A Lightweight and Attack-Proof Bidirectional Blockchain Paradigm for Internet of Things

Chenhao Xu, Youyang Qu, *Member, IEEE*, Tom H. Luan, *Senior Member, IEEE*, Peter W. Eklund, Yong Xiang, *Senior Member, IEEE*, and Longxiang Gao, *Senior Member, IEEE*

***Abstract*—Diverse technologies, such as machine learning and big data, have been driving the prosperity of the Internet of Things (IoT) and the ubiquitous proliferation of IoT devices. Consequently, it is natural that IoT becomes the driving force to meet the increasing demand for frictionless transactions. To secure transactions in IoT, blockchain is widely deployed since it can remove the necessity of a trusted central authority. However, the mainstream blockchain-based IoT payment platforms, dominated by Proof-of-Work (PoW) and Proof-of-Stake (PoS) consensus algorithms, face several major security and scalability challenges that result in system failures and financial loss. Among the three leading attacks in this scenario, double-spend attacks and long-range attacks threaten the tokens of blockchain users, while eclipse attacks target Denial of Service. To defeat these attacks, a novel bidirectional-linked blockchain (BLB) using chameleon hash functions is proposed, where bidirectional pointers are constructed between blocks. Furthermore, a new committee members auction (CMA) consensus algorithm is designed to improve the security and attack resistance of BLB while guaranteeing high scalability. In CMA, distributed blockchain nodes elect committee members through a verifiable random function. The smart contract uses Shamir's secret-sharing scheme to distribute the trapdoor keys to committee members. To better investigate BLB's resistance against double-spend attacks, an improved Nakamoto's attack analysis is presented. In addition, a modified entropy metric is devised to measure eclipse attack resistance across different consensus algorithms. Extensive evaluation results show the superior resistance against attacks and demonstrate high scalability of BLB compared with current leading paradigms based on PoS and PoW.**

***Index Terms*—Bidirectional blockchain, double-spend attack, eclipse attack, Internet of Things (IoT), long-range attack, scalability.**

I. Introduction

INTERNET of Things (IoT) is experiencing a fast booming in recent years, along with which IoT devices are already ubiquitous, such as mobile devices, car terminals, wearable devices, etc. Not surprisingly, the proliferation of IoT devices meets the increasing demands of contactless payment via IoT devices, which attracts growing attention from both academia and industry [1], [2]. For example, Samsung has launched its IoT payment platform on smart and wearable devices, TVs, fridges, and even more. At the same time, automobile giants such as SAIC Motor have embedded their cars with a comprehensive mobile payment system. On account of the popularization of IoT devices, machine-to-machine (M2M) payment as a paradigm is playing an ever-growing important role in the IoT [3].

In M2M payments, centralized transaction management central has relatively poor performances due to the distributed nature of the IoT. Collecting all the transaction information to a central server causes incredibly massive communication overhead, which leads to delayed transactions and low efficient operation. Moreover, the centralized operation mode is vulnerable to single-point failure, while various man-in-the-middle attacks are unceasingly launched due to the financial value of transaction information. Thus, a decentralized and autonomous payment architecture better meets the needs of the IoT. Blockchain, as an emerging distributed ledger technology (DLT), is decentralized and allows for secure, anonymous, and immutable transactions [4]–[7]. Therefore, it is seen as one of the most promising solutions for M2M IoT payments.

However, several serious challenges remain to put blockchain into practice. For example, there are several security vulnerabilities and corresponding attacks launched on existing blockchain-based solutions. Only from July 2019 to February 2020, at least 18 double-spend attacks on four cryptocurrencies were observed by the Reorg Tracker [8]. A double-spend attack, which manifests in blockchain networks using Proof-of-Work (PoW) consensus, is an attack where malicious users spend the same tokens at least twice [9]. Consequently, PoW-based blockchain systems are forced to sacrifice computing power and transaction efficiency to improve security. An instance is Bitcoin, in which about 10 min is required to generate a block, and a merchant has to wait for at least six confirmations of a transaction (meaning that six subsequent blocks of transactions were added to the blockchain) before the transaction is safely assumed as

Manuscript received May 6, 2021; revised July 1, 2021; accepted August 4, 2021. Date of publication August 9, 2021; date of current version March 7, 2022. *(Corresponding author: Longxiang Gao.)*

Chenhao Xu, Youyang Qu, Yong Xiang, and Longxiang Gao are with the Deakin Blockchain Innovation Lab, School of Information Technology, Deakin University, Geelong, VIC 3220, Australia (e-mail: xueri@deakin.edu.au; y.qu@deakin.edu.au; yong.xiang@deakin.edu.au; longxiang.gao@deakin.edu.au).

Tom H. Luan is with the School of Cyber Engineering, Xidian University, Xi'an 710126, China (e-mail: tom.luan@xidian.edu.cn).

Peter W. Eklund is with the School of Information Technology, Deakin University, Geelong, VIC 3220, Australia (e-mail: peter.eklund@deakin.edu.au).

Digital Object Identifier 10.1109/JIOT.2021.3103275

2327-4662 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

valid [10]. Therefore, PoW-based blockchain systems fail to be applied in high-frequency trading scenarios despite various advantageous features. Similarly, a long-range attack is manifest in Proof-of-Stake (PoS)-based blockchain networks targeting double-spending tokens [11]. Moreover, there are also eclipse attacks [12] that cause Denial of Service (DoS), especially for IoT networks [13]. Beyond the aforementioned vulnerabilities of existing blockchain systems, the scalability of IoT devices has always been a critical bottleneck [3], [14], because a broadcasting consensus algorithm is usually highly time consuming.

To address the above challenges, some existing research has been conducted. To prevent double-spend attacks, a double-spending prevention mechanism for Bitcoin zero-confirmation transactions is proposed [15]. However, it is only applicable to Bitcoin or UTXO models. To defend against long-range attacks, checkpoints are adopted to define the correct chain periodically in [16]. However, it is vulnerable to DDoS attacks, especially when creating checkpoints. In terms of eclipse attacks, an eclipse-attack detection model for Ethereum is proposed [12]. Nevertheless, it is only responsible for detecting attack traffic based on the two selected features. To the best knowledge, a generalized and lightweight blockchain paradigm that is able to defeat all of the above attacks has not yet been fully considered.

Motivated by the related researches, a novel bidirectional-linked blockchain (BLB) and a tailor-made Committee Members Auction (CMA) consensus algorithm for a secure and scalable IoT-based payment system are proposed. In the proposed model, a chameleon hash function (CHF) [17] is introduced for an extra reverse pointer in the blockchain, which enables BLB to resist double-spend attacks by adding a pointer from the previous block to the next block. Secret sharing [18] is used for the distribution of the trap-door key, which eliminates eclipse attacks in BLB. Finally, a verifiable random function (VRF) [19] is utilized for committee members election. After the election, elected nodes (i.e., committee members) are responsible for cross-verifying transactions in that period (named "term"). Since the committee members elected in each term are random, it is difficult for attackers to predict or control the next term's committee members and impossible for specific committee members to dominate the consensus process. This helps the blockchain resist long-range attacks. All in all, the joint integration of the novel reverse pointer and the CMA consensus algorithm can improve the security and scalability of blockchain. In order to demonstrate this claim, Nakamoto's analysis of the success rate of double-spend attacks [9] is improved by defining the probability of an attacker finding the next block under the PoS and CMA consensus algorithms. In addition, entropy in information theory is adopted to measure the randomness of nodes participating in transactions' verification for different consensus algorithms (CMA, PoW, and PoS). Higher entropy means more uncertain nodes participating in transaction verification, that is, higher resistance to eclipse attacks. Furthermore, abundant experiments are conducted, and the results testify to the improved attack resistance and higher scalability of BLB.

The main contributions of this article are as follows.

1) A novel BLB and a specially designed CMA consensus algorithm are devised using advanced cryptography tools. The effective integration can significantly and comprehensively improve the security of blockchain while ensuring scalability.
2) Theoretical analysis using improved Nakamoto's double-spend attack analysis and modified entropy metric is conducted. The selected methods quantify the security protection levels of CMA, PoW, and PoS consensus algorithms and testify the improved performance of CMA;
3) Extensive experiments to evaluate BLB have been conducted. The results demonstrate that the security and scalability of the proposed paradigm are superior compared with existing leading ones, such as PoS and PoW-based blockchain systems.

The remainder of this article is organized as follows. In Section II, related works are presented. Section III describes the structure of BLB and the CMA consensus algorithm. In Section IV, the security of the proposed model is analyzed. In Section V, the experiment results prove that the security of the proposed model is higher than PoW and PoS-based blockchain models, and the scalability of the proposed model is also significantly competitive. Finally, Section VI presents conclusions and the future work.

II. Related Works

The related works on the defense of blockchain attacks are illustrated in this section. In addition, the cryptography tools utilized by the proposed model are CHFs, VRFs, and secret sharing. The research relevant to these cryptography tools are also presented in this section.

A. Cryptography Tools

The chameleon-hash function is a hash function that involves a trapdoor, the knowledge of which allows one to find arbitrary collisions in the domain of the function [17]. The CHF is first applied in blockchain to create a redactable blockchain [20]. In this article, the authors mention that the shares of the trapdoor key could be distributed among several authorities, but no further explanations are elaborated.

The VRFs [19] are pseudorandom functions that provide publicly verifiable proofs for the correctness of the output. VRFs are introduced by Algorand to select committee members [21]. Algorand is a blockchain framework adopting committee-based PoS Byzantine consensus protocol and is able to efficiently scale to billions of users. However, the users in Algorand are weighted based on the balance of tokens in wallets, which means a user with more tokens are more vulnerable to DDoS attacks and cause the performance of blockchain downgraded.

The secret-sharing scheme is a method by which a dealer distributes shares to parties such that only authorized subsets of parties can reconstruct the secret, which is first presented by Shamir and Blakley separately in 1979 [18]. In Shamir's scheme, each participant gets a unique part of the secret. When the number of participants is larger than a given threshold, the

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

original secret can be reconstructed. Shamir's scheme provides a secure trapdoor keys management solution for the proposed model.

B. Attacks and Current Solutions

A double-spend attack is defined as a deliberately fraudulent strategy with users spending the same tokens at least twice in PoW-based blockchain networks [9]. A lightweight countermeasure against double-spend attacks is proposed in 2016 [22], which detects double-spend attacks in fast transactions by using a listening period and inserting observers. Subsequently, a double-spending prevention mechanism is proposed for Bitcoin zero-confirmation transactions [15] by inserting observers of the transactions and setting up an appropriate penalty to prevent users from launching an attack. But in general, as a P2P network, the message delivery between nodes is often not so timely, and the order of messages is not guaranteed, which makes their observers unreliable. Through the analysis and experiments, it is proved that to maintain the performance of the blockchain against the adaptive double-spend attack, a larger number of confirmation blocks for validating a transaction are required [23]. Therefore, a longer transaction confirmation time is required to be waiting by merchants before validating the transaction, and the performance of the blockchain cannot be guaranteed.

A long-range attack is defined as the minority stakeholders in the PoS-based blockchain produce a valid alternative history over a long time span and become majority stakeholders [16]. Checkpoints refer to a block that is considered immutable and is utilized to limit the range of long-range attacks [11]. However, the checkpoint mechanism relies on a centralized server to define a correct chain periodically. An improved checkpoint solution is that the placement of the next checkpoint is determined by the node creating the previous checkpoint [16]. But no further experiments are conducted to measure the security of the node selection algorithm when the attacks came at the time of checkpoint creating. Besides, two PoS protocols preventing attackers from long-range attacks are proposed [11] based on a specific hardware component, such as Intel's SGX or ARM's Trustzone.

An eclipse attack involves an attacker isolating a node in a blockchain network, preventing it from communicating with other nodes [24]. Countermeasures are first presented that make eclipse attacks more difficult [25]. After that, an eclipse-attack detection model for Ethereum is proposed [12]. However, the detection model is trained based on the information in the attack packets, and cannot guarantee the effect on other blockchain platforms or other types of attack data packets. In [26], two protocols are proposed to detect eclipse attacks on Bitcoin clients. The first is an eclipse attack detection protocol that examines suspicious block timestamps. The second is an improved gossip protocol to reduce average attack detection time. However, the experiments to prove the effectiveness of the detection of eclipse attacks are not conducted.

In conclusion, to immunize a blockchain from attack, previous works mainly focus on improving PoW or PoS consensus algorithms, including adding logical steps to existing consensus algorithms or introducing hardware assistance. But these models lead to a decline in the scalability or generality of the blockchain. In this article, a novel BLB with the CMA consensus algorithm is proposed, which has higher performance on the security against attacks and scalability than PoW or PoS-based blockchain models.

TABLE I
NOTATION TABLE

Notation	Description
HashPrev	The hash of the previous block (the forward pointer)
HashNext	The hash of the next block (the reverse pointer)
chash	Chameleon hash
rhash	Regular hash
vhash	Verifiable hash
n	The term/block number
u	The user of blockchain
r	The random number filled in Randomness
m	The content on the block
tk	The trapdoor key
π	The VRF proof
η	The number of participant members in blockchain
τ	The number of committee members in blockchain
sk	The private key of the user
pk	The public key of the user
seed	A random number for each term
γ	A value used to verify committee members
ζ	The proportion of hashes required on the new block
τ	The term period for CMA
z	The number of blocks that the merchant will wait
p	The probability an honest node finds the next block
q	The probability the attacker finds the next block
λ	The blocks producing rate of the attacker
D	The probability the attacker catch up to the honest miners
α	The number of nodes controlled by the attacker
P	The probability of a user participating in the consensus
N_m	The number of miners in PoW
H	The entropy

III. SYSTEM MODEL

The system model is a combination of BLB and the CMA consensus algorithm. All of the notations used are listed in Table I.

A. Bidirectional-Linked Blockchain

Similar to other blockchain models, only appending operation is allowed in the proposed model, which is the foundation of the immutable feature of blockchain. However, BLB has two pointers: 1) a forward pointer, from the next block to the previous block and 2) a reverse pointer, from the previous block to the next. As shown in Fig. 1, Block$_{n+1}$ is a newly generated block, which is appended to the previous Block$_n$. The contents of the block include *HashPrev*, *Transactions*, *HashNext*, and *Randomness*. The hash of the previous block is stored in *HashPrev*, which is used for the forward pointer. *Transactions* store all the transaction information packaged

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

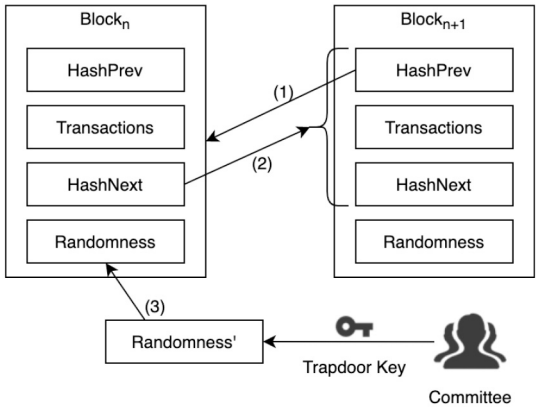


Fig. 1. Newly generated block Block$_{n+1}$ is appended to the previous block Block$_n$. The append steps include: 1) forward pointer construct; 2) reverse pointer construct; and 3) forward pointer repair. After committee members agreed, *Randomness*$'$ is generated by the trapdoor key.

in this blockchain. The hash of the next block (except *Randomness*) is stored in *HashNext*, which is used for the reverse pointer. There is no nonce in the block, instead, *Randomness* is a feature that represents the result of the consensus reached by the distributed participants, since only after all of the members in the committee agree, the trapdoor key of the CHF is constructed. At that point, a new *Randomness* will be calculated, and a pointer from the previous block to the next block will be generated.

Since there are reverse pointers in the proposed model, the process of appending new blocks to the chain can be divided into three steps: 1) the construct of the forward pointer; 2) the construct of the reverse pointer; 3) and the repair of the forward pointer. The specific processes of these three steps are as follows.

1) *Forward Pointer Construct:* The proposed model uses the chameleon-hash function to calculate the hash value (denoted as *chash*) of Block$_n$, and store *chash* into *HashPrev* on Block$_{n+1}$.
2) *Reverse Pointer Construct:* The proposed model calculates the regular hash value (denoted as *rhash*) of Block$_{n+1}$ (except field *Randomness*), and store *rhash* into *HashNext* on Block$_n$.
3) *Forward Pointer Repair:* Subsequently, the value of *HashPrev* on Block$_{n+1}$ will be wrong. However, with the help of the trapdoor key (trapdoor keys are managed by committee members and will be explained in Section III-B), a *Randomness*$'$ is calculated, making the entire hash value of Block$_n$ unchanged. After repair, the forward pointers and reverse pointers point to the correct block.

When constructing the forward pointer on Block$_{n+1}$, operate as follows to find the value for *Randomness*$'$ (which is notated as r'_n): Suppose the original content on Block$_n$ is m_n. After changing the *HashNext* on Block$_n$, m_n becomes m'_n. The trapdoor key of *chash* on Block$_n$ is tk_n. The original random number filled in *Randomness* on Block$_n$ is r_n. According to [27], r'_n can be calculated by

$$r'_n = \frac{m_n + tk_n r_n - m'_n}{tk_n}. \tag{1}$$

B. Committee Members Auction

Based on the data structure of BLB, the CMA consensus algorithm is designed. The process of CMA can be divided into three steps: 1) the election of committee members; 2) the proposal of a new block; and 3) the process of reaching consensus and generating the new block. In addition, with the consideration of the system's randomness, the preparation and renewal of seeds are explained in detail.

1) Election of Committee Members: The consensus process is based on a periodic election that requires all distributed participants to have synchronized clocks. Each election period is called a "term". A seed for each term provides randomness to the consensus algorithm. Users can acquire their verifiable hash vhash and π (π is a VRF proof [27]) by the seed of each term and their respective private keys. π can be used to verify the authenticity of the corresponding vhash. If vhash falls into a specific range γ, the owner is treated as a committee member for this term. Later, this will be explained in detail.

In Algorand [21], users are preferenced as validators based on the number of tokens held in their account. However, this method can result in a system vulnerable to eclipse attacks when tokens are unevenly distributed, i.e., the user holding more tokens is more likely of being responsible for generating new blocks and is, therefore, more vulnerable to eclipse attacks. According to the Pareto Principle, not all things are equal [28], the minority of users (about 20%) in CMA protocol owns the majority of the stakes (about 80%) and are more vulnerable to eclipse attacks. Therefore, CMA treats all users (i.e., participants), irrespective of the number of tokens they hold, equally. All users have the same opportunity to campaign for committee membership. When a new user wants to join a blockchain network, the public key of the new user will be proposed by a recommender (a user that is already in the blockchain network). The new user will become a participant member of the blockchain after being approved by committee members.

Assume that the number of participant members in a blockchain network is η. The number of committee members is τ ($\tau \leq \eta$). A user is selected as the committee member with a probability (τ/η). Each user gets its vhash by computing $VRF(sk, \text{seed}) \rightarrow \langle m, \pi \rangle$, where sk is the private key of the user.

The committee members' range for the vhash is designed as follow: first, the interval [0, 1) is divided into two consecutive intervals $I_0 = [0, 1-p)$ and $I_1 = [1-p, 1)$. Then, γ is calculated as

$$\gamma = \frac{\text{hash}}{2^{\text{hashlen}}} \tag{2}$$

(*hashlen* is the bit-length of *hash*). If γ falls into the interval I_1, then the user holding this hash is elected as a member of the committee for this term.

Committee membership can be verified using its hash value based on π, pk (the public key of the user), and *seed*. Without having ever needing to know sk, the identity of the committee member can be protected, and simultaneously verified.

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 Trapdoor Key Splitting Algorithm

Input: trapdoor key tk, the number of committee members τ, committee member number k
Output: divided secrets *share*

// convert tk charset to integers s_i
$s_i = 0$
define *charset* as all printable string
for *char* in tk **do**
    find the *index* of *char* in *charset*
    $s_i = s_i * len(charset) + index$
end for
Generate new prime number p
// Generate random polynomial coefficients $C[]$
define coefficients $C[]$ as an array
insert s_i into $C[]$
generate τ random integers and append to $C[]$
// generate τ random points based on $C[]$
define *points*[] as an array
for $x = 1; x < \tau + 1; x = x + 1$ **do**
    $y = C[0]$
    for $i = 1; i < len(C); i = i + 1$ **do**
        $exp = (x^i) \bmod p$
        $term = (C[i] \times exp) \bmod p$
        $y = (y + term) \bmod p$
    end for
    append (x, y) to *points*[]
end for
define *shares*[] as an array
for *point* in *points* **do**
    convert x, y in *point* into string s
    append string s to *shares*
end for
return *shares*[k]

After the committee members are selected, the smart contract will automatically fill in the *Randomness* on the last block. Since the forward pointers need to be protected from being tampered by attackers, the trapdoor keys are divided into η parts and distributed to committee members through the secure multiparty computation (MPC) protocol [20]. The MPC protocol is implemented by the smart contract, which is deployed on each node. Under the control of the MPC protocol, only the smart contract on committee members engaged in each round runs the trapdoor key splitting algorithm in parallel. Each part of the trapdoor key stored on the individual node cannot function on its own, which ensures the security of the trapdoor key. After a new valid block is generated, the smart contract will gather all of the parts and repair the forward pointer automatically. The steps of dividing the trapdoor keys are designed based on the secret-sharing algorithm [29], which is shown in Algorithm 1.

2) Propose a New Block: After committee members are elected, they propose new blocks based on the transactions they received through the gossip protocol. As mentioned earlier, the newly proposed block (Block_{n+1}) contains a *HashPrev* pointing to the previous block (Block_n), which is based on the Chameleon hash.

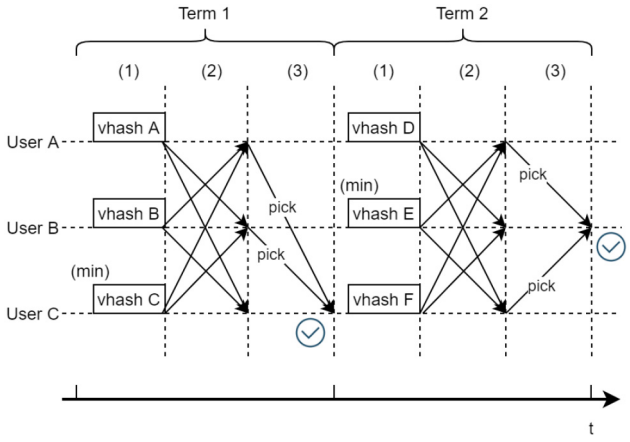


Fig. 2. Period of consensus is divided into three: 1) election of committee members; 2) proposing a new block; and 3) reaching consensus. The block generated by the user who owns the minimal verifiable hash will be accepted.

There are also priorities among committee members. To avoid conflict over generating blocks, whoever has the lowest vhash has the highest block generating priority. When a user receives a block from a higher priority user, it will automatically accept the block. Otherwise, it broadcasts its block through the gossip protocol.

3) Reaching Consensus: When the committee reaches agreement, committee members submit their respective parts of the shared trapdoor key with the hash of the new block Block_{n+1} to the smart contract. When enough secrets are collected, the smart contract can reconstruct the trapdoor key and repair the *Randomness* on Block_n. At this point, the reverse pointer is set to point at Block_n with a new *HashNext*. Finally, Block_{n+1} is appended to the chain with both the forward and the reverse pointer.

4) Preparing the Seed: Before the election of new committee members, a seed is set randomly for each term. This action is very important for the system security because without randomness, committee members can be predicted and the subject of eclipse attack. Since users proposed a vhash_n on the latest block Block_n, a new seed for the next term is generated by hashing the vhash_n. This also means that a user will know the seed for the next term after receiving a new block.

For the seed of the first term, the administrator who is responsible for initializing and deploying the blockchain network can randomly specify one at the beginning of the initialization utilizing the distributed random number generation algorithm [30].

C. Consensus Process

The process of consensus is shown in Fig. 2. There are two terms called "Term 1" and "Term 2". For each term, three-time slices further divide them: 1) election of a committee member; 2) proposing a new block; and 3) reaching consensus. Although the three-time slices in the figure are drawn to be equal in size, in reality, this may not be the case. The time to calculate a verifiable hash is generally very short, and the time

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

to propose a new block and reach consensus is usually longer depending on the size of the blockchain network, transmission speed, and network latency.

Assume several users are participating in this blockchain network, and User A, User B, and User C are elected as committee members during Terms 1 and 2. During Term 1, they calculate their verifiable hashes based on their secret keys and get vhash A, vhash B, and vhash C, respectively. Assume that vhash C is the minimum among the three hashes. So after proposing the new blocks, User A and User B verify the legitimacy of transactions in the new block comes from User C and choose to accept it. Then, send back a "pick" message to User C. The "pick" message includes the vhash of User A and User B. After collecting hashes from a considerable number (which will be explained later) of the committee members, User C then broadcasts the new block to the blockchain network. This new block will be accepted and stored by other users. During Term 2, User B gets a minimal verifiable hash (vhash E), so after proposing the new block, User A and User C verify the new block and send a "pick" message to User B. Finally, User B broadcasts his new block to the blockchain network. There is an overlap between the calculation of verifiable hashes and broadcast of the new block, but for users, these two actions are carried out simultaneously in step (1), so they are not drawn in detail in Fig. 2

During step (3) in Fig. 2, there is a verifiable hash collection task for every user. The requirement of the proportion of hashes collected by the winner is defined as ζ. Assuming that the number of committee members in Term n is τ^n, the number of hashes collected by user u is $\sum \text{vhash}_u^n$, so that

$$\zeta_u^n = \frac{\sum \text{vhash}_u^n}{\tau^n}. \tag{3}$$

It is clear that user u cannot prove himself as a winner during the auction in term n unless $\zeta_u^n > 0.5$. Because if and only if receiving the majority of "pick" messages from committee members, user u knows the vhash of him in this term is the smallest among all committee members. However, ζ cannot be simply set to 1.0 since there are network delays in a P2P network. The specific ζ value setting strategy for different sizes of blockchain networks needs further experimentation and research. Due to page limitation, this question is left as future work.

D. Exception Handle

An appropriate setting of η can only solve part of the network delay problem. Other abnormal conditions will also be encountered when the blockchain network is running. For example, a user could encounter an "out of service" condition at any time.

As shown in Fig. 3, assume that user C, who calculated the smallest vhash during Term 1, is out of service in step (3). Following that condition, user A and B try to send him a "pick" message but get no reply from user C. After the end of Term 1, nobody receives a new block. The result is that user A and user B will calculate their verifiable hashes (vhash D and vhash E) for Term 2. In this way, the loss of user C does not affect the other two users at all. They continue to work as normal and finally reach a consensus at step (3) in Term 2. The transactions that occurred in step (1) of Term 1 are not lost since all of them will be repackaged in step (1) of Term 2. A proper setting of the term period for CMA, which is defined as τ, is also important. If the value of τ is too large (or too small), it will lead to a decrease in transaction efficiency. For example, if τ is too large, the system's response speed to "out of service" conditions will decrease. On the other hand, if τ is too small, each user needs to calculate more verifiable hashes and broadcast them in a higher frequency, which, in turn, brings a higher burden to the blockchain network. Due to space limitations, this part of the research is left as future work.

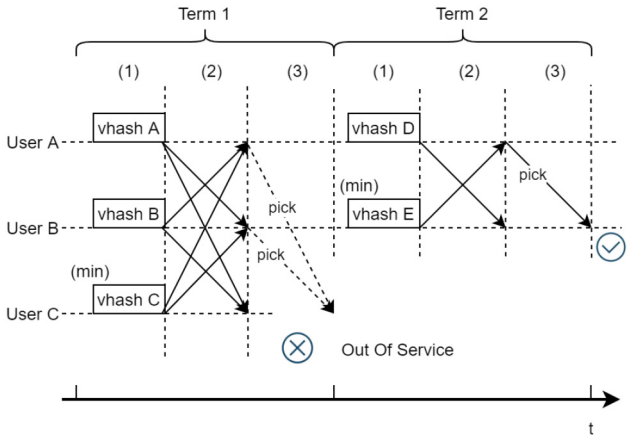


Fig. 3. During step (3) in Term 1, user C is out of service. Neither user A nor user B can send him a "pick" message successfully. However, the offline user C will not hinder the subsequent Term 2.

In the case of a poor network environment, a large-scale auction within a certain time limit may not be supported. In order to address the above challenges, a *hash carry* operation is introduced. Assuming user u_b holds a verifiable hash vhash_{ub}^n during Term n, and u_b receives an auction message m_c from user u_c, carrying a verifiable hash vhash_{uc}^n. As long as $\text{vhash}_{uc}^n > \text{vhash}_{ub}^n$ and u_b knows there is a user u_a whose verifiable hash vhash_{ua}^n is less than vhash_{ub}^n, u_b must forward m_c to u_a. Regarding how much additional load "hash carry" will bring to blockchain networks is left for future work to investigate.

Additionally, malicious users can disrupt the auction. For example, u_c can repeatedly send auction messages to u_a and u_b attempting to disrupt the generation of blocks during the Term n. However, with the hash carry option, u_b will not incorrectly count the auction messages send from u_c, and u_a will normally count the number of verifiable hashes that are less than vhash_{ua}^n. So that such attacks will not bring any other side effects to blockchain networks, except network transmission load. In addition, this kind of attack can be prevented by limiting the frequency of requests from the same IP.

Moreover, the security foundation of the proposed model is asymmetric encryption and verifiable hash. So a new user must broadcast the public key to the blockchain network before joining. Based on that unique public key, the verifiable hashes sent from him can be verified by everyone.

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

IV. Security Analysis

Since the security of the Chameleon hash scheme has been proved, including collision-resistant, message hiding, semantic security, and key-exposure-free [17], the security of the Chameleon hash can be guaranteed. While for the distribution of the trapdoor key, the correctness and security of the secret-sharing scheme have also been proved based on Lagrange's interpolation theorem [18]. Therefore, for the rest of this section, the security of the proposed model is analyzed from two aspects: 1) double-spend attack/long-range attack resistance and 2) eclipse attack resistance.

A. Double-Spend Attack and Long-Range Attack Resistance

Both double-spend attacks and long-range attacks are caused by uncertainty about newly added blocks and the subsequent blocks. However, with the novel reverse pointer design, the subsequent direction of any block can be determined, i.e., starting from the genesis block, the entire chain is undisputed. Long-range attacks are completely ineffective against the proposed model. The only possible stage of the proposed model getting attacked by double spending is when generating the reverse pointers.

As Nakamoto analyzed in [9], the double-spend attack could be treated as Gambler's ruin problem. The probability the attacker could catch up to the honest miners (denoted as D) can be calculated as

$$D(q,z) = 1 - \sum_{k=0}^{z} \frac{\lambda^k e^{-\lambda}}{k!}\left(1 - \left(\frac{q}{p}\right)^{z-k}\right) \tag{4}$$

where z is the number of blocks that the merchant will wait for before handing over physical goods. p is the probability an honest node finds the next block. q is the probability the attacker finds the next block. λ is the blocks producing rate of the attacker during the interval that honest miners produce z blocks, which is calculated by

$$\lambda = z\frac{q}{p}. \tag{5}$$

Based on (5), to find out the probability that the attacker could overtake the honest miners (which means that the double-spend attack happens), z is replaced with $z+1$

$$D(q,z) = 1 - \sum_{k=0}^{z+1} \frac{\lambda^k e^{-\lambda}}{k!}\left(1 - \left(\frac{q}{p}\right)^{z+1-k}\right). \tag{6}$$

For PoW, q is the proportion of computing resources owned by the attacker. For PoS, q is defined as the proportion of stakes owned by the attacker. In CMA, q is defined as the probability that all committee members are controlled by the attacker for each Term. To identify this probability, the number of committee nodes in a Term is defined as τ^n, and the number of nodes controlled by the attacker is defined as α. When $\alpha \geq \tau^n$, all of the committee members in this Term may be controlled by the attacker, and there is a probability that the attacker controls the generation of this term's block. At this time, double-spend attacks may occur and q can be calculated as

$$q = \frac{C_{\eta-\tau^n}^{\alpha-\tau^n}}{C_{\eta}^{\alpha}}. \tag{7}$$

Otherwise, if $\alpha < \tau^n$, several committee members are not controlled by the attacker (named as honest committee members). Honest committee members do not provide their part of the trapdoor keys to the smart contract if they disagree with the newly generated block, and double-spend attacks cannot occur. At this time, q is 0, which in turn leads to $D(q,z) = 0$. In Section V, the *Monte Carlo* method is adopted to verify the performance of CMA that is resistant to the double-spend attack compared with PoW and PoS.

B. Eclipse Attack Resistance

As mentioned in Section I, eclipse attacks will cause deny-of-service of the blockchain. DDoS attacks can be classified into two categories [31]: 1) network/transport-level DDoS flooding attacks and 2) application-level DDoS flooding attacks. On the network/transport level, it is difficult for an adversary to predict committee members of the next term and launch eclipse attacks. In fact, it is ineffective to launch attacks against participant members other than committee members since it will not hinder the consensus process. On the application level, if a transaction request consumes too many resources, the committee member who submits the request will stall. However, this does not affect other committee members in proposing their own transactions and continuing to reach consensus. For example, if user u_a (whose verifiable hash vhash_{ua}) stalls due to the calculation of transaction tx, the user u_b (whose vhash_{ub} is greater than vhash_{ua} but less than others) will continue to propose his transaction block. Eventually, committee members will send auction messages to u_b and reach a consensus on the block proposed by u_b.

In information entropy, the average information per emitted symbol is denoted with $H(X)$

$$H(X) = -\sum_{i=1}^{n} P(x_i) log_b P(x_i) \tag{8}$$

where $P(x_i)$ is the probability mass function, and b is the base of the logarithm used. In this article, $H(X)$ is used to measure the entropy of the blockchain system. Higher entropy means better performance in terms of security.

In order to facilitate a comparison, $P(x_i)$ is defined as the probability of user x_i participating in the consensus algorithm. In the CMA consensus algorithm, $P(x_i)$ is the probability that user i is selected as a member of the committee in each term. The calculation of the entropy of CMA refers to (8).

For PoW, miners act as consensus maintainers. In each term of transactions, the fastest miner will verify the transactions, generate a new block, and broadcast it to everyone. Therefore, P is defined as the proportion of this miner to all miners in the blockchain network. Assume the number of miners is N_m, and the entropy in this article is measured in bits so that the

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

TABLE II
COMPARISONS OF SECURITY PROPERTIES

Properties	PoW	PoS	CMA
Double-spend attack resistance	Low	Medium	High
Long-range attack resistance	-	Low	High
Eclipse attack resistance	Low	Medium	High

entropy can be calculated as

$$H(X) = -N_m \cdot \frac{1}{N_m} \cdot \log_2\left(\frac{1}{N_m}\right) = -\log_2\left(\frac{1}{N_m}\right). \quad (9)$$

As for PoS, users with more stake will have a higher probability to participate in the consensus. So $P(x_i)$ is defined as the proportion of stakes owned by user i. In Section V, the parameter setting of experiments and the entropy calculated for different consensus algorithms are introduced.

All in all, from security analysis in Section IV-A, it is obviously that CMA has a lower q than PoW or PoS and shows that CMA has a higher double spend/long-range attack resistance. From the analysis in Section IV-B, the probability of user in CMA participating in the consensus process is higher than PoW or PoS, which shows that CMA has a higher entropy and a higher eclipse attack resistance. The results of the analysis are summarized in Table II. In Section V, experiments are conducted to verify our analysis.

C. Further Discussion

1) Defense Potential: A Sybil attack is defined as an attack where an adversary creates numerous fake identities to reduce throughput, or even gain control of a blockchain network [32]. Since the CMA consensus algorithm selects committee members based on the public and private key pairs (identities) of each participant, a Sybil attack may reduce the security of blockchain [33]. The current solutions to defend against Sybil attacks can be summarized as follows: trusted certification, resource testing, recurring costs and fees, and trusted devices [34]. To mitigate the impact of Sybil attacks, some periodic resource tests (similar to the computing power test in PoW) are needed. The question then becomes, how to set resource test rules to effectively resist Sybil attacks without affecting the scalability of blockchain? This is a challenging research topic but one that is beyond the scope of this article. When the abnormal nodes are detected, the way to treat the abnormal node is similar to the exception handle mentioned in Section III.

All in all, if the proposed model is adopted in a consortium or private environment (where nodes trust each other), Sybil attacks are not relevant. However, in a public blockchain environment, it is necessary to set up periodic resource tests to resist Sybil attacks.

2) Attacks Limitation: There are other attacks against blockchains that present security risks, such as selfish mining [35], bribery attack [36], and block withholding attacks [37]–[39]. In this article, only the resistance of three mainstream attacking methods, including double-spend attacks, long-range attacks, and eclipse attacks, is considered, which to a degree proves the advantages of the proposed model from the security aspect compared to other blockchain models. To some extent, the experiment results also demonstrate that the proposed model can resist 51% attacks (which is defined as the majority of the network's computing resources are held by the attackers so that they can manipulate the blockchain [40]). The resistance to all other attack types cannot be analyzed and verified completely in this article due to space limitation, which could be left as future work.

TABLE III
SIMULATION PARAMETER SETTINGS

Parameter	Value
The number of participant members (η)	2000
The incoming transaction speed (tx/s)	50
Total incoming transaction number	1500
The number of double-spend attack targets (α)	1000
The number of committee members (τ)	1000
The number of DDoS attack targets	200
The number of transactions in a block	20
The size of a block (KB)	5
The speed of the network (KB/s)	200
The average delay of the network (ms)	100
The standard deviation of the delay of the network	100
The propagation rate of the network	2
The difficulty of PoW	1.5×10^{-5}
The standard deviation of the stake distribution in PoS	15
The consensus period of CMA (s)	2
The number of blocks the merchant wait	3
The number of transaction terms in CMA	100
The proportion of miners in PoW	0.07%

V. PERFORMANCE EVALUATION

In this section, simulation experiments are conducted to evaluate the security and scalability of the proposed model, which verifies the aforementioned analysis.

A. Environment and Parameters Setting

Our simulation environment is based on the Ubuntu 18.04 Operating System. Hardware configuration includes an Intel Core i7-8650U 4 Cores processor and 16-GB RAM. In terms of software configuration, python 3.6 is used to simulate PoW, PoS, and CMA consensus algorithm-based blockchain networks. The default parameters setting of the simulation experiment is shown in Table III. In the remainder of this section, the parameters not mentioned are set according to this table.

In experiments, assume that the latency of blockchain networks obey a normal distribution, whose standard deviation is 100. In order to simulate the worst network environment, the propagation rate is assumed to be 2, which means that a message sent by a node will only be accepted by the other two nodes. Considering the computing power of the hardware, the difficulty of PoW is adjusted to 1.5×10^{-5}.

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

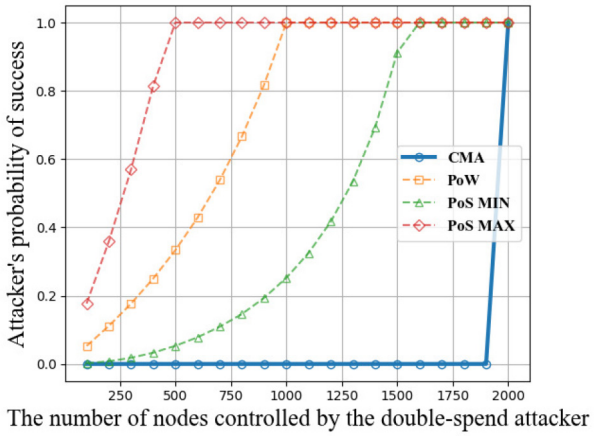


Fig. 4. When the number of nodes controlled by the double-spend attacker increases, the attacker's probability of success under CMA maintains a very low level compared with PoS and PoW.

In addition, assume that all nodes have the same compute power. The standard deviation of the stake distribution across nodes is set to 15 by default. In double-spend attack experiments, "PoS MIN" means that α nodes with the smallest stakes launch double-spend attacks in the PoS consensus algorithm. On the contrary, "PoS MAX" means α nodes with the most stakes launch double-spend attacks in the PoS consensus algorithm. The consensus period of CMA is set to 2 s, which means that all incoming transactions within 2 s will be packaged in a block and broadcast throughout the blockchain network.

B. Double-Spend Attack Resistance

According to the previous analysis, experiments are conducted to analyze how the attacker's double-spend success probability (D) changes as the number of nodes controlled by the attacker α changes. As shown in Fig. 4. The number of nodes in the blockchain network is 2000. After 200 of the nodes are controlled by the attacker (this means, 10% of the nodes are controlled), for PoS MAX, the attacker's probability of success increases dramatically. For PoW, when more than 500 of the nodes are controlled, the attacker's probability of success is more than 30%, which is unbearable. This means PoW can defend against 25% or fewer nodes being hacked under double-spend attacks when the merchant waits for three blocks to confirm the transaction. As for PoS MIN, after 1000 nodes (half of the nodes in the blockchain network) with the fewest stakes are controlled by the attacker, the attacker's probability of success has increased to more than 30%. Compared with PoS MAX, PoS MIN is more likely to happen in reality because nodes that hold more stakes are less likely to do evil (such as launch double-spend attacks). However, PoS MIN is still vulnerable to double-spend attacks (the attacker's probability of success increases to more than 20%) when half of the nodes are controlled by the attacker, which is 1000 in the experiment. In this circumstance, CMA can still guarantee that the attacker's probability of success remains at around 0, which shows that CMA can resist double-spend attacks more effectively than PoS MIN. Finally, when the proportion of nodes under attacker's control reaches 100%, which is 2000 nodes in the experiment, all three consensus algorithms cannot handle the situation that all of the nodes launch double-spend attacks so that the attacker's probability of success increases to 100%.

Not only that the well-known 51% attack can also be resisted by CMA. Fig. 4 shows that after majority of the nodes (which is 1000 in the experiments) are controlled by the attacker, only CMA can keep the double-spend attacker's probability of success at 0%. That means, even though the majority of the network's computing resources are held by the double-spend attackers, they cannot manipulate the CMA-based blockchain network as well.

In order to determine the relationship between the scale of blockchain and the performance of double-spend attack resistance, experiments are conducted. The results are shown in Fig. 5. For Fig. 5(a), the number of nodes controlled by the attacker is set to 10% of the number of participants ($\alpha = 0.1 \times \eta$). In this situation, PoS MAX has a higher attacker's probability of success than the others. The reason for its constant fluctuation is that the distribution of stakes in PoS is randomly generated for different sizes of blockchain networks. PoS MAX selects the nodes who own the largest stakes, so the randomness of the stakes held by the selected nodes leads to fluctuations in the attacker's probability of success. For Fig. 5(b), as the number of nodes controlled by the attacker increases to 25% of participants, most of the stakes are controlled by the attackers so that a more stable attacker's probability of success is reached compared with Fig. 5(a). What is more, the attacker's probability of success in PoW has a slight increase. When α increases to half of the number of participants in the blockchain network, as shown in Fig. 5(c), the attacker's probability of success in PoW and PoS MAX reaches and stabilizes at 100%. After a small fluctuation, the attacker's probability of success in PoS MIN stabilizes at around 5%. The slight fluctuation is also caused by the random distribution of stakes under different blockchain network scales. However, under any circumstances, the attacker's probability of success in CMA is kept at 0%, which proves that CMA is very effective in defencing double-spend attacks and improving security.

In order to figure out the relationship between the number of committee members and double-spend attack resistance, the experiments are conducted and the results are shown in Fig. 6. As mentioned before, the number of nodes participating in the blockchain network is set to 2000 by default. When the number of committee members grows (from 1 to 7), the attacker's probability of success drops drastically. As the number of blocks that merchants will wait for before confirming transactions increases (the increase of z), the attacker's probability of success also decreases. More precisely, when the number of committee members is greater than 6 (the proportion of committee members reaches 0.3% or higher), the attacker's probability of success is less than 0.1%, no matter how many blocks the merchant waits.

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

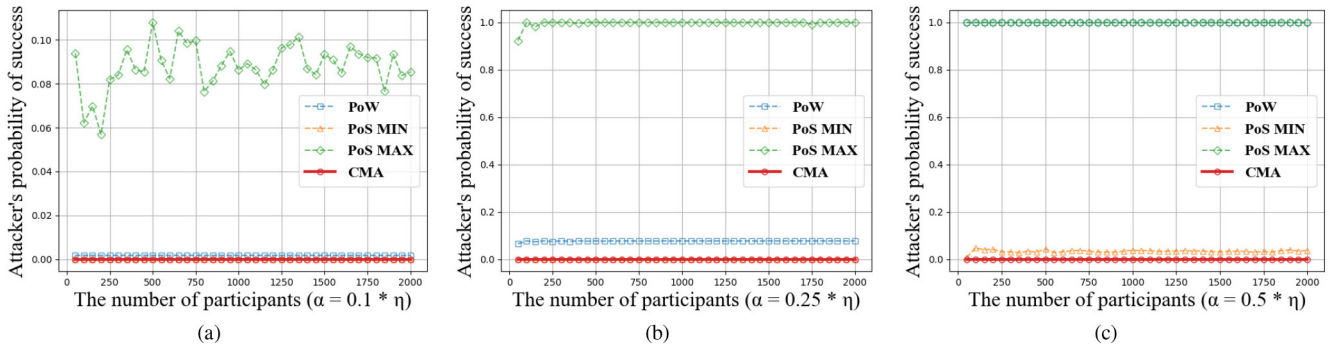


Fig. 5. With the increase of the number of participants and the nodes controlled by the attacker, the attacker's probability of success in CMA is maintained at a very low level compared with PoS and PoW. (a) 10% double-spend attack. (b) 25% double-spend attack. (c) 50% double-spend attack.

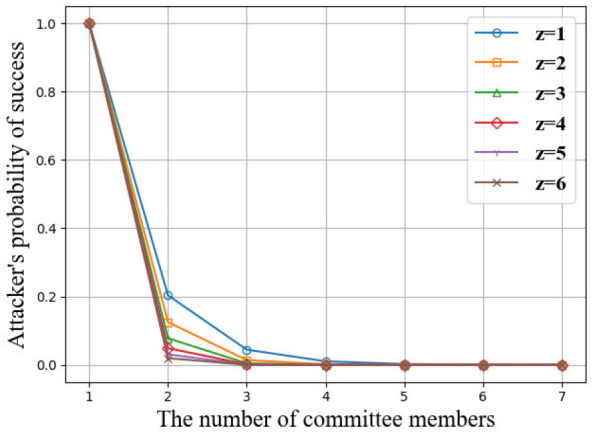


Fig. 6. Attacker's probability of success decreases when the number of committee members increases. Increasing the number of blocks that merchants will wait for before confirming transactions (z) will also help reduce the attacker's probability of success.

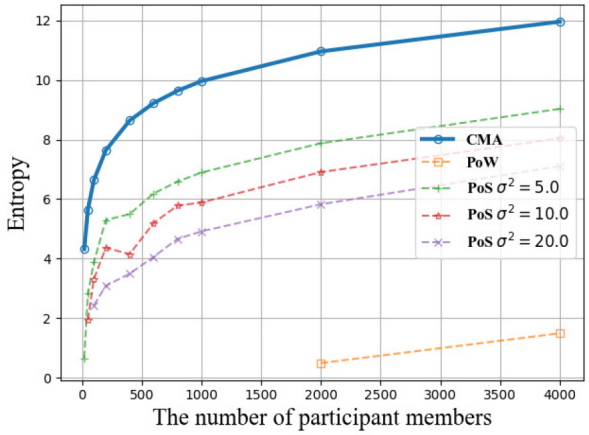


Fig. 7. The entropy comparison of CMA, PoW, and PoS. Regardless of the number of participants, compare with PoW and PoS, CMA has a higher entropy, namely, better security.

C. Eclipse Attack Resistance

As analyzed before, the entropy for CMA, PoW, and PoS needs to be calculated to compare their ability to resist eclipse attacks. To calculate the entropy of CMA, the proportion of committee members is set as 50%. This means that half of the participant members will become committee members in each election term. The number of transaction terms in CMA is set to 100. The experiments show that the changes to this value do not have much impact on the experimental results. To calculate the entropy value of PoW, Bitcoin is treated as an example. The number of miners and wallet active users are set to 10 018 [41] and 14 280 000 [42], respectively. The proportion of miners (these users have the opportunity to participate in transaction verification) in bitcoin is set to 0.07%. Finally, in order to calculate the entropy value of PoS, assume that the stakes held by different users comply with a normal distribution, with a standard deviation of the stakes set from 5.0 to 20.0.

As shown in Fig. 7, the entropy increases as the number of participant members in the blockchain network increases. The number of participant members in the simulation blockchain network starts from 20 to 4000, which covers most of the blockchain network scales in practice. The entropy of CMA increases from 4 to 12, which is higher than PoW (increases from 0.2 to 1.6) and PoS (increases from 1 to 9). For PoS, as the standard deviation of stakes σ^2 decreases, the entropy of PoS increases. This means, the more even the distribution of stakes, the more random the people participating in the PoS consensus, which is consistent with the knowledge in practice. But in fact, the distribution of the stakes in PoS cannot reach a full average. For PoW, the low proportion of miners in bitcoin leads to lower entropy than PoS and CMA. To conclude, the results show that CMA has higher entropy than PoW and PoS, and exhibits a better eclipse attack resistance.

D. Scalability

Security is important but so too the scalability requirements. In this section, simulation experiments are conducted to compare the throughput of CMA with PoS and PoW. For the following experiments, the total time consumption is defined as the time consumption of hash calculation, new block generation, and new block propagation time in the network. In particular, for CMA, the hash calculation time includes time consumption of committee member election, forward pointer construct, reverse pointer construct, and forward pointer repair.

The time consumption of transactions of different consensus algorithms under different numbers of blockchain participants

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

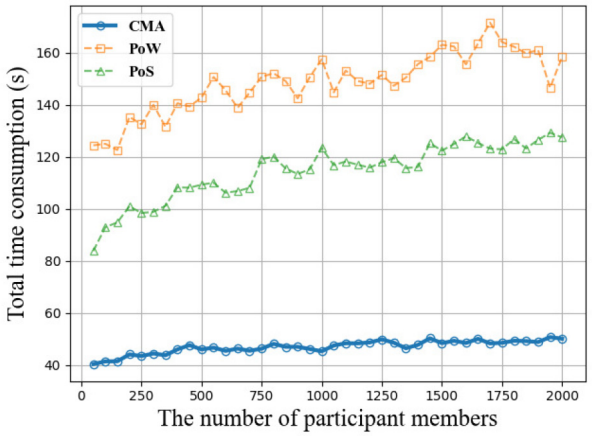


Fig. 8. When the number of participants in the blockchain network increases from 50 to 2000, the total time consumption of CMA is lower than PoW and PoS.

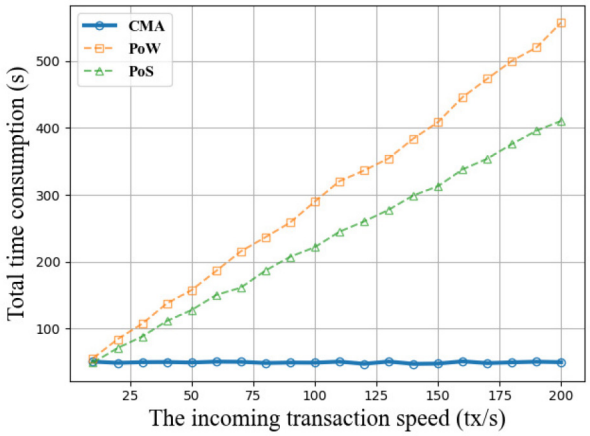


Fig. 9. When the frequency of incoming transactions increases from 10 to 200 tx/s, the total time consumption of CMA is lower than PoW and PoS.

is recorded, as shown in Fig. 8. The incoming transaction speed is set to 50 transactions per second, and the total incoming transaction number is set to 1500, as mentioned in Table III. With an increase of participants from 50 to 2000, the time consumption of PoW increases from 120 to 170 s. For PoS, time consumption increases from 80 to 130 s. This is because as more nodes participate in a blockchain network, it takes more time for the block to be broadcast and confirmed by all nodes. For CMA, the total time consumption is less than PoW and PoS, increasing from 40 to 50 s. The growth rate of CMA is smaller than that of PoW and PoS when the number of participants increases. This is because in the CMA consensus process, it is not necessary for all nodes, but half of the nodes (committee members), to participate in the consensus, and generate and confirm the newly generated block. When the network scale increases, CMA can effectively mitigate the delay of P2P network propagation. In the case that fewer than 2000 nodes, CMA costs less time to process transactions and has a better throughput than PoW and PoS.

In the case of a constant number of participants (2000 nodes) in the blockchain network, the incoming transaction speed is adjusted (from 10 to 200 transactions per s) and the total time consumption of CMA, PoW, and PoS is recorded. As shown in Fig. 9, when the transaction frequency is ten transactions per second, the total time consumption of CMA has no obvious advantages over PoW or PoS. This may be because no matter how many transactions arrive in a second, CMA needs to wait for a period of 2 s before packaging transactions into a new block. However, PoS and PoW are not subject to this restriction. With the increase of incoming transaction speed, the advantage of using CMA to performance increases. As the incoming transaction speed is increased to 200 transactions per second, the total time consumption of PoW and PoS reaches 560 and 410 s, respectively. However, the time consumption of CMA is maintained at about 50 s. This is due to the fact that in the high transaction frequency case, a block size limit (5 KB) will cause more blocks to be generated within a certain

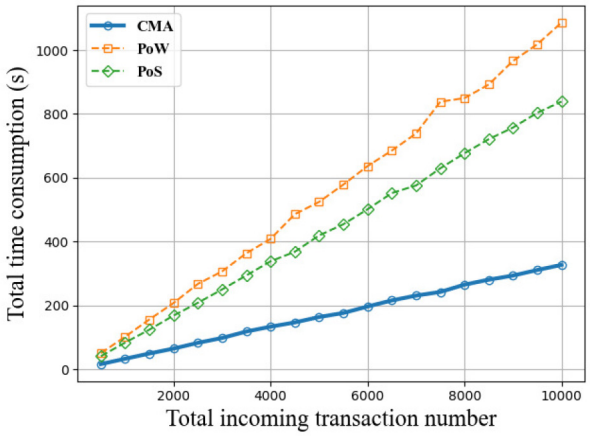


Fig. 10. With the increase of the incoming transaction number, the total time consumption of CMA is always lower than PoW and PoS.

time frame. The network delay increases as there are more blocks that need to be broadcast while consensus.

From another perspective, when the frequency of transactions is set as a constant (50 transactions per second) and only the total incoming transaction number is increased (from 500 to 10 000 transactions), the total time consumption of three consensus algorithms increases as well. As shown in Fig. 10, for CMA, the total time consumption increases linearly with the increase of incoming transaction duration (from 16 to 330 s). For PoW, this number increases from 50 to 1100 s. For PoS, this number increases from 40 to 840 s. It is obvious that CMA consumes the least time to process transactions and has the smallest growth rate, which means that CMA has a higher transaction efficiency than PoW or PoS.

To determine the impact of DDoS attacks on the scalability of CMA, experiments are conducted. The results are shown in Fig. 11. The number of participants (targets) attacked by DDoS is adjusted from 0 to 960 (The total number of participants in the blockchain network is 2000 by default.). From the figure, it is clear that regardless of how many nodes are DDoS attacked, CMA maintains the total transaction time consumption at around 50 s. This means that when less than equal half of the nodes are controlled by the DDoS attacker, the

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

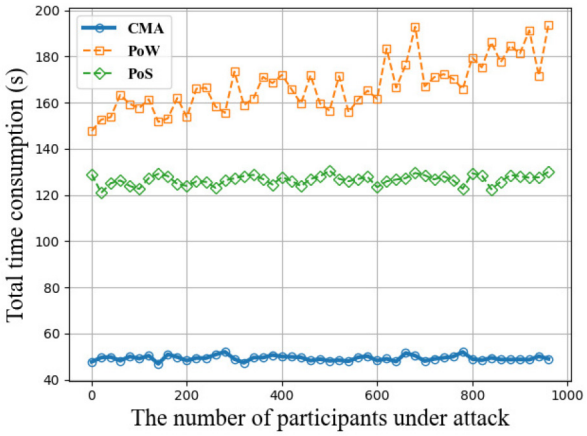


Fig. 11. With the increase of the number of nodes that under the DDoS attack, the total time consumption of CMA is almost unaffected and lower than PoW and PoS.

efficiency of CMA is barely affected. For PoW, the total transaction time consumption increases from 150 to 190 s. For PoS, the total time consumption increases slightly from 120 to 130. The cause of the fluctuations is the randomness of stakes held by the nodes under DDoS attacks. From this point of view, the transaction time consumption of PoS is more dependent on the stakes held by the DDoS attacked nodes, rather than the number of nodes attacked by the DDoS. In general, compared with CMA, PoW and PoS are both affected by DDoS attacks to varying degrees, resulting in an increase in total time consumption. In addition, under the same level of DDoS attacks, CMA has a smaller transaction time consumption compared with PoW and PoS.

All in all, experiments were conducted from different aspects. The results show that the proposed BLB can resist attacks better than PoW and PoS (especially for double-spend attack/long-range attack and eclipse attack). Not only that the time cost of CMA to process transactions is always the least compared with PoW and PoS, which means that CMA has a higher throughput than PoW and PoS in multiple situations, even under eclipse attacks.

VI. Summary and Future Works

In this article, a lightweight and attack-proof BLB with a custom-built CMA consensus algorithm is proposed for IoT payment systems. To eliminate double-spend attacks, long-range attacks, and eclipse attacks while ensuring scalability, bidirectional links between blocks in the blockchain are constructed based on the Chameleon-hash function, whose trapdoor keys are split through distributed smart contracts and hold by committee members. The scalability and security of the committee members are ensured by the VRF. What is more, the exceptions during consensus are also identified and handled. Improved Nakamoto's double-spend attack analysis and early efforts to introduce the concept of entropy in information theory as a measurement of the eclipse attack resistance are carried out correspondingly. Finally, experiments are conducted to testify that the security and scalability of the proposed paradigm are better than those based PoW and PoS. Future work is in progress to consider the probability of cross-chain based on BLB with PoW or PoS-based blockchains to help improve their scalability or security, and a reasonable resource test rule to mitigate the impact of Sybil attacks.

References

[1] Y. Qu, S. R. Pokhrel, S. Garg, L. Gao, and Y. Xiang, "A blockchained federated learning framework for cognitive computing in industry 4.0 networks," *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2964–2973, Apr. 2021.

[2] Y. Qu *et al.*, "Decentralized privacy using blockchain-enabled federated learning in fog computing," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5171–5183, Jun. 2020.

[3] F. Chen, Z. Xiao, L. Cui, Q. Lin, J. Li, and S. Yu, "Blockchain for Internet of Things applications: A review and open issues," *J. Netw. Comput. Appl.*, vol. 172, Dec. 2020, Art. no. 102839.

[4] B. L. Nguyen *et al.*, "Privacy preserving blockchain technique to achieve secure and reliable sharing of IoT data," *Comput. Mater. Continua*, vol. 65, no. 1, pp. 87–107, 2020.

[5] C. Li, G. Xu, Y. Chen, H. Ahmad, and J. Li, "A new anti-quantum proxy blind signature for blockchain-enabled Internet of Things," *Comput. Mater. Continua*, vol. 61, no. 2, pp. 711–726, 2019.

[6] B. Bordel, R. Alcarria, D. Martin, and A. Sanchez-Picot, "Trust provision in the Internet of Things using transversal blockchain networks," *Intell. Autom. Soft Comput.*, vol. 25, no. 1, pp. 155–170, 2019.

[7] L. Gao, T. H. Luan, B. Gu, Y. Qu, and Y. Xiang, "Blockchain based decentralized privacy preserving in edge computing," in *Privacy-Preserving in Edge Computing*. Singapore: Springer, 2021, pp. 83–109.

[8] D. J. Moroz, D. J. Aronoff, N. Narula, and D. C. Parkes, "Double-spend counterattacks: Threat of retaliation in proof-of-work systems," 2020. [Online]. Available: arXiv:2002.10736.

[9] S. Nakamoto. *Bitcoin: A Peer-to-Peer Electronic Cash System*. Accessed: 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[10] D. Liao, H. Li, W. Wang, X. Wang, M. Zhang, and X. Chen, "Achieving IoT data security based blockchain," *Peer Peer Netw. Appl.*, to be published.

[11] W. Li, S. Andreina, J.-M. Bohli, and G. Karame, "Securing proof-of-stake blockchain protocols," in *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Cham, Switzerland: Springer, 2017, pp. 297–315.

[12] G. Xu *et al.*, "Am I eclipsed? A smart detector of eclipse attacks for ethereum," *Comput. Security*, vol. 88, Jan. 2020, Art. no. 101604.

[13] X. Tang, Q. Zheng, J. Cheng, V. S. Sheng, R. Cao, and M. Chen, "A DDoS attack situation assessment method via optimized cloud model based on influence function," *Comput. Mater. Continua*, vol. 60, no. 3, pp. 1263–1281, 2019.

[14] Y. Liu, Y. Qu, C. Xu, Z. Hao, and B. Gu, "Blockchain-enabled asynchronous federated learning in edge computing," *Sensors*, vol. 21, no. 10, p. 3335, 2021.

[15] C. Pérez-Solà, S. Delgado-Segura, G. Navarro-Arribas, and J. Herrera-Joancomartí, "Double-spending prevention for bitcoin zero-confirmation transactions," *Int. J. Inf. Security*, vol. 18, no. 4, pp. 451–463, 2019.

[16] I. A. I. AlMallohi, A. S. M. Alotaibi, R. Alghafees, F. Azam, and Z. S. Khan, "Multivariable based checkpoints to mitigate the long range attack in proof-of-stake based blockchains," in *Proc. 3rd Int. Conf. High Perform. Compilation Comput. Commun.*, 2019, pp. 118–122.

[17] M. Khalili, M. Dakhilalian, and W. Susilo, "Efficient chameleon hash functions in the enhanced collision resistant model," *Inf. Sci.*, vol. 510, pp. 155–164, Feb. 2020.

[18] A. Beimel, "Secret-sharing schemes: A survey," in *Proc. Int. Conf. Coding Cryptol.*, 2011, pp. 11–46.

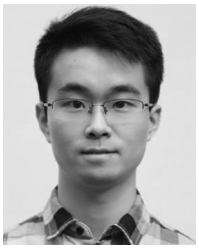
[19] N. Bitansky, "Verifiable random functions from non-interactive witness-indistinguishable proofs," *J. Cryptol.*, to be published.

[20] G. Ateniese, B. Magri, D. Venturi, and E. Andrade, "Redactable blockchain–or–rewriting history in bitcoin and friends," in *Proc. IEEE Eur. Symp. Security Privacy (EuroS&P)*, 2017, pp. 111–126.

[21] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich, "Algorand: Scaling byzantine agreements for cryptocurrencies," in *Proc. 26th Symp. Oper. Syst. Principles*, 2017, pp. 51–68.

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

[22] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, 2016, pp. 3–16.
[23] G. Ramezan and C. S. Leung, "An analysis of proof-of-work based blockchains under an adaptive double-spend attack," *IEEE Trans. Ind. Informat.*, vol. 16, no. 11, pp. 7035–7045, Nov. 2020.
[24] N. Anita and M. Vijayalakshmi, "Blockchain security attack: A brief survey," in *Proc. IEEE 10th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT)*, 2019, pp. 1–6.
[25] E. Heilman, A. Kendler, A. Zohar, and S. Goldberg, "Eclipse attacks on bitcoin's peer-to-peer network," in *Proc. 24th USENIX Security Symp. (USENIX Security)*, 2015, pp. 129–144.
[26] B. Alangot, D. Reijsbergen, S. Venugopalan, and P. Szalachowski, "Decentralized lightweight detection of eclipse attacks on bitcoin clients," 2020. [Online]. Available: arXiv:2007.02287.
[27] S. Goldberg, L. Reyzin, D. Papadopoulos, J. Vcelák, "Verifiable random functions (VRFs)," Fremont, CA, USA, draft-irtf-cfrg-vrf-05, 2019.
[28] R. Dunford, Q. Su, and E. Tamang, *The Pareto Principle*, Plymouth, U.K.: Publishamerica, 2014.
[29] M. H. Dehkordi and S. Mashhadi, "An efficient threshold verifiable multi-secret sharing," *Comput. Stand. Interfaces*, vol. 30, no. 3, pp. 187–190, 2008.
[30] T. Nguyen-Van *et al.*, "Scalable distributed random number generation based on homomorphic encryption," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, 2019, pp. 572–579.
[31] Q. Yan, F. R. Yu, Q. Gong, and J. Li, "Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 602–622, 1st Quart., 2015.
[32] P. Otte, M. de Vos, and J. Pouwelse, "TrustChain: A sybil-resistant scalable blockchain," *Future Gener. Comput. Syst.*, vol. 107, pp. 770–780, Jun. 2020.
[33] C. Huang *et al.*, "RepChain: A reputation-based secure, fast, and high incentive blockchain system via sharding," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4291–4304, Mar. 2020.
[34] B. N. Levine, C. Shields, and N. B. Margolin, *A Survey of Solutions to the Sybil Attack*, Univ. Massachusetts Amherst, Amherst, MA, USA, 2006.
[35] R. Yang, X. Chang, J. Mišić, and V. B. Mišić, "Assessing blockchain selfish mining in an imperfect network: Honest and selfish miner views," *Comput. Security*, vol. 97, Oct. 2020, Art. no. 101956.
[36] H. Sun, N. Ruan, and C. Su, "How to model the bribery attack: A practical quantification method in blockchain," in *Proc. Eur. Symp. Res. Comput. Security*, 2020, pp. 569–589.
[37] A. Kaci and A. Rachedi, "Toward a machine learning and software defined network approaches to manage miners' reputation in blockchain," *J. Netw. Syst. Manag.*, vol. 28, no. 3, pp. 478–501, 2020.
[38] A. Kaci and A. Rachedi, "PoolCoin: Toward a distributed trust model for miners' reputation management in blockchain," in *Proc. IEEE 17th Annu. Consum. Commun. Netw. Conf. (CCNC)*, 2020, pp. 1–6.
[39] C. Tang, L. Wu, G. Wen, and Z. Zheng, "Incentivizing honest mining in blockchain networks: A reputation approach," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 67, no. 1, pp. 117–121, Jan. 2020.
[40] J. Yu, D. Kozhaya, J. Decouchant, and P. Esteves-Verissimo, "RepuCoin: Your reputation is your power," *IEEE Trans. Comput.*, vol. 68, no. 8, pp. 1225–1237, Aug. 2019.
[41] A. Yeow. (2018). *Bitnodes*. [Online]. Available: https://bitnodes.earn.com
[42] A. Lielacher, "How many people use bitcoin in 2019," *Bitcoin Market J.*, vol. 643, p. 32, May 2019.



Chenhao Xu received the B.S. degree in software engineering from Beijing Institute of Technology, Beijing, China, in 2018. He is currently pursuing the Ph.D. degree with the School of Information Technology, Deakin University, Geelong, VIC, Australia.

His research interests include blockchain, federated learning, and IoT.

Youyang Qu (Member, IEEE) received the B.S. degree in mechanical automation and the M.S. degree in software engineering from Beijing Institute of Technology, Beijing, China, in 2012 and 2015, respectively, and the Ph.D. degree from the School of Information Technology, Deakin University, Geelong, VIC, Australia, in 2019.

He is currently a Research Fellow of Deakin Blockchain Innovation Lab. His research interests focus on dealing with security and customizable privacy issues in social networks, machine learning, IoT, and big data.

Dr. Qu is active in communication society. He is also the Publicity Chair of SPDE2020. He has served for a TPC Member for IEEE flagship conferences, including IEEE ICC and IEEE Globecom.

Tom H. Luan (Senior Member, IEEE) received the B.Eng. degree from Jiao Tong University, Xi'an, China, in 2004, the M.Phil. degree from The Hong Kong University of Science and Technology, Hong Kong, in 2007, and the Ph.D. degree from the University of Waterloo, Waterloo, ON, Canada, in 2012.

He is currently a Professor with the School of Cyber Engineering, Xidian University, Xi'an. He has authored/coauthored more than 40 journal papers and 30 technical papers in conference proceedings, and he has received one U.S. patent. His research mainly focuses on content distribution and media streaming in vehicular *ad hoc* networks and peer-to-peer networking, and the protocol design and performance evaluation of wireless cloud computing and edge computing.

Peter W. Eklund received the Honours degree (First Class) in mathematics from the University of Wollongong, Wollongong NSW, Australia, in 1985, the M.Phil. degree from Brighton University, Brighton, U.K., in 1988, and the Ph.D. degree in artificial intelligence from Linköping University, Linköping, Sweden, in 1992.

He is a Professor of AI and Machine Learning with the School of Information Technology, Deakin University, Geelong, VIC, Australia. For many years, he was supported by defence intelligence sources both in Australia and the USA. His current work is on the scalability of blockchain technology, and its applications to future logistics. This followed from his work on "embedding knowledge in Web documents," pioneering and influential semantic Web research. Since then he has been developing an international profile in applied artificial intelligence. Following a large grant from CSIRO's ICT Centre in 2010, he diversified into pervasive computing and intelligent transport systems, including applications of cyber–physical systems in supply chain logistics.

Prof. Eklund won the Inaugural Australian Smart Infrastructure Research Award from the Federal Department of Infrastructure, Transport, Regional Development and Local Government in 2010. He has been a Co-Founder of three tech start-ups and he is current on the advisory board of GenuTex, a company that offers a unique hybrid blockchain solution to authenticate the supply-chain of pharmaceuticals and a Copenhagen-based fintech company called ZTLment, who enable cross-border trade between small- and medium-sized enterprises via programmable money. He is an elected fellow of The Australian Computer Society.

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.

Yong Xiang (Senior Member, IEEE) received the B.E. and M.E. degrees from the University of Electronic Science and Technology of China, Chengdu, China, in 1983 and 1989, respectively, and the Ph.D. degree from the University of Melbourne, Parkville, VIC, Australia, in 2003.

He is a Professor with the School of Information Technology, Deakin University, Geelong, VIC, Australia, where he is also an Associate Head of School (Research) and the Director of the Artificial Intelligence and Data Analytics Research Cluster. He has obtained a number of research grants (including several ARC Discovery and Linkage grants from the Australian Research Council) and published numerous research papers in high-quality international journals and conferences. He is the Co-Inventor of two U.S. patents and some of his research results have been commercialized.

Prof. Xiang is the editor/guest editor of several international journals. He has been invited to give keynote speeches and chair committees in a number of international conferences, review papers for many international journals and conferences, serve on conference program committees, and chair technical sessions in conferences.

Longxiang Gao (Senior Member, IEEE) received the Ph.D. degree in computer science from Deakin University, Geelong, VIC, Australia, in 2014.

He is currently a Senior Lecturer with the School of Information Technology, Deakin University. Before joining Deakin University, he was a Postdoctoral Research Fellow of IBM Research and Development Australia. He has over 70 publications, including patents, monographs, book chapters, and journal and conference papers. Some of his publications have been published in the top venues, such as IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE INTERNET OF THINGS, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. His research interests include data processing, mobile social networks, fog computing, and network security.

Dr. Gao received the 2012 Chinese Government Award for Outstanding Students Abroad (Ranked No. 1 in Victoria and Tasmania consular districts). He is active in IEEE Communication Society. He has served for the TPC co-chair, a publicity co-chair, a organization chair, and the TPC member for many international conferences.

Authorized licensed use limited to: ULAKBIM UASL - CUKUROVA UNIVERSITESI. Downloaded on June 13,2022 at 03:39:23 UTC from IEEE Xplore. Restrictions apply.