IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 May 2023, accepted 6 June 2023, date of publication 9 June 2023, date of current version 21 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3284471

RESEARCH ARTICLE

Lightweight Blockchain-Based Architecture for 5G Enabled IoT

MOHAMMAD MAROUFI[1], REZA ABDOLEE[2], BEHZAD MOZAFFARI TAZEKAND[3], AND SEYED AMIR MORTEZAVI[4]

[1]Wireless Laboratory, Department of ECE, University of Tabriz, Tabriz 51666, Iran
[2]IoT Laboratory, California State University Channel Islands, Camarillo, CA 93012, USA
[3]Department of ECE, University of Tabriz, Tabriz 51666, Iran
[4]Cryptography Laboratory, Department of ECE, University of Tabriz, Tabriz 51666, Iran

Corresponding author: Mohammad Maroufi (m.maroufi@tabrizu.ac.ir)

ABSTRACT This paper introduces a lightweight Blockchain-based architecture for 5G-enabled Internet-of-Thing (IoT) networks that employs a low-complexity consensus algorithm suitable for resource-constrained IoT devices. By combining 5G technology with a lightweight Blockchain consensus algorithm, the proposed architecture guarantees high availability, real-time data delivery, security, reliability, and low-latency connectivity. Two transaction types are considered in this architecture, i.e., local and public. Local transactions are exchanged within devices located in the same Small Cell (SC) or Macro Cell (MC) private Blockchains, while public transactions exchange data among different MCs in the public Blockchain and store verified data in the distributed ledger. Performance evaluation reveals that the proposed architecture outperforms conventional 5G (without Blockchain) regarding security against data manipulation and fraud. The proposed architecture improves hashing and encryption protocols compared to conventional 5G but slightly reduces the data traffic rate and increases local transaction processing time. In contrast, the proposed architecture reduces the consensus processing time in the public Blockchain compared to Proof of Elapsed Time (PoET) by about thirty percent due to adding a Distributed Trust Algorithm (DTA). We evaluate the proposed architecture's performance against conventional 5G and PoET in terms of processing time and power consumption. The results indicate that the proposed architecture provides superior performance, and the DTA algorithm's addition enhances the public transaction's consensus processing time.

INDEX TERMS Blockchain, DTA, IoT, lightweight, PoET, scalability, security, transaction.

I. INTRODUCTION

The number of new Internet-of-Thing (IoT) devices linked to the Internet has grown exponentially in recent years. IoT devices will be anticipated to surpass 20 billion by 2025, increasing the generated traffic to six hundred ZettaBytes [1], [2]. According to Cisco's prediction, half of the networked devices will be IoT devices by the end of 2023, and a third will be wireless. Almost all smartphones and mobile devices provide embedded IoT features in wireless nodes. Although about eleven percent of them will be 5G capable, they generate three times greater net traffic than 4G cell devices [3]. Therefore, suggesting a 5G-enabled platform for IoT devices is essential. In addition to the steadily growing demand for IoT applications, the COVID-19 pandemic worldwide has proven the importance of the convergence of IoT with the latest technologies, including the Internet, 5G, Artificial Intelligence (AI), and Blockchain. During this period, many Internet-based businesses flourished, and several traditional companies were prompted to provide new intelligent solutions to protect their business [4].

By growing the IoT network's size and taking over more essential and sensitive tasks by IoT devices, security breaches, privacy concerns, and connectivity issues have become the main drawbacks and threats to current IoT technologies. Gartner predicts that in 2025, over 25 percent of attacks on enterprises will be targeted at IoT devices [5]. The IoT networks need to design

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

for future requirements and additional features. In other words, they must keep their scalability and functionality when the IoT devices and applications increase exponentially [6]. The predictions and available statistics on IoT network vulnerability motivate recent research toward innovations for creating secure, scalable, and high-speed networks aligned and compatible with 5G requirements and objectives.

Most IoT networks use a centralized architecture which causes security and scalability issues, as a failure in the central server can cause system failure or bottleneck [7], [8]. Decentralized or distributed architecture, such as Edge and Fog computing, can mitigate these issues by transferring a part of computational operations to edge nodes or devices within the network [9], [10], [11]. Edge computing allocates part of the processing load to IoT devices and gateways, while fog computing transfers the computation load from the central servers to local area networks [12], [13]. These distributed computing technologies can create a highly scalable network with lower latency, reduce computation complexity, and keep storage resources, bridging the gap between centralized clouds and distributed IoT devices. Fog and Edge computing could potentially enhance the performance of IoT networks [14], [15].

Recent researches suggest that distributed peer-to-peer (p2p) solutions utilizing Blockchain technology can overcome the centralized architecture and Fog/Edge computing drawbacks in IoT networks, which require permanent support computing [16], [17], [18]. Blockchain technology can transform the centralized architecture into a p2p distributed ledger, offering immutability, security, and data privacy [19]. This technology can utilize the resources of all participating nodes and eliminate many-to-one traffic flows, decreasing latency and solving the single point of failure problem in the centralized design [20]. Blockchain also provides a secure platform for IoT devices by offering massive trust, as most network participants have to agree to validate transactions based on a consensus protocol [21].

Despite its benefits, the well-known Blockchain consensus methods, such as Proof of Work (PoW) [22] and Proof of Stake (PoS) [23] impose significant computation overhead, limited scalability, and delays to the network, making it challenging to fulfill IoT requirements [24]. However, new Blockchain-based solutions have emerged to address these issues in recent years. These solutions include lightweight Blockchain algorithms and Hybrid combinations of distributed ledger and Fog/edge computing [25]. Another alternative is to reconfigure an IoT network by combining single public and multiple private Blockchains with different consensus protocols. For instance, [26] proposes a Lightweight Scalable Blockchain (LSB) in smart homes that utilizes two-layer, local and overlay, to reduce data overload with central local manager to manage communication, while [27] uses Byzantine Fault Tolerance (BFT) to interconnect multiple sub-Blockchain representatives to minimize consensus computation overload and latency. Similarly, [28] connects multiple enterprise private Blockchains using hybrid consensus nodes to create a public Blockchain that facilitates data exchange within enterprises.

Although Blockchain and IoT integration provides several benefits, some IoT devices require real-time connectivity, low latency, and high speed to transmit critical data [29]. The fifth-generation mobile network, which uses advanced wireless technologies like Small Cell (SC), mm-wave communications, Software-Defined Networks (SDN), and Network Function Virtualization (NFV), aims to overcome the limitations of previous cellular standards for IoT devices [30], [31].SC can handle high traffic volumes by utilizing the mmWave band, while SDN can provide services for IoT edge devices [32], [33]. A novel Blockchain-based distributed cloud architecture has been introduced in [34], which uses SDN-enabled controller fog nodes at the edge of the network. NFV is also a promising approach to reducing network control traffic and costs [35].

This paper introduces a Blockchain-based architecture that combines SC, Macro cell (MC), and 5G network core with a lightweight security consensus algorithm to improve real-time data delivery, security, privacy, resiliency, and low latency in IoT networks. While cellular networks offer a stable connection with higher data rates, they are not ideal for small IoT networks in nearby areas due to the 5G centralized architecture, which asks 5G access devices to forward the computation and routing to the 5G network core. This results in increased data blocking rate, internet traffic, and End to End (E2E)h delay, along with higher transmission costs compared to low-power wide area networks. The proposed architecture addresses the issues small IoT networks face in a small area by securely exchanging data in private Blockchains, reducing internet traffic, blocking rates, and delays. Also, a lightweight consensus algorithm is used to protect the security of exchanging data within these private networks, which reduces computation complexity well-suited for IoT devices while maintaining data privacy.

Our proposed architecture involves utilizing 5G SCs to reconfigure IoT devices at the network edge, establishing local Blockchains within their coverage range, and MC manages a second-level local Blockchain to preserve network scalability. Each SC centrally manages packets and maintains an immutable ledger, while each MC handles second-tier local transactions and acts as a local Blockchain representative for out-of-range cases. All MCs participate in transaction verification and consensus processes and keep a copy of the immutable ledger. The proposed architecture uses lightweight encryption and consensus algorithms to reduce computation load, making it suitable for IoT. The main contribution of this research can be summarized below:

- Propose a novel three-layer (two private and one public) Blockchain based on the current 5G cellular network.
- Protect data security and preserve privacy with IoT-suitable consensus algorithm in the proposed architecture.

- Reduce the Internet traffic and down-level the local transaction management via private Blockchains.
- Provides power, resource consumption, and cost-effective architecture for cellular systems.
- Capable of implementing on current 5G cellular network based on software patches.

The remainder of the paper is structured as follows: Section II introduces the preliminary concepts and related works. Section III explains the system model, transaction structures, and consensus mechanisms. Section IV evaluates the performance of the proposed architecture in different scenarios and compares it with conventional 5G and 5G Proof of Elapsed Time (PoET) in terms of latency, consensus time, blocking ratio, and security. Finally, Section V provides the paper's conclusion and future works.

II. RELATED WORKS

IoT devices can be categorized as either resource-rich or resource-constrained. While devices like smartphones and Raspberry Pi boards can perform complex operations due to their sufficient resources, most IoT devices have limitations in power, processing, and memory due to their small size and mobility [36], [37]. These constraints make implementing advanced security measures and cryptography algorithms to safeguard data challenging, thereby leading to potential security and privacy risks. Additionally, gathering sensitive personal data in centralized untrusted entities can aggravate the privacy risk associated with IoT platforms [38].

In this regard, Blockchain suggests a way to record transactions or any digital interaction designed to be secure, transparent, highly resistant to outages, auditable, and efficient, which encourages IoT companies to enhance current networks to Blockchain-based technology [39]. However, conventional Blockchains are well-suited to address security, privacy, and centralized bottlenecks issues, but they can not fulfill the IoT devices' requirements regarding resource consumption, delay, and scalability [24]. In [40], researchers discuss the Blockchain idea and relevant characteristics to provide a detailed study of potential security attacks and present current solutions to evade such attacks, which let Blockchain developers counter security vulnerabilities. In recent years, there has been extensive research on integrating Blockchain and IoT, which tries to find out the well-suited Blockchain approaches for authorization, authentication, privacy protection, security, scalability, and power consumption [41], [42], [43].

Public Blockchains may disclose essential information and threaten user privacy protection, especially in financial transactions. On the other hand, private Blockchains try to keep sensitive information within a small group of pre-approved participants and not publicly share data due to restricted users who verify the transactions. These properties led researchers to suggest hybrid Blockchains that combine private and public Blockchains. For instance, most people prefer to keep their financial information confidential in digital auctions. Reference [44] proposes a hybrid Blockchain architecture that restricts access to open, sensitive bids on a private Blockchain for anyone except the auctioneer. The public Blockchain is used to announce the winner, and payments and smart contracts deployed on the Blockchain guarantee trustful bids.

The first essential item in integrating Blockchain in IoT networks is the authentication of the connected node to the Blockchain and additional hashing overhead to the IoT devices. Reference [45] considered standard and lightweight hashing functions used in Blockchain-based applications regarding the area, power, energy, security, and throughput. The study found on Field Programmable Gate Arrays (FPGA) platforms that SPONGENT [46] provides the best protection and throughput, while QUARK [47] consumes the least power but offers lower security. Another protocol proposed in [48] introduces a device manager entity to connect IoT devices, which acts as an intermediary to connect IoT devices with Blockchain networks, and it is demonstrated to be robust to various attacks.

Although expanded network access and enhanced connectivity between devices in IoT networks provide many benefits, it increases the risk of cybersecurity attacks. These networks are vulnerable to Cyber-attackers with different digital targets, including wireless and mobile networks and related infrastructures shared between independent services. In [49], researchers proposed a Blockchain Random Neural Network to protect users regarding digital and physical cybersecurity threats and channel authentication methods. This approach keeps user identity secret by employing neural weights to codify the user information. In other words, Blockchain-based authentication is used to keep IoT devices' user privacy.

The smart home is one of the most well-known and considerable IoT that can be employed as a small-scale testbed to model the IoT platforms. Reference [26] introduces a two-layer method. First, in the smart home layer, Local Block Managers (LBM) centrally manage the local immutable ledgers of different IoT devices via establishing shared keys for communication and process requests. They are responsible for generating, verifying, and storing individual transactions. Second, the overlay layer constitutes various high-resource entities known as overlay nodes, including the LBM, mobile devices, service provider servers, and local cloud storage, forming a public Blockchain to achieve decentralization. LSB proposed the PoET consensus algorithm to decrease computing overload and verification time. Reference [50] suggests a hybrid architecture that combines the Hyperledger Composer Fabric Blockchain with edge nodes to control access to EHR data. The Blockchain-based controllers apply the identity and access control policies and immutable log of access events stored on the Blockchain. The Edge nodes keep the off-chain EHR datasets and let authorized users access the patients' records to execute smart contracts and access list (ACL) policies.

The majority of current IoT platforms have centralized architectures, leading to high maintenance costs, low time-critical IoT applications compatibility, and security and trust issues. Decentralized smart objects cooperate to achieve distributed consensus in the IoT world. Hybrid-IoT [27] and LSB are similar approaches that include multiple sub-Blockchains. However, Hybrid-IoT attempts to protect its decentralization by executing PoW in each sub-Blockchain and using the BFT inter-connector framework (Polkadot) to provide connections within PoW-based sub-Blockchains. Hybrid Flowchain [28] enables machine learning on IoT Blockchain through private permissioned and public permissionless Blockchains and guarantees data privacy while allowing multiple organizations to perform collaborative data analytics and machine learning. In other words, hybrid consensus nodes represent organizations' private Blockchain in PoS-based mining transactions. Lightweight-BIoV [51], is a lightweight Blockchain-based architecture for the Internet of Vehicles (IoVs) that enhances security and privacy by using a hybrid consensus mechanism (PoW and Proof of Authority (PoA)) to achieve a balance between security and efficiency. The performance of the proposed architecture is evaluated through simulation experiments using the Cooja simulator, demonstrating its ability to efficiently handle large amounts of data and resist various types of attacks.

In [34], a proposed solution for the challenges of high availability, real-time data delivery, scalability, security, resilience, and low latency in the IoT network is a Blockchain-based distributed cloud with SDN-enabled controller fog nodes at the network's edge. Fog nodes, which are distributed fog computing entities, allow the deployment of fog services and are composed of multiple computing resources at the edge of the IoT network. The proposed solution shows promising results in reducing delays, improving response time, increasing throughput, and detecting real-time attacks in the IoT network with low-performance overheads.

To provide network connectivity, heterogeneous IoT devices utilize various wireless technologies, such as 2G/3G/4G, Wi-Fi, and Bluetooth [29]. However, according to Nokia's prediction [52], by 2025, around 10 billion IoT devices will require cellular technologies for connectivity, which need to be fully optimized for IoT applications. Therefore, 5G technology is expected to provide the necessary infrastructure for these devices to meet IoT requirements, including higher speeds, lower latency, reduced energy consumption, increased data traffic capacity, and expansion of cell sites.

In addition to the 5G features, other promising technologies, such as SDN and NFV, are used in 5G implementation to enhance network flexibility and management. SDN and NFV are complementary approaches. They suggest strategies to design, deploy and manage the network and its services. While SDN separates the network control and data forwarding planes to provide a centralized view of the distributed network for more efficient orchestration and automation of services, NFV focuses on optimizing these services by taking network functions from dedicated hardware appliances to run them as software in the same device to accelerate service innovation, and provisioning [53], [54].

III. PROPOSED ARCHITECTURE

This section presents the proposed architecture, which implements a Blockchain-based model on the existing 5G cellular system. The first step comprehensively introduces the system model, layers, connections, interfaces, and functionalities. A detailed analysis of the structure and different local/public transaction types is then performed. The consensus algorithm and related methods to reduce latency and complexity are introduced.

A. SYSTEM MODEL

Figure 1 illustrates our proposed system model that consists of three main layers: Device, Edge, and Core. This architecture attempts to employ the Blockchain-based software platform on the existing 5G architecture without making structural changes in the infrastructure to increase security and reduce Internet traffic. Each layer in the proposed architecture has a tier of public/private Blockchain, described in detail in the following.

1) DEVICE LAYER

In the 5G architecture, SCs play essential roles in connecting end devices and mobile devices to the network. Each SC has appropriate power and computation resources to manage and forward transactions compared to connected devices. In this layer, SCs create multiple sub-Blockchains of connected IoT devices and centrally manage the local immutable ledger and all incoming/outcoming transactions from network/IoT devices as Small Cell Blockchain Manager (SCBM). Each sub-Blockchain formed around the SC coverage area performs as a private Blockchain and lets IoT devices join the SCBM to secure 5G network access via tamper-proof elements that provide end-to-end anonymity. SCBM manages transactions locally and adds them to the SCBM immutable ledger if they perform within SCBM and IoT devices or among two IoT devices located in the same SC. Otherwise, they will be forwarded to the MC if the transaction destination is not in the SCBM table. All SCs directly connect to the MC as a Core network gateway through wired or microwave backhaul in the proposed architecture. Due to the resource-constrained nature of IoT devices, local transactions in SC are encrypted using Diffie-Hellman protocol [55] as a symmetric encryption to securely exchange cryptographic keys over a public channel and SPONGENT as a lightweight cryptographic function. The structure and various local transactions are discussed in the following subsection.

2) EDGE LAYER

This layer includes the multiple 5G MC, responsible for two main tasks. First, as SCs in the device layer, MC creates

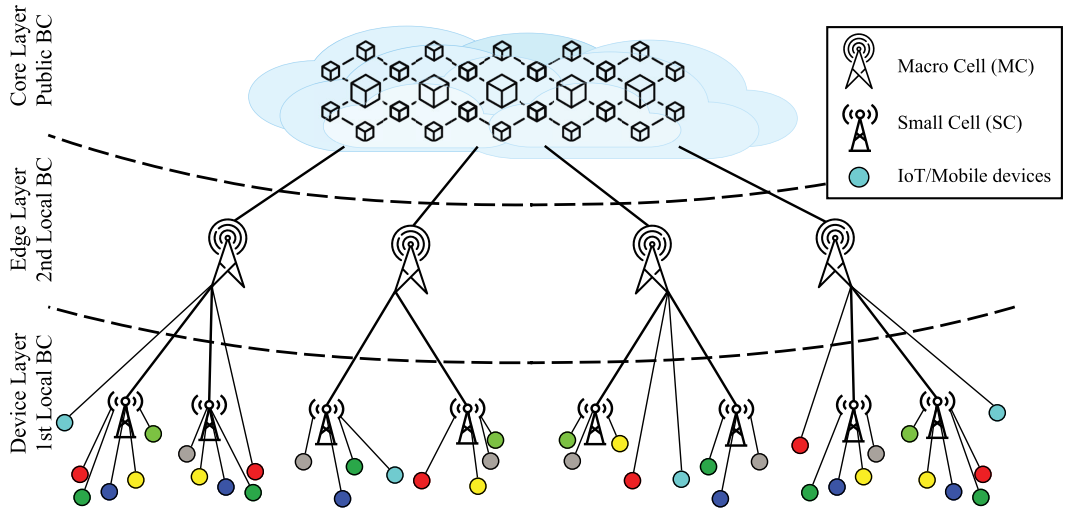


FIGURE 1. Proposed architecture.

sub-Blockchains of connected SCBMs and centrally manages the MC local immutable ledger, collecting associated devices' information table and transactions as MC Blockchain Manager (MCBM). Second, each MCBM participates in the public Blockchain to handle transactions in a distributed ledger. Therefore, MCBM generates two types of transactions in the proposed architecture: the local transaction, which performs within MCBM and IoT devices or among two IoT devices/SCs in the same MC. Second, the public transaction commits beyond the MCBM coverage area or end devices via public Blockchain in the other location. However, MCBM manages local transactions the same as the SCBM process, but from a public transaction viewpoint, each MCBM is known by a public key, and they use a distinct public key for any transaction to ensure anonymity. The MCBM generates secured transactions using asymmetric encryption, digital signatures, and cryptographic hash functions (e.g., SHA256) to provide fully protected communication. The structure and various types of public transactions are discussed in the following subsection.

3) CORE LAYER

This layer consists of a high-performance distributed SDN controller and NFV orchestrator, representing a path toward more generic network hardware and open software. SDN works on layers 2 and 3 of the OSI model [56] to separate the data plane from the control plane and redefine network architecture. On the other hand, NFV works on layers 4 to 7 of the OSI model to separate network software from hardware and redefine network equipment architecture. Therefore, the core layer aims to separate data/control traffic and enhance network performance and flexibility via orchestration. Although the transaction data or payload are instantly sent from the data plane, the transaction control packets are transferred via the control plane and waiting for consensus to verify the transaction data. Moreover, core nodes can access the distributed cloud over the internet to flexibly deploy the application service and computing availability. The Blockchain-based cloud in this layer provides secure, low-cost, and on-demand access to the competitive computing infrastructures proposed in [34]. Clients can search, find, supply, use, and automatically free up all the computing resources they need, such as servers, data, and applications.

B. PUBLIC/LOCAL TRANSACTIONS

Based on source and destination end devices, the local and public transactions are formed by SCBM/MCBM in unique structures to cover system model requirements in the proposed architecture. The public and local transactions' structure and their characteristics are explained in this section.

1) PUBLIC TRANSACTIONS

Depending on the requested actions, public transactions can be classified as single-signature or multi-signature transactions, containing only the requester's signature or both the requester's and responder's signatures. Figure 2 illustrates the structure of public single/multi-signature transactions.

Each multi-signature public transaction contains five essential parts. The first part exposes transaction information, such as the hash of the transaction (transaction ID) and the previous transaction hash of the same requester node, to track a chain of requester's transactions, monitor transaction validation rate, and protect Blockchain from cyber security attacks such as block overgeneration, modification, and Distributed Denial of Service (DDoS). In the second/third part, the public key and signature of the requester/responder are set in their fields. The responder appends its signature to the transaction when it receives the request from the requester. The Fourth part includes transaction verification information and records set by the requester. This part

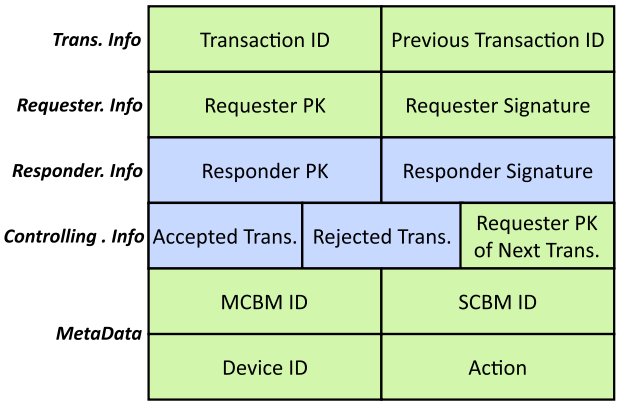


FIGURE 2. The structure of public transaction.

contains the following three entries: (I) The number of verified transactions that the requester generated for the responder, (II) The number of rejected transactions of the responder, (III) The Public key (PK) of the requester in the next transaction. The first two fields are used in the Distributed Trust Algorithm (DTA) [26] explained in the next section to reduce transaction verification participants and consensus latency. The last field is essential for requester verification due to the dynamic PK nature of MCBM in the subsequent transactions. The final part of a multi-signature transaction provides metadata about the desired action and target IoT device, SCBM, and MCBM. In the case of single-signature transactions, no need for any data or confirmation from the responder side; the public transaction structure is similar to multi-signature, excluding the responder fields, highlighted with blue in figure 2.

Public transactions are responsible for various functionalities and tasks in the MCBM. The main public transactions are classified as follows:

- *Genesis transaction:* Each MCBM or Blockchain-based cloud node requires a Genesis transaction to join the public Blockchain. Certificate authorities can generate this transaction to append them to the Blockchain. After verification, the MCBM advertises the Genesis transaction to other MCBMs participating in the Blockchain to add to the distributed ledger. Due to previous experience with the Cell Global Identity (CGI), other certificate authorities can assign Genesis transactions to each node.
- *Cloud storage:* Any MCBM node can generate this transaction to store data in the cloud storage. Any user who needs to accumulate data has to create an account in a cloud storage provider and use the public/private key to store data and access transactions. In this transaction, the IoT device sends the request to the MCBM and asks to store data in the cloud storage. After authorization via checking ACL checking, MCBM forwards this request to the cloud that contains the cloud storage public key for authentication. If authentication is completed, the cloud storage sends the ID of its MCBM to IoT devices (Figure 3(a)).
- *Access:* The MCBM node generates an access transaction to ask for stored data of devices for some specified time. Any user requiring access to any device sends a request to its MCBM. The requester MCBM generates and broadcasts an access transaction to ask the responder to send its information to the user. It should be broadcasting transactions to find responder MCBM and SCBM. The responder SCBM authorized the requester via ACL and obtained the data access from the local or cloud storage for the requester user if it matches. SCBM routes the data to the requester directly in the data plane, signs the access transaction from the requester, and sends it to its MCBM to be stored in the public Blockchain (Figure 3(b)).
- *Monitor:* In the monitor transaction, we have the same process as an access one, but the requester asks for the real-time information of the responder. Therefore, in the responder SCBM, after the authorization process, the requester user directly receives real-time data from devices instead of cloud or local storage (Figure 3(b)).

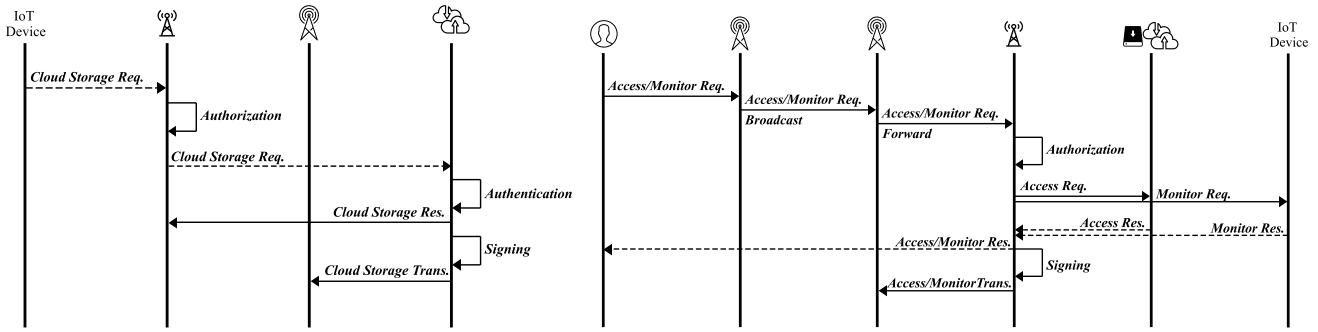
2) LOCAL TRANSACTIONS TYPE

The local immutable ledger records all local and public transactions that their responders are located in the same SCBM/MCBM. Due to the differences between the local and public transaction structure, we provide the form of local transactions, Access policy List, and local immutable ledger example in Figure 4.

Each SCBM/MCBM stored two main structures to manage transactions within its coverage area: The local transaction and ACL roles in characterizing the information of the requested transaction and controlling the access policies, respectively. The local transaction contains five fields to store its information as below: (I) The transaction ID for tracking the sequence of transactions in the local ledger. (II) The device ID stores the requester's device ID. (III) The previous transaction ID for the same device creates a transactions chain for this device. (IV) Transaction types or tasks. (V) The device signature is appended to the transaction frame if needed. As shown in Figure 4, the access policy defines rules for processing local and public transactions.

The access policy contains four parameters: (I) The PK/ID of the requester refers to the generator of incoming public/local transactions. (II) Responder device ID. (II) The access type determines the proper operation for the requester. (IV) The action defines that policy to allow or deny the request. As same as public transactions, the local ones also provide different functionalities, which can be classified as below:

- *Association (Genesis):* Any IoT device requires an association transaction to join the SC/MC and let them add the device to the immutable ledger. This process is divided into two main functions: As standard 5G



(a) Cloud Storage Transaction diagram (b) Access/Monitor Transaction diagram

FIGURE 3. The public transactions diagram.

Structure of Transactions

Ledger Trans. Number	Device ID	Device Previous Trans. Number	Trans. Type	Device Signature (If needed)

Access Policy

PK/ID of Requester	Responder Device ID	Access Type	Action

Local Immutable Ledger Example

9	Light01	0	Associate	-
10	Temp11	1	Access	g6hq2u...
11	Fan15	2	Monitor	dsf32g...
12	Sen13	1	Data Exch.	kdj3f4...
13	Sen01	1	Store	yo6pe7...

Access Policy Example

jhe21a...	Fan15	Access	Allow
All	Cam01	Monitor	Allow
Light01	Sen07	Exchange	Allow
lh3a7j...	Sen13	Monitor	Deny
tq9jv4...	Temp11	Access	Deny

FIGURE 4. Structure of local transaction and access policy.

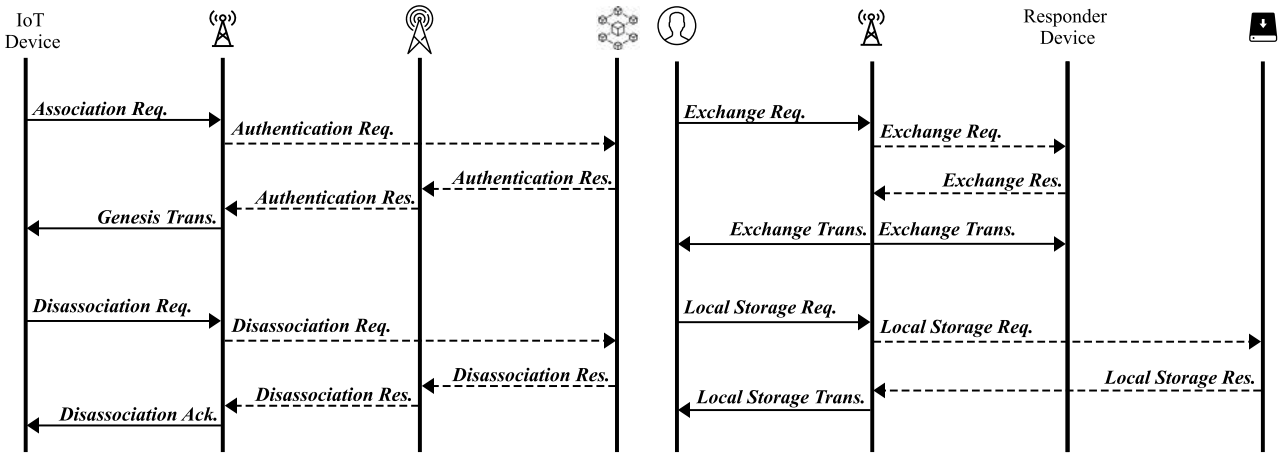
nodes, each device is authenticated via the Access and Mobility Management Function (AMF) for stable connectivity and handover management. Second, the SCBM/MBSM generates a shared key to encrypt its communications directly with the IoT device and stores Genesis transactions in its local immutable ledger (Figure 5(a)).

- *Disassociation:* If the connected device loses its connection with the associated SC for a while or hand over to another cell, the AMF manages this process and send the disassociation notification to update the MC/SC tables. The MCBM/SCBM adds the disassociation transaction to the local ledger and removes PK from its list (Figure 5(a)).
- *Local storage:* When IoT devices need to store data locally, they request the SCBM/MCBM to generate a shared key between the IoT device and the local storage. In the next transactions, local storage use this key to authenticate the IoT device, and SC/MC lets the IoT device store the data in local storage directly(Figure 5(b)).
- *Exchange Data:* Usually, a significant part of the transaction occurs within geographically closed IoT devices. Therefore, When an IoT device needs to communicate with another device inside SC/MC, it allocates shared keys to the devices that request to share data. In the next transaction, they can exchange data with this valid shared key to exchange data via SC directly (Figure 5(b)).

C. CONSENSUS PROTOCOL

Although Blockchain offers several benefits to address IoT obstacles, existing Blockchain consensus algorithms are unsuitable for the IoT context in terms of complexity, scalability, latency, and throughput. For instance, miners in the Bitcoin Blockchain must solve the PoW puzzles to satisfy a particular arbitrary condition. Each block has the SHA-256 hash initiated by a certain number of zeros defined as difficulty level. IoT devices need several minutes to solve PoW puzzles, even with a low difficulty level. Therefore, solving such consensus algorithms imposes significant time and power consumption on resource-constrained IoT devices. These reasons reveal the need for an alternative approach to fit IoT device requirements.

The PoET consensus mechanism was introduced by Intel on the Hyperledger Sawtooth platform to decide which node has the right to mine and determine the winning block in Blockchains. In this mechanism, each node in the network must wait a random amount, and the first node that finishes the waiting time validates the next block. This mechanism in hyperledger sawtooth is used to solve the BFT validation node limitation and requires Intel's Software Guard Extensions (SGX) assistance to determine the next leader in generating the block [57]. SGX offers hardware-based memory encryption to isolate data in memory from application codes and create private spaces called enclaves for a granular level of control and protection [58]. Although the PoET requires dedicated Intel hardware to build trust but provides a more energy-efficient and secure mechanism due to the nodes' long rest and SGX hardware-based encryption for waiting time, respectively. In this paper, the modified LSB, a conceptual PoET consensus-based algorithm in Hyperledger Sawtooth, is employed and altered due to our conditions in the proposed architecture. The modified LSB consensus is organized based on the following strategies:



(a) Association/Disassociation Process (b) Exchange data and locally storage Process

FIGURE 5. The SCBM local transactions diagram.

1) SELECT BLOCK GENERATOR RANDOMLY

In the public Blockchain, as same as PoET, each MCBM node must wait for a random time before generating a new block. After these waiting times, the first MCBM finishes the waiting time and can start the generating block process to validate the generated block. When another MCBM receives the newly developed block, any MCBM should check its transaction pools and remove duplicate transactions. We determine the maximum waiting time in our scheme at 1000 *ms*.

2) RESTRICTED GENERATED BLOCK IN EACH CONSENSUS PERIOD

The Blockchain protects from malicious MCBM by restricting it from generating many blocks and avoiding fake transactions. Any MCBM can generate only one block over a consensus period. The PoET-based mechanisms are at risk of breaking the waiting timer by attackers and consistently winning the lottery. This strategy can mitigate the threat of SGX drawback on the PoET mechanism.

3) DYNAMIC CONSENSUS PERIOD

Distributed throughput management (DTM) algorithm is used to ensure the public Blockchain's utilization rate keeps in range. The algorithm determines the utilization rate by calculating the proportion of the total transactions generated during consensus time Tr_{CT} to the maximum number of added transactions Tr_{max}. Adjusting the consensus time CT can modify the utilization rate. Algorithm 1 presents a pseudocode for determining the consensus period. The algorithm first calculates the utilization rate α and checks if it is within the minimum α_{min} and maximum utilization rate α_{max} limits. The algorithm then computes a new consensus time CT_{new}. If the new consensus time is less than the minimum consensus time, the current consensus time is updated to the minimum consensus time. Finally, the updated consensus time is returned by the algorithm.

Algorithm 1 Dynamic Consensus Period

Input: Consensus Time (CT), Transactions in CT (Tr_{CT}), Max Transaction (Tr_{max}), Max and Min Utilization ($\alpha_{max}, \alpha_{min}$)

Output: New Consensus Time (CT_{new})

Calculate utilization rate $\alpha = \dfrac{Tr_{CT} * CT}{Tr_{max}}$

if $\alpha \geq \alpha_{max}$ ***or*** $\alpha \leq \alpha_{min}$ **then**

  $CT_{new} = \dfrac{CT * \alpha_{avg}}{\alpha}$

  if $CT_{new} \geq CT_{min}$ **then**

    $CT \leftarrow CT_{new}$

  else

    $CT \leftarrow CT_{min}$

    end

  end

return CT

4) DISTRIBUTED TRUST ALGORITHM

Each MCBM must verify the received new block before adding it to the public Blockchain by validating the block generator signature and each transaction in the block. Proving such a significant number of transactions and blocks needs numerous computational resources. Therefore, the amount of transaction that needs to be verified should be reduced via a DTA that addresses scalability issues in the IoT context. An MCBM can have direct evidence about another MCBM if it has previously validated a block generated by it. However, if it lacks direct evidence, it can obtain indirect evidence from a third-party MCBM. In the consensus algorithm, each MCBM increases or decreases the direct evidence value when it verifies or rejects a new block respectively. As a result, if a malicious MCBM sends fake transactions, MCBMs must

Algorithm 2 Distributed Trust Algorithm

Input: Direct & Indirect Trust Impact (T_D, T_I), Node i Direct & Indirect Evidence (D_i, I_i)
Output: Validation Rate (VR_i)

$VR_i = (1 - T_D)^{D_i} * (1 - T_I)^{I_i}$
Generate Binary Random value B_i with success probability VR_i
if $B_i = 1$ **then**
  if *Block Validation Result = True* **then**
    $D_i \leftarrow D_i + 1$
    Broadcast the Result for Indirect evidence update
  else
    $D_i \leftarrow 0$
    BroadcasttheResultforIndirectevidenceupdate
    end
else
  if *Other Nodes validate Block* **then**
    $I_i \leftarrow I_i + 1$
  else
    $I_i \leftarrow I_i - 1$
    end
  end
return $VR_i = (1 - T_D)^{D_i} * (1 - T_I)^{I_i}$

validate more transactions. Conversely, MCBM-generated blocks can be verified more quickly if they build trust with other MCBMs. They use a trust table to manage the verification process. If an MCBM has direct evidence about the block generator, it determines the validation fraction. Otherwise, the indirect evidence sets this fraction.

The DTA is presented in pseudocode in Algorithm 2. It computes the validation rate VR_i for a block generator node i based on the direct and indirect trust impact factors (T_D, T_I) which changes in the range of zero and one, and the direct and indirect evidence counter (D_i, I_i) of node i. The validation rate VR_i is used to generate a random binary value B_i with a success probability of VR_i. If B_i is equal to one, the algorithm verifies the block signature. In this case, If the block is validated, the direct evidence D_i of node i is incremented by one, and the result is broadcasted for indirect evidence update. On the other hand, if the block is not validated, the direct evidence D_i of node i is set to zero, and the result is broadcasted for indirect evidence update. If B_i equals zero, the nodes wait for other nodes to validate the block. If other nodes have validated the block, the indirect evidence I_i of node i is incremented by one. If other nodes have not validated the block, the indirect evidence I_i of node i decremented by one. Finally, the algorithm returns the updated validation rate VR_i, which is the product of the probability of block validation based on the direct and indirect evidence of node i.

IV. PERFORMANCE EVALUATION

This section evaluates the various aspects of our proposed architecture in different scenarios. The Network Simulator (NS3) [59] is one of the most flexible software to simulate wireless network layers and evaluate 5G performance in different conditions. Although still, 5G wireless network standards did not release completely in NS3, the initial models are available to simulate features like mm-waves [60]. We simulate fundamental characteristics of the wireless network, such as physical connection, mobility models, protocol, and applications in NS3. Due to the NS3 restriction to afford a complete model for 5G wireless networks on the latest version and using Non-Stand-Alone (NSA) technology in the initial stage of 5G, the 4G LTE model (LENA) [61] is modified to fit the system requirements. Moreover, due to the absence of an NFV robust model in NS3, the Intel low-latency NFV infrastructure processing time results are employed [62] in terms of latency.

To evaluate the performance of the device/edge layer, we use Cooja [63] to simulate low-resource devices before implementation in terms of resource consumption. Our simulations consider a network with 20 MCBM, 50SCBM, and over 200 IoT devices that generate five transactions in 10 seconds with a Poisson distribution. The maximum consensus time is set to be at most 1000ms. Given that in a similar scenario in a different situation, the transaction verification time varies from transaction to transaction. Then, the verification time calculates in the proposed scheme by the average time of all transactions in each scenario. We use the MSI raider 11UG laptop with Core i7 Gen11, 1TB SSD, 32GB DDR4 RAM, and 8GB RTX3070 GPU as an MCBM node, and the Asus K46 laptop with Core i7 Gen3, 128GB SSD, 6GB DDR3 RAM, and 2GB GT740M GPU as an SCBM node. The Raspberry Pi 3 module and Unisoc UDX710 as 5G chipset were considered as IoT node with Cortex A53 Quad-Core 1.2GHz, 16GB SD card, and 1GB LPDDR2 RAM.

Our Cooja analysis shows a negligible growth in energy consumption. Each IoT devices consume energy for three main tasks: (I) computation tasks by CPU/micro-controller, (II) Radio Transmission by Tx antenna, and (III) Listening to the received transactions. However, a proposed architecture increases energy consumption by about 10 and 45 percent in computation and transmission tasks in comparison to conventional architecture due to the additional hashing, encryption, and related overhead data packets, but listening for received data constitutes the main part (99.5%) of device energy consumption increases below one percent. Therefore we can neglect these increments in our proposed model.

In the following, we evaluate scenarios and tasks in public/local Blockchain within different requester/responder situations regarding consensus processing time, delay, and resource consumption. Simulation results concentrate on processing time and resource consumption of transactions as control packets. We use 5G conventional without any additional hashing, encryption, and local immutable ledger to evaluate local transactions in our proposed architecture. On the other hand, in addition to conventional 5G, the PoET consensus Blockchain is used to evaluate our architecture

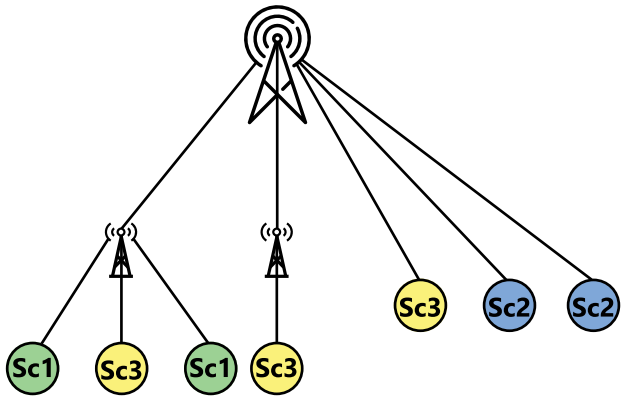


FIGURE 6. Local transactions scenarios.

for public transactions. The following subsections discuss all possible local and public scenarios and simulation results in detail.

A. LOCAL TRANSACTION SCENARIOS

This subsection evaluates the local transaction processing time, which refers to the duration of the request initiated from the IoT device until the related transaction appends to the local immutable ledger. Figure 6 illustrates all the local transaction scenarios. The local transactions can be classified into three main scenarios based on the transaction end nodes' position:

1) SCENARIO 1: BOTH NODES ARE CONNECTED DIRECTLY TO THE SAME SCBM

In this scenario, the connections of IoT devices with the SCBM are established and terminated through association and disassociation transactions, respectively. The exchange of data among connected IoT nodes (either users or IoT devices) within the same SCBM or stored in local storage is accomplished through exchange data and local storage transactions. The performance of the proposed architecture is evaluated using NS3, and the processing time for each transaction type is calculated by taking the average time taken to execute all transactions. As depicted in Figure 7, the association process requires more processing time in our proposed architecture compared to conventional architecture. This disparity is due to the extra hash, encryption, and local ledger recording steps added to the conventional approach in our proposed architecture, which increases the processing time. However, the SCBM can manage other transactions independently, acting as a centralized manager through rules stored in its local immutable ledger.

2) SCENARIO 2: BOTH NODES ARE CONNECTED DIRECTLY TO THE SAME MCBM

In this scenario, the IoT devices are connected directly to the MCBM instead of the SCBM, which results in a more efficient data exchange between the connected IoT nodes, as the MCBM has more robust computing resources than the

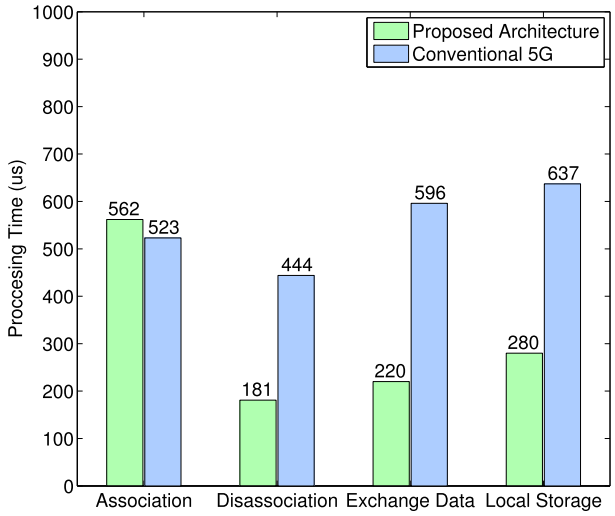


FIGURE 7. Scenario 1 processing Time.

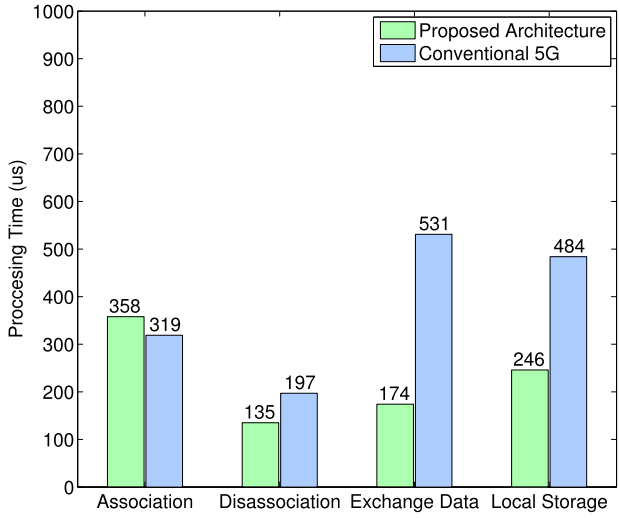


FIGURE 8. Scenario 2 processing Time.

SCBM. The processing time for each transaction is calculated by averaging the times taken for all transaction executions, and the results are shown in Figure 8. The simulation results indicate that the processing time for all transaction types has decreased, confirming the hypothesis that the MCBM would provide faster processing due to its more powerful computing resources. It is worth noting that the direct connection to the MCBM enables the transactions to be executed more efficiently, resulting in a quicker processing time for all transactions.

3) SCENARIO 3: THE FIRST NODE CONNECTS MCBM VIA SCBM1 WHILE ANOTHER CONNECTS THE SAME MCBM DIRECTLY OR VIA SCBM2

In this scenario, local transactions are exchanged between two nodes connected to the same MCBM. One of these nodes is connected through the first SCBM, while the other is either

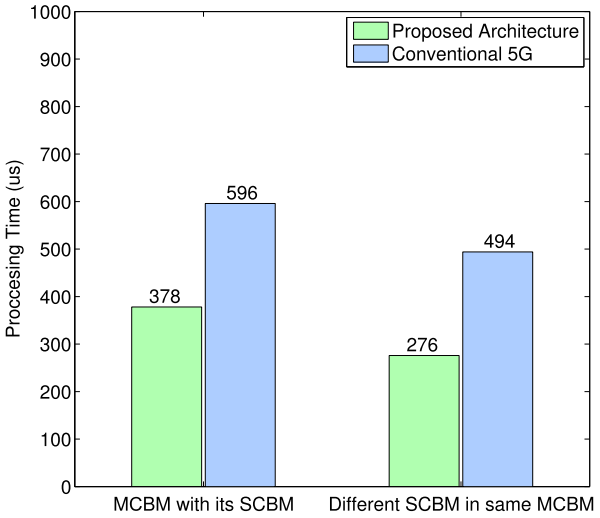


FIGURE 9. Scenario 3 processing Time.

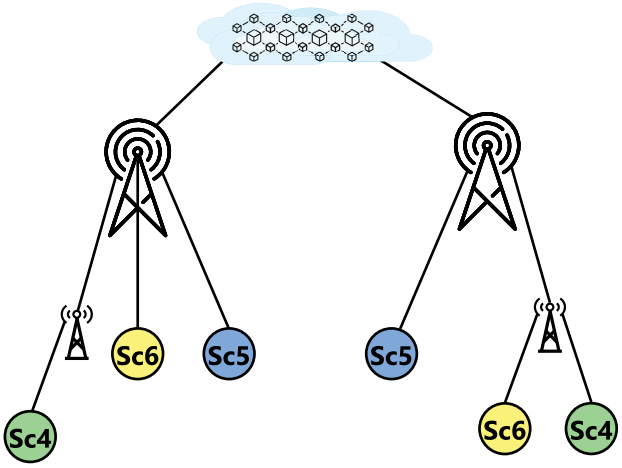


FIGURE 10. Public transactions scenarios.

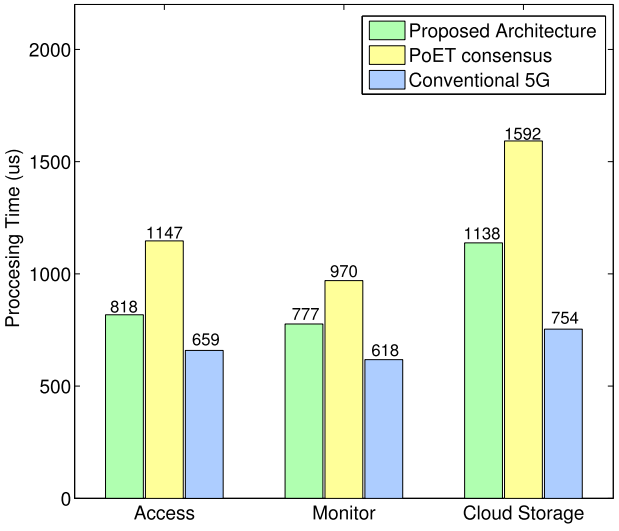


FIGURE 11. Scenario 4 processing Time.

connected through the second SCBM or directly connected to the MCBM. The exchange data transactions between these two nodes in different scenarios are studied, while the association, disassociation, and local storage transactions are already covered in previous local scenarios. The processing time for the exchange data transactions in both scenarios can be seen in Figure 9. Despite the added overhead from hash and encryption in the proposed architecture, the processing time for the exchange data transactions is still lower than that in conventional 5G networks due to the direct exchange of data within the local network rather than through the internet.

B. PUBLIC TRANSACTION SCENARIOS

Public transactions occur between two nodes, either a User or an IoT device, located in different MCs. The modified LSB algorithm checks all transactions and adds those confirmed to the permanent public ledger. This section examines the time it takes to process public transactions, which starts when the requester initiates the transaction and ends when the responder sends back the multi-signature transaction for verification to the public Blockchain. The PoET algorithm is used in the public Blockchain, in addition to using conventional 5G as a reference to compare the performance of our proposed consensus protocol with other lightweight consensus protocols suitable for IoT context. Figure 10 illustrates all the public transaction scenarios. The public transactions can be classified into three main scenarios based on the transaction end nodes' positions as follows:

1) SCENARIO 4: BOTH NODES ARE CONNECTED TO DIFFERENT MCBMs VIA SCBMs

In this scenario, public transactions are carried out between two IoT devices or users connected to different MCBMs through SCBMs. The transactions require the responding device to perform tasks such as accessing, monitoring, or storing data in the cloud, as requested by the initiating device. Despite the increased security and immutability offered by the proposed architecture, the processing time for these transactions is significantly increased due to the additional hash loads, routing process, and the need for the MCBM to check its P2P-associated devices table. The processing time for this scenario is depicted in Figure 11.

The figure depicts that public transactions require more time to process in the proposed architecture compared to the traditional 5G architecture. A comparison of the processing time between using the PoET consensus protocol instead of the modified LSB algorithm shows that the DTA and DTM algorithms can reduce this time by at least 40% based on the simulation results. It's important to note that as the number of transactions and confirmed transactions increases, the confirmation time will decrease with the formation of the trusted table in DTA. Furthermore, the results indicate that it's faster for the user to access IoT device data through

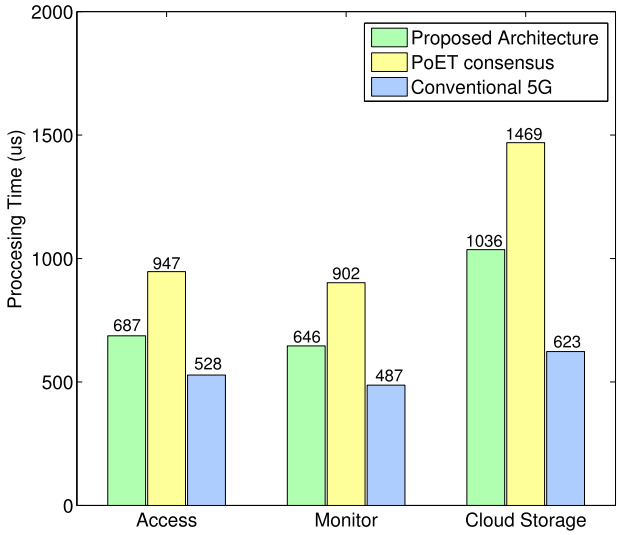


FIGURE 12. Scenario 5 processing Time.

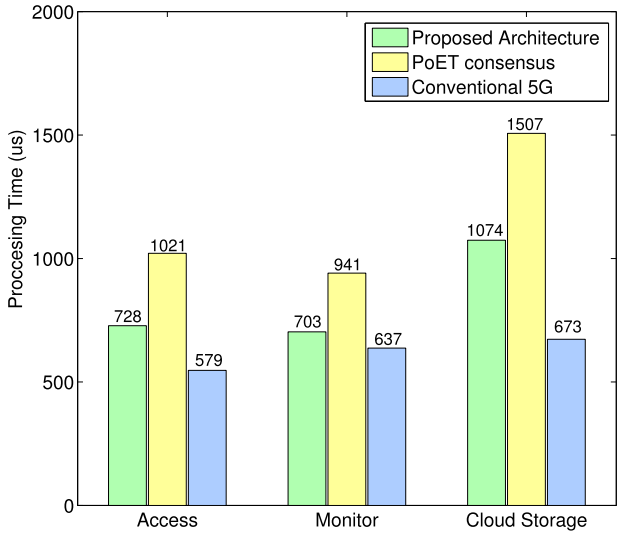


FIGURE 13. Scenario 6 processing Time.

monitoring, rather than requesting it from local or cloud storage.

2) SCENARIO 5: BOTH NODES ARE CONNECTED TO DIFFERENT MCBMS DIRECTLY

In this scenario, the IoT nodes are directly connected to their MCBMs instead of connecting them via SCBMs. Due to fewer process needs by removing the SCBMs and their additional local transaction process, and MCBM's more powerful computing resources in this scenario, it anticipated the processing time decreased compared to scenario 4. The simulation results in Figure 12 prove our hypothesis.

3) SCENARIO 6: JUST ONE OF THE NODES IS CONNECTED TO THE MCBM DIRECTLY, AND ANOTHER NODE CONNECTS DIFFERENT MCBM VIA SCBM

This scenario is a mixture of the previous two scenarios. The public transaction is exchanged within two nodes located on the different MCBM; one is connected to the MCBM directly, while another connects different MCBM via SCBM. The simulation results in Figure 13 reveal that processing tasks perform quicker than in the first scenario and slower than in the second one.

C. CONSENSUS PERIOD

The consensus period is a crucial aspect of the transaction verification process as it determines the amount of time it takes to verify transactions and add them to the distributed ledger. Minimizing the consensus time in IoT networks is essential for adding verified transactions to the Blockchain in real-time. The DTM algorithm regulates the utilization rate, defined as the ratio of new transactions generated to the number of transactions added to the Blockchain, by adjusting the consensus period to maintain a reasonable network utilization rate.

To demonstrate the relationship between the consensus period and transaction rate, we simulated the network using 20 MCBMs. The simulation assumed that out of 1000 generated transactions, only 250 must be stored on the distributed ledger. The minimum and maximum utilization ratios were set to 0.5 and 1, respectively. Also, direct and indirect trust impact factors were set to 0.1 and 0.05, respectively.

The simulation results are illustrated in Figure 14, which shows that the transaction rate starts with 20 transactions generated per second and increases to 25 within a $500 - 5000ms$ period. The rate then increases to 40 until $7000ms$, then decreases to 20, 15, and 10.

The results of the simulation show that the DTM algorithm effectively controls the consensus period and maintains a balance between the transaction rate and the consensus period. When the utilization ratio exceeds the maximum threshold, the DTM algorithm reduces the consensus period to prevent the transactions from piling up in the queue. Similarly, when the utilization ratio drops below the minimum threshold, the DTM algorithm increases the consensus period to compensate for the reduced transaction rate. This way, the DTM algorithm ensures that the network is efficient and operates optimally. The results show that the consensus period fluctuates between $4000ms$ and $7500ms$, a reasonable time frame for adding transactions to the Blockchain. In conclusion, the simulation results demonstrate that the proposed consensus protocol based on the DTM algorithm effectively controls the consensus period and ensures a balance between security and efficiency in the IoT network. The results show that the proposed architecture provides a secure and efficient way for exchanging transactions in the IoT network and is a promising solution for the challenges faced by IoT networks.

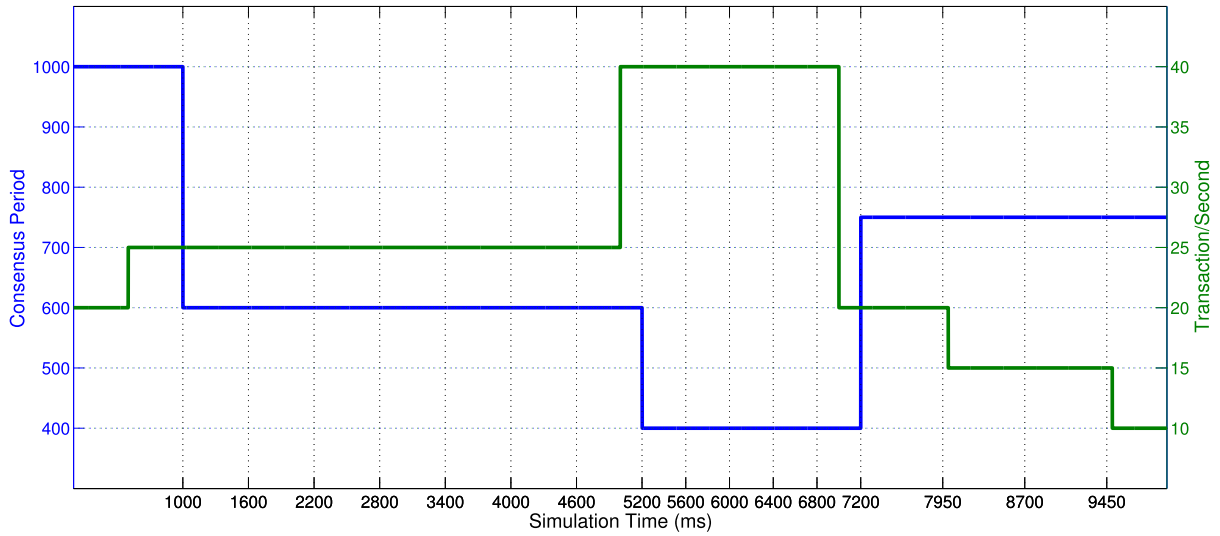


FIGURE 14. Consensus period relationship with transaction rate.

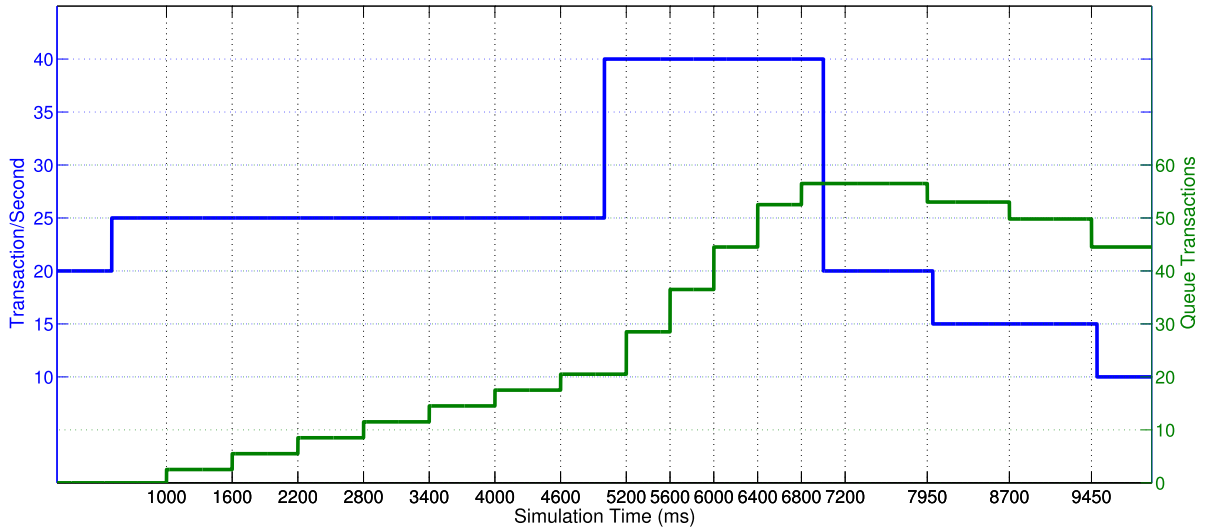


FIGURE 15. Consensus period relationship with queuing transactions.

D. QUEUE TRANSACTIONS AND BLOCKING RATE

As mentioned before, although MCBMs have greater power and computation resources than IoT devices, they still have restricted resources to participate in the transaction verification process. The simulation results reveal that depending on the node's processing power in the proposed Blockchain, some requests are queued to be processed in each consensus cycle, which increases the time required to approve the transactions.

Figure 15 illustrates the queue transactions waiting for verification in each cycle. Therefore, the queuing transactions are calculated after the consensus cycle finishes. Based on our simulation, the verification rate is about ten transactions per second. Two transactions wait for verification when the first consensus period ceases in 1000*ms*. The DTM algorithm changes the consensus period to avoid creating a queue. The queue length of the subsequent consensus periods reaches 5, 8, 11, 14, 17, and 21 transactions. When the transaction rate increases twice, the queue length increases to 56 transactions until the transaction rate and queue length decrease and queued transactions are verified. At the same time, we can use the blocking rate (queuing transaction per all transaction) as a metric to show how many transactions are verified. Figure 16 illustrates the blocking rate in each cycle.

E. CYBER SECURITY ANALYSIS

As previously stated, using Blockchain technology in IoT networks provides a secure means of protecting IoT devices against various cyber attacks. Different security threats target IoT devices, including SCBM, local storage, MCBM, and

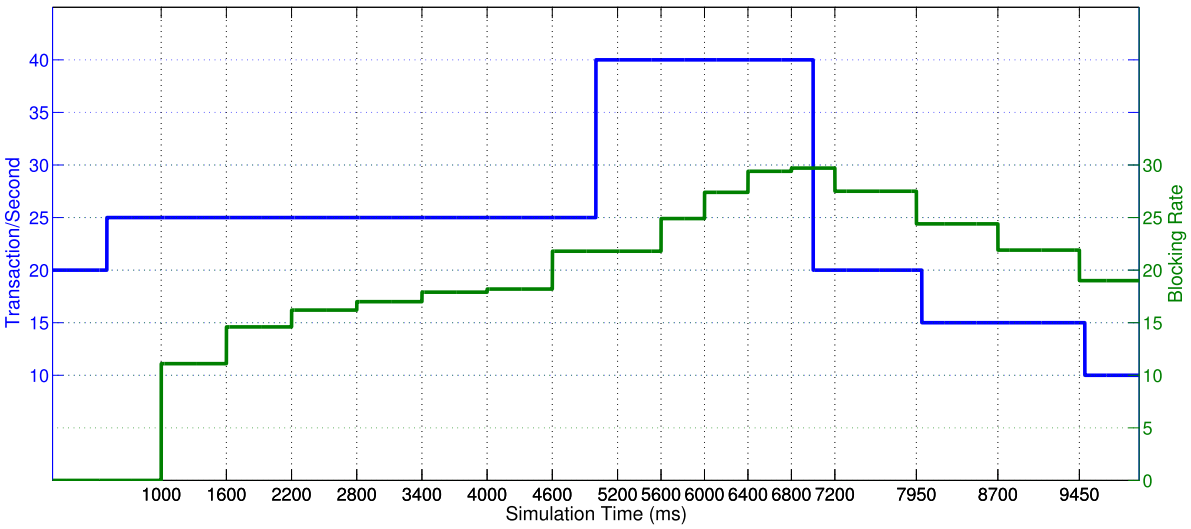


FIGURE 16. Consensus period relationship with blocking rate.

cloud storage. Hackers may attempt to intercept transactions, manipulate validated transactions, or validate false transactions. IoT networks are particularly susceptible to specific security attacks. In this section, we will examine these security attacks and evaluate the robustness of our proposed PoET-based consensus algorithm, which attempts to add an additional security level to Hyperledger Sawtooth. Here are some common cybersecurity attacks and how our proposed architecture can protect IoT networks against them:

1) 51% OR MAJORITY ATTACK

The proposed Blockchain relies on PoET consensus. As mentioned, PoET is a lottery-based consensus mechanism, meaning that the privilege to validate a block is randomly assigned to a node based on the Intel SGX mechanism to generate random waiting times. This process makes it difficult for a malicious actor to control more than half of the network and execute a 51% attack because they cannot predict when their node will be selected to validate a block. If a malicious actor tried to control the network, they would have to spin up many nodes in the hope that one would be selected to validate a block. This solution would require significant computational resources, time, and money. Therefore, PoET provides a robust mechanism for protecting against majority attacks. Although the DTA algorithm may cause decay on the block generator signature validation, indirect evidence help the Blockchain network keep safe.

2) SYBIL ATTACK

The Intel SGX protects the network from Sybil protection due to securely generating random wait times for each validator. This random waiting time makes it difficult for an attacker to control a significant portion of the network and launch a Sybil attack. Additionally, the consensus algorithm uses a digital signature from the rusted execution environment (TEE) to ensure the integrity and authenticity of the wait time to prevent an attacker from manipulating the wait time and creating multiple identities in the network.

3) MAN-IN-THE-MIDDLE (MITM) ATTACK

In this attack, an attacker intercepts and potentially alters the transactions exchange between two parties. In our proposed Blockchain, digital signatures and encryption can protect against MITM attacks by ensuring the authenticity and confidentiality of communications. Also, the PoET-based consensus protocol with a secure and verifiable lottery process for the determination of the following block creator adds a layer of security to the network.

4) BLOCKCHAIN TAMPERING ATTACK

In our PoET-based Blockchain, the consensus protocol detects any attempt to tamper with the validated transactions and block data. The consensus mechanism verifies each transaction before it is added to the ledger, and any suspicious or invalid transactions will be rejected. Additionally, consensus uses cryptographic signatures and public key infrastructure to ensure the authenticity of transactions and blocks. The distributed nature of the network makes it difficult for an attacker to modify a significant number of blocks to tamper with the ledger without being detected. To tamper with the ledger, the attacker would need to have control over a majority of nodes in the network, which is difficult to achieve in a decentralized and distributed Blockchain network.

5) DENIAL OF SERVICE (DoS) AND DISTRIBUTED DOS (DDoS)

The hacker floods the fake transactions to the SCBM/MCBM via a connected device to halt the process of actual transactions in them. The proposed architecture is protected by restricting floods to other nodes via its shared key list

in SCBM/MCBM and keeping the rate of the connected device under the specified threshold. Also, in the PoET-based consensus algorithm, each participant generates a random wait time generated by SGX to ensure that participants cannot cheat by changing the waiting time and must wait for that duration before becoming eligible to create a block. In the end, each block should be verified by the other participants before being added to the Blockchain to prevent DoS attacks by ensuring that malicious blocks are not added to the chain. In DDoS attack, multiple nodes simultaneously flood the network with requests, overwhelming the system and preventing it from functioning correctly. Here, the consensus process provides some protection against DDoS attacks by the PoET lottery-based algorithm of the next block creator selection. This algorithm makes it more difficult for an attacker to launch a DDoS attack using many nodes. Many cryptographic signatures provide integrity and authenticity guarantees, ensuring only valid blocks are added to the chain. Additionally, our approach can use various DoS protection approaches to mitigate the threat of DDoS attacks, including rate limiting, traffic filtering, and load balancing.

6) RACE ATTACK

Our proposed consensus protects MCBMs against race attacks by using random waiting time. The PoET-based consensus algorithm ensures that all participants have an equal opportunity to participate in the block creation process by randomly selecting the leader node using a lottery system, which means that a malicious participant cannot control a large number of nodes to monopolize the block creation process, as in a race attack. The lottery system is deterministic, ensuring that all participants have a fair chance of becoming the leader node, thus preventing race attacks in the consensus process.

7) PRIVACY OR LINKING ATTACK

In a privacy attack, a malicious actor attempts to extract sensitive information, such as user identities or transaction details, from a network. Our consensus algorithm protects privacy using cryptographic techniques such as encryption, hashing, and digital signatures. The transactions are encrypted and stored in a hashed format to prevent unauthorized access and tampering. Additionally, digital signatures are used to validate transactions, adding an extra layer of security.

8) CONSENSUS PERIOD ATTACK

The consensus period adjusts according to the network traffic, but attackers may try to change this value to increase the blocked or queued transactions. For this, the attackers must involve at least half of the MCBMs in the public Blockchain.

From the above explanations, it is evident that the PoET-based Intel SGX technique in the proposed architecture intercepts most attacks. Also, other techniques like the restricted generated block in each consensus period, DTM and DTA, offer much better protection against cyber-attacks than conventional PoET architecture.

V. CONCLUSION

This paper introduced a lightweight Blockchain-based architecture for 5G-enabled IoT to fulfill IoT requirements in terms of availability, real-time data delivery, scalability, security, resiliency, and latency. We proposed a hybrid architecture to integrate the Blockchain, 5G, and IoT devices. The proposed system contains a public Blockchain and multiple local Blockchains. Public Blockchain is established by 5G MCs as their local area representative or cluster head, whereas MCs and SCs centrally manage local Blockchains. In addition, we utilize SDN and NFV technology to separate data from the control plane and software from hardware. The modified LSB algorithm is proposed to fit the 5G architecture, decrease the transaction verification rate and protect the network from cyber-attacks. We evaluate the performance of IoT devices in all possible scenarios via network simulation software such as NS3 and Cooja in terms of processing time and energy consumption. The simulation results show that our proposed architecture outperforms 5G conventional architecture in a local transaction. In public transactions, our scheme processes the transactions slower than 5G conventional architecture due to the massive routing and table checking. At the same time, the proposed consensus algorithm reduces the average processing time compared to the PoET algorithm. Nevertheless, the protection of the proposed Blockchain architecture against cyber attacks and the security of IoT devices are investigated with eleven different attacks.

Future research on the proposed architecture and consensus algorithm can enhance the scalability, resiliency, and transaction rate. One approach is to explore new consensus mechanisms that can support higher throughput and lower latency in the network. Another potential solution is to modify the current algorithm to decrease the processing time and overhead while maintaining high security. This could involve developing more efficient algorithms for block validation, network propagation, and transaction processing or optimizing the parameters of the existing consensus algorithm. Also, new Distributed Ledger Technologies (DLT) such as Directed Acyclic Graph (DAG) and Hashgraph can be explored as public Blockchain consensus approaches to provide a more scalable and resilient network with a higher transaction rate. The proposed architecture and consensus algorithm can be further improved by pursuing these research directions, making them better suited for decentralized applications and enabling a more efficient and robust blockchain system.

REFERENCES

[1] *Undrstanding the IoT Explosion and Its Impact on Enterprise Security*, Fortinet, Sunnyvale, CA, USA, 2017, pp. 1–4.
[2] I. Lee and K. Lee, "The Internet of Things (IoT): Applications, investments, and challenges for enterprises," *Bus. Horizons*, vol. 58, no. 4, pp. 431–440, Jul. 2015.
[3] Cisco. (2020). *Cisco Annual Internet Report (2018–2023) White Paper*. Accessed: Mar. 26, 2021. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/whitepaper-c11-741490.html

[4] V. Chamola, V. Hassija, V. Gupta, and M. Guizani, "A comprehensive review of the COVID-19 pandemic and the role of IoT, drones, AI, blockchain, and 5G in managing its impact," *IEEE Access*, vol. 8, pp. 90225–90265, 2020.
[5] M. Hung, "Leading the IoT gartner insights on how to lead in a connected world," 2017, pp. 1–29. [Online]. Available: https://www.gartner.com/imagesrv/books/iot/iotEbook_digital.pdf
[6] A. Gupta, R. Christie, and P. R. Manjula, "Scalability in Internet of Things: Features, techniques and research challenges," *Int. J. Comput. Intell. Res.*, vol. 13, no. 7, pp. 1617–1627, 2017.
[7] A. V. Dastjerdi, H. Gupta, R. N. Calheiros, S. K. Ghosh, and R. Buyya, "Fog computing: Principles, architectures, and applications," in *Internet of Things*. Amsterdam, The Netherlands: Elsevier, 2016, pp. 61–75.
[8] P. Raj and A. C. Raman, *The Internet of Things: Enabling Technologies Platforms and Use Cases*. Boca Raton, FL, USA: CRC Press, 2017.
[9] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.
[10] P. Corcoran and S. K. Datta, "Mobile-edge computing and the Internet of Things for consumers: Extending cloud computing and services to the edge of the network," *IEEE Consum. Electron. Mag.*, vol. 5, no. 4, pp. 73–74, Oct. 2016.
[11] R. Buyya and A. V. Dastjerdi, *Internet of Things: Principles and Paradigms*. Amsterdam, The Netherlands: Elsevier, 2016.
[12] A. Alrawais, A. Alhothaily, C. Hu, and X. Cheng, "Fog computing for the Internet of Things: Security and privacy issues," *IEEE Internet Comput.*, vol. 21, no. 2, pp. 34–42, Mar. 2017.
[13] Y. Ai, M. Peng, and K. Zhang, "Edge computing technologies for Internet of Things: A primer," *Digit. Commun. Netw.*, vol. 4, no. 2, pp. 77–86, Apr. 2018.
[14] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *Proc. 1st, Ed., MCC Workshop Mobile Cloud Comput.*, Aug. 2012, pp. 13–16.
[15] S. Yi, C. Li, and Q. Li, "A survey of fog computing: Concepts, applications and issues," in *Proc. Workshop Mobile Big Data*, 2015, pp. 37–42.
[16] A. Yousefpour, C. Fung, T. Nguyen, K. Kadiyala, F. Jalali, A. Niakanlahiji, J. Kong, and J. P. Jue, "All one needs to know about fog computing and related edge computing paradigms: A complete survey," *J. Syst. Archit.*, vol. 98, pp. 289–330, Sep. 2019.
[17] O. Salman, I. Elhajj, A. Kayssi, and A. Chehab, "An architecture for the Internet of Things with decentralized data and centralized control," in *Proc. IEEE/ACS 12th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Nov. 2015, pp. 1–8.
[18] I. A. Ridhawi, M. Aloqaily, A. Boukerche, and Y. Jaraweh, "A blockchain-based decentralized composition solution for IoT services," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.
[19] O. Alfandi, S. Khanji, L. Ahmad, and A. Khattak, "A survey on boosting IoT security and privacy through blockchain," *Cluster Comput.*, vol. 24, no. 1, pp. 37–55, Mar. 2021.
[20] J. C. Song, M. A. Demir, J. J. Prevost, and P. Rad, "Blockchain design for trusted decentralized IoT networks," in *Proc. 13th Annu. Conf. Syst. Syst. Eng. (SoSE)*, Jun. 2018, pp. 169–174.
[21] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1495–1505, Apr. 2019.
[22] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 3–16.
[23] J. Siim, "Proof-of-stake," Res. Seminar Cryptogr., Univ. Tartu, Tartu, Estonia, 2017.
[24] M. Maroufi, R. Abdolee, and B. M. Tazekand, "On the convergence of blockchain and Internet of Things (IoT) technologies," *J. Strategic Innov. Sustainability*, vol. 14, no. 1, pp. 101–119, 2019.
[25] L. Bittencourt, R. Immich, R. Sakellariou, N. Fonseca, E. Madeira, M. Curado, L. Villas, L. DaSilva, C. Lee, and O. Rana, "The Internet of Things, fog and cloud continuum: Integration and challenges," *Internet Things*, vols. 3–4, pp. 134–155, Oct. 2018.
[26] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "LSB: A lightweight scalable BlockChain for IoT security and privacy," 2017, *arXiv:1712.02969*.
[27] G. Sagirlar, B. Carminati, E. Ferrari, J. D. Sheehan, and E. Ragnoli, "Hybrid-IoT: Hybrid blockchain architecture for Internet of Things-PoW sub-blockchains," 2018, *arXiv:1804.03903*.
[28] J. Chen, "Flowchain: A distributed ledger designed for peer-to-peer IoT networks and real-time data transactions," in *Proc. 2nd Int. Workshop Linked Data Distrib. Ledgers*, 2017, pp. 1–11.
[29] N. Saxena, A. Roy, B. J. R. Sahu, and H. Kim, "Efficient IoT gateway over 5G wireless: A new design with prototype and implementation results," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 97–105, Feb. 2017.
[30] Q. V. Khanh, N. V. Hoai, L. D. Manh, A. N. Le, and G. Jeon, "Wireless communication technologies for IoT in 5G: Vision, applications, and challenges," *Wireless Commun. Mobile Comput.*, vol. 2022, pp. 1–12, Feb. 2022.
[31] G. A. Akpakwu, B. J. Silva, G. P. Hancke, and A. M. Abu-Mahfouz, "A survey on 5G networks for the Internet of Things: Communication technologies and challenges," *IEEE Access*, vol. 6, pp. 3619–3647, 2018.
[32] C. Mavromoustakis, G. Mastorakis, and J. M. Batalla, *Internet of Things (IoT) in 5G Mobile Technologies*. Cham, Switzerland: Springer, 2016.
[33] M. Samaniego and R. Deters, "Using blockchain to push software-defined IoT components onto edge hosts," in *Proc. Int. Conf. Big Data Adv. Wireless Technol.*, Nov. 2016, p. 58.
[34] P. K. Sharma, M. Chen, and J. H. Park, "A software defined fog node based distributed blockchain cloud architecture for IoT," *IEEE Access*, vol. 6, pp. 115–124, 2018.
[35] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "NFV: State of the art, challenges and implementation in next generation mobile networks (vEPC)," 2014, *arXiv:1409.4149*.
[36] W. Mougayar, *The Business Blockchain: Promise, Practice, and Application of the Next Internet Technology*. Hoboken, NJ, USA: Wiley, 2016.
[37] A. Haroon, M. Ali, Y. Asim, W. Naeem, M. Kamran, and Q. Javaid, "Constraints in the IoT: The world in 2020 and beyond," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 11, pp. 252–271, 2016.
[38] A. Musaddiq, Y. B. Zikria, O. Hahm, H. Yu, A. K. Bashir, and S. W. Kim, "A survey on resource management in IoT operating systems," *IEEE Access*, vol. 6, pp. 8459–8482, 2018.
[39] M. Conoscenti, A. Vetrò, and J. C. De Martin, "Blockchain for the Internet of Things: A systematic literature review," in *Proc. IEEE/ACS 13th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Nov. 2016, pp. 1–6.
[40] S. Singh, A. S. M. S. Hosen, and B. Yoon, "Blockchain security attacks, challenges, and solutions for the future distributed IoT network," *IEEE Access*, vol. 9, pp. 13938–13959, 2021.
[41] A. Reyna, C. Martín, J. Chen, E. Soler, and M. Díaz, "On blockchain and its integration with IoT. Challenges and opportunities," *Future Gener. Comput. Syst.*, vol. 88, pp. 173–190, Nov. 2018.
[42] A. A. Sadawi, M. S. Hassan, and M. Ndiaye, "A survey on the integration of blockchain with IoT to enhance performance and eliminate challenges," *IEEE Access*, vol. 9, pp. 54478–54497, 2021.
[43] L. Lao, Z. Li, S. Hou, B. Xiao, S. Guo, and Y. Yang, "A survey of IoT applications in blockchain systems: Architecture, consensus, and traffic modeling," *ACM Comput. Surv.*, vol. 53, no. 1, pp. 1–32, Jan. 2021.
[44] H. Desai, M. Kantarcioglu, and L. Kagal, "A hybrid blockchain architecture for privacy-enabled and accountable auctions," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 34–43.
[45] S. Abed, R. Jaffal, B. J. Mohd, and M. Al-Shayeji, "An analysis and evaluation of lightweight hash functions for blockchain-based IoT devices," *Cluster Comput.*, vol. 24, no. 4, pp. 3065–3084, Dec. 2021.
[46] A. Bogdanov, M. Knežević, G. Leander, D. Toz, K. Varıcı, and I. Verbauwhede, "SPONGENT: A lightweight hash function," in *Proc. Int. Workshop Cryptograph. Hardw. Embedded Syst.* Nara, Japan: Springer, 2011, pp. 312–325.
[47] J.-P. Aumasson, L. Henzen, W. Meier, and M. Naya-Plasencia, "Quark: A lightweight hash," in *Proc. Int. Workshop Cryptograph. Hardw. Embedded Syst.* Santa Barbara, CA, USA: Springer, 2010, pp. 1–15.
[48] C.-M. Chen, X. Deng, W. Gan, J. Chen, and S. K. H. Islam, "A secure blockchain-based group key agreement protocol for IoT," *J. Supercomput.*, vol. 77, no. 8, pp. 9046–9068, Aug. 2021.
[49] W. Serrano, "The blockchain random neural network for cybersecure IoT and 5G infrastructure in smart cities," *J. Netw. Comput. Appl.*, vol. 175, Feb. 2021, Art. no. 102909.
[50] H. Guo, W. Li, M. Nejad, and C. Shen, "Access control for electronic health records with hybrid blockchain-edge architecture," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 44–51.
[51] A. A. Laghari, A. A. Khan, R. Alkanhel, H. Elmannai, and S. Bourouis, "Lightweight-BIoV: Blockchain distributed ledger technology (BDLT) for Internet of Vehicles (IoVs)," *Electronics*, vol. 12, no. 3, p. 677, Jan. 2023.
[52] Nokia Corporation, "Evolution for IoT connectivity," Nokia, LTE, White Paper. Accessed: Mar. 28, 2021. [Online]. Available: https://halberdbastion.com/sites/default/files/2017-06/Nokia_LTE_Evolution_for_IoT_Connectivity_White_Paper.pdf

[53] J. Matias, J. Garay, N. Toledo, J. Unzilla, and E. Jacob, "Toward an SDN-enabled NFV architecture," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 187–193, Apr. 2015.
[54] D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015.
[55] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Trans. Inf. Theory*, vol. 22, no. 6, pp. 644–654, Nov. 1976.
[56] H. Zimmermann, "OSI reference model—The ISO model of architecture for open systems interconnection," *IEEE Trans. Commun.*, vol. COM-28, no. 4, pp. 425–432, Apr. 1980.
[57] L. Chen, L. Xu, N. Shah, Z. Gao, Y. Lu, and W. Shi, "On security analysis of proof-of-elapsed-time (PoET)," in *Proc. Int. Symp. Stabilization, Saf., Secur. Distrib. Syst.* Boston, MA, USA: Springer, 2017, pp. 282–297.
[58] V. Costan and S. Devadas, "Intel SGX explained," IACR Cryptol. ePrint Arch., Boston, MA, USA, 2016, vol. 2016, no. 86, pp. 1–118.
[59] (2019). *NS3-Nsnam*. [Online]. Available: https://www.nsnam.org/
[60] M. Mezzavilla, S. Dutta, M. Zhang, M. R. Akdeniz, and S. Rangan, "5G MmWave module for the ns-3 network simulator," in *Proc. 18th ACM Int. Conf. Modeling, Anal. Simulation Wireless Mobile Syst.*, Nov. 2015, pp. 283–290.
[61] N. Baldo. (Mar. 2011). *The NS-3 LTE Module by the LENA Project*. Accessed: May 23, 2019. [Online]. Available: https://www2.nsnam.org/tutorials/tutorials/consortium13/lte-tutorial.pdf
[62] T. Long and P. Veitch. *A Low-Latency NFV Infrastructure for Performance-Critical Applications*. [Online]. Available: https://software.intel.com/en-us/articles/low-latency-nfv-infrastructure-for-performance-critical-applications
[63] *Cooja Network Simulator*. Accessed: May 21, 2020. [Online]. Available: http://www.contiki-os.org/start.html

MOHAMMAD MAROUFI received the B.Sc. degree in electrical engineering from Bu-Ali Sina University, Hamedan, Iran, in 2013, and the M.Sc. degree from the University of Tabriz, Iran, in 2015, where he is currently pursuing the Ph.D. degree with a focus on researching DLT, including blockchain, directed acyclic graph (DAG), and hashgraph. His investigations aimed to enhance the security and efficiency of the IoT networks by utilizing DLT technologies in conjunction with cellular networks. Furthermore, he is exploring the application of DLT in wireless sensor networks (WSN) and studying the signal processing of underwater acoustic (UWA) systems. His research interests include the intersection of DLT and the IoT-enabled devices and networks.

REZA ABDOLEE received the Ph.D. degree in computer and electrical engineering from McGill University, in 2014.

He is an accomplished computer science and electrical engineering professional with an extensive experience in both academia and industry. He is currently the CEO of Novesh Cybersecurity Solution, where he provides innovative security solutions to operational technology (OT) and critical infrastructure sectors. He is also an Associate Professor with California State University Channel Islands (CSUCI), where he inspires and guides the next generation of engineers and computer scientists. He was a Faculty Member with the Department of Electrical Engineering, University of California Santa Barbara (UCSB), from 2019 to 2022. He has collaborated on numerous NSF-funded projects, and his contributions to wireless communications and the IoT have led to several patents and inventions. He has received numerous scholarships and awards for his work, including the USDA Grant, the DoD Award, the NSF and NSERC Postgraduate Award, the FQRNT Doctoral Research Scholarship, the DAAD-RISE Professional Scholarship, the ReSMiQ Supplementary Scholarship, and the SYTACom Industrial Collaboration Award. In addition to his expertise in cybersecurity, his research interests also include computer architecture and design, with an emphasis on performance analysis and optimization.

BEHZAD MOZAFFARI TAZEKAND received the B.Sc. degree from the University of Tabriz, in 1993, the M.Sc. degree from the K. N. Toosi University of Technology, in 1996, and the Ph.D. degree from the University of Tabriz, in 2006. He is a distinguished professor of communication engineering, with a wealth of knowledge and experience in his field. He is highly regarded for his expertise and accomplishments in the field of communication engineering. His contributions were widely recognized. Throughout his education and career, he has developed a strong focus and expertise in wireless communication, orthogonal frequency division multiplexing (OFDM) systems, and signal processing for communication systems. He has dedicated himself to understanding these complex topics and has made significant contributions to the field. His research interest includes testament to his passion for advancing the field of communication engineering. His expertise and achievements in this area have earned him recognition and respect among his peers.

SEYED AMIR MORTEZAVI received the B.Sc. degree in electrical engineering from Tabriz University, in 2008, and the M.S. and Ph.D. degrees in cryptography from the Sharif University of Technology, Tehran, Iran, in 2010 and 2018, respectively. He is currently an Assistant Professor with the University of Tabriz. His research interests include security, design, and cryptanalysis of cryptographic algorithms and protocols, provable security, and distributed systems.