,* "COVID-Safe: An IoT-based system for automated health monitoring and surveillance in post-pandemic life," *IEEE Access*, IEEE, vol. 8, 2020.

[20] H. A. Khattak, H. Arshad, S. ul Islam, G. Ahmed, S. Jabbar *et al.,* "Utilization and load balancing in fog servers for health applications," *EURASIP Journal of Wireless Commununication Networks*, Springer, vol. 91, pp. 1–12, 2019.

[21] G. J. Lakshmi, "Cloud-based IoT smart healthcare system for remote patient monitoring," *EAI Endorsed Transactions on Pervasive Health and Technology*, SJR, vol. 7, no. 28, 2021.

[22] L. M. Dang, M. Piran, D. Han, K. Min and H. Moon, "A survey on internet of things and cloud computing for healthcare," *Electronics*, MDPI, vol. 8, no. 7, pp. 768, 2019.

[23] M. Hajvali, S. Adabi, A. Rezaee and M. Hosseinzadeh, "Software architecture for IoT-based healthcare systems with cloud/fog service model," *Cluster Computing*, Springer, vol. 25, no. 1, pp. 91–118, 2022.

[24] H. Gupta, A. V. Dastjerdi, S. K. Ghosh and R. Buyya, "Ifogsim: A toolkit for modeling and simulation of resource management techniques in the internet of things, edge and fog computing environments," *Software: Practice and Experience*, vol. 47, no. 9, pp. 1275–1296, 2017.

[25] P. N. Srinivasu, M. F. Ijaz, J. Shafi, M. Woźniak and R. Sujatha, "6G driven fast computational networking framework for healthcare applications," *IEEE Access*, vol. 10, pp. 94235–94248, 2022.

[26] K. N. Tun and A. M. M. Paing, "Resource aware placement of IoT devices in fog computing," in *Proc. of ICAIT*, Yangon, Myanmar, pp. 176–181, 2020.