Preserving Edge Knowledge Sharing Among IoT Services: A Blockchain-Based Approach

Preserving Edge Knowledge Sharing among IoT Services: A Blockchain-based Approach

Gaolei Li, Mianxiong Dong, Laurence T. Yang, Kaoru Ota, Jun Wu, Jianhua Li

***Abstract*—Edge computational intelligence, integrating artificial intelligence (AI) and edge computing into the Internet of Things (IoT), will generate many scattered knowledge. To enable auditable and delay-sensitive IoT services, this knowledge will be shared among decentralized intelligent network edges (DINEs), end users, and supervisors frequently. Blockchain has a promising ability to provide a traceable, privacy-preserving and tamper-resistant ledger for sharing edge knowledge. However, due to the complicated environments of network edges, knowledge sharing among DINEs still faces many challenges. Firstly, the resource limitation and mobility of DINEs impede the applicability of existing consensus tricks (e.g., Poof of Work, Proof of Stake, and Paxos) of blockchain. Secondly, the adversaries may eavesdrop the content of edge knowledge or entice the blockchain to forks using some attacking models (like man-in-the-middle attack, denial of services, etc.). In this paper, an user-centric blockchain (UCB) framework is proposed for preserving edge knowledge sharing in IoT. Significant superiorities of UCB benefit from the proof of popularity (PoP) consensus mechanism, which is more energy-efficient and fast. Security analysis and experiments based on Raspberry Pi 3 Model B demonstrate its feasibility with low block generating delay and complexity.**

***Keywords*—Edge computational intelligence; Internet of things (IoT); blockchain; knowledge sharing; proof of popularity (PoP).**

I. Introduction

Edge computational intelligence (ECI), more generally intelligent computing at edges [1, 2, 3, 4, 5], that forms the fundamental paradigm for the great majority of solutions related to Internet of things (IoT) [6, 7]. Fog/edge computing and artificial intelligence (AI) are the most important components of ECI. To promote the deployment of ECI in IoT, the basic functions including task replacement and computation offloading has been given full considerations in [8, 9, 10]. By leveraging ECI to process large-scale sensing data, the decentralized intelligent network edges (DINEs) are of ability to contribute their localized efforts to discover and accumulate the valuable knowledge, which is defined as "edge knowledge" in this article. For brief understanding, these edge knowledge can be perceived as the output of ECI in heterogeneous IoT networks. Edge knowledge will be shared among DINEs, end users and supervisors frequently to enforce time-sensitive IoT applications as well as the auditability of decision-making at edge devices. Besides, some implicit clues can be inferred precisely by associating the shared edge knowledge.

The existing systems such as [11] and [12] have studied the raw data sharing in standard distributed computing systems, where some security assumptions were pre-defined. However, for geo-distributed IoT scenarios such as city transportation, smart grid, and healthcare, edge knowledge sharing faces many challenges [13, 14, 15]. Firstly, the subsystems in geo-distributed IoT scenarios are often maintained by multiple stakeholders. Each stakeholder occupies edge knowledge as their own property. Secondly, edge knowledge is often at risk of being attacked. Once it has tampered, the correct decisions can not be made, leading to a huge loss. Blockchain is a promising technology that allows developers to exploit the cryptography to ensure data security in a distributed ledger. By mining, it provides an unprecedented paradigm that encourages each IoT device to take responsibility for its data security. Along with the removal of AI to edges, blockchain has great potentials to preserve edge knowledge of IoT devices. Therefore, the motivation of this paper is to preserve the security of edge knowledge sharing among IoT services by using blockchain-based approach. We aim to deal with the following security problems.

Weak copyright protection. As knowledge discovering often occupies more computational resources than data sensing, resource-limited edges are more willing to protect the copyright of their knowledge. Most of the existing studies are designed for standard distributed computing, where participants are trusted, synchronous and static [16, 17, 18, 19]. Recent notable studies focus on data trading [20, 21], the trust of edge computing, and energy trading [22, 23]. These studies have identified that the participants of data and resources sharing in edge computing-based IoT are dynamic, asynchronous and unknown. Without strong copyright protecting, DINEs can not achieve reasonable benefits so that they have no enthusiasm to execute the arduous learning tasks actively.

Untrusted accounting at edges. Knowledge sharing among DINEs is originally proposed to improve the generalization ability of deep neural networks (DNNs) by sharing some information among several prediction tasks appropriately [24, 25, 26]. With this goal, a trusted branching procedure (BP) is defined to select the participants and initialize the system configuration in a centralized way. For example, C. Liu, et

G. Li is with the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University and Shanghai Key Laboratory of Integrated Administration Technologies for Information Security, Shanghai 200240, China. And also he is with the Muroran Institution of Technology, Muroran 050-8585, Japan.

L. T. Yang is with Department of Computer Science, St. Francis Xavier University, Antigonish, NS, Canada.

M. Dong and K. Ota are with Muroran Institute of Technology, Muroran 050-8585, Japan.

J. Wu and J. Li are with the School of Cyber Security, Shanghai Jiao Tong University and Shanghai Key Laboratory of Integrated Administration Technologies for Information Security, Shanghai 200240, China.

Corresponding author: Mianxiong Dong (mx.dong@csse.muroran-it.ac.jp)

TABLE I:
TYPICAL CONSENSUS MECHANISMS FOR DATA, FEATURE, ENERGY AND KNOWLEDGE SHARING

Work	Participant	Scope	Object	Trusted center	Limitations
Grouping [24]	Known, static	Localized, synchronous	Features	$\surd$	Trustless, failure
MTFS [26]	Unknown, dynamic	Localized, synchronous	Features	$\surd$	Scalability, delay
Cascading [25]	Known, static	Distributed, linear	Features	$\times$	Trustless, delay
Blockchain [27, 28]	Known, static	Multi-party, cloud	Raw data	$\times$	Overhead, redundancy
PriWatt [22]	Anonymous, token	Decentralized	Energy	$\times$	Virtual coin
Consensus [33]	Unknown, dynamic	Byzantine, self-organizing	Knowledge	$\times$	Location-independent
PoP (this paper)	Edge, vulnerable	Geo-distributed, asynchronous	Knowledge	$\times$	————

al. proposed a multi-task feature selection (MTFS) scheme based on graph-clustered feature sharing in [26]. The MTFS can select the most valuable features from multiple computing tasks automatically in a dynamic scenario, where the feature correlations between different tasks are not unknown. Notably, knowledge is a high-level abstract and orchestration of features, which is produced after the baptism of large-scale end users. Under edge environments, there is no trusted BP to monitor and control the knowledge sharing. Moreover, considering the cyber-physical mobility and vulnerability, DINEs can not provide trusted accounting for edge knowledge sharing in IoT.

Inefficient consensus. Blockchain's superiority closely depends on its consensus mechanisms. Any IoT data that can be formulated as transactions can be shared securely using blockchain [22, 29, 30]. However, the cost of data sharing increases sharply along with the growth of data scale in IoT [31]. Reference [32] advocated to enforce blockchain-based data sharing in AI-powered networks, and further Study [33] proposes a proof of creditability (PoC) to make big data at edges open. TABLE I shows the comparison of latest consensus mechanisms. Traditional consensus mechanisms relying on static wealth are inefficient for DINEs in IoT.

To address these problems, we propose an user-centric blockchain (UCB) framework, which enables secure and efficient edge knowledge sharing among IoT services. In UCB, DINEs with different data processing models are responsible to provide intelligent edge services for end users and cooperatively discover knowledge at edges. The DINE in UCB is endowed with four different functions: 1) publish the discovered edge knowledge to its known DINEs; 2) collect the scattered edge knowledge from the whole network and write them on a pending localized ledger as transactions; 3) calculate the correlation matrix between the collected edge knowledge and transfer them into proposals; 4) elect a DINE that discovers the most valuable knowledge to generate block.

Strength of proposed UCB framework is that it provides an efficient, scalable and decentralized ledger for edge knowledge. Different from traditional mining, the goal of UCB's mining is to find the most popular knowledge rather than a random number. UCB's mining adopts proof of popularity (PoP) algorithm, which is proposed to replace traditional consensus mechanisms. The PoP treats the usage of edge knowledge as DINE's wealthy and it is of ability to make consistent decisions to fulfill the requirements of edge knowledge sharing even if the minority DINEs in UCB are of absence. Main contributions of this paper are summarized as follows:

- A flexible UCB framework is proposed for IoT, which enables peer-to-peer secure edge knowledge sharing between untrusted DINEs. In the UCB framework, both copyright and privacy of edge knowledge are protected.
- A proof of popularity (PoP) consensus algorithm is proposed, which can enhance the fault-tolerance and security of blockchain in edge environment. Different from existing consensus mechanisms, the PoP treats the DINEs' knowledge usage as wealth to compete the privilege of knowledge block generating.
- The superiority of our schemes is validated by security analysis and experimental evaluation. The experiment is implemented on Raspberry Pi 3 Model B.

The organization of our work is introduced as follows. In Section II, we review the architectural evolution to ECI, and then introduce the related decentralized consensus algorithms. Section III provides a scenario statement including network model, adversary model, and security requirements of edge knowledge sharing in IoT. To help readers to understand the application scope of our work, the Part A of Section IV introduces the key components of UCB framework and their functions. And then, the workflow of the PoP consensus mechanism is described in Part B of Section IV. Security-aware decentralized knowledge ledger between DINEs is explained in Part C of Section IV. Security analysis and experimental evaluation based on real datasets and Raspberry Pi 3 are demonstrated in Section V. Finally, we conclude our work and envision the future work in Section VI.

II. RELATED WORK

Edge entities as important infrastructures that close to end users have great potential to produce useful knowledge precisely. As the rapid development of ECI in various industries, knowledge sharing between DINEs should be paid more attention. Although this paper is a further study based on many previous approaches, it has an essential difference from existing approaches. In this section, we will review the recent development of ECI and blockchain in IoT to highlight the value of our work.

A. Edge Computational Intelligence

Recently, edge computational intelligence (also named as edge learning or edge AI) has attracted increasing attention. The ECI consists of edge computing and machine learning. Literature [35] envisioned that the ECI may enable computation-oriented multiple access for ultra-fast data aggregation, importance-aware resource allocation for agile intelligence acquisition, and learning-driven signal encoding for high-speed data-feature transmission. Machine learning at edges has been utilized to predict the number of end users, accelerate network speed and provide personalized networking services in 5G cellular networks [36, 37]. Especially, H. Li, et al study that deep learning can be applied in the edge environment by reasonably distributing the neural network layers to different edges [38] and T. Wang, et al propose a three-layer privacy-preserving cloud storage scheme based on computational intelligence in fog computing [39].

This paper is the first to study how to share knowledge among DINEs securely. Before this, features sharing is always a hot topic in the field of AI that often needs to process multiple tasks simultaneously. A. Torralba, et al. [27] contributed a notable achievement, in which the common features were selected and shared among procedures in an overlapping way. The most popular framework consists of two different kinds of learning models: 1) Weak learning model C_n, and 2) strong learning model C^*. By aggregating the knowledge of multiple weak learning models, C^* provides more robust performance:

$$C^*(D, Y_m) = \sum_{i=1}^{n} C_i(D_i, Y_m) \tag{1}$$

where D is the input dataset and $\{D_1, D_2, ..., D_{n-1}, D_n\}$ is a group of sub-datasets of D. $C^*(D, Y_m) = log\frac{P(y='Y'_m,'Y'_m \in Y)}{1-P(y='Y'_m,'Y'_m \in Y)}$ is the log value of being in category $'Y'_m$. Y denotes a set of data labels $\{Y_1, Y_2, ..., Y_m, ..., Y_M\}$.

Subsequently, features sharing has always been improved and extended to support object classification and attribute prediction. Y. Lu, et al. [24] proposed a thin network model for improving the adaptivity of feature sharing. Unfortunately, these approaches still have no deployment in real industries for the real industrial environments are not as static as laboratory settings. In this paper, we will leverage this framework and extend it to edge environment, which will improve its applicability on real IoT systems.

B. Blockchain and Decentralized Consensus

For the future smart city, major things will be interconnected by IoT. Each IoT node will be equipped with several computational intelligence technologies. Since the emergence of blockchain has fulfilled many security, maintenance, and authentication requirements of IoT systems, blockchain-based IoT systems are seen from the past few years [41, 42].

In order to apply blockchain in future IoT systems, literature [43] studies how to offload the computation-intensive mining tasks to the neighboring DINEs. Due to the mismatch between resource limitations of IoT devices and the high cost of mining tasks, any wealth depending on static resources can not guarantee the trust of participants. For edge environment, a novel resource-efficient decentralized consensus mechanism named as proof of collaboration (PoC) is proposed in [34]. In PoC, the data flows are treated as transactions and the collaboration credit is abstracted as virtual coins for trading data between edge entities. However, as the growth of data scale in IoT, the length of blockchain increases sharply [44]. In our work, a proof of popularity (PoP) scheme is proposed, which will reduce the cost of mining tasks on edge devices.

Recently, the decentralized consensus has gained increasing attention. With the swift development of blockchain in various scenarios, research on decentralized consensus forms two key branches. The first is the proof-based scheme. The most famous schemes of this branch include Proof of Work (PoW) and Distributed Proof of Stake (DPoS). Another branch is the voting-based scheme, typical schemes include Ripple, Practical Byzantine Fault Tolerance (PBFT) and Paxos. W. Wang, et al. [45] compare the advantages and disadvantages of these decentralized consensus mechanisms, which comprehensively reviewed the studies on the recent development of blockchain's consensus mechanisms.

Different from existing studies, we propose a UCB framework, which presents joint optimization for blockchain in IoT by enabling secure and efficient edge knowledge sharing. In UCB, each DINE is of ability to act as a respective well-equipped node that can run the learning models, mining tasks and block generating. To enhance the security of edge knowledge sharing in IoT, we put forward proof of popularity (PoP) consensus algorithm. The PoP algorithm is independent of the computation-consuming miners (e.g., puzzle resolving). The significant innovation of PoP is that the knowledge blocks are generated according to the elections among users rather than mining.

III. System Model

In this section, the system model of blockchain-based edge knowledge sharing in IoT is stated comprehensively. The network model is demonstrated as shown in Fig. 1. The adversary model is described in part B of this section. Besides, we highlight the security requirements of edge knowledge sharing.

A. Network Model

The network model is considered with multiple DINEs, multiple miners, and many end users. Peer-to-peer communication is selected to acts as the basic network protocol of the proposed UCB framework. Throughout this paper, the symbol κ, τ and $\hbar$ are used to denote the space of DINEs, miners, and end users, orderly. The DINEs, miners, and end users have their universally unique identifiers (UUIDs). We denote the DINE i with the symbol κ_i, miner j with τ_j and end user k with $\hbar_k$. Multiple DINEs can work on one Raspberry Pi 3 Model B (Rasp3+) if the size of Rasp3+'s secure digital (SD) is large. Edge knowledge of each fog domain is maintained by a miner in the blockchain network. Each miner has two different procedures 1) knowledge monitoring procedure and 2) PoP consensus procedure.

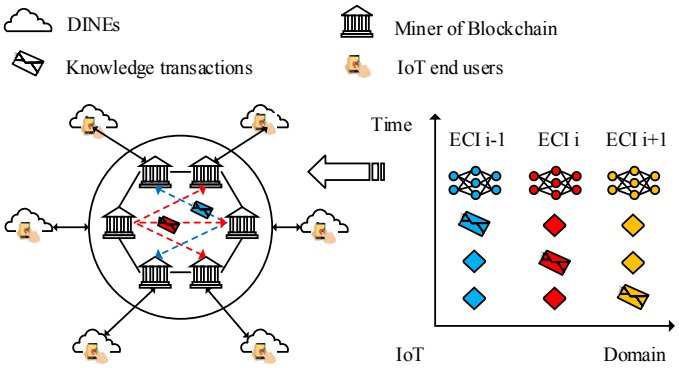


Fig. 1: Network model of proposed UCB for edge knowledge sharing in IoT.

The edge knowledge is discovered by local DINEs, and then published to the knowledge monitoring procedure of neighboring DINEs. Each miner can analyze the received knowledge deeply and map them into proposals for end users. We consider the raw sensing data can be sent to multiple DINEs and each edge knowledge can be subscribed by multiple DINEs. Besides, in our framework, each end user is assumed as a "rational man", which will accept the most suitable proposal according to his demands. In addition, we assume that there is a known access control matrix among the DINEs and the social distance between DINEs is less than six hops. Such an assumption has been formulated as a small world theory.

B. Adversary Model

The main goal of an attacker may tamper the contents of edge knowledge that they are not permitted to. Both unauthorized DINEs and external intruders can be treated as adversaries. Due to the limited resources, DINEs in IoT are vulnerable to strong attacks. There are four different attacking methods that may be adopted by adversaries: 1) inject false knowledge into the network edges, 2) disturb the formal working of DINEs by launching dense of services (DoS), 3) tamper the content of knowledge in the blockchain network. In addition, 4) DINEs with a higher security level may leak the confidential edge knowledge to the ordinary DINEs.

C. Requirements of Edge Knowledge Security

With respect to edge knowledge sharing in IoT, we recognize the following unique security requirements of edge knowledge sharing. In particular, the security risks of knowledge transactions in IoT blockchain.

1) Edge Knowledge Desensitization: In many IoT scenarios, especially for the face, fingerprint, voice recognition-based cases, the knowledge discovered by DINEs will be sensitive data that should be well controlled. Before sharing the knowledge with the other DINEs, edge knowledge desensitization should be completed to avoid the disclosure of both knowledge content and the parameters of DINEs. For this purpose, we first enforce an edge knowledge desensitization policy, by transferring edge knowledge into various topics. Notably, the edge knowledge desensitization policy is of ability to distinguish a unique group of subscribers that the DINE can trust because the known access control matrix among the DINEs is used to orchestrate the topic and blocker.

2) Pseudo-DINE Resistance: As described in adversary model, malicious attacks may register a pseudo-DINE to publish false edge knowledge and participate the blockchain network to compete for the privilege of knowledge block generating or eavesdrop the edge knowledge traffic from DINEs with higher security levels. This requires our user-centric blockchain (UCB) framework to be resilient to the pseudo DINEs attacking in the sense that the UCB will not give them additional advantages.

3) 51% Attacking Defence: For traditional blockchain in IoT, if a participant's wealth is equal to the sum of other participants' wealth, this participant will be easy to preempt the block and modify his own transactions at any time. For Bitcoin, this problem is formulated as 51% attacking. Actually, the traditional consensus mechanism is not suitable to edges of IoT because the wealth (including the computing, caching, networking resources) of DINEs is usually further less than the wealth of cloud. Moreover, the state of DINEs is unpredictable so that any existing wealth of DINEs in IoT can not be used as proof during the consensus process of the traditional blockchain. Edge knowledge sharing with UCB will require a novel consensus mechanism that can defend 51% attacking.

4) Dense of Services Mitigation: Dense of services (DoS) attacking has a high success rate by utilizing the distributed and large-scale nature of IoT devices. DoS attacking may incur two main kinds of disasters for edge knowledge sharing: 1) ledger overflow, 2) consensus interrupting. Each kind of disasters can result in unpredictable results. This requires the UCB and PoP consensus algorithm is robust when several DINEs are suffering from the DoS attacks.

IV. Workflow of User-centric Blockchain

The key components of UCB framework and their functions that involved with the ECI are described in detail in Part A. Part B introduces the proof of popularity (PoP) consensus algorithm. Fig. 2 show the workflow of UCB framework.

A. Components and Functions

Here, we would like to bring out the workflow of UCB framework briefly. Comprehensively, entities of DINEs can be specified as routers, base stations, gateways, personal devices (e.g., smartphones, cars), even public infrastructures (e.g., street lights, buses, and big boards) in a city. DINEs can be treated as transitional nodes between Internet service providers (ISP) and local users. Due to the limitation of the resources, it is common to see that an additional controller is deployed in DINE to monitor and control the states of connected devices. In UCB, we implement a stronger DINE based on this controller to boost the accuracy of edge knowledge associating. In UCB, each DINE can receive other DINE's knowledge. Therefore, each DINEs has the ability of C^*. For a clear explanation of

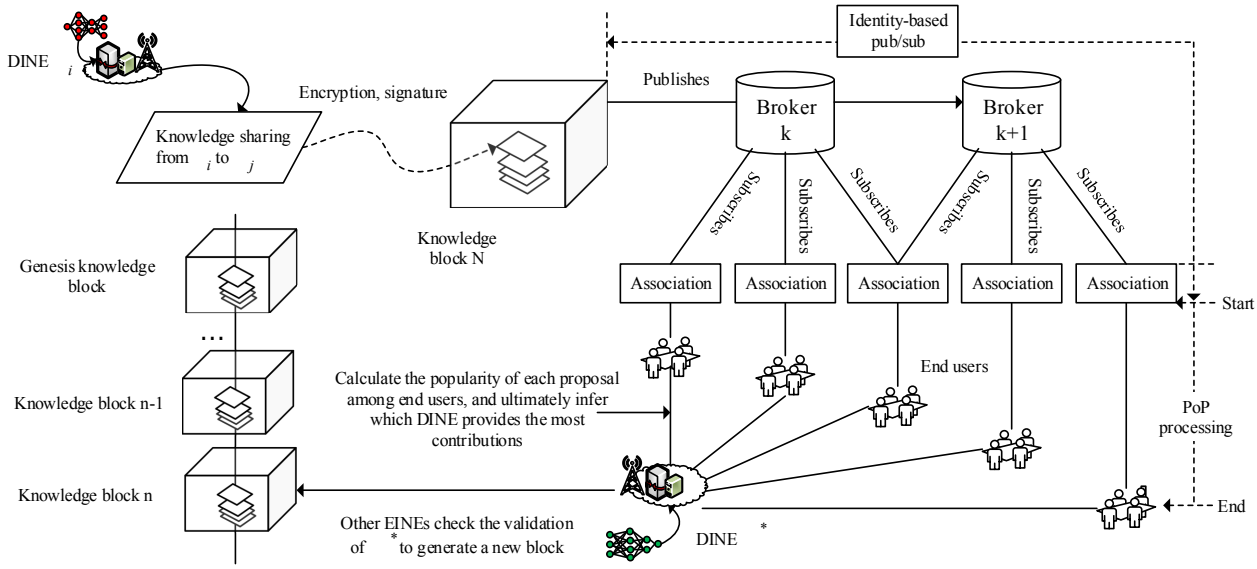


Fig. 2: The workflow of UCB framework. Therein, different from the Proof of Work (PoW) solution, the PoP consensus algorithm exploits the edge knowledge association and end users' confirms to achieve the consistency of DINEs.

our proposal, the proposed UCB framework is divided into three layers: 1) Edge knowledge discovering layer; 2) Self-incentive layer; 3) Association layer.

1) Edge knowledge discovering: With the help of ECI, edge knowledge discovering layer will process the aggregated IoT data and discover the valuable knowledge at IoT edges. These ECIs mainly include many pre-trained learning models (such as the least absolute shrinkage and selection operator (LASSO), k-nearest neighbor (KNN), naive bias (NB), support vector regression (SVR), convolutional neural networks (CNN)). Characteristics of edge knowledge discovering layer focus on unbalanced resources and heterogeneous networking of DINEs. DINEs try their best efforts to recognize and accumulate knowledge at IoT edges to serve for end users. In order to achieve better performance and reduce the costs, DINEs are encouraged to share their own knowledge. The common operations in edge knowledge discovering layer are listed as follows:

- Pre-processing operation: execute the pre-processing functions (e.g., data cleaning, resource offloading, instance selection, data normalization, and transformation). With these pre-processing operations, the aggregated raw data from various local sensors can be formulated as the standard data of observed objects.
- Monitoring operations: Monitor the resources (such as computing, caching, and networking) and resource states of DINEs in the local network and schedule these resources for efficient and accurate edge knowledge discovering.
- Recognizing operation: Recognizing operation impute the output of pre-processing operations into the pre-trained learning models for discovering knowledge at edges.
- Provisioning operation: DINEs act as service providers at edges to provision low-latency services for local end users. It also can provision edge knowledge subscribing services for the neighboring DINEs. Besides, DINEs provide an auditability interface for the supervisors.

2) Edge knowledge sharing: The edge knowledge sharing layer is self-incentive. It is designed to encourage each DINE to share their knowledge. By introducing the role of end users $\hbar$ into the blockchain consensus process, edge knowledge sharing becomes a self-incentive system, where DINEs are willing to share their knowledge to attract more end users. It can be imagined that a DINE who has more valuable knowledge will attract more end users in IoT. DINE's profits will rise up because the average service costs will be reduced with the rapid growth of end users. Self-incentive makes the UCB be independent of virtual coins and proof of work. In such a self-incentive system, the value of knowledge can be modeled with its popularity. Self-incentive makes the UCB be independent of virtual coins and proof of work. In our proposal, the DINEs will consume the value of their edge knowledge rather than computation. The value of one DINE's edge knowledge is denoted as $\jmath$ and defined as the following equation:

$$\jmath = \frac{\sum_{k=1}^{K}\sum_{w=1}^{W} F_{(D_k,Y_m)}}{K \times W \times N} \tag{2}$$

where N represent the number of DINEs in the network model and W is the maximum number of knowledge discovering functions on one DINE. K is the knowledge number threshold that one knowledge discovering function can discover during the timeslot of generating two neighboring blocks. $F_{(D_k,Y_m)}$ is the usage number of one knowledge when C^*. According to the definition of $\jmath$, the value of $\jmath$ will be raised up automatically

if there are more DINEs to join the blockchain networks in the proposed UCB framework. A network with $\jmath < 1/N$ will be not valid because there is at least one end users will accept the knowledge.

Initially, the UCB will reset $\jmath$ into the initial state. Thus, each DINE begins to collect the edge knowledge from the whole network and calculate the value of $\jmath$. Once a DINE's $\jmath$ exceeds the pre-defined threshold $\jmath_0$, this DINE gains the authority to stamp a block and broadcasts them to others through the non-tamper ledger.

3) Edge knowledge associating: Association layer presents an automatic knowledge value adding model for different DINEs. In this layer, each DINE can selectively download the readable knowledge that the local end users can access, and then associate them to produce better proposals for the local end users. To evaluate the value of each knowledge, the UCB allows DINEs to monitor if the proposal is accepted by local end users. Once a proposal is accepted by local end users, the value of related edge knowledge will be added. Ultimately, the value of edge knowledge is equal to its usage number among local end users. Notably, knowledge associating is also under the known access control matrix.

B. Proof of popularity (PoP) Consensus

The decentralized consensus algorithm is the basis of all different blockchains, which form the most important part of the blockchain platform. During the process of decentralized consensus, every participant broadcasts its transactions to the others and simultaneously receives the others' transactions. Therefore, every transaction can be achieved and validated by all participants in the whole blockchain network.

For proof of work (PoW) consensus, participants need to act as miners to find a random number to solve the pre-defined puzzle. The first miner that finds this random number is of ability to write the transactions into the blocks. For proof of stake (PoS) consensus, the generator of the next block is elected by validating their wealth and qualifications. Due to the limited resources and dynamics, both PoW and PoS are not adaptive to mobile edge computing. For voting-based consensus (e.g., Paxos and Raft), a candidate that receives votes from a majority of the full clusters will become the new illegal leader to maintain the transactions. Both voting-based consensus and proof-based consensus emphasize the consistency of participants in the whole blockchain network.

For edge knowledge sharing between multiple DINEs in IoT, not all DINEs will participate in the public blockchain networks. Some of DINEs may be self-organized with multiple private blockchain networks in a geo-distributed way. Even, there may be only one DINE in a small private blockchain network. In this case, the value of knowledge in such a small private blockchain network can not be maintained by DINEs and should be supervised by the local end users. In our work, we propose a proof of popularity (PoP) consensus mechanism, which elects block generator by validating their knowledge wealth rather than resources or votes.

We formulate two different kinds of ranking, 1) knowledge ranking, and 2) proposal ranking. Actually, the proposal ranking can be easily calculated by static adding up the end user feedback. The knowledge ranking is inferred by analyzing knowledge ranking and local usage records of involved knowledge. Therefore, the popularity of both knowledge and proposals should be computed.

TABLE II

Algorithm 1: User-centric proposal value ranking and electing.

Input: $P_i^*(j), \{\kappa_i\}, Counter_i^j = 0, i \in [1, N], j \in [1, s]$
Output: $\phi_i^{P^*(j)}(R_j)$
1: Collecting proposals in the whole network $P_i^*(j)$
2: Monitoring DINEs' confirm signals for proposal acceptance
3: **for** $i = 1 : N$
4: **if** $\exists\kappa$ that accepts the proposal in $P_i^j, \kappa \in \kappa_i$
5: $Counter_i^j + +$;
6: **else** $\{P_i^*(j)$ is added to the pending list $PList$
7: $Counter_i^j = 0\}$
8: **end**
9: **end**
10: **if** Length of the pending list $Len(PList) < s$
11: **for** i=1:N
12: **for** j=1:$Len(PList)$
13: 1) Calculate the the popularity ranking $R_j, j \in [1, s]$
14: 2) Estimate $\phi_i^{P^*(j)}(R_j)$ according to equation (5), (6)
15: **end**
16: **end**
17: Broadcast $\phi_i^{P^*(j)}(R_j = 1)$ in the whole network
18: Electing the most popular proposal $\phi_i^{P^*(j)}(R_j = 1)$
19: **end**

Stage 1: Popular Proposal Identification. In the traditional blockchain network, end user's payment is usually executed after the transactions are generated and validated. The work of miners is to find a participant to store large-scale transactions in a distributed way. Therefore, the performance of traditional consensus algorithms is independent of the end user's payment services so that the second-scale transaction latency can be accepted. In the edge environment, the DINEs treat the edge knowledge as transactions. The essential difference is that end users do not need to pay anything for these transactions, but they need to identify if the received edge knowledge can be associated as proposals and if the proposals will be accepted. Thus, after the end users receive the proposals produced by DINEs, they can select the satisfactory proposal and return the corresponding confirm signals to DINEs. The popularity of one proposal among end users reflects the value of this proposal.

Each DINE κ_i in proposed UCB framework is of ability to maintain a strong leaning model C_n^* to optimize the edge services for end users. The output of a strong learning model C_n^* can be mapped into P^* following the equation (3) and equation (4). By using a mobile edge computing platform, these proposals can be accessed by all of the authorized local end users. If there is one local end user accepts a proposal, the value of this proposal is updated by adding "1". To calculate the proposal value ranking, the IoT service popularity

TABLE III:
OPEN KNOWLEDGE TO EACH DINE

Popularity	Proposal	Involved knowledge	Ranking
0.96	P_3^2	$\mu_2, \mu_{12}, \mu_{22}$	
0.92	P_1^{14}	$\mu_4, \mu_8, \mu_{10}, \mu_{14}$	
0.86	P_{10}^3	$\mu_8, \mu_{12}, \mu_{14}$	$\mu_8 : 8 > \mu_{12} : 5$
0.83	P_2^{71}	$\mu_3, \mu_8, \mu_{12}, \mu_{31}$	
...	...		
0.68	P_4^{12}	$\mu_1, \mu_4, \mu_{16}, \mu_{26}$	$> \mu_4 : 3 > \mu_{14} : 2$
0.42	P_3^{21}	μ_6, μ_9, μ_{22}	
0.25	P_3^{49}	μ_7, μ_8	
0.14	P_5^{80}	$\mu_8, \mu_{12}, \mu_{36}$	
...	...		
0.08	P_3^{52}	μ_4, μ_8, μ_{19}	

TABLE IV:
HIDDEN KNOWLEDGE TO PARTIAL DINES

Popularity	Proposal	Involved knowledge	knowledge ranking
0.94	$[P_3^2]$	$\mu_2, [\mu_{12}], \mu_{22}$	
0.92	P_1^{14}	$\mu_4, \mu_8, \mu_{10}, \mu_{14}$	
0.88	P_{10}^3	$\mu_8, \mu_{12}, \mu_{14}$	$\mu_8 : [6] > \mu_{12} : [4]$
0.84	P_2^{71}	$\mu_3, \mu_8, \mu_{12}, \mu_{31}$	
...	...		
0.45	P_4^{12}	$\mu_1, \mu_4, \mu_{16}, \mu_{26}$	$> \mu_4 : 3 > \mu_{14} : 2$
0.42	$[P_3^{21}]$	μ_6, μ_9, μ_{22}	
0.35	$[P_3^{49}]$	$\mu_7, [\mu_8]$	...
0.24	P_5^{80}	$[\mu_8], \mu_{12}, \mu_{36}$	
...	...		
0.11	$[P_3^{52}]$	$\mu_4, [\mu_8], \mu_{19}$	

is formulated with Zipf's law.

$$Z_i^{P^*(j)}(R_j) = \frac{(\sum_{r=1}^{s-1} \frac{1}{r^\beta})^{-1}}{R_j} \tag{3}$$

where R_j is the popularity ranking of $j-th$ proposal and $Z_i^{P^*(j)}(R_j)$ presents the proposal j's usage number in the local network. In different local networks, the number of end users may have a great impact on $Z_i^{P^*(j)}(R_j)$. Therefore, we standardize the $Z_i^{P^*(j)}(R_j)$ with Z-score normalization as follows:

$$\phi_i^{P^*(j)}(R_j) = \frac{Z_i^{P^*(j)}(R_j) - \overline{Z_i^{P^*(j)}(R_j)}}{\sigma} \tag{4}$$

where $\overline{Z_i^{P^*(j)}(R_j)}$ is the mathematical mean of $Z_i^{P^*(j)}(R_j)$ and σ is the mathematical variance of $Z_i^{P^*(j)}(R_j)$.

At this step, each DINE κ_i needs to calculate the proposal popularity ranking among local end users and anonymously publish the ranking to the other edge entity for popular proposal identification. Ultimately, the most popular proposal will be identified from $\{\phi_1^{P^*(j)}(R_j), \phi_2^{P^*(j)}(R_j), ..., \phi_N^{P^*(j)}(R_j)\}$.

Popular proposal identification treats users' preference as one kind of wealth of DINEs. This approach establishes a bridge for jointly optimizing the user experience and consensus performance. Furthermore, this approach can reduce the transaction redundancy for the related knowledge of unaccepted proposals will be deleted before the block is stamped.

The user-centric proposal value ranking and electing algorithm is shown as illustrated in Table II. Initially, DINEs κ_i self-checks their proposals $P_i^*(j)$ and set a two-dimensional counter array to record the usage frequency of each proposal. When a piece of new knowledge is discovered, the DINE will associate this new knowledge with all the knowledge it has collected to generate new proposals. Before publishing these new proposals, it will traverse all the neighboring DINEs that it can access to monitor if the previous proposals have been accepted by end users. Ultimately, the DINE will calculate the popularity ranking of previous proposals according to the two-dimensional counter array.

Particularly, the difficulty of block generating can be well controlled by specifying which proposal is used to decentralized knowledge ledger. If we choose the most popular proposal and use it to find valuable knowledge, the average time interval for block generating will be very short. Otherwise, the average time interval for block generating will increase.

Stage 2: Security-aware block generating. Section V demonstrates that the PoP is beneficial to 51% attacking defense and Pseudo DINE resistance, which also can protect blockchain security. However, before writing into the blockchain, the edge knowledge still has a possibility to be exposed. Although the DINEs can be anonymous in the blockchain network, it exists a big security risk for the edge knowledge is open to each DINE. For the propose of secure edge knowledge sharing between DINEs in IoT, the UCB enables the security-aware block generating.

Different from stage 1, where the DINE needs to exchange information with end users, stage 2 requires each DINE to communicate with the other DINEs. As formulated by the equation (3) and equation (4), each consensus procedure of the DINE is responsible to map the aggregated edge knowledge into proposals. Thus, once a proposal is accepted by end users, we can backtrack the related edge knowledge of this proposal.

Each edge knowledge will be configured with several authorities for different DINEs' access and each DINE will be defined with different security levels. The role-based access control (RBAC) can be used to match DINE's security level with knowledge's authorities to decide if the DINE can operate the received knowledge. In this case, the blockchain network of UCB seems to be divided into a lot of small blockchain networks (SBNs). The relationship between these SBNs is modeled with the small-world theory, which is a famous law in human society. To abstract the real scenarios, we can give two reasonable hypotheses:

- The geo-distribution of knowledge is observable but the capacity of each DINE to discover one knowledge is uneven.
- All participants (e.g., end users and DINEs) are rational

economic men, that means each participant desires to maximize his profits or minimize his costs.

In our UCB framework, the producer of each knowledge is hidden for security and privacy. A DINE only can associate partial knowledge that it can read and map them into proposals. At time t, all readable knowledge for a DINE is denoted as available knowledge space $\mu = \{\mu_1, \mu_2, ..., \mu_s\}$, where s is the length of available knowledge space μ. knowledge association in our framework is defined as the following equations:

$$Len(P^*) = 2^s \tag{5}$$

$$P^* = H(\mu) \tag{6}$$

where, $H(\mu)$ is the function for mapping from available knowledge space into proposal space. P^* is the proposal space, $\{[\Pi(u,0)], [\Pi(u,1)], ..., [\Pi(u,s-1)], [\Pi(u,s)]\}$. $\Pi(u,x)$ is the association between x available knowledge in the available knowledge space μ. $Len(*)$ is the length getting function.

Table III and Table IV give a glance to the process of knowledge popularity prediction in different schemes. When the knowledge is open to each DINE, all knowledge received from the network can be beneficial to optimize the proposals. We can trace the association process and calculate the usage ranking of knowledge. For example, according to the listed data in Table II, μ_8 gains the top usage frequency.

In a real DINE scenario, access control and encryption is a must to defend knowledge privacy leaking. Therefore, most of the DINEs has no ability to read all the knowledge data. For instance, μ_8 and μ_{12} are produced by κ_3 and it applies a copyright protection service for its knowledge. Thus, it will only share with the DINEs who have purchased the copyright. Finally, P_3^2, P_3^{49} and P_3^{52} can not be generated.

However, in UCB, P_3^2, P_3^{49} and P_3^{52} are also considered. This advantage benefits from the decentralization and society relationship of HEAIs. For this purpose, we propose a security-aware block generating algorithm. As mentioned in the previous, the small world theory is used to model the relationship between DINEs in the networks. In order to bring out this idea, we first define an access control matrix M^a for all DINE κ in the network and a neighboring matrix M^s. The value of each tuple M_{ij}^a in M^a can be set 'read+' or 'read-'. 'read+' means DINE κ_i can read the knowledge sent from DINE κ_j and 'read-' means DINE κ_j can read the knowledge sent from DINE κ_i. The value of each tuple M_{ij}^s in the social matrix M^s represents the communication distance between DINE κ_i and DINE κ_j. Following the small world theory, the value of M_{ij}^s is less than 6.

Consider the DINE in IoT is geo-distributed, each DINE will maintain a local M_{ij}^s and M_{ij}^a. When κ_i participates the blockchain network to compete for the block, it will check the local knowledge and identify the readable knowledge based on the M_{ij}^s. Else if the value of $M_{ij}^s < 6$ and the knowledge is unreadable, the DINE κ should find a DINE $\kappa_{i'}$ that can make $M_{i'i}^a ==' read+'$ and $M_{i'j}^a ==' read+'$. And then add the proposal $P^*(j)$ into $LIST_i$. Else if the value of $M_{ij}^s > 6$, it will traverse the neighboring DINEs κ_i^N of κ_i to find a DINE close to DINE κ_j. In this case, the algorithm must go back to Step 4.

TABLE V

Algorithm 2: Security-aware block generating.

Input: $M_{ij}^a, M_{ij}^s, \phi_i^{P^*(j)}, LIST_i = [\,], Indutor = set()$
Output: κ_b, New_block
1: Initially, load the parameters of inputs and start all DINEs
2: Randomly select a DINE to generate the genius block and then broadcast it to all the other DINEs
3: **for** all DINEs **in** κ:
4: **Thread 1:**
5: **if** $M_{ij}^a ==' read+'$:
6: $LIST_i.add(P^*(j))$
7: **elif** $M_{ij}^{s\,*} \leq 6$:
8: Traverse κ to find a DINE $\kappa_{i'}$ that can make $M_{i'i}^a ==' read+'$ && $M_{i'j}^a ==' read+'$
9: $LIST_{i'}.add(P^*(j))$
10: **else**:
11: Traverse the neighboring DINEs κ_i^N of κ_i to find a DINE close to DINE κ_j
12: **Go to:** Step 4
13: **end**
14: **Thread 2:**
15: **if** Inductor==1
16: Calculate $\phi_i^{P^*(j)}$ by executing **Algorithm 1**
17: **end**
18: **Thread 3:**
19: Monitor $\phi_{-i}^{P^*(j)}$ issued by other DINEs
20: Calculate the most popular proposal $\phi_i^{P^*(j)}$ at local
21: **if** $\phi_i^{P^*(j)} > \phi_{-i}^{P^*(j)}$
22: Generating New_block
23: **end**
24: **end**

The proposed algorithm is shown in **Thread 1** of Table V. The other two threads are also included in this algorithm. Firstly, $Indutor$ is defined as a flag, when the $Inductor$ is equal to 1, the **Thread 2** will execute the algorithm 1 to calculate $\phi_i^{P^*(j)}$. Secondly, as the goal of algorithm 2 is to elect a valid node to generate a new block in a decentralized network paradigm, the **Thread 3** of each DINE needs to monitor the other DINEs' popular proposals and adjust if it has found the most valuable knowledge.

V. Analysis and Evaluation

A. Experimental Settings

Under cloud paradigm, researchers can easily develop some experiments by following several steps: 1) buy cloud services from providers, 2) upload image data produced by multiples devices of users, 3) select reasonable learning models to deal with data analysis, 4) orchestrate security policy according to the existing security library. However, for edge computing paradigm, data processing models are also shifted from the cloud into edge devices together with the computation resources. This brings three big challenges for developing experiments under edge computing paradigms. Firstly, a single DINE can not access all the image data as free and stable

as a cloud due to the untrusted communication conditions between edge devices. Secondly, the profits of attacking edge knowledge are more than DINE's raw data so that the designed experiments will need to defect more premeditated attacks. Thirdly, different from the traditional consensus approaches, the privilege of DINEs to generate blocks reply on the end users' voting but not other DINEs' voting. Privacy-preserving of users also should be considered during the phase of experiment design.

Leveraging the popular architecture of computer vision-based smart applications, the main components of our experiments also contain cameras and base station, but it is not imperative to purchase resources and services from cloud providers. To adapt to the edge computing environment, the base station is established with Raspberry Pi 3 Model B (noted as Rasp3+). The configuration parameters of Rasp3+ are 1) 1.4GHz 64-bit quad-core processor, 2) dual-band wireless LAN, 3) faster Ethernet, and 4) Power-over-Ethernet (with separate PoE HAT). A micro SD card with NOOBS is installed on the Rasp3+. In memory of each SD card is 1GB.

We develop a monitoring procedure and a consensus procedure for each Rasp3+. The monitoring procedure loads a learning model and executes it for object recognition. The consensus procedure publishes local knowledge to its known Rasp3+ and subscribes the other Rasp3+'s knowledge, and simultaneously interacts with users to ensure which Rasp3+ win the right of block generating. The input of the monitoring procedure is image data and the output is recognition. The input of the consensus procedure is knowledge, previous block, and user's votes, while the output of the consensus procedure is proposals, current block, and voting results. There are two different kinds of deploying methods for these two procedures. The first is integrating the monitoring procedure into the camera and the second is installing it on the Rasp3+ together with consensus procedure.

Communication between the mentioned procedures adopts XML remote procedure call (XML-RPC), which enables heterogeneous operating systems. XML-RPC uses HTTP as the transport protocol and XML language as the coding format. Usually. XML-RPC involves a client and a server. The XML-RPC client sends requests to the server using HTTP messages. The corresponding server will respond to it after request handling. In our experiment, we modify the handling process of the HTTP request. When the consensus procedure receives a message sent by the monitoring procedure, it will parse the content of this message, associate it with local knowledge, votes for the proposals, and then calculate its popularity among users.

Based on XML-RPC, we establish a peer-to-peer network to transport the blockchain with complicated data structure between multiple Rasp3+. In our experiment, two different scenarios are considered. For the static scenario, the topology of the peer-to-peer network is fixed and which Rasp3+ will participate in the consensus process is known. For the dynamic scenario, the relationship between Rasp3+ is unknown and each Rasp3+ is free to join and quit the proposed consensus scheme.

It is easy to understand that, in the battlefield, an outstanding

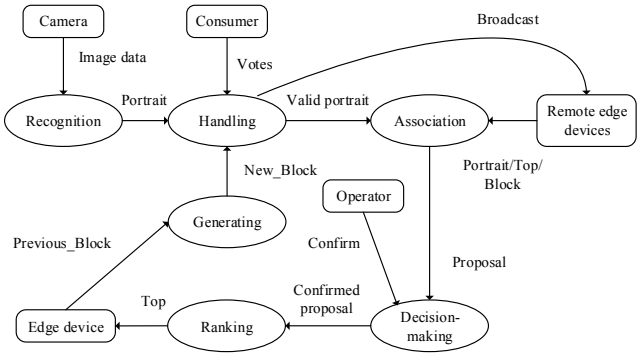


Fig. 3: Data flow diagram of designed experiment.

scout not only needs to have great wisdom to collect the information about enemy number, habits and fire configuration but also needs to protect this collected information at all cost to ensure that the collected information can be sent to their general. Before the general precisely and timely receives the collected information, it will face many unpredicted emergencies. For example, the scout may be captured, delayed and deceived. Moreover, the information may be destroyed, tampered and tarried.

The edge knowledge should be protected as secure as the observed information. In our experiment, a reasonable hypothesis is that the Rasp3+ can not tamper data unless it has been captured by attackers. The design goal of our experiment is to validate the consensus is convinced, the privacy of participates can be protected and the cost of consensus is low. In our experiment, six main functions are developed. For readers' easy understanding, a view of the data flow diagram (DFD) is provided as shown in Fig. 3, which aims to bring out the data processing model of the designed experiment.

Following the arrows in DFD, we can see that there are three parameters should be imputed to Handling function: 1) knowledge recognized by learning models, 2) current block generated by Generating function, and 3) votes sent by consumers. Correspondingly, this function provides various services such as broadcasting, validity checking and votes calculating. The ECI in one group can share and backup their recognitions with each other over the XML-RPC based peer-to-peer network. The relationship between peers is formulated as small-world theory, that means the social distance between any two peers is less than 6. Therefore, we set that the knowledge only can be forwarded within 6 times.

Association function executes association analysis of valid knowledge beyond the constraint of small-world theory. Valid knowledge recognized by edge devices in one group is mapped into various unconfirmed proposals. It seems a recommendation system where operators choose one proposal as their decisions. Once one new proposal is confirmed by operators, the popularity ranking of confirmed proposals will be updated, and ultimately the top one popular confirmed proposal will be selected. At the last step of the POP consensus process, a Rasp3+ will be selected by parsing the contribution of each edge to the most popular proposal.

B. Low RAM Usage

In our experiments, we considered two different KPIs and obtained two kinds of simulations results: 1) RAM usage; 2) block generating delay. Before giving a further analysis of the simulation results, we would like to explain why we select RAM as the KPI. Predominate functions of the DINE are to execute the AI model to process the local big data and provide real-time services for time-sensitive end users. An available security policy should not occupy too many resources. RAM usage is an important resource occupation inductor. A procedure with high RAM usage will impact the response performance of other procedures. Moreover, if a procedure is occupying too many RAM resources, it is easy to be impacted especially when the DoS attacking appears.

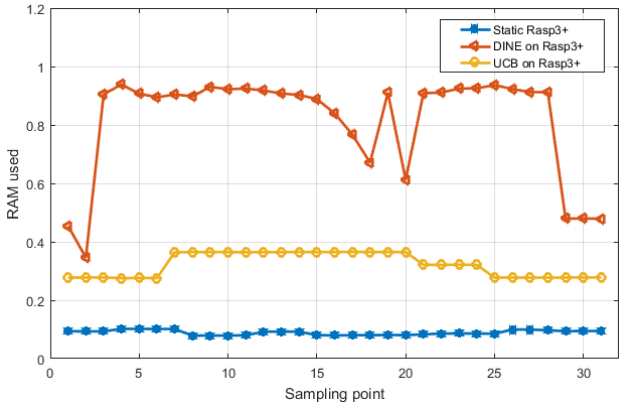


Fig. 4: Comparison of RAM usage.

Fig. 4 shows the RAM used versus sampling slot with three different scenario configurations. In the scenario of static Rasp3+, Rasp3+ is not equipped with any DINEs or consensus mechanisms. We use the RAM usage of static Rasp3+ as the baseline compared with the other two scenarios. DINE on Rasp3+ installs one pre-trained learning model on the Rasp3+ to discover knowledge in the local network. During the knowledge discovery period, we uniformly produce 32 sampling points to measure the RAM usage of the Rasp3+. It can be found that the RAM usage volume of Rasp3+ is about 0.93GB/1GB when the DINE is running. To measure the RAM usage of proposed PoP, we collect the output of DINEs as the pending consensus transactions. Therefore, each client procedure reads the pending consensus transactions from a pre-defined file. The experiment results show that the RAM usage of the proposed PoP mechanism is about 0.39GB/1GB. Compared to the DINE on Rasp3+, the RAM usage of the proposed PoP mechanism is very low.

1) Mitigation of Unknown DoS attacking: Due the decentralization nature of IoT, it is common to see unknown DoS attacking. For both end users and DINEs can participate the PoP consensus, the PoP consensus mechanism with low RAM usage can mitigate this attack. Low RAM usage is brought by introducing end users into PoP consensus algorithm. This can bring three significant benefits. Firstly, it will mitigate the working burdens of each DINE because the validation of edge knowledge transactions is shifted from DINEs into the end users' devices. Secondly, It will mitigate the ledger overflow when it is suffering from DoS attacks, for only the new edge knowledge can trigger block generating procedure. Thirdly, the consensus process won't be interrupted even if a DINE is attacked by DoS attacking, because the end users in this local network can access the neighboring DINEs. Therefore, unknown DoS attacks only can impact edge knowledge discovering and publishing on some DINEs, but it has no impact on PoP consensus.

During the DoS attacking phase, an attacked DINE can not discover new knowledge, calculate knowledge popularity and compete for the privilege of block generating. In our design, each DINE should monitor both local knowledge and remote knowledge. Once a knowledge is received, the DINEs will add it into a temporary knowledge list. Such design provides a backup for knowledge so that failure of one DINE has no impact on the consistency of PoP mechanism except for the block generating delay. Simultaneously, a normal DINE should check the validation of previous block generators. An attacked DINE can not compete for the privilege of block generating, leading to several blocks missing.

C. Fast Block Generating

The block generating speed is an important performance parameter for blockchain in IoT. Faster block generating means that more knowledge transactions can be processed. We consider a blockchain network with three groups of DINEs. Each group consists of 4 DINEs and 10 end users. The DINEs are connected without any loops or branches. For each DINE, the average delay of submitting a knowledge is 100ms (which is equal to the delay requirement in 5G edge network [41]. We illustrate the delay of block generating in Fig. 5. The block generating delay of UCB with 3 DINEs increase faster than the block generating delay of UCB with 4 DINEs. Notably, it also can be found that the block hight UCB with 3 DINEs is less than UCB with 4 DINEs.

1) Desensitise edge knowledge releasing: The knowledge is learned by different DINEs, which are usually deployed in the local area network distributively. Spurred on by the profiteering, the adversaries may try various means to steal the content of knowledge or infer the identity of its owner. Therefore, the knowledge must be desensitized before they are released to share with other DINEs.

For the purpose of knowledge desensitization, the UCB enables anonymous knowledge releasing. Moreover, it maps the content of each knowledge into a hashing value when a new block will be generated. Further, the encrypted knowledge will be orchestrated as topics in UCB framework. Only the trusted subscribers can gain the permissions (such as read & write) to operate the content of knowledge. Such an orchestration prevent a corrupt DINE with higher security level from sending private knowledge to nodes with a lower security level.

2) Weakening the pseudo DINE: The proposed framework will not give the pseudo DINEs additional advantages because the PoP solution treats knowledge popularity as wealthy to

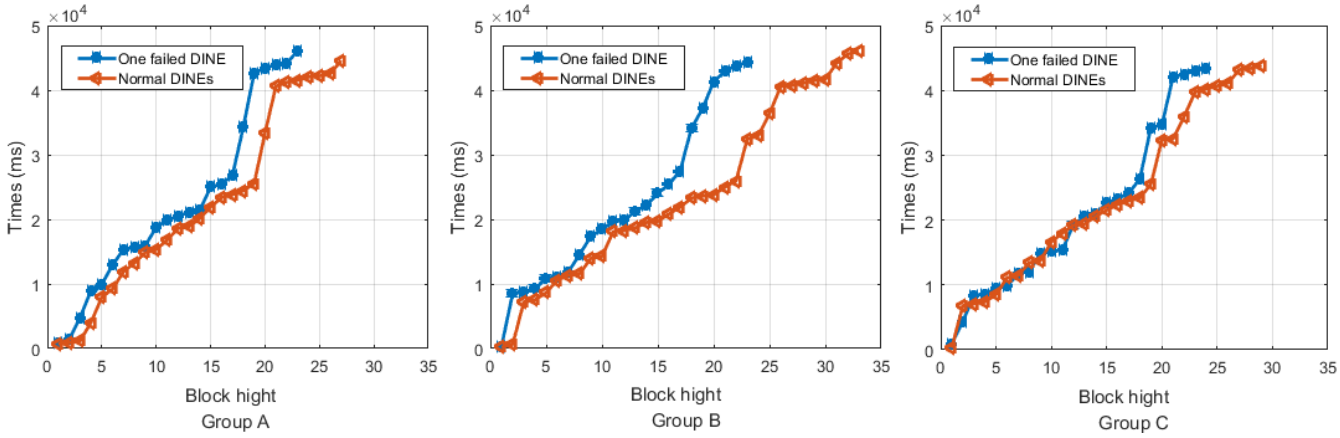


Fig. 5: Comparison of block generating delay. When one DINE suffer from DoS attacking and fails to work, the blocks that should be generated by it originally can not be generated but the transactions can be recorded on the next block.

compete the privilege of block generating. Firstly, in order to release the false knowledge, a pseudo-DINE needs to spend more costs to build a social relationship with the normal DINEs. If there are no real DINEs that are willing to subscribe the false knowledge, this false knowledge can not be sent into the blockchain network. Secondly, it is extremely hard for a pseudo-DINE to acknowledge what is the most valuable knowledge in the blockchain network because normal knowledge are usually encrypted. Without acknowledging the most valuable knowledge, the pseudo DINE can not gain the privilege of block generating. Thirdly, even if the pseudo-DINE happens to release a valuable knowledge and gain the privilege of block generating, it only can tamper the transactions between this pseudo-DINE and its neighboring DINEs. This will result in redundant knowledge sharing, which can be inspected and reduced by using the existing transaction filtering approaches easily.

3) Low successful rate of 51% attacking: As mentioned in Section IV, the proposed PoP mechanism enables flexible interactions between end users and DINEs. Different from the existing consensus mechanisms which only take care of what the blockchain nodes, the proposed PoP mechanism shifts the focus of blockchain from what the blockchain nodes have into what is valuable in a DINE-maintained local network.

In our framework, the 51% attacking has a low successful rate. Firstly, any DINE is impossible to attract more than half of end users because it only can maintain local end users. Secondly, the interests of end users are dynamic, any DINE is impossible to continuously gain the privilege of block generating.

VI. Conclusion

In this paper, we studied the security of ECI in IoT. When the outputs of each DINE are modelled as important knowledge, service providers may not trust each other, directly leading to difficulty of knowledge sharing among DINEs. Moreover, due to the cyber-physical vulnerability and mobility of edge computational infrastructures, the edge knowledge sharing faces with many security threats. Different from the existing studies that focus on sharing large-scale IoT data, a novel user-centric blockchain (UCB) scheme is proposed to share edge knowledge in IoT. To preserve the security of edge knowledge sharing, end users are introduced into the blockchain consensus mechanism, which is formulated as proof of popularity (PoP) consensus. In UCB, the scattered edge knowledge is stitched as proposals automatically and broadcast to trusted partners by using Pub/sub protocol. To evaluate the superiority of proposed schemes, we state the security analysis to fulfill the mentioned security requirements and then demonstrate the experiment evaluation, which is developed based on Raspberry Pi 3 Model B. Our work will promote the application of computational intelligence in network edges of IoT.

Acknowledgements

This work is supported by National Natural Science Foundation of China (Grant No.61431008) and partially supported by JSPS KAKENHI Grant Numbers JP16K00117, JP19K20250, Leading Initiative for Excellent Young Researchers (LEADER), MEXT, Japan and KDDI Foundation.

References

[1] P. G. Lopez, A. Montresor, D. Epema, et al. “Edge-centric Computing: Vision and Challenges, ” *Acm Sigcomm Computer Communication Review*, vol. 45, no. 5, pp. 37-42, 2015.

[2] S. B., Calo, D. C., Verma, and E. Bertino, “Distributed Intelligence: Trends in the Management of Complex Systems ,” *Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies (ACMT)*, Indianapolis, USA, June. 2017.

[3] TX. Tran, A. Hajisami, P. Pandey, D. Pompili, “Collaborative Mobile Edge Computing in 5G Networks: New Paradigms, Scenarios, and Challenges, ” *IEEE Communications Magazine*, vol. 55, no. 4, pp. 54-61, 2017.

[4] W. Zhang, Z. Zhang, H.C. Chao, “Cooperative Fog Computing for Dealing with Big Data in the Internet of Vehicles: Architecture and Hierarchical Resource Management, ” *IEEE Communications Magazine*, vol. 55, no. 12, pp. 60-67, 2017.

[5] Z. Sheng, C. Mahapatra, et al, "Energy efficient cooperative computing in mobile wireless sensor networks" *IEEE Transactions on Cloud Computing*, vol. 6, no. 1, pp. 114-126, 2018.

[6] T. Wang, Q. Yang, K. Tan, et al. "DCAP: Improving the Capacity of WiFi Networks with Distributed Cooperative Access Points," *IEEE Transactions on Mobile Computing*, vol. 17, no. 2, pp. 320-333, 2018.

[7] X. Lyu, C. Ren, W. Ni, H. Tian, and R. P. Liu, "Distributed optimization of collaborative regions in large-scale inhomogeneous fog computing," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 574-586, 2018.

[8] Y. Harchol, A. Mushtaq, J. McCauley, A. Panda, S. Shenker, "CESSNA: Resilient Edge-Computing, " *Proceedings of the SIGCOMM Workshop on Mobile Edge Communications*, pp. 1-6, Budapest, Hungary, August 20, 2018.

[9] Z. Ning, P. Dong, X. Kong, F. Xia, "A Cooperative Partial Computation Offloading Scheme for Mobile Edge Computing Enabled Internet of Things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4804-4814, 2018.

[10] G. Li, J. Wu, J. Li, K. Wang, and T. Ye, "Service Popularity-based Smart Resources Partitioning for Fog Computing-enabled Industrial Internet of Things, " *IEEE Transactions on Industrial Informatics* , vol. 14, no. 10, pp. 4702-4711, 2018.

[11] M. Hogan, F. Esposito, "A Portfolio Theory Approach to Edge Traffic Engineering via Bayesian Networks, " *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, Snowbird, Utah, USA, Oct. 16-20, 2017.

[12] B., Tang, Z., Chen, et al., "Incorporating intelligence in fog computing for big data analysis in smart cities, " *IEEE Transactions on Industrial Informatics*, vol. 13, no. 5, pp. 2140-2150, 2017.

[13] Y. Jin, Z. Qian, S. Guo, S. Zhang, X. Wang, S. Lu, "ran-GJS: Orchestrating Data Analytics for Heterogeneous Geo-distributed Edges, " *Proceedings of the 47th International Conference on Parallel Processing*, pp. Eugene, OR, USA. August 13 -16, 2018.

[14] R. Yu, J. Ding, S. Maharjan, S. Gjessing, Y. Zhang, D.H.K. Tsang, "Decentralized and Optimal Resource Cooperation in Geo-Distributed Mobile Cloud Computing, " *IEEE Transactions on Emerging Topics in Computing*, vol. 6, no. 1, pp. 72-84, 2018.

[15] J. Dold and J. Groopman, "The future of geospatial intelligence, " *Geospatial Information Science*, vol. 20, no. 02, pp. 151-162, 2017.

[16] F. Li, B. Luo, P. Liu, D. Lee, C. Chu, "Enforcing Secure and Privacy-Preserving Information Brokering in Distributed Information Sharing, " *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 6, pp. 888-900, 2013.

[17] H. Yang, J. Lee, "Secure Distributed Computing With Straggling Servers Using Polynomial Codes, " *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 1, pp. 141-150, 2019.

[18] D.M. Shila, W. Shen, Y. Cheng, X. Tian, X.S. Shen, "Toward a Secure Autonomic Mobile Ad Hoc Cloud Computing System, " *IEEE Wireless Communications*, vol. 24, no. 2, pp. 74-81, 2016.

[19] A. Castiglione, P. DArco, A.D. Santis, R. Russo, "Secure group communication schemes for dynamic heterogeneous distributed computing, " *Future Generation Computer Systems*, vol. 74, pp. 313-324, 2017.

[20] X. Ren, P. London, J. Ziani, A. Wierman, "Datum: Managing Data Purchasing and Data Placement in a Geo-Distributed Data Market, " *IEEE/ACM Transactions on Networking*, vol. 26, no. 2, pp. 893 - 905, 2018.

[21] I. Psaras, "Decentralised Edge-Computing and IoT through Distributed Trust," *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, pp. 505-507, Munich, Germany, June 10-15, 2018.

[22] N. Z. Aitzhan, D, Svetinovic, "Security and Privacy in Decentralized Energy Trading Through Multi-Signatures, Blockchain and Anonymous Messaging Streams, "*IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 5, 840-852, 2018.

[23] W. Hou, L. Guo, Z. Ning, "Local Electricity Storage for Blockchain-based Energy Trading in Industrial Internet of Things, " *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, 3610-3619, 2019.

[24] Y. Lu, A. Kumar, S. Zhai, Y. Cheng, T. Javidi, R.S. Feris, "Fully-adaptive Feature Sharing in Multi-Task Networks with Applications in Person Attribute Classification, " *CVPR*, 2017

[25] L.N. Le, D.L. Jones, "Feature-Sharing in Cascade Detection Systems With Multiple Applications, " *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 3, pp. 466-478, 2017.

[26] C. Liu, C.T. Zheng, S. Wu, Z. Yu, H.S. Wong, "Multitask Feature Selection by Graph-Clustered Feature Sharing, " *IEEE Transactions on Cybernetics*, vol. PP, no. 99, pp. 1-13, 2018.

[27] A. Torralba, K.P. Murphy, W.T. Freeman , "Sharing Visual Features for Multiclass and Multiview Object Detection, " *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 5, pp. 854 - 869, 2007.

[28] S.J. Hwang, F. Sha, K. Grauman, "Sharing features between objects and their attributes, " *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, pp. 1761-1768, June 20-25, 2011.

[29] B.K. Zheng, L.H. Zhu, et al, "Scalable and Privacy-Preserving Data Sharing Based on Blockchain, " *Journal of Computer Science and Technology*, vol. 33, no. 3, pp. 557567, 2018.

[30] M.A. Ferrag, M. Derdour, et al, "Blockchain Technologies for the Internet of Things: Research Issues and Challenges, " *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2188-2204, 2018.

[31] J. Pan, J. Wang, A. Hester, I. Alqerm, Y. Liu, Y. Zhao, "EdgeChain: An Edge-IoT Framework and Prototype Based on Blockchain and Smart Contracts, " *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4719-4732, 2018.

[32] G. Zhang, T. Li, Y. Li, P. Hui, D. Jin, "Blockchain-Based Data Sharing System for AI-Powered Network Operations, " *Journal of Communications and Information Networks*, vol. 3, no. 3, pp 18, 2018.

[33] EAP. Alchieri, A. Bessani, F. Greve, and JDS. Fraga, "Knowledge Connectivity Requirements for Solving Byzantine Consensus with Unknown Participants, " *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 2, pp. 246-259, 2018.

[34] C. Xu, K. Wang, P. Li, S. Guo, J. Luo, B. Ye, M. Guo, "Making Big Data Open in Edges: A Resource-Efficient Blockchain-based Approach, " *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 4, pp. 870-882, 2019.

[35] H. Li, K. Ota, M. Dong, "Deep Reinforcement Scheduling for Mobile Crowdsensing in Fog Computing, " *ACM Transactions on Internet Technology (TOIT)*, vol. 19, no. 2, 2019.

[36] M. Polese, R. Jana, V. Kounev, K. Zhang, S. Deb, M. Zorzi, "Machine Learning at the Edge: A Data-Driven Architecture with Applications to 5G Cellular Networks, " *arXiv:1808.07647v3*, 2018.

[37] J. Wu, M. Dong, K. Ota, J. Li, Z. Guan, "FCSS: Fog-Computing-based Content-Aware Filtering for Security Services in Information-Centric Social Networks, " *IEEE Transactions on Emerging Topics in Computing*, DOI: 10.1109/TETC.2017.2747158, 2018.

[38] H. Li, K. Ota, M. Dong, "Learning IoT in Edge: Deep Learning for the Internet of Things with Edge Computing, " *IEEE Network*, vol. 32, no. 1, pp. 96-101, 2018.

[39] T. Wang, J. Zhou, X. Chen, G. Wang, A. Liu, Y. Liu, "A Three-Layer Privacy Preserving Cloud Storage Scheme Based on Computational Intelligence in Fog Computing, " *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 1, pp. 3-12, 2018.

[40] Z. Ning, J. Huang, X. Wang, "Vehicular Fog Computing: Enabling Real-Time Traffic Management for Smart Cities, " *IEEE Wireless Communications*, vol. 6, no. 1, pp. 87-93, 2019.

[41] B. Yu, J. Wright, S. Nepal, L. Zhu, J. Liu, R. Ranjan, "IoTChain: Establishing Trust in the Internet of Things Ecosystem Using Blockchain, " *IEEE Cloud Computing*, vol. 5, no. 4, pp. 12-23, 2018.

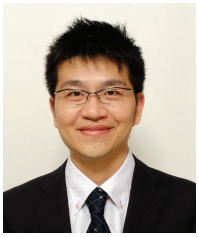
[42] A. Meloni, S. Madanapalli, et al, "Exploiting the IoT Potential of Blockchain in the IEEE P1931.1 ROOF Standard, " *IEEE Communications Standards Magazine*, vol. 2, no. 3, pp. 38-44, 2018.

[43] M. Liu, R. Yu, Y. Teng, V.C.M. Leung, M. Song, "Computation Offloading and Content Caching in Wireless Blockchain Networks with Mobile Edge Computing, " *IEEE Transaction on Vehicular Technology*, vol. 67, no. 11, pp. 11008 - 11021, 2018.

[44] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, Z. Han, "When Mobile Blockchain Meets Edge Computing, " *IEEE Communications Magazine*, vol. 15, no. 2, pp. 33 - 39, 2018.

[45] W. Wang, D. T. Hoang, Z. Xiong, D. Niyato, P. Wang, P. Hu, Y. Wen, "A Survey on Consensus Mechanisms and Mining Management in Blockchain Networks, " *arXiv:1805.02707v1*, pp. 1-33, May. 07, 2018.

Gaolei Li (S'16) received the B.S. degree in electronic information engineering from Sichuan University, Chengdu, China and he is pursuing the Ph.D. degree in School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China. He is a visiting student at Muroran Institution of Technology, Muroran, Japan, supported by the China Scholarship Council (CSC) Program from October 2018 to September 2019. His research interests are focusing on edge computing, blockchain, and artificial intelligence.



Mianxiong Dong received B.S., M.S. and Ph.D. in Computer Science and Engineering from The University of Aizu, Japan. He became the youngest ever Professor of Muroran Institute of Technology, Japan where he currently serves Advisor to Executive Director, and Vice Director of Office of Institutional Research. He was a JSPS Research Fellow with School of Computer Science and Engineering, The University of Aizu, Japan and was a visiting scholar with BBCR group at University of Waterloo, Canada supported by JSPS Excellent Young Researcher Overseas Visit Program from April 2010 to August 2011. Dr. Dong was selected as a Foreigner Research Fellow (a total of 3 recipients all over Japan) by NEC C & C Foundation in 2011. His research interests include Wireless Networks, Cloud Computing, and Cyber-physical Systems. He has received best paper awards from IEEE HPCC 2008, IEEE ICESS 2008, ICA3PP 2014, GPC 2015, IEEE DASC 2015, IEEE VTC 2016-Fall, FCST 2017, 2017 IET Communications Premium Award and IEEE ComSoc CSIM Best Conference Paper Award 2018. Dr. Dong serves as an Editor for IEEE Transactions on Green Communications and Networking (TGCN), IEEE Communications Surveys and Tutorials, IEEE Network, IEEE Wireless Communications Letters, IEEE Cloud Computing, IEEE Access, as well as a leading guest editor for ACM Transactions on Multimedia Computing, Communications and Applications (TOMM), IEEE Transactions on Emerging Topics in Computing (TETC), IEEE Transactions on Computational Social Systems (TCSS). He has been serving as the Vice Chair of IEEE Communications Society Asia/Pacific Region Information Services Committee and Meetings and Conference Committee, Leading Symposium Chair of IEEE ICC 2019, Student Travel Grants Chair of IEEE GLOBECOM 2019, and Symposium Chair of IEEE GLOBECOM 2016, 2017. He is the recipient of IEEE TCSC Early Career Award 2016, IEEE SCSTC Outstanding Young Researcher Award 2017, The 12th IEEE ComSoc Asia-Pacific Young Researcher Award 2017, Funai Research Award 2018 and NISTEP Researcher 2018 (one of only 11 people in Japan) in recognition of significant contributions in science and technology. He is currently the Member of Board of Governors and Chair of Student Fellowship Committee of IEEE Vehicular Technology Society, and Treasurer of IEEE ComSoc Japan Joint Sections Chapter. He is Clarivate Analytics 2019 Highly Cited Researcher (Web of Science).

Laurence T. Yang received the BE degree in computer science and technology and BSc in applied physics both at Tsinghua University, China, and the Ph.D degree in computer science from University of Victoria, Canada. He is currently a professor and W.F. James Research Chair at St. Francis Xavier University, Canada. His research has been supported by the National Sciences and Engineering Research Council, and the Canada Foundation for Innovation. His research interests include parallel and distributed computing, embedded and ubiquitous/pervasive computing, big data and cyber-physical-social systems.

Kaoru Ota was born in Aizu-Wakamatsu, Japan. She received M.S. degree in Computer Science from Oklahoma State University, USA in 2008, B.S. and Ph.D. degrees in Computer Science and Engineering from The University of Aizu, Japan in 2006, 2012, respectively. She is currently an Associate Professor with Department of Sciences and Informatics, Muroran Institute of Technology, Japan. From March 2010 to March 2011, she was a visiting scholar at University of Waterloo, Canada. Also she was a Japan Society of the Promotion of Science (JSPS) research fellow with Kato-Nishiyama Lab at Graduate School of Information Sciences at Tohoku University, Japan from April 2012 to April 2013. Her research interests include Wireless Networks, Cloud Computing, and Cyber-physical Systems. Dr. Ota has received best paper awards from ICA3PP 2014, GPC 2015, IEEE DASC 2015, IEEE VTC 2016-Fall, FCST 2017, 2017 IET Communications Premium Award and IEEE ComSoc CSIM Best Conference Paper Award 2018. She is an editor of IEEE Transactions on Vehicular Technology (TVT), IEEE Internet of Things Journal, IEEE Communications Letters, Peer-to-Peer Networking and Applications (Springer), Ad Hoc & Sensor Wireless Networks, International Journal of Embedded Systems (Inderscience) and Smart Technologies for Emergency Response & Disaster Management (IGI Global), as well as a guest editor of ACM Transactions on Multimedia Computing, Communications and Applications (leading), IEEE Internet of Things Journal, IEEE Communications Magazine, IEEE Network, IEEE Wireless Communications, IEEE Access, IEICE Transactions on Information and Systems, and Ad Hoc & Sensor Wireless Networks (Old City Publishing). She is the recipient of IEEE TCSC Early Career Award 2017, and The 13th IEEE ComSoc Asia-Pacific Young Researcher Award 2018.She is Clarivate Analytics 2019 Highly Cited Researcher (Web of Science).

Jun Wu (S'08-M'12) received the Ph.D. degree in information and telecommunication studies from Waseda University, Japan, in 2011. He was a Postdoctoral Researcher with the Research Institute for Secure Systems, National Institute of Advanced Industrial Science and Technology (AIST), Japan, from 2011 to 2012. He was a Researcher with the Global Information and Telecommunication Institute, Waseda University, Japan, from 2011 to 2013. He is currently an Associate Professor with the School of Cyber Security, Shanghai Jiao Tong University, China. He is also the Vice Director of the National Engineering Laboratory for Information Content Analysis Technology, Shanghai Jiao Tong University. He has hosted and participated in a lot of research projects, including the National Natural Science Foundation of China (NFSC), the National 863 Plan and 973 Plan of China, and the Japan Society of the Promotion of Science Projects (JSPS). His research interests include the advanced computing, communications and security techniques of software-defined networks (SDN), information-centric networks (ICN) energy Internets, and the Internet of Things (IoT). He is also the Chair of the IEEE P21451-1-5 Standard Working Group. He has been a Guest Editor of the IEEE SENSORS JOURNAL. He is an Associate Editor of the IEEE ACCESS.

Jianhua Li gets his BS, MS and Ph.D. degrees from Shanghai Jiao Tong University, in 1986, 1991 and 1998, respectively. He is a professor/Ph.D. supervisor of School of Cyber Security, Shanghai Jiao Tong University, Shanghai Key Laboratory of Integrated Administration Technologies for Information Security, Shanghai, China. He is the chief expert in the information security committee experts of National High Technology Research and Development Program of China (863 Program) of China. He got three First Prize of Technology Progress Awards of Shanghai in 2003, 2004, and 2013. He got the Second Prize of National Technology Progress Award of China in 2005. He was the Chief Expert in the Information Security Committee Experts of National High Technology Research and Development Program of China. His research interests include cyberspace security, data science, etc.