Hindawi
Security and Communication Networks
Volume 2021, Article ID 6671132, 11 pages
https://doi.org/10.1155/2021/6671132

Research Article

A Novel Classified Ledger Framework for Data Flow Protection in AIoT Networks

Daoqi Han,[1] **Songqi Wu**,[1] **Zhuoer Hu**,[1] **Hui Gao**,[1] **Enjie Liu**,[2] **and Yueming Lu**[1]

[1]*Key Laboratory of Trustworthy Distributed Computing and Service, Ministry of Education, Beijing University of Posts and Telecommunications, Beijing, China*
[2]*University of Bedfordshire, Institute for Research in Applicable Computing (IRAC), Luton, UK*

Correspondence should be addressed to Yueming Lu; ymlu@bupt.edu.cn

Received 26 December 2020; Revised 11 January 2021; Accepted 4 February 2021; Published 19 February 2021

Academic Editor: Liguo Zhang

The edge computing node plays an important role in the evolution of the artificial intelligence-empowered Internet of things (AIoTs) that converge sensing, communication, and computing to enhance wireless ubiquitous connectivity, data acquisition, and analysis capabilities. With full connectivity, the issue of data security in the new cloud-edge-terminal network hierarchy of AIoTs comes to the fore, for which blockchain technology is considered as a potential solution. Nevertheless, existing schemes cannot be applied to the resource-constrained and heterogeneous IoTs. In this paper, we consider the blockchain design for the AIoTs and propose a novel classified ledger framework based on lightweight blockchain (CLF-LB) that separates and stores data rights at the source and enables a thorough data flow protection in the open and heterogeneous network environment of AIoT. In particular, CLF-LB divides the network into five functional layers for optimal adaptation to AIoTs applications, wherein an intelligent collaboration mechanism is also proposed to enhance the across-layer operation. Unlike traditional full-function blockchain models, our framework includes novel technical modules, such as block regenesis, iterative reinforcement of proof-of-work, and efficient chain uploading via the system-on-chip system, which are carefully designed to fit the cloud-edge-terminal hierarchy in AIoTs networks. Comprehensive experimental results are provided to validate the advantages of the proposed CLF-LB, showing its potentials to address the secrecy issues of data storage and sharing in AIoTs networks.

1. Introduction

Current Artificial Intelligence Internet of Things (AIoTs) systems, 5G communications, and cloud computing are deeply integrated, which effectively enhances wireless ubiquitous connectivity, data acquisition, and analysis capabilities. In particular, in the IoT environment, various wireless networks, such as 5G, WiFi, and BLE, can achieve high-speed point-to-point communication and then conduct various functional subnetworks, enabling collaboration among heterogeneous nodes to support various types of services [1–3].

Through ubiquitous sensing and connectivity, all kinds of data can be efficiently collected and transported at high speed. Meanwhile, data security risks are also increasing. Specifically, in the new "cloud-edge-terminal" environment, although the central cloud has massive data storage capacity [4–7], it is unable to protect the terminal and user-side source data in an open environment and lacks a mechanism to promote peer-to-peer secure data sharing among users. During the process of data transportation, the data are vulnerable to malicious modification and duplication, and data rights are difficult to be guaranteed. To this end, there is an urgent need to study the mechanisms and methods for validating and managing the data rights throughout the life cycle in the AIoTs.

Blockchain technology conducts a network in which everyone can autonomously store transaction data anonymously and securely, is open, self-organizing, and difficult to be attacked. Furthermore, it can solve the abovementioned management issues of data rights, thus receiving increasing attention from industry and academia. In particular,

blockchain technology uses different policies that are replicated to each node as much as possible [8], open distributed storage network services [9], value incentive and flow capabilities [10], and self-organizing fault tolerance [11], in order to enable data protection mechanisms for open environments. However, traditional blockchain technology using a decentralized homogeneous model leads to a high volume of communication overheads. Moreover, disordered competition leads to high energy consumption of consensus. Thus, the traditional blockchain cannot take full advantage of the characteristics of the IoT to slice functional modules and flexibly deploy them to various layers so as to adapt to environmental constraints. Meanwhile, it is unable to classify and store different structural ledgers and inefficient book keeping and retrieval. Therefore, there is a need to design lightweight frameworks for flexible deployment in MEC.

A number of lightweight IoT-oriented blockchain designs have emerged recently. The new designs are able to basically achieve on-demand elastic scheduling of computing resources [12], weighing data transmission and storage between terminal networks, edge networks, and cloud platform networks so as to form blockchain storage capabilities that are suitable for this edge-computing architecture. Among them, simplification of the blockchain's consensus, ledger structure, P2P, and other mechanisms are the focus of the current research attention. Specifically, Liu et al. [13] study lightweight blockchain systems for the Industrial Internet of Things (IIoTs), propose green Co-PoW green collaborative consensus, and design LightBlock's lightweight data structure to simplify broadcast content which uses irrelevant block offload filters to reduce ledger blocks. However, the lack of systematic analysis of the cloud-edge-terminal collaboration limits scalability. Then, Liu [14] also proposes the heterogeneous and resource-limiting features of IoTs, constructs an independent Tornado P2P network layer, utilizes the current sensor network and management network hierarchical structure of IoT, and divides two levels to compute and pack macro-/microblocks, respectively, which uses space-structured ledger to extend the performance, reaching the maximum throughput of 3464.76 transactions per second. However, it is still energy-consuming of elections, synchronized consensus and ledger across the network, and full data storage ledger. Ferrag et al. initially investigate and propose future research areas such as dynamic adaptive security framework, social network and trust management, and specific blockchain infrastructure in terms of security goals, performance, limitations, computational complexity, and communication overheads [15]. However, there is a lack of viable framework recommendations, failing to propose specific functions of blockchain infrastructure and how to achieve orderly management in conjunction with edge computing.

In summary, the relevant research has achieved some innovations, which are limited by the current state of IoTs infrastructure. In addition, the new challenges we consider cannot yet be adequately addressed, such as the problem of efficient data storage and the coherent protection of privacy and data rights in AIoTs scenarios. In this paper, we propose a multiledger framework (CLF-LB) for future 5G/6G fully connected scenarios to address the problems of the current blockchain framework, such as no classification structure, no management, and disordered competition. Specifically, we propose a "cloud-edge-terminal" division of multilayer and multiclassification storage networks. Then, we lighten the blockchain ledger for local real-time processing. We also realize that orderly management can reduce workload by gradually reinforcing the security level on the demand. The results can take full advantage of the new architecture's full connectivity, intelligent perception, data portrayal, and other characteristics.

This research focuses on analyzing the real-time classification storage requirements of localized blockchains in the IoT network. Different ledger structures are constructed based on the stored data subjects and data rights.

Compared with the existing unimodal [10], unordered competitive consensus [16], and all-data storage blockchain for IoTs [14], CLF-LB has the following novel contributions and main advantages:

1.1. Multilayer and Multiclassification Framework. A hierarchical and clustered multiledger blockchain network, with a convergent approach that takes both local efficiency and the massive processing power of the cloud platform into account, focuses on the extraction and protection of data interests.

1.2. Real-Time Data Rights Ledger. A ledger daily inherits and implements regenesis strategies with forkless, fixed, prefabricated minute blocks.

1.3. Iterative Reinforcement of Consensus. A lightweight and incremental consensus can maximize the security in resource-constrained scenarios.

The remainder of this paper is organized as follows: In Section 2, we present related work and our contribution. In Sections 3 and 4, we propose our algorithms and models. The experimental results and a scheme comparison are described in Section 5. We make a conclusion in Section 6.

2. Related Work

In terms of decentralized blockchain design, unimodal networks lack authoritative mechanisms, and the competitive consensus strategy is adopted to ensure the consistency and correctness of each transaction on all ledger nodes; however, the resource consumption is quite high. The commonly used consensus algorithms include proof-of-work (PoW), proof-of-stake (PoS) [16], and delegated proof-of-stake (DPoS) [17]. The PoW mechanism consumes a large amount of energy, which severely limits the transaction throughput. In addition, PoS shortens the time it takes to reach consensus among nodes and avoids the massive waste of resources caused by mining; however, it undermines fairness by discouraging "poorer" participants and allowing the "richest" stakeholder to have complete control over the block generation.

Moreover, based on PoS, the DPoS mechanism solves the high-energy consumption problem of POW and avoids the "trust-balance" bias possible under PoS, but is not decentralized enough. In addition, the zero-knowledge proof algorithm [18, 19] simplifies the PoW and also increases the difficulty of attacks, which further enhances the privacy and security of consensus strings.

In terms of lightweight blockchain design, with the increasing number of IoTs devices, there are huge security risks associated with storing massive amounts of information. Due to high-energy consumption and large processing overhead, the existing blockchain architectures are not suitable for IoTs scenarios. A lightweight blockchain architecture can be achieved on the premise of ensuring data security and privacy through strategies such as heterogeneous environment adaptation, RAFT consensus [20], eliminating tailoring means such as miner and value records [21], and dividing subnets [14].

In terms of adaptation to heterogeneous environments, MEC has recently gained widespread attention for data processing, data analysis, and data storage in heterogeneous Internet of Things (H-IoTs) scenarios [22]. Furthermore, MEC takes full advantage of the computing power of edge nodes, greatly reducing the computational pressure on data centers and facilitating the storage and processing of big data. Due to the lack of management of distributed nodes, edge nodes are easy targets of hacking. Differential privacy-based machine learning strategies address the privacy issues in edge computing from both data aggregation as well as data mining [23]. Wang et al. [24] studied the integrated framework of computational offloading and interference management in MEC-enabled wireless cellular networks to support the accomplishment of indivisible computational tasks. Dinh et al. [25] show that it is possible to perform computational tasks without requiring prior system knowledge, but instead, with the help of emerging reinforcement learning (RL) algorithms that can optimally learn the dynamic computational triage strategy. Yan et al. [26] model the process of offloading and caching to ensure that both edge nodes and edge computing service providers obtain the maximum profit based on game theory and auction theory. Gong et al. [27] present an intelligent cooperative edge (ICE) computing in IoTs networks to achieve a complementary integration of AI and edge computing. Wei et al. [28] used a deep reinforcement learning algorithm, which combines RL method Q-learning with the deep neural network (DNN) to approximate the value functions for complicated control applications, and the optimal policy will be obtained when the value function reaches convergence. They find that the computation offloading and content caching achieve a better solution using localized AI.

In terms of tailoring adaptation, the transaction throughput of traditional blockchains is too low to meet the high scalability requirements of IoTs. The tailor-made lightweight blockchains simplify the processing of key modules and enhance the scalability of the blockchain without changing the model structure. Karlsson [29] proposed a blockchain with a directed acyclic graph (DAG) structure, which supports asynchronous concurrent writing of single-user transactions, thereby reducing the storage requirements of data, while enabling the tracking of data sources and the creation of shared tamper-proof data repositories. Since it is impossible to simultaneously ensure consistency, availability, and partition tolerance in a blockchain system, the system's design often needs to weaken the guarantee of a particular feature. Karlsson also adopted a partition-tolerant blockchain "Vegvisir blockchain" for power-constrained, network-connected IoTs environments. Hassija et al. [30] proposed an IoTs network based on a lightweight blockchain protocol using the Tangle data structure, which replaces the traditional chained data structure to record transactions in the network in a secure and scalable manner. The model is highly scalable as it does not require extensive computation to add transactions to a block, nor does it require any transaction compensation charges. As the data in a block grows dramatically, it can incur significant communication and storage overhead. Therefore, a fragmented ledger is used to store the relevant detailed data, and the storage structure of the shared ledger is lightened to achieve reliable and traceable event analysis with minimal storage and processing overhead [31]. Based on this, Yang W [32] proposed a social-based data simplification approach, a directed acyclic graph (DAG) lightweight blockchain model, where each node stores only the data of interest and ignores irrelevant data to reduce the number of duplicate data in the blockchain. The experimental results showed that the model saves 97.13% of the storage space. With a lightweight blockchain, Xin Jiang et al. [33] proposed a new blockchain-based authentication protocol for WLAN mesh security access. It takes the user's authentication request as a transaction, considers all the authentication records in the mesh network as the public ledger and realizes the effective monitoring of the malicious attack. For ensure the security of data transmission in IoTs, Hui et al. [34] applies a new chaotic secure communication scheme to address the security problem of data transmission. The scheme is based on the synchronization of different-structure fractional-order chaotic systems with different orders.

Our scheme proposes using heterogeneous communication chips to directly extract metadata and equity data for uplink, balancing security, and resource consumption to achieve a new hierarchically deployed multimodal blockchain network.

3. System Model

In this section, we take the AIoT scenario as a case study of the CLF-LB scheme. Next, we give a detailed description of the essential multimodal network. Finally, we design a light block structure called regenesis. Table 1 lists the explanation of the symbols associated with the CLF-LB scheme.

It is known that Bitcoin and Ethereum are already severely limited by the amount of data because there are too many historical blocks. Hundreds of gigabytes of data and days-long block download tasks have prevented new full-featured nodes from being added. The workloads for data traceability, package validation, and energy consumption are

Table 1: Notations in the CLF-LB scheme.

Symbol	Notation
CRS	Common reference strings set
HCF	Hierarchical and classified framework
MNET	Multimodal network
AN	Access control network
CN	Wireless cache network
DN	Decision network
PN	Proof-of-work network
SN	Storage network
BRG	Block regenesis
TBLOCK	Temporary timestamp blocks
RBLOCK	Reinforcement emphasis blocks
CBLOCK	Permanent chain blocks
RPOW	Iterative reinforcement of proof-of-work
EDCC	Extract data rights on the chain by chip
BT	Bittorrent
BLE	Bluetooth low energy
WMN	Wireless mesh network
Zk-SNARKs	Zero-knowledge succinct noninteractive argument of knowledge algorithms

increasing. Light weighting the data that need to be processed in real time on the same day is a key point for the adaptability of blockchain technology to IoTs environments.

3.1. Hierarchical and Classified Framework. The proposed framework is depicted in Figure 1. In IoTs sensors, multimode communication chips can be used as a specific infrastructure to perform uplink processing directly. The smart devices are equipped with intelligent grouping and application access control; they also participate in blockchain services such as caching, packaging, and PoW task processing, as well as providing multiledger cluster caching services.

We implement a local-area blockchain network at the network-edge layer, lightly pregenerate a total of 1440 fixed blocks per minute, and regenerate previous blocks daily. We elected high-performance nodes to run multiple ledgers, schedule the PoW network to intelligently decompose tasks, and assign them to idle nodes for execution.

In a cloud storage system, raw data and metadata such as data rights and historical blocks are stored, and the daily chain is compressed into a single block that is concatenated day-by-day based on the structure of the public chain and PoW consensus.

To achieve a timely and flexible security level, we define the TBLOCK process of PoW for small-scale AIoT to find nonce as follows:

$$\begin{aligned}&\text{SHA256}\,(\text{SHA256}\,(\text{gensis meta data} + \text{block header}\\&\quad + \text{nonce})) < \text{target}\,(20\,\text{bit zero}).\end{aligned} \tag{1}$$

The indicators of difficulty and resource utilization control the effectiveness of the PoW network. For the process of confirming block to writing into the blockchain, we sample every ten minutes to strengthen the RBLOCK, defining the PoW as follows:

$$\begin{aligned}&\text{SHA256}\,(\text{SHA256}\,(\text{block header} + \text{root of merkel tree}\\&\quad + \text{nonce})) < \text{target}\,(24\,\text{bit zero}).\end{aligned} \tag{2}$$

Eventually, a CBLOCK storage in a cloud system should merge 1441 fixed blocks. It builds a new Merkel tree with the hash of each block. We define the PoW as follows:

$$\begin{aligned}&\text{SHA256}\,(\text{SHA256}\,(\text{block header}\\&\quad + \text{root of merkel tree} + \text{nonce}\,)) < \text{target}\,(28\,\text{bit zero}).\end{aligned} \tag{3}$$

3.2. Multimodal Network. We build a hierarchical and clustered multiaccount blockchain network for the three-layer structure, which is composed of an IoTs terminal, edge network, and cloud platform. We delineate five categories of different characteristics and multimodal regions to select the appropriate data processing network so as to match the appropriate participating nodes' networking in different environments.

We distributed the deployment of five different kinds of node networks in three layers to collaborate on data protection services as follows:

3.2.1. A Network (Access Control Network). This network is used mainly to strengthen authentication services in IoTs and protect data [35, 36]. It utilizes blockchain's end-to-end zero-knowledge proof for verification and consensus, to defend against network layer attacks and malicious tampering.

3.2.2. C Network (Wireless Cache Network). The C network is a real-time transaction caching network that clusters transactions in terms of ledger type, caches real-time transactions, and realizes batch verification of transactions.

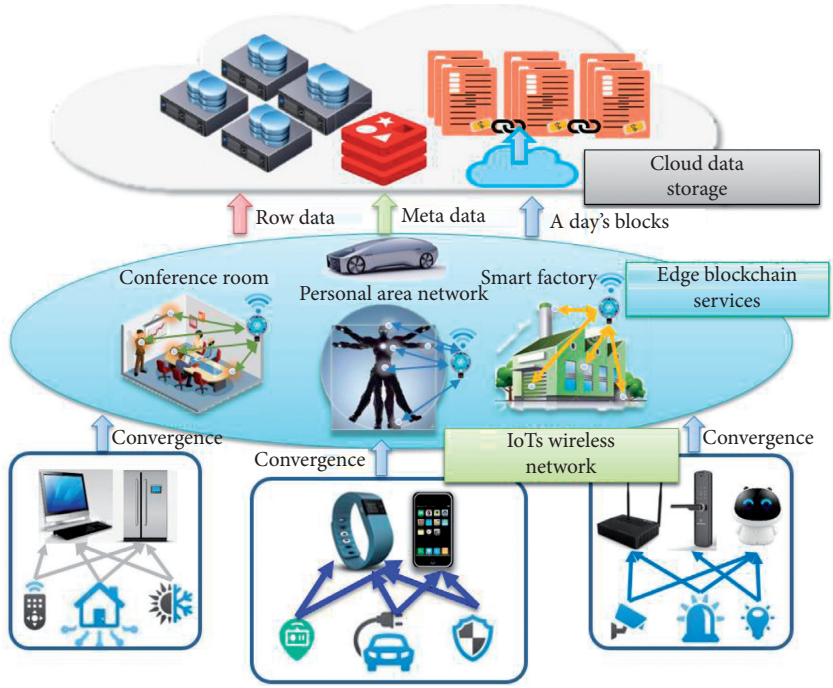

Figure 1: Hierarchical classified blockchain storage based on edge computing.

3.2.3. D Network (Decision Network). This is a local, lightweight, and packaged block network. We pack multichain block data in parallel across multiple centers and store equity data. This network realizes lightweight transmission, computation, and storage of data based on the requirements of the IoTs resource-constrained environment. Once the data are packaged, the cache will be cleaned periodically, and the real-time data in the backup cycle, and daily block formation files will be asynchronously uploaded to cloud storage.

3.2.4. P Network (PoW Network). This is a task-oriented PoW computing network. The network scheduling approach combines computing resources for producing proof-of-workload strings and discovering trusted and efficient nodes. A self-organizing approach is implemented to manage the nodes and addresses of the distributed network and realize the optimization of hierarchical connections.

3.2.5. S Network (Storage Network). The S network is a massively distributed storage network on the cloud platform designed to achieve massive and efficient storage of backup data, thereby solving the capacity constraint problem. The network stores the daily packed backup block files and data files. The blocks are concatenated into a permanent super chain via block references for multiple cycles. The PoW consensus and block-based Merkel Tree are used to protect integrity.

Our model is constructed from the tuple (A, C, D, P, and S). C->D->S cyclic iterations are used to accumulate historical operational data for subsequent decision analysis. The C and P networks collaborate to provide the basis for judging pending tasks and processing capacity. In the D network, intelligent nodes can dynamically assign tasks based on task queue length and adjust the difficulty of workload calculations. The D and S networks collaborate to balance network occupancy and local storage volume thresholds and are responsible for asynchronous gradual offloading of data to cloud storage. The P and D networks are mutually constrained to optimize the control of the D network by participating in the voting election process of the nodes.

In particular, the collaborative process of C, D, and S requires constant interaction with the environment to obtain multiparty states. It is a complete-information and cooperative-game model of the supply-chain pipeline. We define the following reward function (4) that can train the Deep Reinforcement Learning (DRL) model optimizing actions policy balancing the utilization of storage and network resources. The expected resource-occupancy rate is stable in the range of 30% to 70%. The corresponding positive reward is the proportional value of the positive rate of the change generated by the scheduling algorithm, and vice versa. When the occupancy rate is more than 70%, the model should accelerate the offloading action. When the occupancy rate is less than 30%, the model should increase the new tasks. The defined offload is the reward generated by the offloading action; the defined upload is the reward generated by the adding task action; and the defined loss is the state value to be deducted when the computing, storage, and network resources exceed 70% of the threshold.

$$R = \text{offload}(\text{ao}) + \text{upload}(\text{at}) - \text{loss}(\text{over\_cpu}, \text{over\_storage}, \text{over\_net}). \quad (4)$$

The matching DRL method, asynchronous advantage actor-critic (A3C) [37], can iteratively optimize the

scheduling strategy for hierarchical caching and transmission of data. The advantage function A (s, a) trains the actor-police model to maximize the benefits of the action. The loss function Q (s, a)-V (s) trains the critic-value model to evaluate the value of the current state. Finally, a unified model can be trained to output the probability distribution of actions and the value of the state, respectively. Using the loss function (5) composed of the following three parts (6–8) such as policy loss, value loss, and regularization with policy entropy, the deep learning framework minimizes this summarized loss.

$$L = L_\pi + \alpha L_v + \beta L_r, \tag{5}$$

$$L_\pi = -\frac{1}{n}\sum_{i=1}^{n} A(s_i, a_i) \cdot \log \pi(a_i|s_i), \tag{6}$$

$$L_v = \frac{1}{n}\sum_{i=1}^{n}\left(\left(\sum_{j=0}^{k-1}\gamma^j r_{i+j} + \gamma^k V(s_{i+k})\right) - V(s_i)\right)^2, \tag{7}$$

$$L_r = -\frac{1}{n}\sum_{i=1}^{n} H(\pi(s_i)). \tag{8}$$

Note that the collaboration between D and P is a typical two-stage Stackelberg game. The model needs to maximize utility for flexibly control workload while adapting the appropriate level of security. On the supply side, P maximizes the reception and completion of tasks and accumulates the workload for improving the level gradually. On the demand side, D needs to maximize scheduling reward and accumulates voting credit by honestly recording the contributions of the worker in the P network.

3.3. Block Regenesis. The CLF-LB scheme includes a mechanism to regenerate blocks on a daily basis by collaboration. The six-step processing flow of the data stream is shown in Figure 2.

3.3.1. Phase of Aggregation of Transaction Records

Step 1. The security network manages access control and privileges of its applications. Here, we organize IoT nodes, manage identities and permissions, and provide access control mechanisms. Mainly, we strengthen authentication services in the IoT to protect data and set up public reference strings to reinforce confidentiality. Next, we schedule the PoW network according to the security-level requirements to produce the PoW required for the corresponding level of authentication string.

Step 2. The real-time transaction caching network receives the broadcasted transaction records. Various types of nodes in the IoT can simultaneously send the certificate of deposit transactions to all nodes in the caching network via a wireless broadcast mechanism so as to cache real-time transactions. A transparent infrastructure layer is formed through chip-cured communication management, data entitlement extraction, and an uplink process. To be able to respond to end users in real time, an in-memory caching network is built on the smart-device or edge-gateway side. Real-time clustering is used to cache different transaction records and real-time streaming data to verify data consistency, integrity, and authenticity.

3.3.2. Phase of Blockchain-Service Processing

Step 3. The local lightweight block network sorts blocks. Multiple local chains storing different ledgers will elect the appropriate central node by RAFT consensus. The multiple centers pack multiple chains and different classifications of block data in parallel and store equity data. The central nodes first lightly prefabricate fixed blocks per minute of the current day and then regenerate the blocks on a daily basis after inheriting the previous day's balance pencils and other summary information.

We assign tasks to the PoW network to produce workload-proof strings, ranging from easy to difficult. The transaction records in the cache are sorted by timestamp order and recorded into blocks with different sequence numbers at regular intervals every minute. The status is determined 10 minutes later; the final block to be packaged is fixed and generated for posting to the following nodes.

Step 4. The local lightweight block network parcels blocks. Subsequent nodes in the RAFT network accept the blocks to be packaged for local storage. This step includes the P2P network method, which actively propagates the block header chain table to all participating PoW nodes, and the master node modifies the status to pack complete according to the number of acknowledgments in the RAFT network. This step also provides download services for individual blocks and individual transaction records.

3.3.3. Phase of Storing in Cloud for Recycling

Step 5. The periodic tasks submit data to the cloud storage network. The video stream in the IoTs needs to be segmented and cached locally. The corresponding metadata and data entitlements are recorded in the block, and after completing the packaging and archiving, the task is submitted asynchronously. The backend thread progressively advances the task and is responsible for uploading each segmented data to the cloud storage network. After uploading, the local backup file can be deleted.

Step 6. The leader packs the chain file daily submitting it to the cloud storage network. Historical data in daily blocks are packaged and compressed into a chain file, as a block on the permanent super chain, which is then uploaded to the cloud storage network. The daily packaged backup block file is stored in the cloud and concatenated into a permanent super chain by block referencing for multiple cycles, thereby protecting the integrity and preventing data tampering through PoW consensus. As the public chain policy,

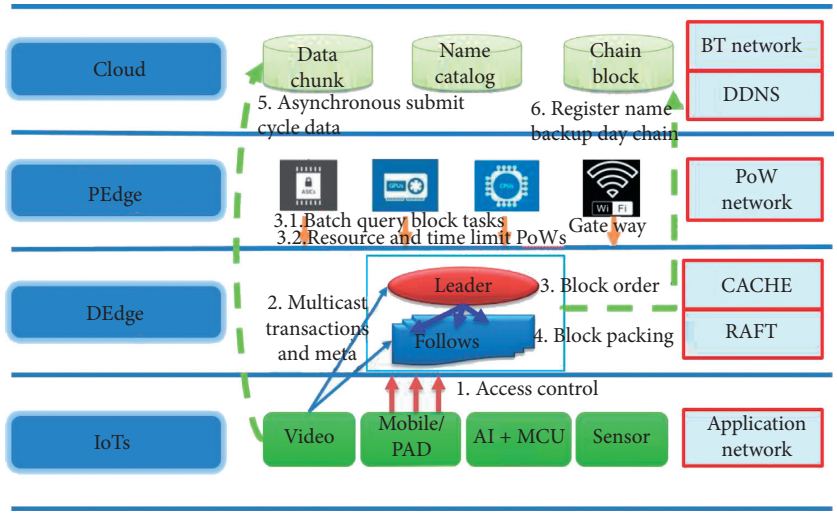

Figure 2: Process of data block regenesis.

consensus concatenates multiday blocks to ensure that data for more than six days is essentially impossible to roll back. Super chains are stored in the cloud distributed storage, reducing local storage space pressure and redundancy.

4. Dynamic Adaptivity

4.1. Iterative Reinforcement of PoW. The primary step is to build a dynamically joinable and task-oriented PoW computational network. Participating nodes are able to perform tasks according to their capabilities, either passively or proactively, by querying the blocks to be packaged and submitting workload proofs on a regular basis. The network provides computing power services for prepacking local blocks, applying secure access tokens, PoW consensus, and other secure processing based on PoW.

Unlike PoW networks that maximize network workload goals and set the corresponding difficulty, RPoW determines the benchmark workload by network capacity and security-level definitions and without using PoW to compete for book-keeping rights and rewards. The lightweight and incremental batch production of workload-proof strings gradually increased the verification difficulty to maximum security difficulty in resource-constrained scenarios while meeting the constraints of the task's deadline.

As a security support mechanism, computational nodes focus on participating and providing capacity indicators. Node trustworthiness is enhanced through multidimensional indicators such as capacity, participation time, and contribution. Open mechanisms for sharing computing power and contributing resources can improve nodes and address the management of distributed networks. With the participation of more trustworthy nodes, the connection management capability of each network can be improved, and the synchronization data capability of the network can be improved so that the packaged nodes can be compared and selected.

The reinforcement consensus makes the P network more flexible for the security goal by asynchronous iterating. The D network makes the decision for high-effect utilize the resources of the P network through intelligent perception. The optimization goal of the scheduling algorithm is to maximize the computational difficulty (CD) of the consensus string. The constraints include limited time (LT), limited resources (LR), and the number of tasks that will be submitted at the next time frame (TS). The main parameters with which the schedular judges the current difficulty are the priority sequencing queue length of the task (QT) and the current basic difficulty factor (BF). By accumulating the historical multicycle time (CT) and the number of transactions (CN), the peak time-frame range (PT) by daily predicting is taken as the fixed BF to maintain the minimum input task. Using the current observation value, the Kalman filter algorithm can continuously smooth the estimated value and predict the transaction number (PN) in each minute, which is the basis for scheduling to increase the workload. The optimal workload of the ith minute is calculated by the following formula:

$$\begin{aligned} \max \quad & CD[i] = A(BF, QT) + \gamma(S(P[i])) + \gamma(S(P[i+1] \\ & \quad + \cdots \gamma S(P[n]))), \\ & LT = \text{True}, \\ \text{s.t.} \quad & LR = \text{True}, \\ & TS = \text{True}. \end{aligned} \tag{9}$$

The framework of the algorithm is shown in Figure 3.

4.2. Extract Data Rights on the Chain by Chip. Based on the blockchain computation and connection embedded in the chip, we realized the automatic uplink processing. Embedded in the 5G/WiFi/BLE module chip of IoTs, processing such as trusted registration and statistical service data flow, uplink communication processing, and security cryptographic signature is used to achieve anticounterfeit control and an efficient real-time uplink mechanism. The data generated by resource-constrained terminals can be directly

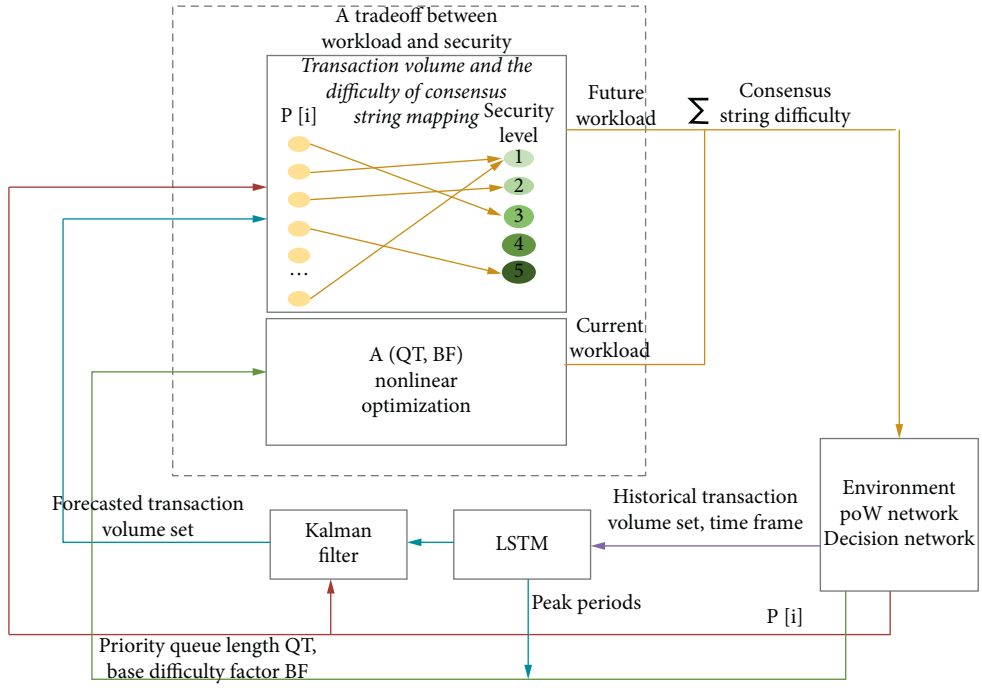

Figure 3: The framework of dynamic adjustment of difficulty factors.

uploaded through the blockchain module embedded in the chip, without the need for application development and occupying terminal resources.

The first step to realizing this is trusted registration processing, registering the identity within the application security network. The unique identification and location information of the chip can be used as the registration evidence of the service to construct the identity of each service for which the communication chip assumes data transmission, including address, public key, and private key.

The second step is to authorize the establishment of an access relationship within the application security network, apply an authentication token, and access the data service.

The chip can classify service data traffic information according to the predefined service types and regularly count the transmitted service data traffic information, including the service object, service start time, service traffic statistics, and service duration. Using the identity of the service, the subsequent uplink processing is triggered at regular intervals. The data volume and service statistics transmitted by the chip are deposited daily into the proof blockchain. The contextual information of these services is trustworthy and difficult for other devices to falsify.

The last step is that the terminal application extracts user rights and privacy information at the data collection stage, collects statistics regarding the characteristics of the collected data, forms a transaction record, calls the chip API interface of the uplink process, and triggers the uplink processing operation of the chip.

The specific uplink communication process is as follows:

(1) The system utilizes a neighbor discovery protocol for wireless communication and discovers and manages various types of cached service addresses in the vicinity. Then, it evaluates and ranks them in terms of service quality.

(2) The user information and privacy fields are encrypted, and the uplink data are signed.

(3) The RPC protocol format is used for encapsulating uplink transactions. The system picks the first n cache service nodes based on the COAP protocol for IoT. It then uses the packet broadcasting protocol to broadcast and send transaction records to multiple cache server addresses.

5. Evaluation

5.1. Implementation. We build a multimachine Fabric1.4 network environment based on RAFT consensus, using the REST API to store evidences, trace the source by caches, and trace by blocks. These services can also trace data replication, along with the addition and deletion of data rights.

5.2. Testbed. The devices comprising the network structure and the computing power of each node are shown in Table 2.

5.3. Influence of the Number of Nodes. To evaluate the influence of the number of nodes, we execute the evidence storage service 1,000 times in each scenario. As shown in Figure 4, the number of participating nodes gradually increases from one node to 40 nodes, which increases the average latency of evidence storage and sharply decreases the performance of TPS.

Table 2: Experimental devices.

Type	Specification	Hashrate (MH/s)	Concurrency	Number
MCU	ARM cortex-A7 CPU @ 1.60 GHz 4C/1G	0.16	2	1
Edge devices	Intel (R) xeon (R) platinum 8269CY CPU @ 2.50 GHz 2C/8G	1.1	1	1
	Intel (R) xeon (R) gold 6278C CPU @ 2.60 GHz 4 C/8G	1.2	3	1
	Intel (R) core (TM) i7-6700HQ CPU @ 2.60 GHz 4 C/8G	1.3	4	1
Server	Intel (R) xeon (R) E5 CPU @ 3.20 GHz 16C/32G	1.4	16	1

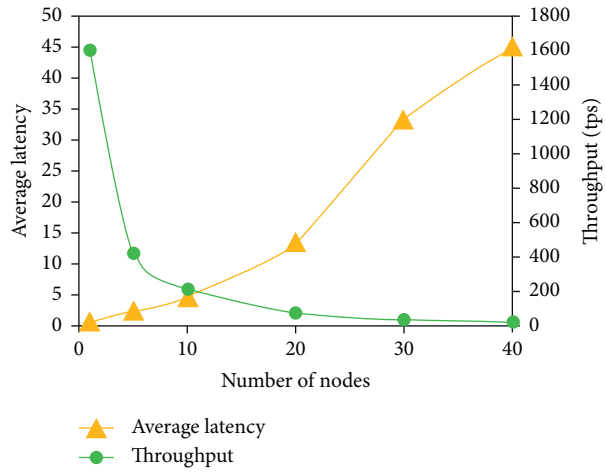

Figure 4: Performance of evidence storage with varying number of nodes.

Therefore, to avoid performance degradation, we adopt the strategy of dividing multiple ledgers to reduce the number of packaging nodes in each subnet.

We design a strategy for voting a master node daily. Having a master node means that we can avoid disordered competition, and this is the only way to maximize serial performance. Hence, the framework implements parallel block data for each master node through hierarchy and classification.

5.4. Performance of the Cache Network. We use the cache method of key-value storage to record the query time of traceability data under different cache sizes. The results are shown in Table 3. As the cache size increases gradually from 10 to 100,000 groups, the time taken to trace is not affected, and the time consumption is always short because we use the unique key identifier. The average trace time was 26.298 ms, and the standard deviation was within [3.01, 3.46] ms.

According to the abovementioned analyses and test results, we select 5000 groups of data to test the C network of this paper and count the probability distribution of the query trace time, as shown in Figure 5. The probability curve obeys the normal distribution. The mathematical expectation of the query trace-time μ is 26.298 ms, and the standard deviation δ is 3.27 ms. The values within the standard deviation range account for 86.12% of the total values. Therefore, the C network can easily create and manage the data in the cache, thus providing an efficient batch verification mechanism for real-time transactions, which is a key component of lightweight blockchains.

Table 3: Query time performance of the cache network.

Cache sizes	Average (ms)	Max (ms)	Min (ms)	Std
10	26.50	46	22	3.46
100	26.30	43	21	3.44
1,000	26.31	45	21	3.13
10,000	26.34	44	22	3.30
100,000	26.04	43	21	3.01

5.5. Inspections of Daily Regenesis. For scenarios of 100, 500, 1000, and 2000 transactions, we compare the impact of having a different number of blocks on traceability query processing. The results are shown in Table 4. The original scheme I controls the generation of 10 blocks per 100 transactions, whereas the improved scheme II generates two blocks for every 100 transactions.

By reducing the number of local blocks, scheme II makes most of the data traceable in the same block, reducing the trace latency of cross block. The daily regeneration strategy can control the number of local blocks and improve the query efficiency of the day. Especially, when new nodes join the network, the scheme improves the processing of block packing, broadcasting, and transaction and reduces the network time consumption and traffic. The experimental results are shown in Figure 6.

Table 4: Traceability performance in relation to the number of blocks.

Trans	100	500	1,000	2,000
Original I blocks	10	10×5	10×10	10×20
Original I times (ms)	557	2,717	5,338	10,687
Improved II blocks	2	2×5	2×10	2×20
Improved II times (ms)	510	2,503	4,953	9,929
Improved II promoted	8.44%	7.88%	7.21%	7.09%

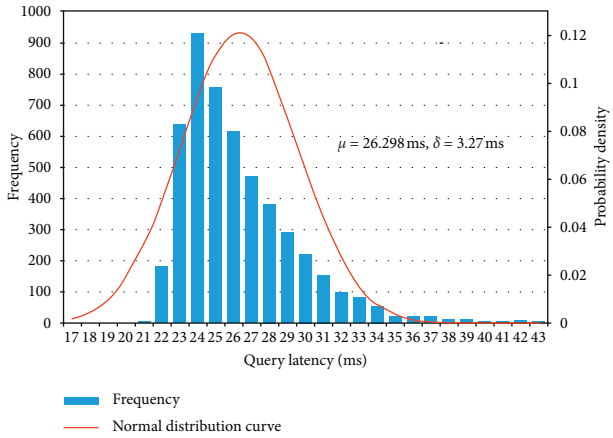

Figure 5: The probability distribution of traceability query time in the C network.

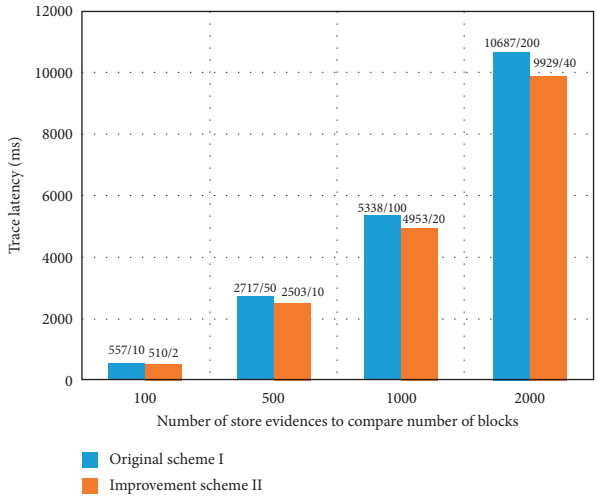

Figure 6: Traceability performance with different number of blocks.

6. Conclusions

In this paper, we propose a lightweight blockchain that can be deployed in heterogeneous, multisource, and multimodal IoTs environments. By dividing the network into different models, intelligent collaboration is promoted. Thereby, the framework can support the daily regeneration of local blocks to lightens the local storage data. Specifically, the fixed structure improves real-time processing ability. To keep a tradeoff between security and cost, the RPoW algorithm iteratively increases computational complexity on demand. Meanwhile, within the communication chips, the data directly upload to the blockchain that sorts and stores the multisource data in the classified ledgers. Finally, the experiments demonstrate that the framework reduces resource consumption, enabling blockchain service in MEC for throughout data flow protection.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

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

Acknowledgments

This work was supported by the Key Technologies of Trusted Sharing of Multi-source and Multi-modal Data Based on Blockchain Project under Grant No. 2019YFB2102403.

References

[1] M. I. Poulakis, A. G. Gotsis, and A. Alexiou, "Multicell device-to-device communication: a spectrum-sharing and densification study," *IEEE Vehicular Technology Magazine*, vol. 13, no. 1, pp. 85–96, 2018.

[2] C. E. Casetti, C. F. Chiasserini, Y. Duan, P. Giaccone, and A. Perez Manriquez, "Data connectivity and smart group formation in wi-fi direct multi-group networks," *IEEE Transactions on Network and Service Management*, vol. 15, no. 1, pp. 245–259, 2018.

[3] Bluetooth, *Bluetooth Core Specification, 5.2*, Bluetooth Special Interest Group Std, Kirland, WA, USA, 2020.

[4] S. A. Weil, S. A. Brandt, E. L. Miller, D. D. Long, and C. Maltzahn, "Ceph: a scalable, high-performance distributed file system," in *Proceedings of the 7th Conference on Operating Systems Design and Implementation (OSDI '06)*, pp. 307–320, Seattle, WA, USA, November 2006.

[5] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The google file system," *ACM SIGOPS Operating Systems Review*, vol. 37, no. 5, pp. 29–43, 2003.

[6] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The hadoop distributed file system," in *Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies*, pp. 1–10, Incline Village, NV, USA, May 2010.

[7] Y. Chen, C. Li, M. Lv, X. Shao, Y. Li, and Y. Xu, "Explicit data correlations-directed metadata prefetching method in distributed file systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 12, pp. 2692–2705, 2019.

[8] R. Van Renesse, D. Dan, V. Gough, and C. Thomas, "Efficient reconciliation and flow control for anti-entropy protocols," in *Proceedings of the 2nd Workshop on Large-Scale Distributed Systems and Middleware (LADIS '08)*, Association for Computing Machinery, New York, NY, USA, September 2008.

[9] J. Benet, "IPFS-content addressed, versioned, p2p file system," 2014, https://arxiv.org/abs/1407.3561.

[10] S. Nakamoto, *Bitcoin: A Peer-to-Peer Electronic Cash System*, HN Publishing, Guldborg, Denmark, 2008, https://bitcoin.org/bitcoin.pdf%20.

[11] P. Maymounkov and D. M. Eres, "Kademlia: a peer-to-peer information system based on the xor metric," in *Proceedings of the 2002 Revised Papers from the First International Workshop on Peer-To-Peer Systems*, Springer-Verlag, Cambridge, MA, USA, March 2002.

[12] Z.-K. Zhang, M. C. Y. Cho, C.-W. Wang, C.-W. Hsu, C.-K. Chen, and S. Shieh, "IoT security: ongoing challenges and research opportunities," in *Proceedings of the 2014 IEEE 7th International Conference on Service-Oriented Computing and Applications*, pp. 230–234, Matsue, Japan, November 2014.

[13] Y. Liu, K. Wang, Y. Lin, W. Xu, and " Lightchain, "$\mathsf {LightChain}$: a lightweight blockchain system for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3571–3581, 2019.

[14] Y. Liu, K. Wang, K. Qian, M. Du, and S. Guo, "Tornado: enabling blockchain in heterogeneous internet of things through a space-structured approach," *IEEE Internet of Things Journal*, vol. 7, no. 2, pp. 1273–1286, 2020.

[15] M. A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, L. Maglaras, and H. Janicke, "Blockchain technologies for the internet of things: research issues and challenges," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2188–2204, 2019.

[16] S. Fahad, "Blockchain without waste: proof-of-stake," *The Review of Financial Studies*, vol. hhaa075, 2020.

[17] L. Jiang, S. Xie, S. Maharjan, and Y. Zhang, "Joint transaction relaying and block verification optimization for blockchain empowered D2D communication," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 828–841, 2020.

[18] D. Gabay, K. Akkaya, and M. Cebe, "Privacy-preserving authentication scheme for connected electric vehicles using blockchain and zero knowledge proofs," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 5760–5772, 2020.

[19] D. Han, X. Du, and Y. Lu, "Trustworthiness and a zero leakage OTMP-P2L scheme based on NP problems for edge security access," *Sensors*, vol. 20, no. 8, p. 2231, 2020.

[20] D. Ongaro and J. Ousterhout, "In search of an understandable consensus algorithm," in *Proceedings of the 2014 ATC USENIX*, Philadelphia, PA, USA, June 2014.

[21] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for IoT security and privacy: the case study of a smart home," in *Proceedings of the 2017 IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 618–623, (PerCom Workshops), Kona, HI, USA, May 2017.

[22] Z. Yan, W. Yuan, M. Hassnaa, D. H. K. Tsang, L.-G. Albert, and U. Javaid, "Multi-access mobile edge computing for heterogeneous IoT," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 12-13, 2018.

[23] M. Du, K. Wang, Y. Chen, X. Wang, and Y. Sun, "Big data privacy preserving in multi-access edge computing for heterogeneous internet of things," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 62–67, 2018.

[24] C. Wang, F. R. Yu, C. Liang, Q. Chen, and L. Tang, "Joint computation offloading and interference management in wireless cellular networks with mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 7432–7445, 2017.

[25] T. Q. Dinh, Q. D. La, T. Q. Quek, and H. Shin, "Distributed learning for computation offloading in mobile edge computing," in *Proceedings of the 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, Dallas, TX, USA, July 2018.

[26] Y. Yan, Y. Dai, Z. Zhou, W. Jiang, and S. Guo, "Edge computing-based tasks offloading and block caching for mobile blockchain," *Computers, Materials & Continua*, vol. 62, no. 2, pp. 905–915, 2020.

[27] C. Gong, F. Lin, X. Gong, and Y. Lu, "Intelligent cooperative edge computing in internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9372–9382, 2020.

[28] Y. Wei, Z. Wang, D. Guo, and F. Richard Yu, "Deep q-learning based computation offloading strategy for mobile edge computing," *Computers, Materials & Continua*, vol. 59, no. 1, pp. 89–104, 2019.

[29] K. Karlsson, W. Jiang, S. Wicker et al., "Vegvisir: a partition-tolerant blockchain for the internet-of-things," in *Proceedings of the 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, pp. 1150–1158, IEEE, Vienna, Austria, July 2018.

[30] V. Hassija, V. Chamola, S. Garg, D. N. G. Krishna, G. Kaddoum, and D. N. K. Jayakody, "A blockchain-based framework for lightweight data sharing and energy trading in V2G network," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 5799–5812, 2020.

[31] M. Cebe, E. Erdin, K. Akkaya, H. Aksu, and S. Uluagac, "Block4Forensic: an integrated lightweight blockchain framework for forensics applications of connected vehicles," *IEEE Communications Magazine*, vol. 56, no. 10, pp. 50–57, 2018.

[32] W. Yang, X. Dai, J. Xiao, and H. Jin, "LDV: a lightweight DAG-based blockchain for vehicular social networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 5749–5759, 2020.

[33] X. Jiang, M. Liu, C. Yang, Y. Liu, and R. Wang, "A blockchain-based authentication protocol for WLAN mesh security access," *Computers, Materials & Continua*, vol. 58, no. 1, pp. 45–59, 2019.

[34] H. Hui, C. Zhou, S. Xu, and F. Lin, "A novel secure data transmission scheme in industrial internet of things," *China Communications*, vol. 17, no. 1, pp. 73–88, 2020.

[35] J. Qiu, Z. Tian, C. Du, Q. Zuo, S. Su, and B. Fang, "A survey on access control in the age of internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 4682–4696, 2020.

[36] H. Chen, W. Wan, J. Xia et al., "Task-attribute-based access control scheme for iot via blockchain," *Computers, Materials & Continua*, vol. 65, no. 3, pp. 2441–2453, 2020.

[37] V. Mnih, A. P. Badia, M. Mirza et al., "Asynchronous methods for deep reinforcement learning," in *Proceedings of the 33rd International Conference on Machine Learning*, New York, NY, USA, June 2016.