IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 21 August 2022, accepted 6 September 2022, date of publication 12 September 2022, date of current version 19 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3205592

RESEARCH ARTICLE

Message Scheduling in Blockchain Based IoT Environment With Additional Fog Broker Layer

ISRAR AHMAD, SAIMA ABDULLAH, MUHAMMAD BUKHSH, ADEEL AHMED, HUMAIRA ARSHAD, AND TALHA FAROOQ KHAN

Department of Computer Science, Faculty of Computing, The Islamia University of Bahawalpur, Punjab 63100, Pakistan

Corresponding author: Adeel Ahmed (adeelmcs@gmail.com)

ABSTRACT Recently researchers and companies have shown significant interest in merging blockchain and the Internet of Things (IoT) to create a safe, reliable, and resilient communication platform. However, determining the proper role of blockchain in existing IoT contexts with minimum implications is a challenge. This work suggests a message schedule for a blockchain-based architecture with two access-level setting filters for incoming messages: critical and non-critical. The proposed work of the researchers divides the fog layer into two parts: action clusters and blockchain fog clusters. Similar to the three-layered IoT architecture, the action cluster and the main cloud data center work together for critical message requests. The blockchain fog cluster is dedicated to only the blockchain application's requirements. In the fog layer, a fog broker is used to schedule critical and non-critical messages in the action and blockchain fog clusters, respectively. The proposed technique is compared to the existing Dual Fog-IoT architecture. The solution is also tested for fog and cloud computing resource utilization. The findings demonstrate that this architecture is feasible for varying percentages of receiving critical and non-critical messages. In addition to the inherent benefits of blockchain, the suggested paradigm reduces the system loss rate and offloads the cloud data center with minimal changes to the existing IoT ecosystem.

INDEX TERMS Internet of Things (IoT), message scheduling, wireless sensor networks, fog computing.

I. INTRODUCTION

The Internet of Things (IoT) is a remarkable revolutionary technology that connects and empowers things to make independent decisions in a smart environment. The modern technology of electronic gadgets and communication technologies has aided in achieving an unexpectedly quick development in its expansion [1]. As a byproduct of IoT, there are billions of gadgets connected to the Internet [1], [2]. It is a reality that with the development of novel hardware and software technologies, the growth of IoT is running beyond predictions made by the business and researchers previously. The engagement of a third party to maintain data in a Centralized Datacenter (CDC) [1], [3] has raised numerous key concerns, and these issues may be acting as a barrier to achieving its long-term goals.

In 2006, Cisco unveiled Fog computing [1], [4], which brings processing capabilities to the edge of the network to solve the difficulties of scalability, latency, cost, and energy consumption that come with IoT design. The fog layer exists at the network's access level to alleviate storage load from a data center and respond to requests with low late. The industry is currently leveraging the three-layer IoT architecture [5]: The device layer contains sensors, actuators, and smart devices; layer two is the fog layer, which provides a quick reaction to critical applications and is made up of devices such as smart gateways, routers, and dedicated fog computing devices; and layer three is data center, which is made up of fog devices connected to cloud gateways. Implementation of a fog layer into cloud infrastructure is expected to decrease latency [2], [6] however, a lack of trust may persist as

The associate editor coordinating the review of this manuscript and approving it for publication was Chunsheng Zhu.

a challenge to the IoT ecosystem. Notably, blockchain features for reliability and transaction transparency have gained interest and appreciation within academia and industry.

Blockchain was introduced in 2008 as a technological breakthrough [3], [7], and it has begun to affect industries including economics, e-healthcare, e-finance, mortgage lending, e-voting, production processes, home automation, and IoT [4], [8]. A blockchain is a decentralized and distributed ledger technology that forms a record of currency or other token-based transactions and contracts which are cryptographically signed. Whenever a block of data is verified and added to the blockchain, it is a data structure that is propagated over a distributed network, similar to a linked list. Due to the numerous copies available in the network, once a block is added to the blockchain, it cannot be tampered with.

The main characteristics of blockchain technology are tamper-evident, safe, maintaining anonymity, and allowing for the creation of a stable network with no downtime [5], [9], while the same characteristics of the currently available IoT ecosystem have a primary concern. As a result, combining these two technologies might be a viable option for meeting the demands of the ever-expanding IoT network. The Trusted IoT Alliance now known as Industrial Internet Consortium was formed by a group of well-known organizations to develop a trustworthy IoT ecosystem employing blockchain. Several systems, including Ethereum, Hyper ledger Fabric, Multichain, Litecoin, Lisk, Quorum, and HDAC, are attempting to reduce the complexity of blockchain to facilitate integration with resource-restricted devices in IoT environments [6], [10]. The consortium also has the purpose of improving the capabilities of the fog layer to enable decentralized IoT technology.

IoT faces privacy and security issues [7], [11] which can be addressed by incorporating blockchain technology. The following key blockchain characteristics [8], [12] are considered instrumental in addressing privacy and security:

- Decentralization- ensures that no single resource controls the entire system. All participating nodes can use their resources to avoid abundant incoming traffic, which eventually solves the problem of a single point of failure and reduces the delay. The decentralized system ensures the system's stability and scalability.
- Inherent Anonymity- protects respondents' privacy by allowing them to share information that cannot be traced back to them. This is usually appropriated among IoT applications when the primary goal is to protect the user's identity.
- Security- tells that the blockchains records are all individually encrypted. Encryption offers an additional degree of protection to the entire blockchain network process. There is no centralized authority which is why adding, updating, or deleting data on the network is not simple.

Message scheduling is a process in which action is carried out by a distributed system's broker (scheduler) mechanism [9], [13], it makes use of message contexts or any other type of information that may be considered according to their priorities. Messages are categorized in the proposed system based on their features and quality of service (QoS) specifications, and applications are divided into two categories: critical and non-critical. An IoT ecosystem can provide different types of services. The Service-Oriented Architecture (SOA) [14] specifies how these services are represented and communicated [10], [15]. SOA is an architectural strategy that improves the service delivery effectiveness of current traditional systems while keeping their most significant characteristics. This technique has attracted the interest of business groups due to its adaptability, particularly in the creation of world-leading services and applications of cloud computing and IoT. New protocols, communication technology, and gadgets are being explored and deployed to provide a sustainable connection among various SOA services. This allows these massive physical world objects to communicate and interact with their surroundings leading to growing computational capacity.

Generally, IoT data sets are sent to cloud services for processing. Time-sensitive IoT applications [11], [16], on the other hand, cannot withstand the considerable latency that data may face when transferred to the cloud. Fog computing-based alternatives for these types of applications are becoming increasingly appealing because of the reduced latency. Growing prevalence of fog base stations, the researchers present in this work a framework for QoS-aware fog service provisioning, which enables IoT application activities to be scheduled on a fog broker. To facilitate IoT applications in meeting their QoS requirements, a fog broker component can apply various scheduling strategies [15], [17], [18], [19], [20]. The simulation results demonstrate that by employing a few basic tactics, it is feasible to maintain low application latency and spread the processing load throughout the fog nodes of the cluster.

The current studies have tended to link blockchain with IoT, and these can be categorized into two types. That one is a complete transfer of IoT to the blockchain, in which all sensor-embedded devices are connected directly with one another without the need for an intermediary. Products like EthEmbeded [16], Ethraspban [17], Raspnode [18], and Bitmain [19] are illustrations of the existing devices. However, a complete transition of IoT to a blockchain is not sustainable since blockchain mining is a complex task that necessitates high computational power, which is currently provided by the application of specific integrated circuit chips [20] and it is difficult to implement blockchain on resource-constrained gadgets in an IoT ecosystem [21]. The second class focuses on improving one of the three levels of the existing IoT ecosystem with a new layer devoted to running blockchain protocols.

The basic goals are transparency, trust, and traceability, and also it eliminates the need for third-party and central authority participation. As a result, data storage is maintained in the fog in a decentralized manner. Three nodes are used to hold the blockchain in the solution, but this number may be changed depending on the need. It adds to the expense of data storage, but it can aid in the implementation of blockchain in an effective way. Since immense volumes of data transported across networks are more likely to trigger security concerns, fog computing decreases the amount of data transferred back and forth across the cloud, lowering latency as a result of local processing while limiting security threats.

The major contributions to this article are as follows: In this context, a fog-level blockchain connection with cloud mining is presented without affecting the conceivable architecture of the existing IoT platform. The suggested architecture also distinguishes between critical and non-critical conditions, moreover, it handles emergency messages in the action cluster of the fog layer, while non-critical messages will be tackled using Blockchain technology. At the fog level, a fog broker is introduced to schedule messages based on critical and non-critical messages.

II. RELATED WORK

A Dual Fog IoT architecture is presented in [10], in which the fog layer is divided into Fog-Mining Clusters (FMC) and Fog-Cloud Clusters (FCC). Three pre-defined settings are defined in this solution: Real-Time (RT), Non-Real Time (NRT), and Delay Tolerant Blockchain (DTB). The Access Point (AP) receives messages from the device layer and filters them based on the parameters indicated above. The FCC receives RT and NRT requests through the AP. As a result of the speedy response, NRT moved even faster towards the cloud layer. DTB requests are held in the AP's local memory until they surpass the block size, at which point the blocks are transmitted to FMC for further blockchain processing. The fog layer handles the blockchain, processing, and storage in this approach. As a result, massive resources are required at fog to process and store the blockchain, message segregation is performed on the access point. On contrary, our solution handles it at the fog broker because access points have limited resources such as processing power, memory, and storage.

In the researchers' solution, the fog broker performs message segregation into critical and non-critical messages, and it also performs the block formation for blockchain. If all the fog broker's designated tasks try to be performed via an access point, it may overburden. The researchers are getting mining services from the cloud, but the above solution mining task is performed on fog, which may increase the operating cost of the network. A blockchain-based IoT architecture is presented in [21] and [22] for optimal data management in a resource-constrained IoT context. The suggested method trains IoT devices to transmit optimum transaction rules using a deep reinforcement learning algorithm. For blockchain deployment, a cloud layer is used for blockchain processing and storage. This system uses deep learning to help with data management, but it ignores notifications that require immediate attention. Second, data access delays may be experienced as a result of cloud storage and blockchain processing.

The network edge is a resource-constrained area, whereas blockchain implementation necessitates a large amount of computational power. Authors in [22] and [23] have presented the notion of accessing blockchain services from the cloud. The suggested proposal is a viable solution for public blockchains, but it gives full accessing power in the hands of a third party, which is undesirable in the case of a private blockchain. The blockchain must be store on an IoT network or in fog in private blockchain systems. The suggested IoT-based architecture in [23] and [24] focuses on message scheduling in IoT clusters. Each group has a designated broker that collects data from the nodes of its members and transmits it to the sink. A scheduler is built at the broker level to determine which message will be transmitted first. Compared to traditional LEACH, the major considerations for selecting a broker are residual energy and distance. The suggested design has a longer network lifetime, lower overall energy dissipation, and faster reaction time.

The authors of this study developed a GMM (Group Message Management) system to efficiently coordinate the delivery of alerts from IoT devices to client end devices [24], [25]. The scheduling module divides the customers into groups based on their requirements and assigns each group an ideal period. The caching module reduces the number of alerts required to group client requests by setting the maximum age value. To reduce bandwidth and energy, the suggested module aggregates alerts from multiple IoT devices requested by the same group request. The integration of edge and cloud infrastructure with IoT to facilitate its execution and demanding computing applications has received attention recently. In terms of security, platform independence, multiple application execution, resource management, and many real-world frameworks strive to provide such integration. The fog integration framework was suggested in this study. It helps the developers to create IoT applications and lets users execute redundant applications at the same time while managing resources [25], [26]. On IoT fog gateway, [26], [27] presented QoS scheduling. Critical and non-critical requests are categorized in this solution based on priority and specified settings. Messages from IoT devices are scheduled from a single queue to various queues based on their priority using a modified version of the Hierarchical Token Bucket (HTB) scheduling method. Researchers in [21] and [22] have proposed the destination prediction algorithm on the Internet of Vehicles (IoV) to predict the location of any vehicle using machine learning. First of all, a real-time prediction framework is proposed to explore the location of a vehicle while traveling. The benefit of the prediction scheduling algorithm is that it chooses the most suitable service provider for resources. In this proposed solution, fog services are used to process and store location-based information. When data is acquired from a network cluster, the use of the Internet of Things raises various security concerns. IoT security, data collection integrity, and data management may be improved by Blockchain technology. In [27] and [28] authors have proposed a context-aware technique for on-chain data allocation in a blockchain-based IoT system. Furthermore, this system is based on fuzzy logic that was developed for data controllers to calculate the rating of allocation value requests while taking into account numerous context elements such as data, network, and quality. To handle the security risks associated with IoT-based infrastructure, such as confidentiality, integrity, availability, and certification, this study proposes a blockchain-based smart home gateway network to protect against future smart home gateway threats. The network is divided into three layers: device, gateway and cloud. To avoid the problem of possible assaults, blockchain is used on the gateway layer, where data is transferred in the form of blocks [28], [29].

Blockchain technology is not only used for cryptocurrency but also in many other fields like government, healthcare, the property market, etc. The article [1], [4] is a review of the literature on how blockchain in IoT may impose many security and privacy issues and challenges. Moreover, it recognizes five key mechanisms with design considerations and challenges that should be considered during Blockchain implementation for IoT-based architectures. This research also considers the holes that obstruct creating a secure blockchain context for IoT. Data blocks are gathered and packetized from a variety of sensors located in various locations, and data is transported over numerous networks in IoT. This study presents a ZigBee-based packetized system for IoT devices. ZigBee is an open global standard designed to meet the demand for low-power, low-cost wireless IoT networks. Data from the sensor is collected, organized, and packed into a packet for transmission in the proposed system. The performance of these methods is addressed, and the implementation of the system is given to show how they may be validated and verified [29], [30]. A blockchain is a distributed and decentralized ledger that stores data in the form of transactions in the form of blocks. The blockchain also ensures that accepted transactions are stored in a tamper-proof manner. The place of blockchain storage in IoT echo system is a challenging task in the implementation of blockchain technology. The hosting platforms for fog and cloud are evaluated in [30] and [31] study.

Blockchain technology is employed in a collaborative manufacturing network because of its security and transparency. Multiple firms participate in this partnership to take benefit from pooled manufacturing. In [31] and [32] a resource scheduler is utilized to assign production to a physical machine after determining whether its capacity is satisfied by available resources. This entire procedure is automated via a smart contract. The method is employed in consensus evidence of authority. IoT devices are unable to supply the essential resources for the blockchain since it is a resource-intensive procedure. A strategy is provided for mobile blockchain, and mining services are conducted using edge computing for offloading mining jobs onto the blockchain. To provide mining services for mobile blockchain, edge resources are properly handled. For the purpose of evaluation, a numerical analysis of the suggested model's effectiveness and efficacy is offered [32], [33]. IoT's rapid rise, as well as the explosion in the quantities of data created by smart devices, has resulted in data outsourcing. However, in order to handle such a massive data storage site, centralized data centers, such as cloud storage, cannot afford to transmit data from an untrustworthy source. The article [33], [34] offers a novel blockchain-based distributed cloud method with software-defined networking (SDN) to overcome some vital problems, allowing control fog nodes at the network's edge to meet the appropriate design criteria. The suggested concept includes a distributed cloud infrastructure based on blockchain technology that provides low-cost, secure, and on-demand access to the cheapest architecture in an IoT-based network.

Light Chain [34], [35] a resource-efficient lightweight blockchain structure described by the authors of this study, is ideal for power-constrained IoT devices. They provide green computing to encourage IoT device support, as well as Light Block, a lightweight data technique that simplifies broadcast data contents and also creates a unique unrelated block offloading filter to keep the blockchain ledger from expanding endlessly while maintaining blockchain traceability. To complete a job, each procedure requires computing. Scheduling is the process of assigning a job to a resource for computation. A privacy-aware, upgraded blockchain-assisted task scheduling protocol is presented to determine the appropriate virtual resource assignment for work. In terms of efficient resource assignment, privacy, message exchange, and the outcomes of this system are better. To get the most out of cloud computing, the article [35] approach employs blockchain as a service. The suggested technique [36] has shown to be a feasible solution to improve wireless sensor network trust and dependability. Reference [36] the study proposes a unique technique for seamless network communication that extends the LEACH protocol with improved network availability, fault tolerance, and energy-efficient message scheduling. By eliminating recurrent sensing, consolidation, and message scheduling operations, this strategy not only handles the problem and promotes availability, but also saves energy for non-broker and broker nodes.

With the support of a decentralized mechanism to govern edge nodes' task execution with heavy resources to reduce task delay, the suggested LAAECHA [37] strategy was developed. The edge network's availability and dependability are ensured by the high availability system, in comparison to traditional cluster-based techniques for group construction with dispersed mode execution.

In [38] the researcher suggested a QoS message scheduling method that is more oriented on service provisioning with the objective of differentiation strategy. Messages are divided into two categories: high priority (HP) and best effort (BE), with the latter transmitting all other non-critical data. The goal is to allow IoT networks to distinguish between mission-critical and non-mission-critical communications, achieving the best possible balance using a cross-layer design process that includes network-layer routing and application-layer QoS-aware message scheduling.

The authors [39] studied and evaluated published articles that combined BC and FC techniques in the survey. It divided the studies into classes based on their category, domain, publication year, BC role, consensus mechanism, and layer wherein the BC was implemented. As a result of the discussion and examination of the publications, they came up with numerous important observations, properties, and outstanding issues surrounding the BC-FC integration.

Research Model [40] provides a load balancing method that accounts for changing resource conditions in fog contexts and moves job requests from one scenario to another where it is most advantageous. The suggested framework accomplishes its goal in two steps: first, it determines whether relocation is viable, and then it chooses a job for relocating and shifts it to a different environment. This framework is custom-made for fog situations, where load balancing is a critical component for maximizing resource efficiency, and bandwidth, and achieving the desired level of service (QoS). The studies in the article [41] discuss the fog computing environment, employ quantization datacenter security and introduce a suitable security-based service broker policy (SbSBP) to allocate the best datacenter(s) to serve users' demands based on cost, time, and security requirements. The concept of reconfigure ability has been included, taking into account the dynamic behavior of fog computing.

The authors of [42] have evaluated the task and resource scheduling problems of multiple tasks for a single application within the DCC environment, taking into account task dependencies and user mobility, and they have introduced a greedy task graph partition GTGP load balance algorithm, in which the task scheduling process is facilitated according to the device computing capabilities with a greedy optimization approach to minimize the tasks communication cost. Furthermore, they create a framework that functions as an SDN and manages the shifting process in a centralized manner. They also develop a compute-intensive software platform for use in the DCC architecture, which primarily consists of infrastructure-based cloudlets, mobile cloudlets, and cloud.

The researchers [46] investigates a trustworthy distributed audit approach for cloud job scheduling, as well as the formulation and construction of a blockchain-based cloud work scheduling system. The cloud task scheduling information is protected and recorded by the system. The blockchain methodology is used to create immutable records, with blocks being formed for each data record that is checked for transparency, authenticity, and validity. The schedule of workloads in the clouds cluster may be regulated in a consistent way using the structure suggested in this research, while possible attacks and information breaches can be prevented at the same time. The system has been implemented and its performance has been reviewed by the researchers. The results indicate that the model is satisfactory.

The remarkable development of social interactions across IoT entities has resulted in a social relationship explosion, which has resulted in computational and communication constraints. The topic of social relationship expansion in IoT is examined in [47], and the researchers show that the nascent ASI has the ability to address the problem. Unlike traditional AI, ASI is combined with computer and communication techniques, allowing it to deal with the explosion of social ties from the standpoint of social computing. IoT devices will be able to use social context to improve. These devices offer and adapt the material. These devices also give thanks to social-centered fusion computing and communication.

The movement of computational from the cloud to the edge of the network is discussed in [42] and [48]. Fog computing works closer to the end-user, on the network edge, providing precise service delivery with a fast reaction time, eliminating delays and network faults that might disrupt or delay the decision-making process and the delivery of healthcare services. The benefits of combining IoT with fog computing are demonstrated through an architectural model and a series of use cases. The authors [43], [49] to lessen the strain on consumers and electricity-producing systems, a three-layered approach cloud is enabled and fog architecture is proposed. The fog server layer is connected to the end-user layer by clusters of buildings. The fog layer serves as a bridge between the end-user layer and the cloud layer. Three load balancing techniques are employed for resource allocation: Round Robin (RR), throttled, and the suggested Particle Swarm Optimization with Simulated Annealing (PSOSA).

The major goal is to improve the performance of the cloud environment by lowering overall costs by routing incoming requests to key nodes. Using a cloud analysis simulator, the article [44] provides an empirical evaluation of both load balance and service broker strategies. The goal of the analysis is to look at three alternative load balancing algorithms and see how they behave (Round Robin, Throttled, and Active Monitoring).

Table 1 presents a comparison of the proposed solution with already proposed literature with respect to the implementation, processing, and storage of the blockchain in the IoT environment. It also compares the different solutions either to perform the differentiation of messages into emergency(critical) and delay-tolerant messages or not. In the proposed solution, the processing is performed on cloud to save the processing resources and storage is done on fog to avoid the delay.

III. PROPOSED FRAMEWORK

As illustrated in Figure 1, blockchain is implemented in the segregation of processing, i.e., mining and storage, at distinct tiers of the proposed IoT three-layered architecture [48], [50]. To prevent placing control of the blockchain in the hands of a third party, blockchain storage is done at the fog layer. Not only does the proposed architecture use blockchain for segregation, but it also handles critical and non-critical messages separately. Fog is divided into two parts: The Fog Action Cluster (FAC), which response quickly to critical messages, and the Fog Blockchain Cluster, which handles blockchain communications. This section contains in-depth descriptions of each layer.

TABLE 1. Comparison of existing work with respect to blockchain message scheduling.

Reference	Implementation of solution on architecture	Blockchain processing at	Blockchain Storage at	Emergency and Delay tolerate message differentiation	Remarks
[43]	IoT 3-level Architecture	Fog Layer	Fog Layer	Yes	• High Computational power required in fog • Message differentiation is performed at the device layer, so the more powerful equipment required at this layer
[18]	IoT 3-level Architecture	I. IoT Layer II. Fog Layer III. Cloud Layer	I.IoT Layer II.Fog Layer III.Cloud Layer	No	• All IoT Layers have to upgrade to implement Blockchain • No differentiation of messages into emergency and delay tolerate. • At each layer, extra storage and computation power have to add which will increase the cost.
[43]	IoT 3-level Architecture	Fog Layer	Fog Layer	No	• No differentiation of messages into emergency and delay tolerate. • At each layer, extra storage and computation power have to add the fog layer
[44]	IoT 3-level Architecture	Fog Layer	Fog Layer	No	• No differentiation of messages into emergency and delay tolerate. • At each layer, extra storage and computation power have to add the fog layer
[45]	IoT 3-level Architecture	Cloud Layer	Cloud Layer	No	• No differentiation of messages into emergency and delay tolerate. • Blockchain storage in the cloud means all data is in the hands of third-party; there may be some vulnerabilities attached to data security.
Proposed Work	IoT 3-level Architecture	Cloud Layer	Fog Layer	Yes	• The message is properly handled with respect to being critical and non-critical • Blockchain storage at fog to keep data secure. • To avoid processing power requirements blockchain processing will be gotten from a cloud. • Message differentiation is performed at fog broker.

A. PROPOSED ARCHITECTURE

The proposed solution structure is based on the three-layered architecture of IoT as shown in Figure 2. The three-layer architecture comprises IoT devices, fog, and cloud layers [49], [51]. The device layer's functionalities are described in this section. The fog layer is made up of an action cluster and a blockchain cluster, as well as a fog broker that schedules messages to the action or blockchain cluster based on requirements. Figure 2, depicts the proposed architecture.

1) DEVICE LAYER

IoT deals with a variety of devices and applications, including the critical and non-critical [50], [52]. The network must respond to a wide range of queries. Critical application requests must be responded quickly, and blockchain has the potential to store significant delay-tolerant requests. For instance, critical communications include fire alarm systems, traffic accident systems, medical crises, etc. If a fire is detected in the building, the fire alarm system disconnects the whole building's electricity and dials an emergency

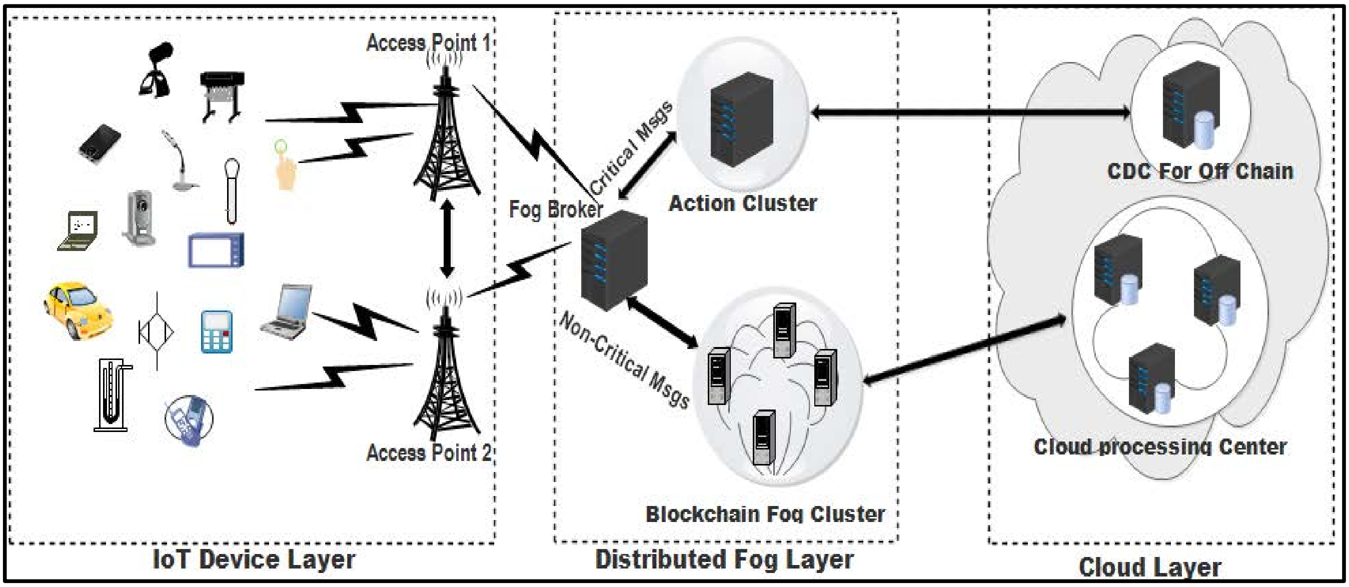

FIGURE 1. Proposed IoT architecture.

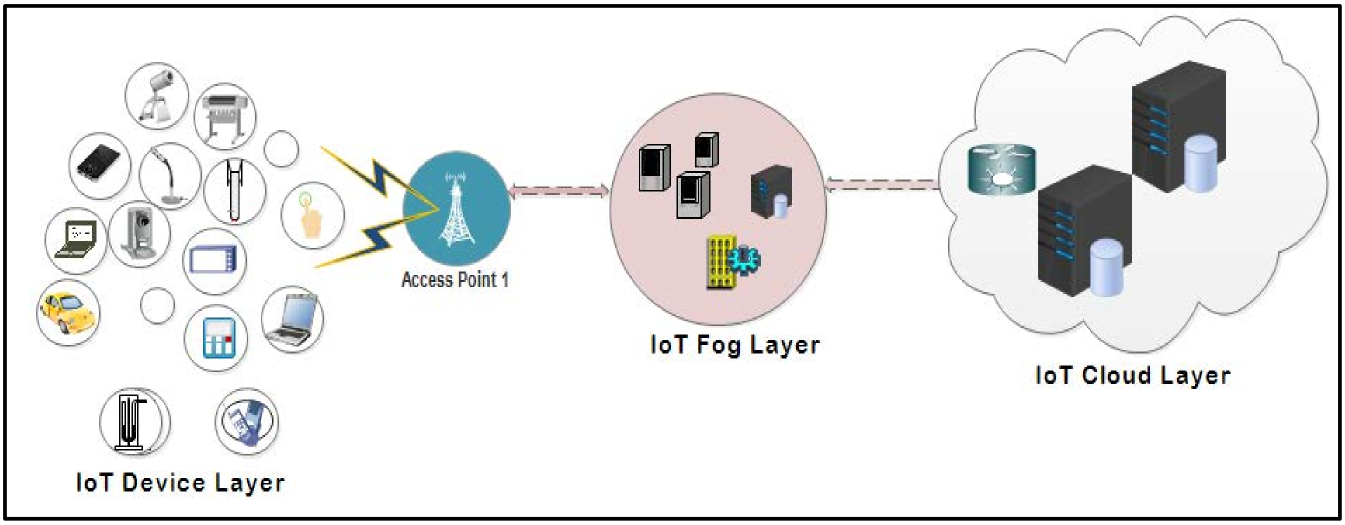

FIGURE 2. IoT existing three level architecture.

number for the fire department and rescue office. Because of the processing delay, critical messages cannot be handled using blockchain, and such notifications must be responded to immediately.

Scientific data analysis, business data mining, business report generators, and other non-critical, delay-tolerant signals require greater processing and storage resources. Non-critical messages are the most suitable candidates for storage in blockchain blocks [51], [53]. In IoT devices, processing, storage, transmission, and battery power are all constrained [52], [54]. Because of the aforementioned limits, implementing blockchain at the device level is very difficult [55]. Blockchain processing is handled by a cloud service, while storage is handled by the fog layer in this system. Some devices, like Bitmain, Raspnode, and Ethraspbian, can run lightweight blockchain client software. In some cases, replacing current IoT devices with blockchain-capable devices is problematic. IoT layer is unaffected by the researchers' suggested method.

2) ACCESS POINT (AP)

The access point is a forwarding device located at the edge of the network. To forward messages to the fog layer for additional processing, all devices obtain the message forwarding services of an access point. As indicated in Figure 3, when a device enters the network, its registry is conducted

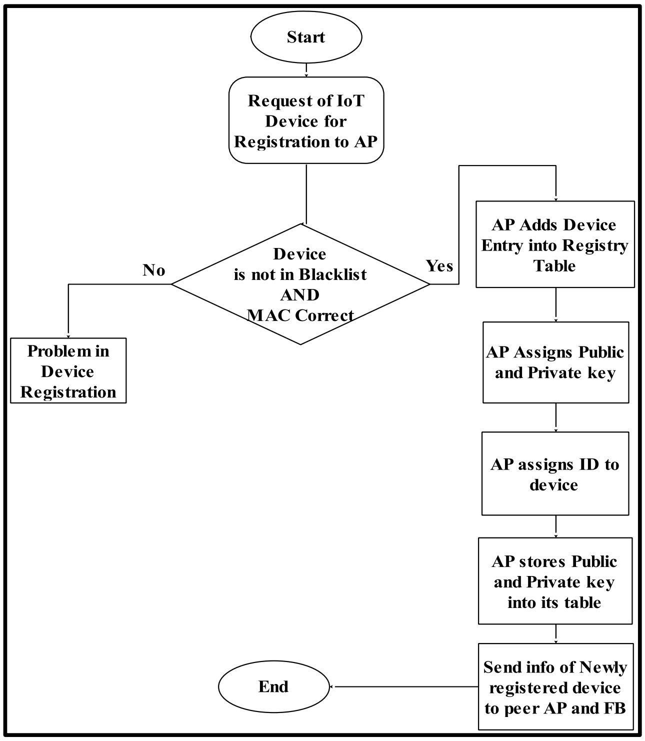
Start
Request of IoT Device for Registration to AP
Device is not in Blacklist AND MAC Correct
No
Yes
Problem in Device Registration
AP Adds Device Entry into Registry Table
AP Assigns Public and Private key
AP assigns ID to device
AP stores Public and Private key into its table
Send info of Newly registered device to peer AP and FB
End

FIGURE 3. Device registration flowchart.

Algorithm 1 Pseudo Code for Message Transmission

```
Receive the message from the selected Active node
Msg$_i$(Msg_header,Msg_Payload,Priority$_i$)
If priority$_i$ == 1// Critical Message
        Put the meg$_{(i)}$ into Q1
        CMCouter++
    Else // non-critical Message
        Put the message into Q2
        NCMCounter++
 End If
flag=false

For
        if (! Empty.Q1)
            flag=true
        else
            flag=false
        If flag=true
            SendMessage() from Q1
            CMCounte–
            CMMSent++
        Else
            SendMessage() from Q2
            NCMCounte–
            NCMSent++
        End If

End for
Procedure SendMessage()
        If Msg$_i$..priority=1
            Send msg$_i$ to action cluster
            Delete Msg$_i$from queue
        Else
    Send Msg$_{i+1}$ Blockchain Fog cluster
    Delete Msg$_{i+1}$from queue
    End If

End Procedure
```

on the access point either automatically or manually by the network administrator [54], [56]. The device sends a network join request to the AP with its MAC address through the automated registry. For future communication, the AP keeps a registry log and issues a public-private key pair and an ID. For synchronization, the AP transmits information about the new device to other APs in the network as well as to the fog broker. Following registration, a device transmits an encrypted message including its private key, MAC address, and ID, i.e., Msg (Message, MAC address, ID).

3) FOG LAYER

The Fog Broker (FB), Action Cluster (AC), and Blockchain Fog Cluster (BFC) are the three primary components of this proposed solution's fog layer. At the fog edge, FB receives all incoming messages and routes them to either AC or BFC based on their QoS requirements. The AC is responsible for important communications, whereas the BFC handles delay-tolerant messages.

a: FOG BROKER

FB has a memory pool where all incoming messages are held until either AC or BFC is selected to send these messages. FB uses preset settings to establish message forwarding rules that take into account application needs. Critical and non-critical message configurations are employed in this approach. Health care applications, traffic control systems, industrial process management, defense systems in the field, and radar systems are examples of critical communications that demand prompt reaction without delay [57].

Non-critical messages provide more flexibility with respect to time constraints. The smart grid, smart farming, selling and buying, online shopping, and supply chain all require high processing power, reliability, confidentiality, authorization, and tamper-proofing. In the traditional blockchain process, each device has its own local memory to hold transactions for block formation, but in the present case, the block is formed on a FB. This architecture implements a publish-subscribe model [56], [58] to forward messages from the IoT layer to the appropriate cluster of the fog layer. A publisher, a broker, and a subscriber are the three components of this model shown in Figure 4. The publisher is a collection of data-generating devices like sensors, computing

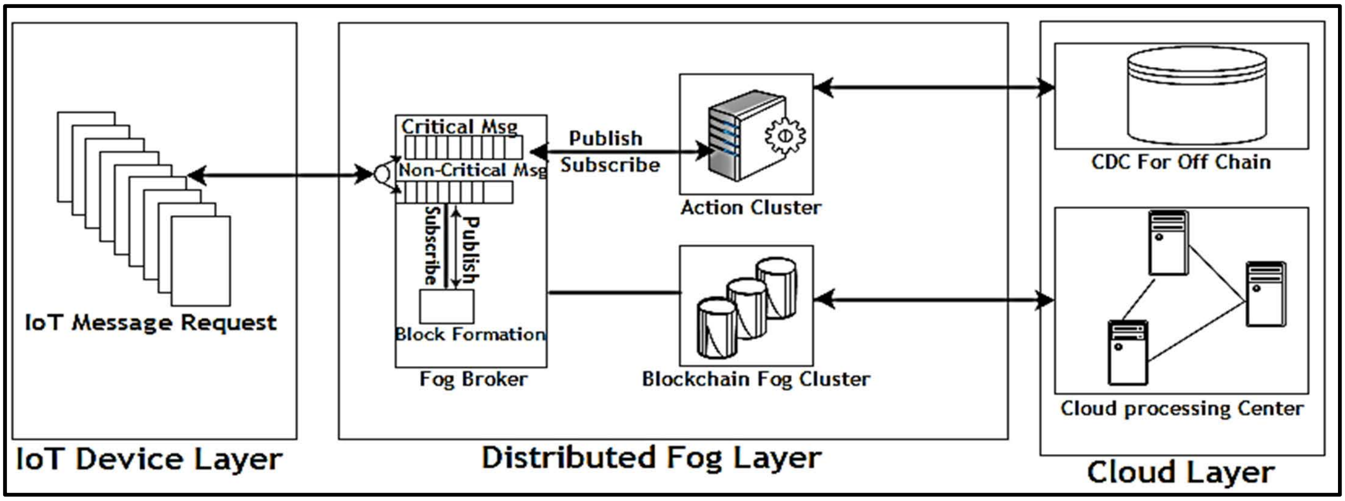

FIGURE 4. System architecture of proposed solution.

devices, or any other object having embedded sensors. A broker is a message filtering device that multiplexes the messages [57], [59] according to the subject subscribed to by some subscribers. A subscriber is a device that receives data generated by the publisher. In the present model, subscribers are AC and BFC that subscribe to critical and non-critical messages respectively.

The proposed algorithm has queue processing and is the main processing operations regarding the time complexity.

One dequeue () operation takes O (1)

To remove N elements from the queue will take O(N) time. Similarly, One enqueue () operation takes O(1) time. To insert N elements in the queue will take O(N) time. Then, T(N) = O(N) + O(N) = 2* O(N).

b: ACTION CLUSTER

The AC is a vital message subscriber. When FB gets a critical notification, it is forwarded to the action cluster, which takes the appropriate action. AC's functionality is comparable to the cloud's Centralized Cloud Data Center (CDC-IoT) architecture.

c: FOG BLOCKCHAIN CLUSTER (FBC)

FBC is a part of the blockchain network; it is responsible for block formation and storage. In this model, blockchain processing is achieved through Mining as a Service (MaaS) from the cloud, and distributed ledgers are stored in fog. In some solutions, devices in the fog are used for solo miner blockchains, and in the end, devices have to be upgraded to a lightweight blockchain wallet. FBC consists of individual nodes called Fog Blockchain Nodes (FBN) and one of these is selected as the Fog Blockchain Cluster Head (FBCH).

(FBN1, FBN2, FBN3…FBNn)

All incoming communications from the AP are received by the FB. In its memory pool, FB has two queues. According to the specified configuration, FB schedules messages and places critical messages in the critical msg queue and non-critical messages in the non-critical msg queue. Message transmission flow is shown in Figure 5. For blockchain processing, the FMC msg queue subscribes messages from non-critical msg queues. It is transmitted to all FBNs when the queue size surpasses the block size. The FBN's FBCH sends blocks to the cloud for mining purposes. The cloud provides mining services for the blockchain. After mining, a block is returned to the FBN. On the other hand, a critical message generated at the device layer reaches the AC. The message is transmitted to the cloud for storage after the relevant process is completed swiftly.

Miners use a variety of mining technologies for this goal, including CPU mining, GPU mining, FPGA mining, mining pools, ASIC mining, and others. The problems can only be solved by trial and error. As a result, in order to find answers rapidly, miners need more processing capacity. Cloud mining entails renting or buying mining equipment from a third-party cloud provider, who is also responsible for maintaining the equipment. Network bandwidth is used to send a block for mining in the cloud after the mining block is received back from the cloud. In this way, blockchain will be stored on fog, which will be accessible via local accesses. In this manner, it can use network bandwidth more effectively. Due to local access from the fog layer, the delay will be minimized.

The Proof of Work (POW) method, often referred to as mining, is one of the most well-known processes for achieving consensus, and miners are nodes that carry out mining. Miners work on difficult mathematical puzzles that need massive processing power. In the present solution, non-critical messages are packed in the form of blocks by the FB, and then the blocks are broadcast to the blockchain fog cluster. One predefined cluster head forward block is sent to the cloud for mining in the cluster. After mining from the cloud, a nonce is received by all cluster nodes. Each node verifies the block hash and if it is correct, it will be included in its blockchain.

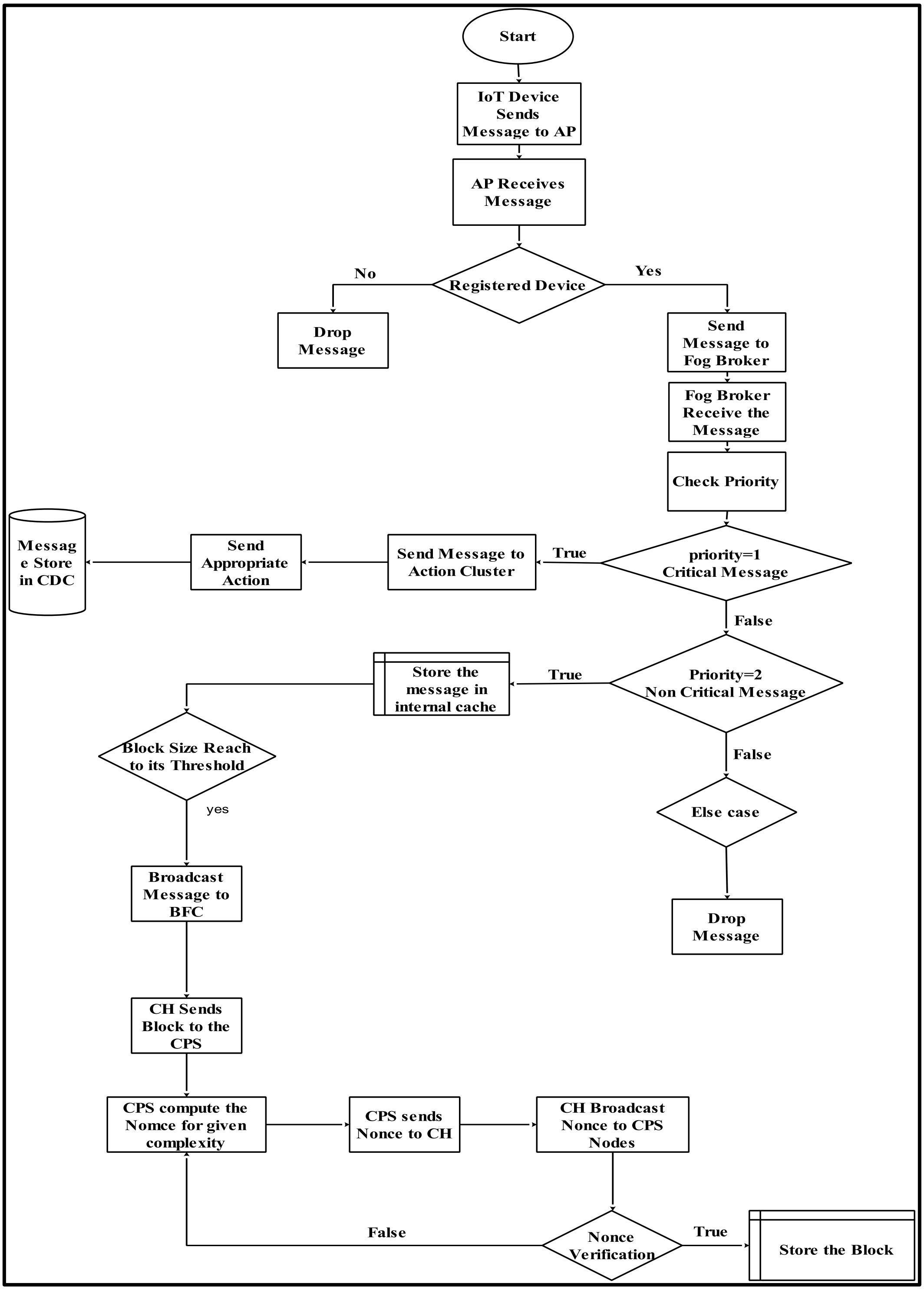
Start
IoT Device Sends Message to AP
AP Receives Message
Registered Device
No
Yes
Drop Message
Send Message to Fog Broker
Fog Broker Receive the Message
Check Priority
priority=1 Critical Message
True
Send Message to Action Cluster
Send Appropriate Action
Message Store in CDC
False
Priority=2 Non Critical Message
True
Store the message in internal cache
Block Size Reach to its Threshold
yes
False
Else case
Drop Message
Broadcast Message to BFC
CH Sends Block to the CPS
CPS compute the Nomce for given complexity
CPS sends Nonce to CH
CH Broadcast Nonce to CPS Nodes
Nonce Verification
False
True
Store the Block

FIGURE 5. Message transmission flow chart.

B. WORKING CONFIGURATION OF THE FOG LAYER

The proposed architecture implements two configurations. In this section, the workings of each configuration are elaborated. Figures 6 and 7 illustrate the corresponding configurations. Configuration 1 is for critical and configuration 2 is for non-critical messages.

By utilizing distributed network design, blockchain not only delivers security, stability, and a trustworthy network

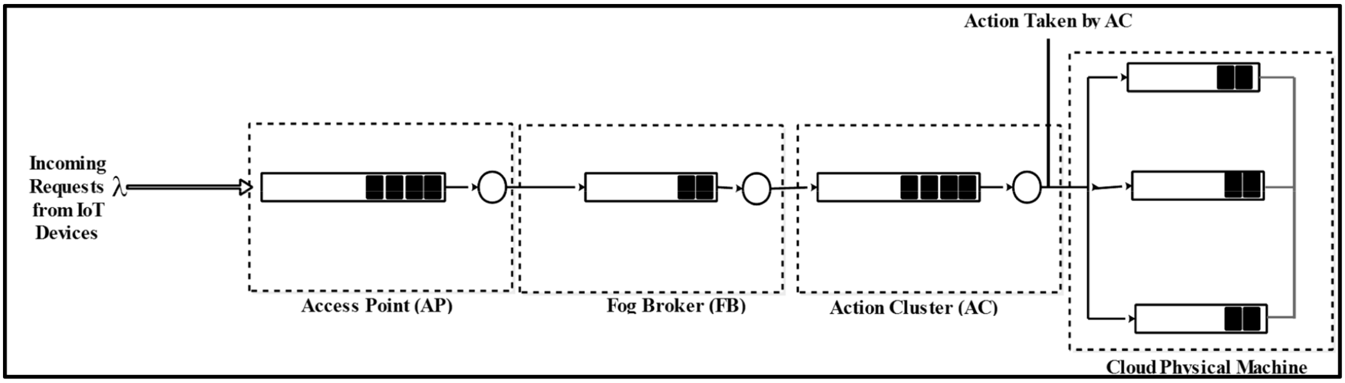

FIGURE 6. Queuing model of critical messages.

but also eliminates the need for a third party. Due to time-consuming mining activities, the blockchain suffers from prolonged latency and delayed transactions. For example, in the case of a critical message requiring immediate response, like a fire in a building, all essential measures must be completed without wasting time.

1) CONFIGURATION FOR CRITICAL MESSAGES

Configuration 1 illustrates the crucial system, as seen in Figure 6. Because a delay in response might cause significant damage, the critical system is responsible for reacting to crucial notifications without wasting time. There may be a slight delay due to some key choices or transmission disruptions; nevertheless, the focus is only on completing the essential activity as quickly as possible. It is to assume that a nuclear reactor's mechanism malfunctions. It must act fast, or else a calamity might arise. The breakdown sensors detect the incident and transmit a message to the access point, which sends the message to the action cluster for immediate response. It could turn off the machine, notify the appropriate department, and sound the alarm to inform people to evacuate the building.

2) CONFIGURATION FOR NON-CRITICAL MESSAGES

As shown in Figure 7, configuration 2 is an illustration of the non-critical system. Due to the processing delay of blockchain, delay-tolerant applications are the best candidates for this system, such as online shopping, scientific data analysis, data mining, banking systems, etc. The message request generated by some trustworthy device is forwarded to the access point and then to the FB after checking the credentials. The FB places such messages in a non-critical msg queue, which holds the transaction until it reaches the block size limit. After reaching the required size, these sets of transactions are sent to the fog blockchain cluster. The fog blockchain cluster head node forwards this message to the cloud service for mining. After calculating the nonce value, the cloud server returns the correct nonce to the cluster head. The cluster head broadcasts the nonce to each node in the blockchain cluster. Each node in this cluster evaluates the received nonce and sends a broadcast to the peer node.

C. MATHEMATICAL FORMULATION OF SYSTEM

1) DESCRIPTION OF THE QUEUING MODEL

The typical Markovian assumptions of inter-arrival and service times are used in this model. The average rate of arrival is λ and the average rate of service is μ. The system's capacity is assumed to be finite, say N. There is just one FB. The queue follows the first-come, first-served principle. When a message enters the queue, it will have to wait for a specific amount of time for the service to begin. With parameters ζ, the waiting times follow an exponential distribution.

TABLE 2. Notations to derive the mathematical model of this problem.

$p_n(t)$	The probability that there are n messages in the system, both waiting and in service, in the transient state.
$p_0(t)$	Empty system probability at time t.
p_n	Probability of n messages in the system in steady-state.
p_0	Probability of no messages in the system in the steady-state.
N	Number of messages in the system, $0 \le n \le N$
N	Maximum capacity of the system.
Ls	Queue size
Lq	Expected queue length
Ws	Waiting time in the system
Wq	Waiting time in the queue

The mathematical model is described by the following assumptions:

1) Messages come one by one to the service facility in a Poisson process with a rate ($\lambda > 0$) and a mean inter-arrival time of $1/\lambda$.
2) Message service times are exponential random variables with rate $\mu > 0$ and mean service time $1/\mu$, $0 < \lambda < \mu$. that are independent and identically distributed.
3) Messages are treated in a first-come, first-served (FCFS) manner.

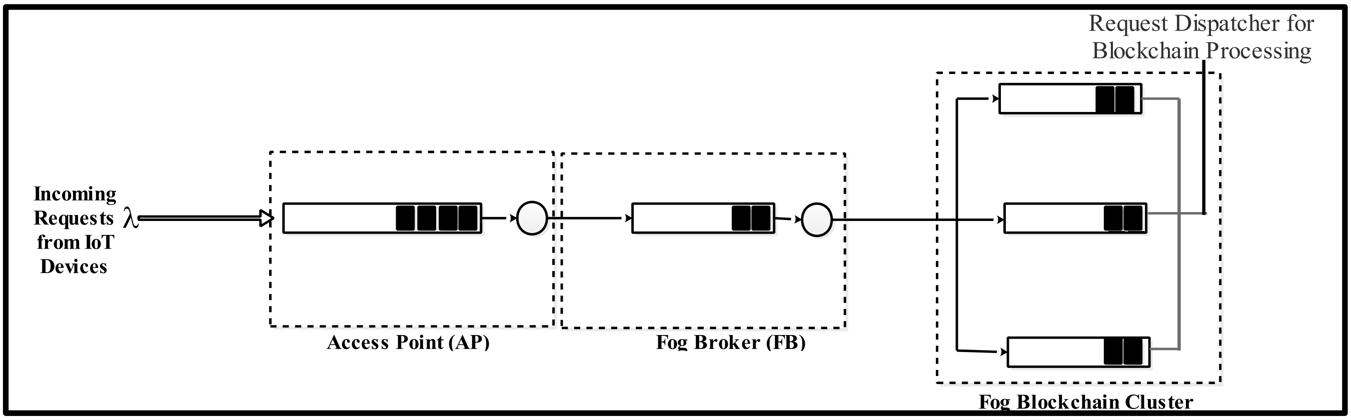

FIGURE 7. Queuing model of non-critical messages.

$P_n(t)$ probability of n messages in the system, 1 message in the service, and n-1 in queues [60], [61]

$$\frac{dp_0(t)}{dt} = -\lambda p_0(t) + \mu p_1(t) \tag{1}$$

$$\frac{dp_n(t)}{dt} = -[\lambda + \mu + (n-1)\zeta p)]p_n(t) + \lambda p_{n-1}(t) \ \forall\, 1 \le n \le N-1 \tag{2}$$

$$\frac{dp_N(t)}{dt} = \lambda P_{N-1} - [\mu + (N-1)\zeta p]P_N(t) \ \ \forall n = N \tag{3}$$

In case of a steady state $\lim_{t-\infty} p_n(t) = p_n$, so $\frac{dp_n(t)}{dt} = 0$ when $t - \infty$

Correspondence of equations 1-3 into steady-state

$$0 = -\lambda p_0 + \mu p_1 \tag{4}$$

$$0 = -[\lambda + \mu + (n-1)\zeta p)]p_n + \lambda p_{n-1} \tag{5}$$

$$0 = \lambda P_{N-1} - [\mu + (N-1)\zeta p]P_N \tag{6}$$

Recursive solution of Eq. 4-5

$$P_n = \prod\nolimits_{k=1}^{n} \frac{\lambda}{\mu + (k-1)\zeta p} p_0 \tag{7}$$

When n=N we get

$$P_N = \prod\nolimits_{k=1}^{N} \frac{\lambda}{\mu + (k-1)\zeta p} p_0 \tag{8}$$

With respect to normalization condition $\sum\limits_{n=0}^{N} p_n = 1$, we will get

$$p_0 = \frac{1}{1 + \sum\limits_{n=1}^{N} p_n \prod_{k=1}^{N} \frac{\lambda}{\mu+(k-1)\zeta p} p_0} \tag{9}$$

Using the above equation derivations following conclusions have been drawn

1. Expected System Size means the total number of messages generated by the IoT network. It also tells the no of messages exist in the waiting queue, no messages in execution process and the no of message in critical and non-critical situation.

$$L_s = \sum_{n=0}^{N} nP_n$$
$$L_s = \sum_{n=0}^{N} n(\prod\nolimits_{k=1}^{N} \frac{\lambda}{\mu+(k-1)\zeta p})p_0 \tag{10}$$

2. Expected queue length means the how many IoT device message can store in queue for execution and waiting interval.

$$L_q = \sum_{n=0}^{N} n(\prod\nolimits_{k=1}^{N} \frac{\lambda}{\mu + (k-1)\zeta p})p_0 - \frac{\lambda}{\mu} \tag{11}$$

3. The expected waiting time in the system means when the message stays at the queue and waits for the allocation of the network resources allocation. In the given below eqs. (12)

$$W_s = \frac{\sum\limits_{n=1}^{N} p_n(\prod_{k=1}^{N} \frac{\lambda}{\mu+(k-1)\zeta p})p_0}{\lambda} \tag{12}$$

4. Expected waiting time in the queue interpret the time required for the allocation of resources. Following equation also determine how much time is required and have to wait for the execution.

$$W_q = \frac{\sum\limits_{n=1}^{N} p_n(\prod_{k=1}^{N} \frac{\lambda}{\mu+(k-1)\zeta p})p_0}{\lambda} - \frac{1}{\mu} \tag{13}$$

IV. SIMULATION SETUP

A. QUEUING MODEL FOR THE PROPOSED ARCHITECTURE

Simulation is performed unconnectedly for comparison of CDC-IoT [62], DualFog-IoT [10], and the proposed architecture. All three simulation resources at the fog layer are identical for the fog layer at CDC-IoT, the fog cloud cluster at DualFog-IoT, and the action layer at the proposed architecture. The comparison of the above-mentioned architectures is revealed before and after blockchain integration. JMT version 1.0.5 is used for simulation modeling on an HP EliteBook with Windows 10 operating system, Intel(R) Core (TM) i5-3320M 2 CPUs @ 2.6 GHz, 8 GB of RAM, and a 500 GB SSD. All incoming requests are forwarded to FB, which sends critical messages to AC and non-critical messages to FBC, P_C is the probability of critical messages, and P_{NC} is the probability of non-critical messages. In Figure 6, a simulation model for critical messages is depicted where critical messages are forwarded to a FB, which takes action locally, and cloud storage is used to store data. In Figure 7, block formation and storage are performed on the BFC, and processing for mining is acquired through cloud service. Configuration 1 belongs to critical messages and configuration 2 is for non-critical messages. In Table 3, simulation parameters are shown that belong to critical messages as per CDC-IoT. Some additional parameters are explained in Table 4, for blockchain integration in IoT. The QoS parameters observed in both models are message drop rate, system utilization, throughput, and response time for several requests in the system.

TABLE 3. Simulation parameters for critical messages.

Parameters	Description	Values
Λ	Request Arrival Rate (Req/s) at AP	300 to 3000
$1/\mu_e$	Mean Fog Broker Computing Service Time (s)	0.005
$1/\mu_g$	Mean Request at Cloud Gateway (s)	0.0003
$1/\mu_c$	Mean Service Time at Cloud	0.02
C_e	The capacity of Request at Fog Broker	∞
C_c	The capacity of Request at Each Cloud Node	500
C_g	The capacity of Request at Cloud Gateway	3000
P_{ac}	Probability of Request Served at Action Cluster	0.6
P_{fbc}	Probability of request Served FBC	0.4
FCC	Number of Stations at Action Cluster	3
Q_e	Queue Policy	FCFS
Q_c	Drop Rule for Action Cluster	Packet Drop

The parameters of critical and non-critical messages are described in Tables 3 and 4, respectively. The Poisson distribution process is the message arrival rate per second at the AP. All queue work adheres to the FCFS model's rules. The queue policy is no longer followed, it implies that as the queue fills up, more inbound messages will be discarded. The block size in this simulation is set at 300.

TABLE 4. Simulation parameters for non-critical messages.

Parameters	Description	Values
P_b	Probability of Request Forwarding to Blockchain	0.4
C_n	The capacity of Request at Fog Broker	∞
$1/\mu_{ch}$	Cluster Head Service Time (s)	0.003
$1/\mu_m$	Mean Service Time for Mining Service from Cloud (s)	0.05
$1/\mu_g$	Mean Request at Cloud Gateway (s)	0.0003
C_m	The capacity of Request at Mining Pool	300
B_{size}	Block Size	300
Q_b	Blockchain Queue Policy	FCFS

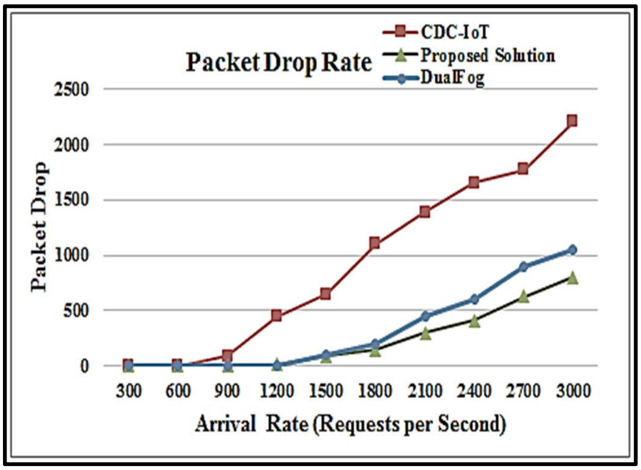

FIGURE 8. Packet drop rate.

V. RESULTS

In this section, results obtained through simulation are compared between DualFog-IoT, CDC-IoT, and the proposed solution. It is obvious that an IoT system having blockchain implementation at any level and in any form decreases the system throughput and also increases the number of messages in the system at any specific time. Along with the disadvantages mentioned above, blockchain has many advantages, such as immutability and the integrity of the untrusted network. Results are obtained 10 times in a scenario by fluctuating the message arrival rate (λ) 300 and increasing the remaining experiment by 300 messages per second to 3000. Results are obtained and depicted in the line chart.

In Figure 8, a comparison of packet drop rates is represented in the cases of CDC-IoT, DualFog, and the proposed solution. The X-axis represents the message arrival rate per second, and Y-axis denotes the packet drop rate. At an arrival rate of 300 and 600 requests per second, the packet drop rate is 0 for all three architectures. But after the arrival rate of 600, CDC-IoT observed a high drop rate. The drop rate of the proposed solution is the lowest compared to CDC-IoT and DualFog. The main reason for the packet drops rate is congestion. It means when a resource is utilized beyond its capacity; extra messages start to drop. In the case of the proposed solution, AP has an unlimited queue length and forwards all messages to FB in the FCFS technique. The proposed solution has the lowest message drop rate.

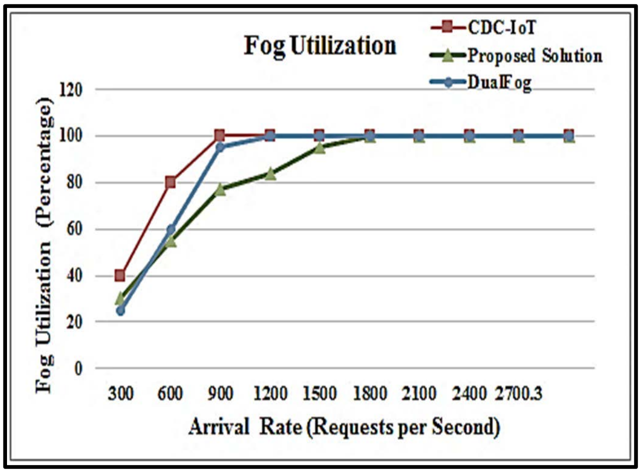

FIGURE 9. Cloud resource utilization.

In the case of CDC-IoT, fog utilization is 100% when the request rate is 600 messages per second. In the case of DualFog architecture, fog utilization is also observed at 100% when the arrival rate reaches 900, but in the case of the proposed solution, when the arrival rate is 1150, it reaches 100% fog utilization. It is proof that, in the case of critical messages, the proposed solution performs better than the other two architectures. In Figures 9 and 10, fog and cloud resource utilization of CDC-IoT, DualFog, and the proposed solution are compared, respectively. In figure 9, fog utilization is equated, and the graph shows that the utilization of fog resources is managed in a more effective way compared with CDC-IoT and DualFog.

At the arrival rate of 300-message per second, 40% of the resources of the fog layer are utilized. 25% and 30% are utilized for CDC-IoT and the proposed solution, respectively. 100% of fog resources are utilized at 900 message arrival rates for CDC-IoT, 1200 message arrival rates for DualFog, and 1800 in the case of the proposed solution.

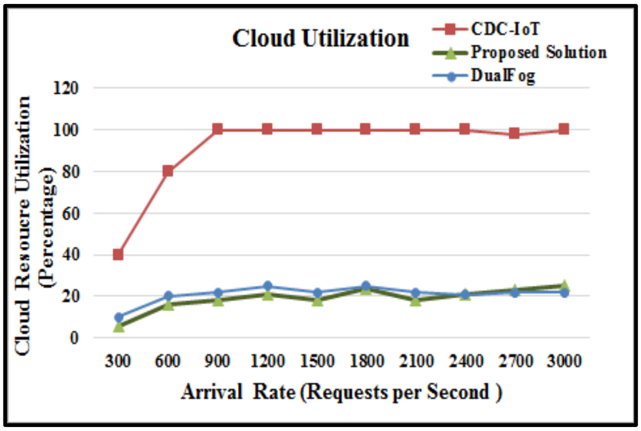

FIGURE 10. Fog utilization.

In Figure 10 Cloud resource utilization is depicted in the graph. CDC-IoT utilizes 40% of cloud resources at a message arrival rate of 300. When the message arrival rate is 600 then 80% of resources are utilized. After 600 messages CDC-IoT utilizes 100% of its resources. In the case of Dual-Fog, cloud resources are utilized far less than CDC-IoT.

At message arrival rate 300 only 10% of cloud resources are utilized at message arrival rate 600, 900, 1200, 1500, 1800, 2100, 2400, 2700, and 300 cloud resource is 20%, 22%, 25%, 22%, 25%, 22%, 21%, 22%, 22% respectively. In the case of the proposed solution utilization of cloud, resources are less than CDC-IoT but a bit greater than Dual-Fog. The reason for fog utilization is due to cloud services of the mining operation. In the proposed solution Blockchain mining is attaining with cloud services i.e., MaaS.

Figure 11 compares CDC-IoT, DualFog, and the suggested approach in terms of system reaction time. The X-axis depicts the number of messages arriving each second, while the Y-axis depicts a time in seconds. When the message arrival rate is 300, the reaction time of the CDC-IoT system is virtually zero. The proposed solution is sandwiched between CDC-IoT and DualFog, with a time of 1.1 seconds. Because CDC-IoT sensing devices collect data from the environment and transfer it straight to the fog layer, scheduling is required at the FB in the proposed system. As a result, there may be some delays. The suggested solution's system reaction time is faster than CDC

IoT and DualFog at a message arrival rate of 400 to 1500 per second. The graph lines of these three solutions are practically parallel after 1500 to 1800 messages per second. In Figure 12, the suggested solutions, CDC-IoT, and DualFog are compared in terms of system throughput. CDC-IoT has the highest throughput of the two remaining solutions. Moreover, the CDC-IoT results are without blockchain implementations. DualFog and the suggested method have almost comparable throughput.

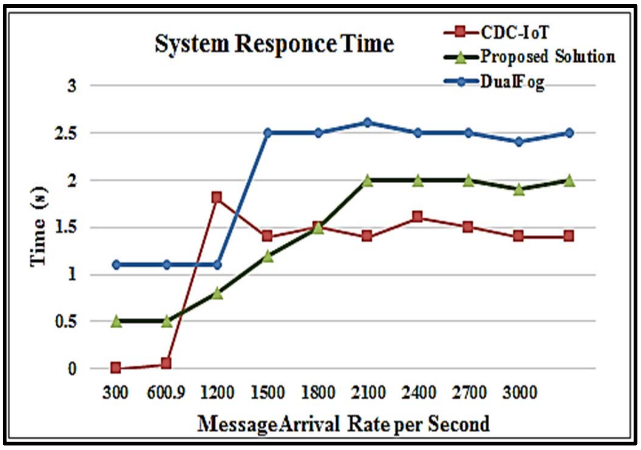

FIGURE 11. System response time.

The throughput graph of proposed solution is slightly below the DualFog line. When the message arrival rate is 300 and 600 per second, the throughput of CDC-IoT is between 300 and 600 per second. When the message arrival

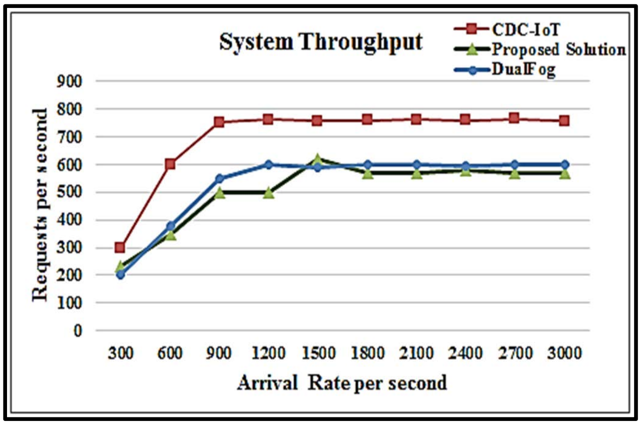

FIGURE 12. System throughput.

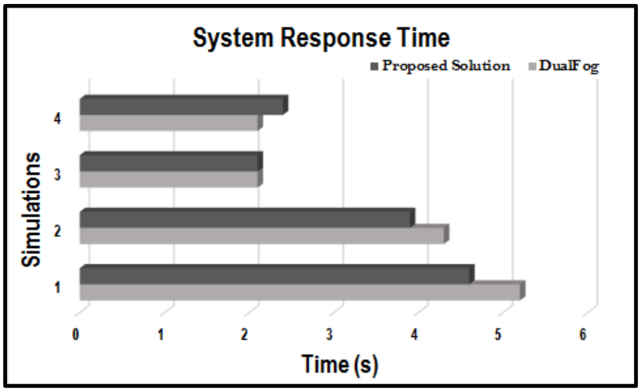

FIGURE 13. System response time.

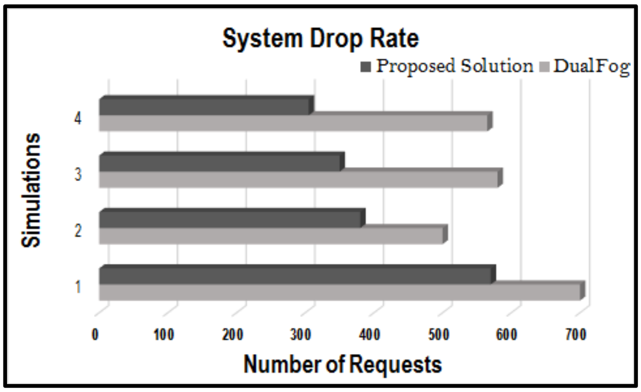

FIGURE 14. System drop rate.

rate is 900 to 3000 messages per second, the CDC-IoT throughput is nearly constant. When the message arrival rate is 300 per second, the throughput with the DualFog solution is 200 per second. Messages arrive at a pace of 900 to 3000 per second, with a fairly constant flow. When the message arrival rate is between 1800 and 3000, the throughput in the suggested method becomes virtually constant before fluctuating according to the message arrival rate. By altering the frequency of receiving critical and non-critical requests at the fog and cloud levels, the IoT model is further evaluated and compared with DualFog forhandling critical and non-critical requests. Critical messages 20% and non-critical 80% are considered in the first simulation; critical 40% and non-critical 60% are considered in the second simulation; critical 60% and non-critical 40% are considered in the third simulation, and critical 80% and non-critical 20% are taken in the fourth simulation. Figure 13 shows the system response time, which clearly shows that the proposed system response time is significantly faster than the DualFog solution when the critical messages are 80% or above, simulation 4 demonstrates

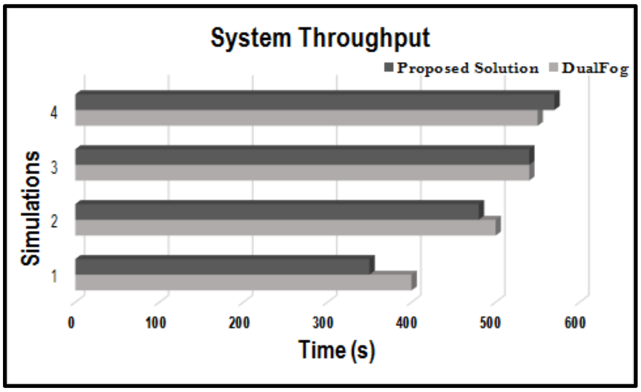

FIGURE 15. System throughput.

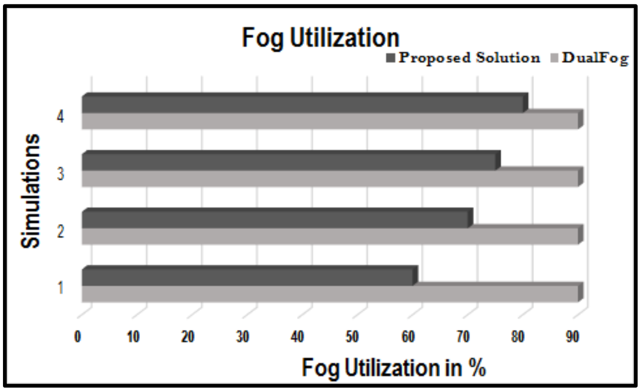

FIGURE 16. Fog utilization.

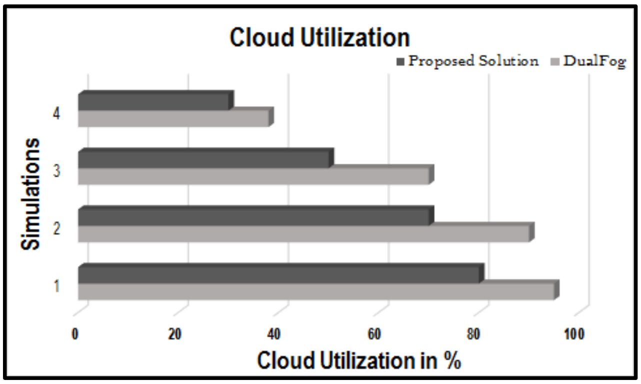

FIGURE 17. Cloud utilization.

the system's response time. The system outperforms DualFog significantly. Figures 14 and 15 depict the system's drop rate and throughput, respectively. In Figure 16, fog utilization for DualFog and the proposed solution are compared. Figure 17 shows cloud utilization. The proposed solution acquires a little bit more cloud resources compare to DualFog because of the mining services being gotten from the cloud.

VI. DISCUSSIONS

It's also worthy of note that all arriving messages from IoT devices in the present simulation model follow a Poisson distribution, and the service time of all the stations in the simulation is exponentially distributed. In real systems, nonetheless, relying on the content and kind of data, incoming requests may vary and follow different patterns (like streaming and burst arrivals, which are common in IoT contexts). Sensors, smartphones, businesses, traffic density, and other sources of information may be used to create the content. Likewise, the professional service time for each type of data may not be exponential in all cases. However, it is worth mentioning that the Poisson arrival and exponential service time have been adopted in the literature to get an appropriate approximation of real systems. As stated in [58], [59], [60], [61], [63], [64], [65], [66], [67], considering blockchain IoT integration would result in latent responses and diminished throughput. It's also difficult to discover a viable method for delivering blockchain solutions in order to build a trustless society with minimal changes to the existing IoT ecosystem. The solution offered in the present research is regarding the limits of blockchain. The suggested architecture is the best option for integrating it into IoT, and the findings in this article back up the claim. On the other hand, the two configurations described in this study are completely based on the types of applications. This sort of integration necessitates additional service layer agreements and regulations.

VII. CONCLUSION AND FUTURE WORK

Blockchain integration with IoT has a lot of research potential, and over the previous decade, several authors have proposed various techniques for it. Blockchain is implemented by some researchers on the fog layer and by others on the cloud layer. When a blockchain is placed in the fog layer, the time it takes to access a blockchain decreases, but the fog layer requires a significant increase in computing power. In this solution, storage is done at the fog layer, while processing is acquired from the cloud. Blockchain fog storage will minimize latency and also cloud computing is more cost-effective. Another thing is that, because of the requests from several heterogeneous systems, this strategy allows for message scheduling at the fog layer. The fog layer is divided into two halves. For critical messages, the first sub-system will operate as a legacy CDC-IoT, while the second will act as a novel blockchain-based environment for non-critical messages. As a result, it can be stated that blockchain incorporation with IoT is feasible and should be pursued; the question is how it is managed properly. The architecture presented here serves as a foundation for future Internet developments. This integration will not only inherit the benefits of blockchain but will also have a significant consequence on the effectiveness of life by lowering the energy consumption of massive CDCs. This idea can be implemented in IoT, healthcare, and smart cities in the future to take advantage of blockchain integration with IoT in a smooth way. It is also worth noting that all arriving messages from connected devices in the present simulation model follow a Poisson distribution, and the response time of all the nodes in the simulation is expressed as the mean. In actual systems, however, depending on the content and kind of data, arriving requests may fluctuate and follow different patterns.

REFERENCES

[1] S. M. Nagarajan, G. G. Deverajan, P. Chatterjee, W. Alnumay, and V. Muthukumaran, "Integration of IoT based routing process for food supply chain management in sustainable smart cities," *Sustain. Cities Soc.*, vol. 76, Jan. 2022, Art. no. 103448.

[2] O. I. Abiodun, E. O. Abiodun, M. Alawida, R. S. Alkhawaldeh, and H. Arshad, "A review on the security of the Internet of Things: Challenges and solutions," *Wireless Pers. Commun.*, vol. 119, no. 3, pp. 2603–2637, 2021.

[3] B. K. Tripathy, S. K. Jena, V. Reddy, S. Das, and S. K. Panda, "A novel communication framework between MANET and WSN in IoT based smart environment," *Int. J. Inf. Technol.*, vol. 13, no. 3, pp. 921–931, 2021.

[4] Y. Kalyani and R. Collier, "A systematic survey on the role of cloud, fog, and edge computing combination in smart agriculture," *Sensors*, vol. 21, no. 17, p. 5922, Sep. 2021.

[5] L. Arnold, J. Jöhnk, F. Vogt, and N. Urbach, "A taxonomy of industrial IoT platforms' architectural features," in *Proc. Int. Conf. Wirtschaftsinformatik*, 2021, pp. 404–421.

[6] S. Rathore, J. H. Park, and H. Chang, "Deep learning and blockchain-empowered security framework for intelligent 5G-enabled IoT," *IEEE Access*, vol. 9, pp. 90075–90083, 2021.

[7] P. V. Kakarlapudi and Q. H. Mahmoud, "A systematic review of blockchain for consent management," in *Healthcare*, vol. 9, no. 2, p. 137, 2021.

[8] R. A. Memon, J. P. Li, and J. Ahmed, "Simulation model for blockchain systems using queuing theory," *Electronics*, vol. 8, no. 2, p. 234, Feb. 2019.

[9] B. Bhushan, C. Sahoo, P. Sinha, and A. Khamparia, "Unification of Blockchain and Internet of Things (BIoT): Requirements, working model, challenges and future directions," *Wireless Netw.*, vol. 27, no. 1, pp. 55–90, Jan. 2020.

[10] R. A. Memon, J. P. Li, M. I. Nazeer, A. N. Khan, and J. Ahmed, "DualFog-IoT: Additional fog layer for solving blockchain integration problem in Internet of Things," *IEEE Access*, vol. 7, pp. 169073–169093, 2019.

[11] M. K. Kagita, N. Thilakarathne, T. R. Gadekallu, P. K. R. Maddikunta, and S. Singh, "A review on cyber crimes on the Internet of Things," in *Deep Learning for Security and Privacy Preservation in IoT*. Springer, 2021, pp. 83–98.

[12] W. Viriyasitavat, T. Anuphaptrirong, and D. Hoonsopon, "When blockchain meets Internet of Things: Characteristics, challenges, and business opportunities," *J. Ind. Inf. Integr.*, vol. 15, pp. 21–28, Sep. 2019.

[13] T. Zhou, D. Tang, H. Zhu, and Z. Zhang, "Multi-agent reinforcement learning for online scheduling in smart factories," *Robot. Comput. Integr. Manuf.*, vol. 72, Dec. 2021, Art. no. 102202.

[14] N. Niknejad, W. Ismail, I. Ghani, B. Nazari, and M. Bahari, "Understanding service-oriented architecture (SOA): A systematic literature review and directions for further investigation," *Inf. Syst.*, vol. 91, Jul. 2020, Art. no. 101491.

[15] W. Viriyasitavat, L. D. Xu, G. Dhiman, A. Sapsomboon, V. Pungpapong, and Z. Bi, "Service Worklow: State-of-the-art and future trends," *IEEE Trans. Serv. Comput.*, early access, Oct. 20, 2021, doi: 10.1109/TSC.2021.3121394.

[16] N. Nasser, N. Khan, L. Karim, M. ElAttar, and K. Saleh, "An efficient time-sensitive data scheduling approach for wireless sensor networks in smart cities," *Comput. Commun.*, vol. 175, pp. 112–122, Jul. 2021.

[17] H. Baniata, A. Anaqreh, and A. Kertesz, "PF-BTS: A privacy-aware fog-enhanced blockchain-assisted task scheduling," *Inf. Process. Manag.*, vol. 58, no. 1, 2021, Art. no. 102393.

[18] J. Ge, B. Liu, T. Wang, Q. Yang, A. Liu, and A. Li, "*Q*-learning based flexible task scheduling in a global view for the Internet of Things," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 8, p. 4111, 2021.

[19] M. Whaiduzzaman, M. J. N. Mahi, A. Barros, M. I. Khalil, C. Fidge, and R. Buyya, "BFIM: Performance measurement of a blockchain based hierarchical tree layered fog-IoT microservice architecture," *IEEE Access*, vol. 9, pp. 106655–106674, 2021.

[20] S. Tuli, S. Poojara, S. N. Srirama, G. Casale, and N. Jennings, "COSCO: Container orchestration using co-simulation and gradient based optimization for fog computing environments," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 1, pp. 101–116, Jan. 2021.

[21] Q. Zhu, S. W. Loke, R. Trujillo-Rasua, F. Jiang, and Y. Xiang, "Applications of distributed ledger technologies to the Internet of Things: A survey," *ACM Comput. Surv.*, vol. 52, no. 6, pp. 1–34, 2019.

[22] Z. Xiong, Y. Zhang, N. C. Luong, D. Niyato, P. Wang, and N. Guizani, "The best of both worlds: A general architecture for data management in blockchain-enabled Internet-of-Things," *IEEE Netw.*, vol. 34, no. 1, pp. 166–173, Jan./Feb. 2020.

[23] M. Samaniego, U. Jamsrandorj, and R. Deters, "Blockchain as a service for IoT," in *Proc. IEEE Int. Conf. Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom) IEEE Smart Data (SmartData)*, Dec. 2016, pp. 433–436.

[24] S. Abdullah, M. N. Asghar, M. Ashraf, and N. Abbas, "An energy-efficient message scheduling algorithm with joint routing mechanism at network layer in Internet of Things environment," *Wireless Pers. Commun.*, vol. 111, no. 3, pp. 1821–1835, 2020.

[25] W.-K. Lai, Y.-C. Wang, and S.-Y. Lin, "Efficient scheduling, caching, and merging of notifications to save message costs in IoT networks using CoAP," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 1016–1029, Jan. 2021.

[26] S. Tuli, R. Mahmud, S. Tuli, and R. Buyya, "FogBus: A blockchain-based lightweight framework for edge and fog computing," *J. Syst. Softw.*, vol. 154, pp. 22–36, Aug. 2019.

[27] F. L. de Caldas Filho, R. L. Rocha, C. J. B. Abbas, L. M. C. E. Martins, E. D. Canedo, and R. T. de Sousa, "QoS scheduling algorithm for a fog IoT gateway," in *Proc. Workshop Commun. Netw. Power Syst. (WCNPS)*, 2019, pp. 1–6.

[28] W. Yánez, R. Mahmud, R. Bahsoon, Y. Zhang, and R. Buyya, "Data allocation mechanism for Internet-of-Things systems with blockchain," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3509–3522, Apr. 2020.

[29] Y. Lee, S. Rathore, J. H. Park, and J. H. Park, "A blockchain-based smart home gateway architecture for preventing data forgery," *Hum.-Centric Comput. Inf. Sci.*, vol. 10, no. 1, pp. 1–14, 2020.

[30] A. Nikoukar, S. Raza, A. Poole, M. Güneş, and B. Dezfouli, "Low-power wireless for the Internet of Things: Standards and applications," *IEEE Access*, vol. 6, pp. 67893–67926, 2018.

[31] T. McGhin, K.-K. R. Choo, C. Z. Liu, and D. He, "Blockchain in healthcare applications: Research challenges and opportunities," *J. Netw. Comput. Appl.*, vol. 135, pp. 62–75, Jun. 2019.

[32] J. Lohmer, "Applicability of blockchain technology in scheduling resources within distributed manufacturing," in *Logistics Management*. Springer, 2019, pp. 89–103.

[33] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, "When mobile blockchain meets edge computing," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 33–39, Aug. 2018.

[34] P. K. Sharma, M.-Y. Chen, and J. H. Park, "A software defined fog node based distributed blockchain cloud architecture for IoT," *IEEE Access*, vol. 6, pp. 115–124, 2017.

[35] Y. Liu, K. Wang, Y. Lin, and W. Xu, "LightChain: A lightweight blockchain system for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3571–3581, Jun. 2019.

[36] M. Bukhsh, S. Abdullah, A. Rahman, M. N. Asghar, H. Arshad, and A. Alabdulatif, "An energy-aware, highly available, and fault-tolerant method for reliable IoT systems," *IEEE Access*, vol. 9, pp. 145363–145381, 2021.

[37] M. Bukhsh, S. Abdullah, and I. S. Bajwa, "A decentralized edge computing latency-aware task management method with high availability for IoT applications," *IEEE Access*, vol. 9, pp. 138994–139008, 2021.

[38] S. Abdullah and K. Yang, "A QoS aware message scheduling algorithm in Internet of Things environment," in *Proc. IEEE Online Conf. Green Commun. (OnlineGreenComm)*, Oct. 2013, pp. 175–180.

[39] H. Baniata and A. Kertesz, "A survey on blockchain-fog integration approaches," *IEEE Access*, vol. 8, pp. 102657–102668, 2020.

[40] M. Kaur, R. Sandhu, and R. Mohana, "Fog load balancing broker (FLBB)," in *Proc. 6th Int. Conf. Image Inf. Process. (ICIIP)*, vol. 6, 2021, pp. 332–337.

[41] D. Arya and M. Dave, "Security-based service broker policy for FOG computing environment," in *Proc. 8th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, 2017, pp. 1–6.

[42] A. Naouri, H. Wu, N. A. Nouri, S. Dhelim, and H. Ning, "A novel framework for mobile-edge computing by optimizing task offloading," *IEEE Internet Things J.*, vol. 8, no. 16, pp. 13065–13076, Aug. 2021.

[43] N. Tariq, M. Asim, F. Al-Obeidat, M. F. Farooqi, T. Baker, M. Hammoudeh, and I. Ghafir, "The security of big data in fog-enabled IoT applications including blockchain: A survey," *Sensors*, vol. 19, no. 8, p. 1788, 2019.

[44] E. Badidi and A. Ragmani, "An architecture for QoS-aware fog service provisioning," *Proc. Comput. Sci.*, vol. 170, pp. 411–418, Jan. 2020.

[45] M. Rehman, N. Javaid, M. Awais, M. Imran, and N. Naseer, "Cloud based secure service providing for IoTs using blockchain," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–7.

[46] H. Zhu, Y. Wang, X. Hei, W. Ji, and L. Zhang, "A blockchain-based decentralized cloud resource scheduling architecture," in *Proc. Int. Conf. Netw. Netw. Appl. (NaNA)*, Oct. 2018, pp. 324–329.

[47] S. Dhelim, H. Ning, F. Farha, L. Chen, L. Atzori, and M. Daneshmand, "IoT-enabled social relationships meet artificial social intelligence," *IEEE Internet Things J.*, vol. 8, no. 24, pp. 17817–17828, Dec. 2021.

[48] F. Andriopoulou, T. Dagiuklas, and T. Orphanoudakis, "Integrating IoT and fog computing for healthcare service delivery," in *Components and Services for IoT Platforms*. Springer, 2017, pp. 213–232.

[49] A. Yasmeen, N. Javaid, O. U. Rehman, H. Iftikhar, M. F. Malik, and F. J. Muhammad, "Efficient resource provisioning for smart buildings utilizing fog and cloud based environment," in *Proc. 14th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jun. 2018, pp. 811–816.

[50] M. M. S. Maswood, M. D. R. Rahman, A. G. Alharbi, and D. Medhi, "A novel strategy to achieve bandwidth cost reduction and load balancing in a cooperative three-layer fog-cloud computing environment," *IEEE Access*, vol. 8, pp. 113737–113750, 2020.

[51] A. Kumari, S. Tanwar, S. Tyagi, and N. Kumar, "Fog computing for healthcare 4.0 environment: Opportunities and challenges," *Comput. Elect. Eng.*, vol. 72, pp. 1–13, Nov. 2018.

[52] U. K. Saba, S. U. Islam, H. Ijaz, J. J. P. C. Rodrigues, A. Gani, and K. Munir, "Planning fog networks for time-critical IoT requests," *Comput. Commun.*, vol. 172, pp. 75–83, Apr. 2021.

[53] S. Sicari, A. Rizzardi, and A. Coen-Porisini, "5G in the Internet of Things era: An overview on security and privacy challenges," *Comput. Netw.*, vol. 179, Oct. 2020, Art. no. 107345.

[54] M. Capra, R. Peloso, G. Masera, M. R. Roch, and M. Martina, "Edge computing: A survey on the hardware requirements in the Internet of Things world," *Future Internet*, vol. 11, no. 4, p. 100, 2019.

[55] L. Tseng, L. Wong, S. Otoum, M. Aloqaily, and J. B. Othman, "Blockchain for managing heterogeneous Internet of Things: A perspective architecture," *IEEE Netw.*, vol. 34, no. 1, pp. 16–23, Jan. 2020.

[56] K. Miyachi and T. K. Mackey, "hOCBS: A privacy-preserving blockchain framework for healthcare data leveraging an on-chain and off-chain system design," *Inf. Process. Manag.*, vol. 58, no. 3, 2021, Art. no. 102535.

[57] R. Gupta, D. Reebadiya, and S. Tanwar, "6G-enabled edge intelligence for ultra-reliable low latency applications: Vision and mission," *Comput. Standards Interfaces*, vol. 77, Aug. 2021, Art. no. 103521.

[58] P. Lv, L. Wang, H. Zhu, W. Deng, and L. Gu, "An IoT-oriented privacy-preserving publish/subscribe model over blockchains," *IEEE Access*, vol. 7, pp. 41309–41314, 2019.

[59] S. E. Kafhali and K. Salah, "Efficient and dynamic scaling of fog nodes for IoT devices," *J. Supercomput.*, vol. 73, no. 12, pp. 5261–5284, Dec. 2017.

[60] M. Seenivasan, R. Senthilkumar, and K. S. Subasri, "M/M/2 heterogeneous queueing system having unreliable server with catastrophes and restoration," *Mater. Today Proc.*, vol. 51, pp. 2332–2338, Jan. 2022.

[61] K. Jeganathan, S. Vidhya, R. Hemavathy, N. Anbazhagan, G. P. Joshi, C. Kang, and C. Se, "Analysis of *M/M/1/N* stochastic queueing—Inventory system with discretionary priority service and retrial facility," *Sustainability*, vol. 14, no. 10, p. 6370, 2022.

[62] S. El Kafhali and K. Salah, "Performance analysis of multi-core VMs hosting cloud SaaS applications," *Comput. Standards Interfaces*, vol. 55, pp. 126–135, Jan. 2018.

[63] R. A. Memon, J. Li, J. Ahmed, A. Khan, M. I. Nazir, and M. I. Mangrio, "Modeling of blockchain based systems using queuing theory simulation," in *Proc. 15th Int. Comput. Conf. Wavelet Act. Media Technol. Inf. Process. (ICCWAMTIP)*, 2018, pp. 107–111.

[64] T. Meng, Y. Zhao, K. Wolter, and C.-Z. Xu, "On consortium blockchain consistency: A queueing network model approach," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 6, pp. 1369–1382, Jun. 2021.

[65] S. M. Alrubei, E. A. Ball, J. M. Rigelsford, and C. A. Willis, "Latency and performance analyses of real-world wireless IoT-blockchain application," *IEEE Sensors J.*, vol. 20, no. 13, pp. 7372–7383, Jul. 2020.

[66] A. Ahmed, S. Abdullah, M. Bukhsh, I. Ahmad, and Z. Mushtaq, "An energy-efficient data aggregation mechanism for IoT secured by blockchain," *IEEE Access*, vol. 10, pp. 11404–11419, 2022.

[67] A. Shifa, M. N. Asghar, A. Ahmed, and M. Fleury, "Fuzzy-logic threat classification for multi-level selective encryption over real-time video streams," *J. Ambient Intell. Hum. Comput.*, vol. 11, no. 11, pp. 5369–5397, 2020.

ISRAR AHMAD received the bachelor's and master's degrees in computer science from the Virtual University of Pakistan. He is currently pursuing the Ph.D. degree with The Islamia University of Bahawalpur, Pakistan. His main research interests include the IoT systems, fog computing, high availability, and blockchain.

SAIMA ABDULLAH received the Ph.D. degree from the Department of Computer Science and Electronic Engineering, University of Essex, U.K. She is currently an Assistant Professor with the Department of Computer Science and Information Technology, The Islamia University of Bahawalpur, Pakistan. She is a member of the Multimedia Research Group, DCS, where she has been involved in efficient and secure communication of multimedia data over future generation network technologies. Her main research interests include wireless networks and communications, future internet technology, and network performance analysis. She has authored around ten papers in the above research areas. She serves as a reviewer of international journals.

MUHAMMAD BUKHSH received the master's degree in information technology from the University of Education Lahore, Pakistan, in 2012, and the M.S. degree in computer sciences from The Islamia University of Bahawalpur, in 2016, where he is currently pursuing the Ph.D. degree. His main research interests include wireless networks and communications, ad-hoc networks, the IoT systems, energy efficiency, edge commuting, high availability, blockchain, and fault tolerance.

ADEEL AHMED received the master's degree in computer science from The Islamia University of Bahawalpur, Pakistan, the M.S. degree in computer sciences from the Virtual University of Pakistan, and the Ph.D. degree from The Islamia University of Bahawalpur. His main research interests include edge computing, the IoT systems, energy efficiency, fuzzy logic, high availability, cloud, wireless sensor networks, and blockchain.

HUMAIRA ARSHAD received the master's degree in information technology from the National University of Science and Technology (NUST), Pakistan, and the Ph.D. degree from the School of Computer Science, University Sains Malaysia. She joined the Faculty of Computer Sciences and IT, in 2004. She is currently an Assistant Professor with the Department of Computer Sciences and IT, The Islamia University of Bahawalpur, Pakistan. Her research interests include digital and social media forensics, information security, online social networks, cybersecurity, intrusion detection, reverse engineering, and semantic web.

TALHA FAROOQ KHAN received the master's degree in computer science from The Islamia University of Bahawalpur (IUB), Punjab, Pakistan, and the M.S. degree in computer science from NCBA&E, Pakistan. He is currently pursuing the Ph.D. degree in computer science with the Department of Computer Science and Information Technology (DCS), IUB. His current research interests include text mining, web mining, machine learning, and deep learning.