


**Please cite the Published Version**

Chen, Jianing, Wu, Jun, Liang, Haoran, Mumtaz, Shahid, Li, Jianhua, Konstantin, Kostromitin, Bashir, Ali Kashif and Nawaz, Raheel  (2020) Collaborative Trust Blockchain Based Unbiased Control Transfer Mechanism for Industrial Automation. IEEE Transactions on Industry Applications, 56 (4). pp. 4478-4488. ISSN 0093-9994

**DOI:** <https://doi.org/10.1109/tia.2019.2959550>

**Publisher:** Institute of Electrical and Electronics Engineers (IEEE)

**Version:** Accepted Version

**Downloaded from:** <https://e-space.mmu.ac.uk/624665/>

**Usage rights:**  In Copyright

**Additional Information:** This is an Author Accepted Manuscript of a paper accepted for publication in IEEE Transactions on Industry Applications published by and copyright IEEE.

**Enquiries:**

If you have questions about this document, contact [openresearch@mmu.ac.uk](mailto:openresearch@mmu.ac.uk). Please include the URL of the record in e-space. If you believe that your, or a third party's rights have been compromised through this document please see our Take Down policy (available from <https://www.mmu.ac.uk/library/using-the-library/policies-and-guidelines>)

# Collaborative Trust Blockchain Based Unbiased Control Transfer Mechanism for Industrial Automation

Jianing Chen, Jun W, Haoran Liang, Shahid Mumtaz, Jianhua Li, Kostromitin Konstantin, Ali Kashif Bashir, *Senior Member, IEEE*, and Raheel Nawaz

## I. INTRODUCTION

**Abstract**—In industrial automation, numerous devices are interconnected in smart factories for further monitor and control. Various infrastructure devices in industrial automation are usually used for control instruction distribution, data collection, and collaboration of the industrial applications. Recent security threats on industrial automation are more frequent and the industrial control systems lack trust mechanism. Blockchain has been introduced due to its decentralization and security promise, but the election results in the original designs could be biased without collaboration trust, which leads the blockchain-based industry applications invalid. In addition, in existing solutions, neither supernodes nor normal nodes in blockchain can transfer their control authorities for disaster backup. To address the aforementioned challenges, this article proposes a collaborative trust based unbiased control transfer mechanism (CTM), which realizes a dynamic assignment of industrial control. First, a collaborative trust based delegated proof of stake consensus is proposed for determining the authorities of control dynamically and unbiasedly, by designing a lightweight trust propagation protocol. Second, a CTM for checking, alarming, and restarting CTM is devised for the disaster backup. The simulation results demonstrate the CTM, which is feasible and effective for industrial automation security.

**Index Terms**—Blockchain, collaborative trust, control transfer, industrial control systems (ICS), security.

This work was supported by the National Natural Science Foundation of China under Grants 61431008 and 61571300. (*Corresponding author: Jun Wu.*)

J. Chen, J. Wu, H. Liang, and J. Li are with the Shanghai Key Laboratory of Integrated Administration Technologies for Information Security, School of Cyber Security, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: jonnychen1996@sjtu.edu.cn; junwuh@sjtu.edu.cn; Sayyoorj@hotmail.com; lijh888@sjtu.edu.cn).

S. Mumtaz is with the Instituto de Telecomunicações, 3810-193 Aveiro, Portugal (e-mail: smumtaz@av.it.pt).

K. Konstantin is with the Department of Physics of Nanoscale Systems, South Ural State University, Chelyabinsk 454080, Russia (e-mail: kostromitinki@susu.ru).

A. K. Bashir is with the Department of Computing and Mathematics, Manchester Metropolitan University, Manchester M15 6BH, U.K., and also with the School of Electrical Engineering and Computer Science, National University of Science and Technology, Islamabad 44000, Pakistan (e-mail: dr.alikashif.b@ieee.org).

R. Nawaz is with the Department of Operations, Technology, Events and Hospitality Management, Manchester Metropolitan University, Manchester M15 6BH, U.K. (e-mail: R.Nawaz@mmu.ac.uk).

WITH the development of industry 4.0, numerous devices are interconnected for enhanced supervision and advanced process control [1], which hope to increase production efficiency and decrease operation disruption [2]. Industrial control systems (ICSs) running on various critical infrastructures are the most important platforms for continuous and stable control in industrial automation. Recently, the security threats on infrastructures are more frequent, critical pieces of US infrastructures were subjected to more than 200 hacking events every year [3]. Therefore, the security strategies of ICS have gained considerable attention.

However, there are two critical limitations of the traditional security solutions of ICS. First, because of the diversity of attack paths, the conventional security strategies based on the industrial firewall, intrusion detection system, and access control can always be broken through, which means the control authorities and instructions to facilities are always at risk of being tampered. For example, Iran's nuclear facilities were attacked by the Stuxnet virus seriously [4] and a steel mill furnace in German was destroyed by hackers in 2014 [5]. Second, the existing cloud-based or center-based regulatory programs [6] are centralized, these control centers are vulnerable to single-node failure, such as Sybil attack [7]. Therefore, how to build a reliable ICS for industrial automation is still an open issue.

Nowadays, some traditional lightweight consensus mechanism based blockchains are widely applied in the industrial field to realize digital identity, distributed security, smart contract, and microcontrol [8], hence, blockchain technology have the potential to construct a secure ICS. Specifically, instructions made by infrastructures can be stored into the blockchains to distribute with tamper-resistance and nonrepudiation properties. Moreover, smart contract technologies can be adopted to realize on-demand instruction configurations to make the control of terminals flexible and automatic.

Nevertheless, these traditional blockchains cannot be employed to construct the ICS directly. First, the consensus mechanism is the cornerstone in blockchain technology, which is expected to select the block generators (control authority owners) fairly. However, in these blockchains, lightweight consensus mechanisms can be intervened by malicious nodes easily, which causes these malicious nodes selected as the controller of the

systems biasedly. Moreover, these blockchain-based systems lack disaster backup mechanisms [9], under the distributed denial of service (DDoS) attacks or physical destruction, neither the supernodes [10] nor normal nodes in blockchain can transfer their control authorities preventing a large failure of the control nodes.

Therefore, it is an urgent problem to design an unbiased mechanism of control authority assignment in ICS. This article proposes a threshold signing based collaborative consensus mechanism to build a novel and unbiased blockchain network for ICS, thus malicious nodes are hard to attack the ICS by manipulating the consensus process. In addition, an unbiased control transfer mechanism (CTM) is also designed for preventing the failure of the control nodes or physical destruction. With these mechanisms, a system can assign control authorities to infrastructures dynamically, if the control nodes fail, the authorities can also be urgently transferred to other infrastructures to maintain the continuity. The contributions of this article are as follows.

- 1) A scalable trust propagation (TP) protocol is devised to enable the monitor and control center (MCC) and infrastructures can receive the publicly verifiable trust values from the terminals.
- 2) A collaborative trust based delegated proof of stake (CT-DPoS) mechanism is proposed, which make sure the blockchain can select control nodes randomly and unbiasedly.
- 3) A CTM is designed by an urgent invoke of the CT-DPoS for disaster backup, which realizes the stability of the ICS.

The rest of this article is organized as follows. Section II introduces the related works, involving the security threatens of the ICS, the current solutions with blockchain technology and their defects, as well as some cryptography algorithms applied in the consensus mechanisms. Section III generalizes the proposed architecture for the CTM, combining with the MCC, infrastructures, and terminals. Section IV shows the cryptographic details of the TP- and CT-DPoS-based blockchain. Section V proposes the threaten assumptions and specific processes of the unbiased CTM. The security analysis and simulations for evaluating the feasibility and effectiveness are provided in Section VI. Finally, this article is concluded in Section VII.

## II. RELATED WORKS

Most of the recent research works about industrial automation mainly focus on the efficiency and stability of the control authority. For example, fog computing is applied to enhance the processing capacity in [11] and a smart resource partitioning scheme [12] is proposed to optimize the control structures. But the inconsistency and interruptions in automation projects have been noticed in [13] and the researchers have proposed a new framework for analyzing the effectiveness of security controls in industrial Internet of things (IIoT) environments, which partly realized the stable control among the numerous devices. In [14], Chen *et al.* proposed a distributed collaborative control scheme, benefited from the introduction of the collaborative mechanism, it maintained robust against inaccurate system parameters.

Nowadays, blockchain technology has gained a considerable development, Fraga-Lamas and Fernandez-Carames have

elaborated the advantages of introducing the blockchain technology into the automotive industry in [15], they believe that the blockchain can enhance the data privacy, traceability, trustworthiness, and authentication, as well as realizing longer sustainability and higher operational efficiency to the whole industry. But the defects about high consumption and low throughput of the traditional blockchain designs are unacceptable. In order to seize the opportunities, various consortium or private blockchains based on lightweight consensus mechanisms have been proposed to match the properties of high throughput, fast response, and low consumption in the industrial field. In [16]–[18], some optimized statistics methods based on the contribution were devised to allocate the stakes in delegated proof of stake (DPoS) and then determined the control authorities depending on the rank of the stakes. Besides, Hassanzadeh *et al.* [13] also modified the structure of a blockchain to enhance the ability to handle the concurrent transactions.

However, there are still various threatens against the blockchain network. First, Lei *et al.* [19] proposed their concern that the dominating members are likely to have larger discourse rights in the voting process. Second, stake bleeding attacks [20] can leverage transaction fees and the standard longest chain rule to completely dominate a blockchain, thus conducting other malevolent behaviors. Moreover, targeted DDoS attacks and majority attacks [21] can cause interruption in a blockchain network.

The aforementioned dilemmas are generated mainly from the over concern about the lightweight mechanism and the neglect of extensive collaboration and fairness. In these respects, the algorithms of cryptography can make up for these shortcomings. First, Ulutas *et al.* [22] proposed a publicly verifiable secret sharing (PVSS) scheme, in which all entities including participators can verify the authenticity of a share through the public information sent from dealers. In the secret sharing phase, a dealer generates an encrypted share for each participator, together with a noninteractive zero-knowledge (NIZK) proof [23] to guarantee the trustworthy. During the secret reconstruction phase, the participators can recover secrets by reasonably decrypting, and all the messages and NIZK proofs will be published to prove that secrets have been decrypted correctly. PVSS allows participators to verify the reality of their shares or secrets without revealing any private content. The issue of PVSS has been widely discussed, and recent research has focused on improving the efficiency and security of the algorithms. Jarecki *et al.* [24] introduced an efficient design with compatible password protection. At last, an algorithm for generating distributed randomness [25] can be borrowed to achieve collaboration among the devices, and Cascudo and David [26] have increased its scalability.

In order to adapt to the scene of industrial automation and control, we need to design a partly decentralized but collaborated and unbiased consensus mechanism to assign control authorities unbiasedly, and then introduce a transfer mechanism for the disaster backup.

## III. PROPOSED ARCHITECTURE FOR CTM

The ICS underpinning several critical infrastructures (e.g., manufacturing, distribution, and transportation) have become

an important component of the IIoT, which are expected to achieve intelligent manufacture and continuous monitor. Hence, the precise operations and synergistic instructions request a better performance of the ICS, for improving the throughput, reducing the latency, and ensuring safety.

Although the decentralization systems are prevalent due to the popularity of bitcoin, the engineers still need a partly decentralized and hierarchical management system to administrate the smart factories [27]. The entire architecture is divided into four levels according to the Industrial Automation and Control Systems Security Standard (ISA-62443) [28]: components, systems, policies and procedures, and general management.

In this article, we mainly focus on the manufacturing processes of the ICS while ignoring the management of external material distribution and merchandise logistics. In the industrial scene, the diverse equipment should be classified according to their functions, computing abilities, and authorities. To make it clearer and simplified, a three-level architecture is proposed as the Fig. 1. shows, which includes: *MCC*, *infrastructures*, and *terminals*.

- 1) *MCC for deployment and configuration*: In smart factories, the engineers still need MCC for deploying general control strategies, transmitting general instructions (such as starting and suspending commands), and grasping global situations from subordinate equipment. In addition, any anomaly will be reported to the MCC for timely assessment and response. In terms of the blockchain-based ICS, the engineers will initialize the configuration files for infrastructures and terminals in MCC layer, then the infrastructures and terminals can be controlled through the blockchain network as well as deploying smart contracts on the blockchain nodes. The files sent from here are uniformly represented by MCC (such as  $\mathcal{F}_{MCC}$ ).
- 2) *Infrastructures for consensus execution*: Instructions for the automatic operations and the data processing are basically done at this layer. Infrastructures such as the computing platforms of machine tools are responsible for the manufacture of the terminals under their jurisdiction. By analyzing the data timely, infrastructures will determine the corresponding cooperative instructions based on the algorithm of the automation software. In order to save energy and improve efficiency, the infrastructure in a region need no work simultaneously during normal operation. The blockchain network designed is mainly built on infrastructures. Depending on the high computing abilities and large capacity, infrastructures are required to run a designed consensus mechanism to select the block generators (control authority owners, controller). These infrastructures are the most vulnerable because of their importance for maintaining the whole ICS. The messages sent from here are uniformly represented by I (such as  $\mathcal{M}_I$ ).
- 3) *Terminals for TP*: Physical manufacturing processes with a huge amount of sensors, facilities, and test devices are carried out at this layer. Terminals will receive instructions from MCC or infrastructures and generate informative data, which supports the synchronous and intelligent

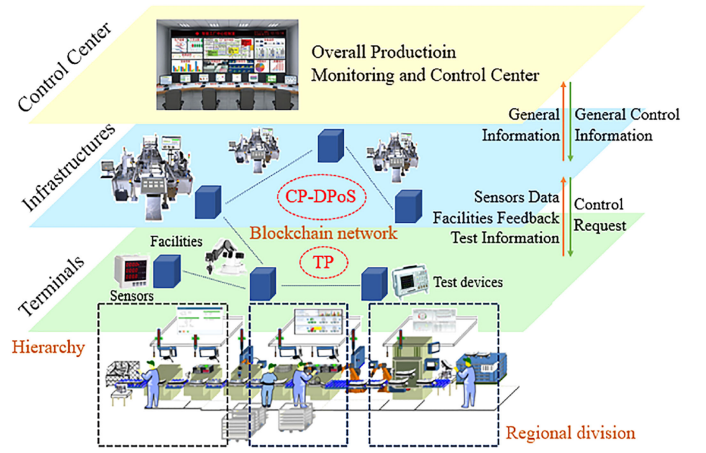


Fig. 1. Hierarchical and decentralized architecture of the ICS.

manufacture of the factories. Terminals are always allocated with lower computing power and storage, thus they can only partly participate in the consensus by providing some basic parameters. Moreover, in order to reduce computational complexity and increase scalability, in this architecture, the terminals as well as the infrastructures they belong to are regionalized according to their different locations on the production line (such as the different workshops for material transportation, scanning, and rough machining). The messages sent from here are uniformly represented by T (such as  $\mathcal{M}_T$ ).

Private blockchain restricts the access authorities of visitors, which result in a slightly weak decentralization, but it still retains most of the novel characteristics of the blockchain (e.g., immutable and traceable), and smart contracts can still be deployed on. These characteristics inspire us to combine the private blockchain and ICS. There are still some considerations about the extent of decentralization. We all know that the most primitive proof of work (PoW) mechanisms are based on the enormous computational power consumed to solve difficult mathematical problems, expecting to achieve broad fairness. However, the drawback of high latency and high energy consumption are not accepted by the industry. In our scenario, we retain the MCC and deploy the critical infrastructures as supernodes, in order to increase efficiency and save energy at the expense of the decentralization. But the cryptography will make up for this, with the terminals participating, a collaborative trust based blockchain consensus is proposed to achieve an unbiased and verifiable control assignment and transfer mechanism.

#### IV. COLLABORATIVE TRUST BASED BLOCKCHAIN

##### A. Methodologies and Assumptions

1) *Publicly Verifiable Secret Sharing*: In order to achieve the collaboration, a TP protocol is designed by introducing a PVSS scheme, the trust values  $T_i$  correspond to the secrets shared among the terminals. First, the  $(t, n)$  secret sharing scheme enables a dealer to share the secret  $S$  among  $n$  participants, where  $t$  is the threshold, and any subset of  $t$  honest participants



---

**Algorithm 1: Discrete Logarithm Equivalence Based Proof.**

---

**Require:**

- three random elements  $\alpha, \beta, \omega \in \mathbb{G}_q, s(x), r(x)$
- determine  $x = s^\alpha, y = r^\alpha$
- get proof functions:  $p^{\text{DLEQ}}(s, r) = (x, s, y, r)$
- 1: certifier sends messages  $m_1 = s^\omega, m_2 = r^\omega$  to verifier
- 2: verifier sends questions  $\beta$  to certifier
- 3: certifier sends answers  $a = \omega - \alpha\beta$  to verifier
- 4: verifier checks if  $m_1 = s^a x^\beta, m_2 = r^a y^\beta$

**Ensure:**

the validity of the message

---

can reconstructs later. Suppose  $s(x)$  is a secret sharing polynomial defined on a  $q$ -ordered Galois field  $\mathbb{G}_q$  with a generator  $\mathcal{G}$  and  $p$  is the large prime number,  $p > n \geq t$ . Second,  $\mathcal{C}$  is a commitment value generator defined on  $\mathcal{G}$ , which used to confirm the source of a message sent by a node. Third, NIZK proofs are referenced to achieve publicly verification, the dealers are requested to generate two  $t - 1$  ordered polynomial  $s(x)$ ,  $r(x)$ , and a proof function  $p(s, r)$  (such as a hash function), all the encrypted results are broadcast to every participants. Finally, participants will reconstructs the secrets depended on if  $p(s(i), r(i)) = p(\widehat{s(i)}, \widehat{r(i)})$ . An NIZK algorithm based on discrete logarithm equivalence (DLEQ) has been already introduced as Algorithm 1.

Trying to reduce the computational complexity, the terminals are already divided into groups and the number of each group is fixed, which means  $n$  is already set.

2) *Consensus Mechanisms on Blockchain*: The entire blockchain system will execute a improved DPoS consensus mechanism. However, the election is no longer based on statistics to the stake (or some alternatives) but deploying the infrastructures within a region as supernodes directly, whose number is determined. The controller will be selected dynamically and randomly. In order to achieve unbiased randomness, the algorithm of practical Byzantine fault tolerance (PBFT) algorithm is also referenced, so that the system can be still stable if at most one-third of the infrastructures in the system are malicious or unavailable.

The terminals are assumed to participate in the TP protocol by generating the trust values (a form of random numbers), and the infrastructures are assumed to invoke the trust values for DPoS elections. Log files  $\mathcal{L}$  are generated in each process in the protocol, which can be verified by any entities. If the protocol does not produce an output or some conditions are not satisfied, the failures will be reported to the MCC for a mandatory restart.

### B. TP Protocol

When the MCC or infrastructures with control authorities need to invoke the messages of terminals in order to further coordinate the control authorities, the MCC should configure public keys PK and secret keys SK corresponding to the machine addresses in advance. Then, the terminals will share and verify their messages with the aforementioned trust information, where

---

**Algorithm 2: Trust Propagation Protocol.**

---

**Require:**

- $n_i, t_i, S_i, \text{PK}_i, \text{SK}_i, i \in \{1, \dots, n\}$
- 1: **for**  $j \in \{1, \dots, n\}, j \neq i$  **do**
- 2:   initialize  $a_j$  randomly
- 3:   generate  $\widehat{s}_{ij} = \widehat{s}_{ij}(0) = \widehat{a}_{j0}, c_i, \widehat{p}_i$
- 4: **end for**
- 5: distribute the messages to each other
- 6: **for**  $j \in \{1, \dots, n\}, j \neq i$  **do**
- 7:   verify if  $s_j(x) = \mathcal{G}^{a_j(x)}$
- 8:   verify if  $p^{\text{DLEQ}} = \text{Booleans}(s^\omega = s^\alpha r^\beta)$
- 9:   **if** any verification process failed **then**
- 10:     report the invalid message and its source to MCC
- 11:   **end if**
- 12: **end for**
- 13: decrypt  $s_i = (\widehat{s}_i^{\text{SK}_i})^{-1}$  and  $p_i$
- 14: reconstruct  $S_j = s_j(0)$

**Ensure:**

The secret from participant  $P_j$ :  $S_j$

---

the TP protocol is necessary. In this model, we unify the devices required to spread trust information as propagators  $P$ . The algorithm details of the TP protocol is introduced below.

1) *Initialization*: In order to check the synergistic and unbiased features, each propagator  $P_i$  should determine the number of participants  $n_i$  and initialize the parameters  $t_i$  and  $a_j$  to form a  $t_i - 1$  ordered trust propagators polynomial  $s_i(x) = \sum_{j=0}^{t_i} a_j x^j$ , the threshold  $t_i = \lceil N_i/3 \rceil + 1$  usually, and  $a_j \in \mathbb{Z}_q^*$ . Each propagator participating in the protocol will publish its public key  $\text{PK}_i$  and retain the secret key  $\text{SK}_i$  for verification.

2) *Distribution*: After each propagator  $P_i$  receiving the propagation polynomials just initialized and the public keys accepted from the other participants,  $j \in \{1, \dots, n\}, j \neq i$ , they will generate a trust value  $\widehat{s}_{ij}$  for each propagator  $P_j$ . Suppose an original trust value to be shared is  $S_i = s_i(0) = a_0$ , in general, the trust value should be encrypted into  $\widehat{s}_{ij} = \text{PK}_i^{s_i(j)}$ . Meanwhile, a commitment value  $c_i = \mathcal{C}^{s_i(0)}$  should be generated independently and propagated together with  $\widehat{s}_{ij}$  in order to verify the validity of the trust value later. At the same time, an encrypted NIZK proof  $\widehat{p}_i = \mathcal{G}^{p_i}$  is also generated to prove the validity of the information. Afterward,  $\widehat{s}_{ij}, c_i$ , and  $\widehat{p}_i$  are broadcast to the whole network.

3) *Verification*: The third part is the verification process, any entity (not just the participants involved) can use  $\widehat{p}_i$  and  $c_i$  to verify if the trust value propagated in the protocol is valid. Taking a participant  $P_i$  as an example, if  $P_i$  receives enough trust values  $\widehat{s}_{ji}, j \neq i$ , accompanied with their  $c_j$ ,  $P_i$  will verify these message according to

$$s_j(x) = \prod_{k=0}^{t-1} c_{jk}^{x^k} = \mathcal{G}^{\sum_{k=0}^{t-1} a_{jk} x^k} = \mathcal{G}^{a_j(x)}. \quad (1)$$

If the result is positive,  $P_i$  release the decrypted trust values  $s_{ji} = (\widehat{s}_{ji}^{\text{SK}_j})^{-1}$  and  $p_j$ .

4) *Reconstruction*: After that, each participant will use  $p_i$  to check the validity of each published  $s_i$  and exclude inauthentic ones. Finally, if  $t$  trust values are valid, all the original trust values can be reconstructed by Lagrangian interpolation

$$s_i(x) = \sum_{i=1}^k s_{ji} \prod_{j=1, j \neq i}^k \frac{x - x_i}{x_i - x_j}. \quad (2)$$

Otherwise, if nodes do not receive a sufficient amount of valid data, an alert will be sent to the MCC.

### C. Collaborative Trust Based DPoS Mechanism

In this part, the CT-DPoS mechanism will be introduced in detail. The CT-DPoS mechanism is still introduced as the basis for this blockchain consensus system because a balance between security and efficiency is expected while achieving security and unbiasedness. Using the values of collaborative trust can avoid problems, such as the Byzantine attacks, which cause access denied or failure.

It is worth mentioning that a private blockchain network is built and the supernodes are introduced here, that is, not counting the equities of all the devices to form an election pool and selecting the block generators in the election pool. Instead, the aforementioned infrastructures are directly used as the supernodes, and the controllers will be directly selected from them. Although reducing the degree of decentralization, this design is more closely matched with the actual industry scene, because obviously, the factories always need hierarchy management control systems, and it is not necessary to publish the control instructions and operations throughout the entire network.

The CT-DPoS mechanism is a client-server protocol based on generating publicly verifiable, unbiased, and scalable random numbers. This mechanism allows infrastructures to aggregate the trust values generated from the participating terminals in TP protocol. It uses a commitment approach to implement the PVSS, and then uses collective signing as a witness mechanism to tie the output of the protocol for preventing client ambiguity. Fig. 2 shows the sequence of the seven steps in the CT-DPoS.

1) *Configuration*: Before the consensus initialization, the MMC will configure the basic information for each participating infrastructure and terminal. Suppose each region  $Z_i$  has  $n_{Z_i}$  infrastructures and each infrastructure  $I_i$  has  $n_{I_i}$  terminals. For infrastructures, the configuration files  $\mathcal{F}_I$  includes the functions  $F_i$ , regions  $Z_i$ , the number  $N_i$  of subordinate terminals, the lists of public keys  $\text{PK} = (\text{PK}_0, \dots, \text{PK}_{n-1})$ , private secret keys  $\text{SK}_i$ , timestamps  $ts_i$ , and identify string  $I_i$ . For terminals, the configuration files  $\mathcal{F}_T$  including the functions  $F_i$ , regions  $Z_i$ , public keys  $\text{PK}_i$ , secret keys  $\text{SK}_i$ , and timestamps  $T_i$

$$\begin{cases} \mathcal{F}_{I_i} = \{F_i, Z_i, N_i, \text{PK}, \text{SK}_i, ts_i, I_i\}_I \\ \mathcal{F}_{T_i} = \{F_i, Z_i, \text{PK}_i, \text{SK}_i, ts_i\}_T \end{cases} \quad (3)$$

2) *Initialization*: Each infrastructure  $I_i$  in the same region will determine the values in the corresponding configuration file and confirm the total number  $N_{Z_i} = \sum_{j=1}^{n_{Z_i}} N_{I_j}$ ,  $i \in \{1, \dots, n_Z\}$ ,  $j \in \{1, \dots, n_{Z_i}\}$  of the subordinate terminals, then

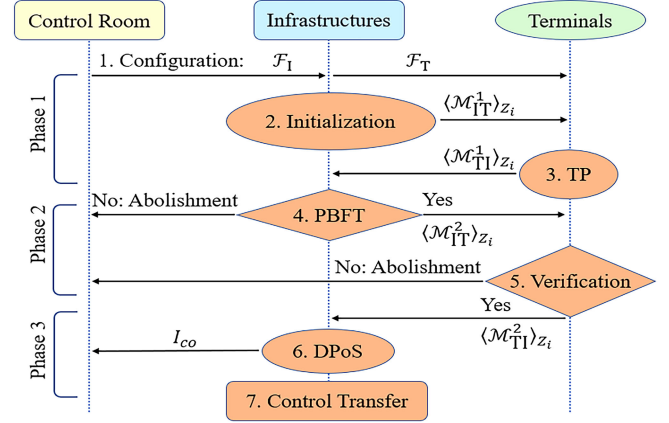


Fig. 2. Collaborative trust based DPoS mechanism.

broadcast

$$\langle \mathcal{M}_{IT}^1 \rangle_{Z_i} = \langle H(\mathcal{F}_{I_i}), N, ts_i \rangle_{Z_i} \quad (4)$$

to all equipment online and record  $\mathcal{M}_{IT}^1$  and  $N_{Z_i}$  in  $\mathcal{L}$ .

3) *Trust Propagation*: After receiving the messages, the terminals will execute the TP protocol for providing the publicly verifiable trust values to infrastructures. The protocol gets a minor modified here.

First, each terminal  $T_i$  will set  $t_i = \lfloor N_{Z_i}/3 \rfloor$ , choose a  $t_i - 1$  ordered TP polynomial  $s_i(x) = \sum_{j=0}^{t_i-1} a_j x^j$  and a  $t_i - 1$  ordered commitment polynomial  $c_i(x) = \sum_{j=0}^{t_i-1} b_j x^j$ . Second, mapping  $H(\mathcal{F}_{s_i})$  to an unidirectional hash function generator  $\mathcal{H}$  on  $\mathbb{G}_q$ . The trust values to be propagated is  $s_i = \mathcal{H}^{s_i(0)}$  and the proofs of NIZK is  $p_i$ . Third, the trust shares and NIZK proofs should be encrypted as  $\widehat{s}_{ij} = \text{PK}_{j_i}^{s_i(j)}$  and  $\widehat{p}_{ij} = \text{PK}_{j_i}^{p_i^{\text{DLEQ}}}$ .

Then, broadcast

$$\langle \mathcal{M}_{TI}^1 \rangle_{Z_i} = \langle H(\mathcal{M}_{IT}^1), (\widehat{s}_{ij}, \widehat{p}_{ij}), c_{ij}, ts_i \rangle_{Z_i} \quad (5)$$

to all equipment online and record  $\mathcal{M}_{TI}^1$  in  $\mathcal{L}$ .

4) *Practical Byzantine Fault Tolerance*: After receiving the messages, infrastructures will execute the PBFT protocol to prevent the large scale of failure. The protocol gets a minor modified here.

First, each infrastructure  $I_i$  will check the amount of received messages  $N_{\mathcal{M}_{TI}^1}$  and judges the amount of failed nodes

$$N_{F_{Z_i}} = N_{Z_i} - N_{\mathcal{M}_{TI}^1_{Z_i}} \quad (6)$$

with the maximum tolerance  $f$  if  $N_{F_{Z_i}} \leq f$ . Sent abolishment message to MCC if the inequality does not hold. Second, generates the corresponding response  $r_i = c_i - p_i s_i$ .

Then, broadcast

$$\langle \mathcal{M}_{IT}^2 \rangle_{Z_i} = \langle H(\mathcal{M}_{TI}^1), (\widehat{s}_{ji}, \widehat{p}_{ji}, H^{s_j(i)}), c_{ji}, Q, ts_i \rangle_{Z_i} \quad (7)$$

to all equipment online and record  $\mathcal{M}_{IT}^2$  in  $\mathcal{L}$ .

5) *Verification*: After receiving the messages, terminals will execute the verification process for preventing the falsification.

First, each terminal  $T_i$  will check the amount of valid commitment  $N_C$  and judge if  $N_C \geq 2f + 1$ . Compute the aggregate

response  $r = \sum r_i$  and create a list of exceptions  $\mathcal{E}$  that contains information on missing server commits and responses. Second, verify all  $\widehat{s}_{ji}$  and  $\widehat{p}_{ji}$  using  $\mathcal{H}^{s_j(i)}$  and  $\text{PK}_i$ .

Then, broadcast

$$\langle \mathcal{M}_{\text{TI}}^2 \rangle_{Z_i} = \langle \text{H}(\mathcal{M}_{\text{TI}}^2), \mathcal{E}, ts_i \rangle_{Z_i} \quad (8)$$

to all equipment online and record  $\mathcal{M}_{\text{TI}}^2$  in  $\mathcal{L}$ . Sent abolishment message to MCC if any condition does not hold.

6) *Delegated Proof of Stake*: After receiving the messages, terminals will execute the DPoS for selecting the control nodes. First, each infrastructure  $I_i$  will decrypt

$$\begin{cases} s_{ji} = (\widehat{s}_{ji})^{x_i^{-1}} = H^{s_j(i)} \\ c_{ji} = (\widehat{c}_{ji})^{x_i^{-1}} = H^{c_j(i)} \end{cases} \quad (9)$$

with  $sk_i$ , check  $s_{ji}$  with  $p_{ji}$ , and retain the valid pairs. Second, reconstruct the trust value

$$t_{i0} = t_i(0) = \sum_{i=1}^k t_i \prod_{j=1, j \neq i}^k \frac{x - x_i}{x_i - x_j} \quad (10)$$

using the Lagrange interpolation. Third, calculate the collaborative trust value  $t_{co} = \prod_{i=1}^k t_{i0}$ ,  $i \in \{1, \dots, n_Z\}$ . The trust value just generated is a random string actually, which must be converted to the corresponding random number in order to specify a serial identify number of the infrastructure next. The conversion standard introduced here is defined in american standard code for information interchange (ASCII), which is represented as a conversion function  $N$  for convenience. By normalize the result and multiply number of nodes in the region  $N_I$ , the collaborative trust identify number

$$I_{co} = \frac{N[t_{co}] - N[t_{i0}]}{\sum_i^{N_T} (N[t_{co}] - N[t_{i0}])^2} N_I N_T \quad (11)$$

corresponding to the actual equipment is generated.

Then, broadcast  $I_{co}$  to all equipment online and record  $I_{co}$  in  $\mathcal{L}$ .

7) *Control Authority Assignment*: After getting the specific identifier, the MMC will reconfigure the configuration files for the specific infrastructure to assign the control authority and transfer the necessary instructions (as well as data) to the new controller.

## V. CONTROL TRANSFER MECHANISM

### A. Threaten Assumption

There are two specific assumed attack scenarios that need the CTM. First, by entering malicious parameters, or requiring repeated runs until achieving beneficial results, the attackers may try to bias the election results, but the network is ostensibly normal. Second, in the case of DDoS attacks or large-scale physical damages, a huge amount of nodes may become inaccessible, which may causes the network broke down.

This mechanism uses the same threat model as the Byzantine attack. Assuming the attackers launch DoS attacks by repeating the TP protocol, which needs the underlying system restarting continuously. In the verification process, malicious nodes cannot transmit a dishonest trust value while imitating the correct

signatures and commitment for the other propagators in the elections of DPoS.

The proposed network includes a log verification way and an enforced restarting mechanism. The log files can ensure the legality of the messages transmitted in the network, the data generated by the terminal, and the instructions executed by the infrastructures. The enforced restarting mechanism can achieve new trust values rapidly for preventing the interruption of the whole ICS. Based on the aforementioned considerations, a CT-DPoS-based, unbiased emergency CTM is proposed for transferring the control authorities to other random infrastructures if the nodes with control authorities currently are unavailable.

The mechanism is divided into five processes. In the first process, the log files are available for any entity (including MCC, infrastructures node and terminals, or even external accessors with authorization). In this way, if an abnormal situation occurs among the devices, the entities who detect the malicious nodes will issue a warning to the entire network and notify the MCC. After the MCC is threatened and the related verifications are performed, a mandatory re-execution of the CT-DPoS command will be sent through the gateway to elect a new infrastructure for control allocation. After the seeds are collected from the terminals and the trust random value is formed, a new manager of the network is selected. Then, in the form of a blockchain transaction, both of the control authorities and the data stored in the failed devices are handed to the new infrastructures.

### B. Control Transfer Process

In general, under the assumption of the aforementioned attack model, the CTM will use the CT-DPoS based blockchain network to select the distributed random numbers for control nodes, which can avoid the negative effects caused by the unavailability of some devices. Fig. 3 shows the sequence of the six steps in the CTM, which will be explained in detail below.

1) *Anomaly Detection*: If any malicious nodes attempt to create threats, the log files

$$\mathcal{L} = \{F, SK_i, \mathcal{M}_{\text{TI}}^1, \mathcal{M}_{\text{TI}}^1, \mathcal{M}_{\text{TI}}^2, \mathcal{M}_{\text{TI}}^1, ts_i\} \quad (12)$$

of them cannot finish the verifiability examination executed by any entity (such as the response is not correct). There are following two considerations that may happen to trigger the abnormal alarm.

- 1) *Single-point malfunction alarm*: When one of the devices is illegally accessed, all operations taken by the intruder will be exposed in  $\mathcal{L}$ , and any entities find the abnormal record will trigger this alarm. Specifically, this situation generally has the following phenomenon: The promised value cannot be confirmed, the decrypted secret value format is incorrect, the signature value does not match the sender, etc.
- 2) *Multiple-point malfunction alarm*: The PBFT algorithm quoted will weaken the influence when encountering the APT or large-scale physical damage. If more than  $f$  nodes whose messages are not accepted or verification failed, this alert will be triggered. Specifically, this situation always cause the authentication of the infrastructures frequently



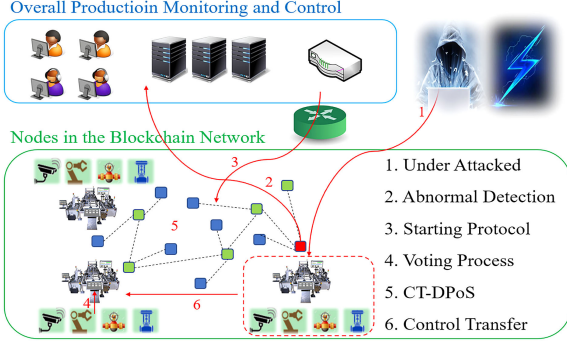


Fig. 3. Control transfer process.

denied, or the timestamps with the infrastructure messages are no longer updated.

2) *Threats Reporting*: If any entity detects an anomaly, the abnormal equipment, its log files, and the anomaly type will all be broadcast to the entire blockchain-based ICS, including the MMC.

If a smart contract is deployed on the infrastructures, the CTM can be started immediately. If not, the administrators of the MMC will further analyze and verify the report, and issue instructions through the gateway according to the result.

3) *Enforced Restarting*: After the administrators verifying the reliability of the report (or sometimes waiting for the feedback for a certain threshold time), they will return a call value for revoking the existing control authorities and restarting the CT-DPoS mechanism. Then, the entire network will abort the current process, modify the configuration file, cancel all access rights and control permissions, pending reassignment.

4) *CT-DPoS*: As the aforementioned mechanism, in order to generate the trust values for DPoS, the TP protocol will be recalled to restart the protocol process among the terminals. The blockchain network will leverage the resulting trust value to repeat the CT-DPoS consensus, and select a new reliable controller of the system.

5) *Control Transfer*: After the controller is selected, the corresponding configuration files including control instructions and data will be handed over to the new controller and the infrastructures elected become responsible for the entire ICS, so that the ICS can return to the normal situation.

Because all the messages, data, and operational instructions are kept in the blockchain network, so most of the original information need not be rewritten or restored, the original controllers just hand over the control authorities and exclude the threats. In some extreme cases, such as the devices are damaged physically, it is inevitable to lost some information, which need not discussed here.

Even if no anomaly is detected, the administrators can still start the CTM at intervals, so that control authorities are dynamically assigned, increasing the security of the ICS.

## VI. SECURITY ANALYSIS AND SIMULATIONS

### A. Security Analysis

This article mainly focuses on the control authorities in ICS, which has three aspects. First, because the contents written in

blockchain are unchangeable, blockchain technology is introduced to prevent the control information from being tampered. In the proposed ICS, the control instructions are broadcast in the form of the transactions as well as the automation configurations are deployed in the form of the smart contracts. Second, some consensus mechanisms of existing lightweight blockchain are flawed and the control authorities generated by them may be biased, which leads to the destruction of the ICS. Thus, the NIZK-based TP protocol is proposed so that a publicly verifiable election process can be realized. Third, for some unavoidable large-scale failures, a disaster backup mechanism for the block-based ICS is also devised, namely the CTM to achieve stable and sustainable industrial control. A detailed analysis of some security features will be introduced in this section.

1) *Feasibility*: The most basic aim of the mechanism is to ensure that the ICS can always maintain an effective control toward the smart factories, even some nodes are under attack. There are two situations and the proposed mechanism can solve both of them. First, the control authorities are always dynamically allocated with the time according to the different configuration files, the intruders can hardly hack into the specific control infrastructures. Moreover, even some nodes have been already hacked in and become malicious, the mechanism of PVSS will exposure them and make reports to the MCC. As a result, an honest infrastructure can successfully initialize the terminals and obtain their trust values with verifiable commitments or signatures, and then unbiased selection can be realized. As to some extreme condition, such as the number of failed nodes is bigger than the threshold, the authentication mechanisms are still helpful, as the simulation shows, the possibility of a successful control transfer is high.

2) *Disclosure Prevention*: Anonymity is one of the most important features of the blockchain. In this blockchain-based system, the anonymity is well preserved to realize the disclosure prevention.

First, multiple security mechanisms have been introduced into this system, such as the strategies of keys distribution and the  $(t, n)$  secret sharing scheme. During the trust values generation, all the trust shares are encrypted by corresponding secret keys, and the public keys can be decrypted various trust shares, which means only the shares encrypted with correct secret keys are valid, whereas the invalid shares from the abnormal nodes will be reported to the MCC. Second, the terminals send the encrypted shares to each other along with the NIZK proofs. Then, an infrastructure will select a subset of inputs from each group, ignoring the terminals, which do not respond on time or with appropriate values. When the infrastructures receive the valid signatures and commitments, the messages will be decrypt and broadcast to the entire network. Third, the threshold mechanism is executed such that any node whose number of received trust shares is less than the set threshold cannot reconstruct the trust values. In actual scenario, this method increases the cost and difficulty of the attacks, because more shares of the trust values must be obtained to eavesdrop the secret.

3) *Counterfeit Prevention*: In the CT-DPoS, all the generated messages are recorded into the log files so that any entities including the third-parties authorized to visit the private ICS can verify the normality of the execution processes. The log



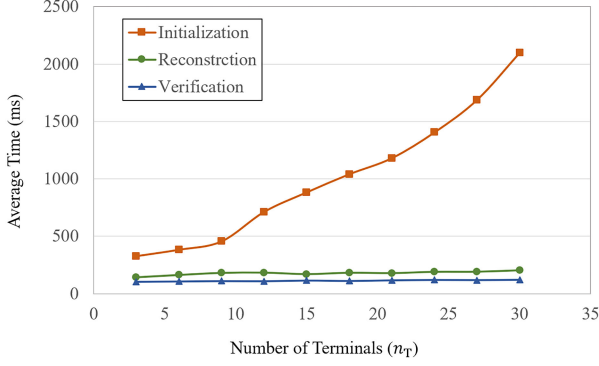


Fig. 4. Average running time depends on terminals.

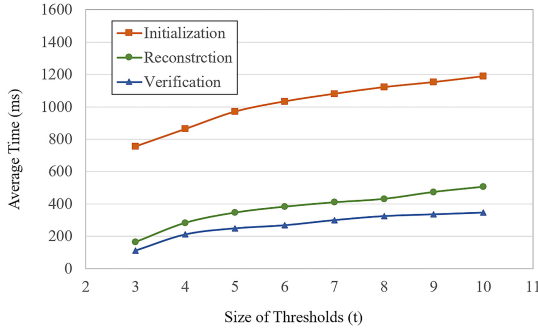


Fig. 5. Average running time depends on threshold.

files contain all messages sent and received during protocol execution, as well as session configuration files. First, the TP protocol is designed based on PVSS, which is the cornerstone of the verifiability for ICS. Second, the secret keys of the equipment are only used to generate the signatures, and the signatures can be verified using the public keys. Therefore, despite the expected verifiability is primitives in PBFT, it is not possible to produce a legal log file unless obtaining both of the public key and the secret key. Third, the log files can be executed by any entity, according to the process written, the result of the execution cannot be legal unless the protocol is run in a legal manner. With these verifiability assurances, this CTM cannot be unbiased by malicious nodes.

### B. Simulations

The proposed TP protocol, CT-DPoS consensus, and the CTM are evaluated with Go language and the network topology is simulated in Network Simulators 3 (NS-3). All computing nodes are equipped with Intel Core 3.20 GHz CPU, 16 GB of RAM, and Ubuntu operation system.

1) *TP Protocol Evaluation*: Three kinds of average running time according to the three procedures (initialization, verification, and reconstruction) in the TP protocol are tested under the different  $(t, n)$  threshold conditions. The comparison of different numbers of terminals  $n$  is shown in Fig. 4, and the comparison with different thresholds  $t$  is shown in Fig. 5, where the number of terminals varies from 3 to 30 and the threshold is set from 3 to 10.

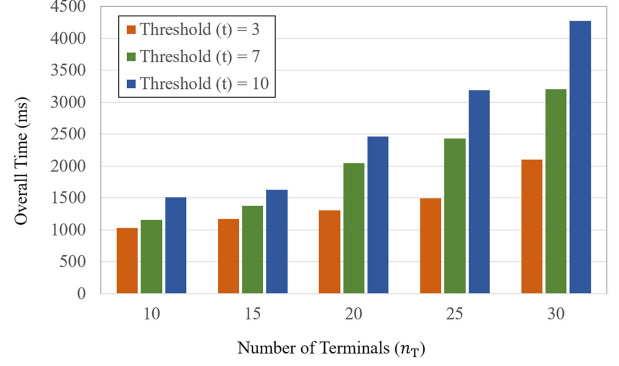


Fig. 6. Overall running time of the TP protocol.

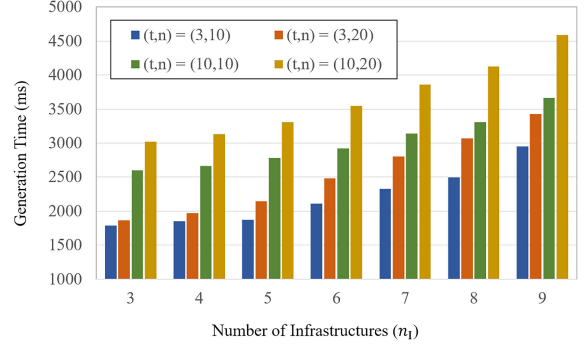


Fig. 7. Average running time of the randomness generation.

In general, most of the time required is spent on initialization. The execution time of the trust value initialization process exponentially increases with the terminal number, whereas the impacts on the verification and reconstruction processes are not significant. This is because the rising number of terminals significantly increases the number of parameters that need to be initialized, whose computational complexity is between  $n$  and  $n^2$ . On the other hand, the slight increase of verification and reconstruction proofs the good capability of this scheme.

The threshold  $t$  determines the orders of the propagation polynomial and the Lagrangian interpolation. With the threshold increasing, the parameters of initialization and the computational complexity of reconstruction should all increase. But the slopes of all three procedures can be noticed getting flattered, which means the effect is becoming feebler with the threshold increasing.

Generally, as the Fig. 6. shows, the overall running time of TP protocol has an approximately linear relationship with the network scales and gets little affection by the threshold, which indicates novel feasibility and expandability.

2) *CT-DPoS Evaluation*: In terms of the CT-DPoS mechanism, the randomness generation program is implemented and invoked in a real proof-of-stake simulation. First, considering the parameters, the number of infrastructures is varied from 3 to 9, the number of terminals distributed to each infrastructure is set to 10 and 20, whereas the corresponding thresholds are determined to be 3 and 10 (which is most representative). Fig. 7. shows the detailed results, it can be seen that in the process of generating

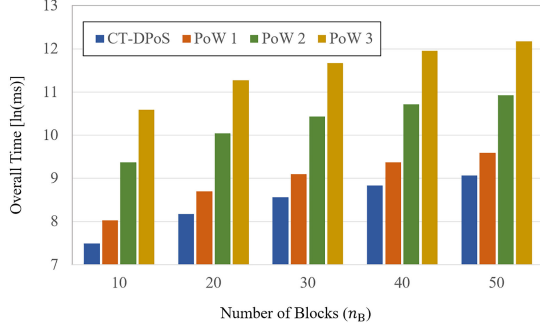


Fig. 8. Overall running time of the CT-DPoS mechanism.

random numbers, the number of nodes and thresholds influences the generating time greatly at the same time. Because on the one hand, the increased numbers of terminals and infrastructures will exponentially increase the complexity of the generated and verified information. On the other hand, the number of thresholds determines the appropriateness of the terminal device to return a valid trust value, which requires some waiting time.

In order to test the performance of the consensus mechanism, the collaborative generation of unbiased random numbers based on trust values in real blockchain network scenarios is invoked for testing the time of generating 10–50 blocks, which is a mature method to reflect the instruction efficiency of this blockchain network. Then, the results are compared with three traditional PoW methods, as shown in Fig. 8. In this comparison, different difficulties for the traditional PoW methods are set, whose evaluation indexes are the numbers of zeros before the target hash values. The indexes of difficulties are 5, 7, and 9 for PoW 1, PoW 2, and PoW 3, respectively. The comparing results shows that the mechanism designed has very efficient performance.

3) *Unbiased Transfer Mechanism Evaluation*: In order to measure the success rate of the CTM under different attack ranges, a certain ratio of nodes are randomly set as malicious nodes (sending error data or rejecting responses), and then count the probability that the control authority is transferred to a normal infrastructure after the CT-DPoS consensus is executed. The range of malicious nodes is set to 1%–63%, because exceeding this probability, most of the equipment in the factory has been invaded, and the transfer has no meaning. Among them, when  $p = 1\%$ , it represents the case of single point invasion. As the Fig. 9. shows, in this case, the probability of successful transfer is above 99%. In terms, there are more than one-third of malicious nodes, the means of PBFT has been invalidated. At this time, the system only relies on the signature-commitment mechanism to maintain security, but the transfer efficiency is still more than 50%.

## VII. CONCLUSION

This article focus on the security threats to the ICS. By devising a TP protocol among the terminals and combining it with blockchain technology, a CT-DPoS consensus is proposed for assigning the control authorities dynamically. Meanwhile, a publicly verifiable CTM is also implemented against the failure. The

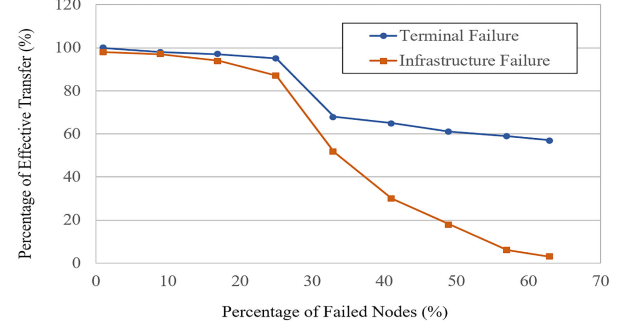


Fig. 9. Percentage of the effective transfer.

simulation results show the TP protocol is scalable according to the good computing performance, and the comparison among the CT-DPoS indicates a more efficient transaction ability than the traditional designs. At last, the control transfer experiment testimony the proposed architecture is effective with a high probability of success. This collaborative trust blockchain based unbiased CTM is of importance to realize a stable and reliable ICS.

In the future, in order to adapt a wider ICS requirement (such as the more meticulous instructions), optimizing the underlying structure for improving the throughput of the blockchain should be further considered. Then, this private blockchain will be extended to a public blockchain for breaking the information estrangement and achieve a true industrial interconnection.

## REFERENCES

- [1] C. Zhang, Z. Liu, B. Gu, K. Yamori, and Y. Tanaka, "A deep reinforcement learning based approach for cost- and energy-aware multi-flow mobile data offloading," *IEICE Trans. Commun.*, vol. E101-B, no. 7, pp. 1625–1634, Jul. 2018, doi: [10.1587/transcom.2017CQP0014](https://doi.org/10.1587/transcom.2017CQP0014).
- [2] C. Zhang and Z. Zheng, "Task migration for mobile edge computing using deep reinforcement learning," *Future Gener. Comput. Syst.*, vol. 96, pp. 111–118, Jul. 2019, doi: [10.1016/j.future.2019.01](https://doi.org/10.1016/j.future.2019.01).
- [3] C. Lin, S. Wu, and M. Lee, "Cyber attack and defense on industry control systems," in *Proc. IEEE Conf. Dependable Secure Comput.*, 2017, pp. 524–526.
- [4] A. Nourian and S. Madnick, "A systems theoretic approach to the security threats in cyber physical systems applied to stuxnet," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 1, pp. 2–13, Jan. 2018.
- [5] "The state of IT security in Germany 2014," Federal Office for Information Security, Bonn, Germany, 2014.
- [6] G. Li, G. Xu, A. K. Sangaiah, J. Wu, and J. Li, "EdgeLaaS: Edge Learning as a service for knowledge-centric connected healthcare," *IEEE Netw.*, vol. 33, no. 6, pp. 37–43, Nov./Dec. 2019.
- [7] K. Zhang, X. Liang, R. Lu, and X. Shen, "Sybil attacks and their defenses in the internet of things," *IEEE Internet Things J.*, vol. 1, no. 5, pp. 372–383, Oct. 2014.
- [8] N. Mohamed and J. Al-Jaroodi, "Applying blockchain in industry 4.0 applications," in *Proc. IEEE 9th Ann. Comput. Commun. Workshop Conf.*, Las Vegas, NV, USA, 2019, pp. 852–858.
- [9] H. Liang, J. Wu, S. Mumtaz, J. Li, X. Lin, and M. Wen, "MBID: Micro-blockchain based geographical dynamic intrusion detection for V2X," *IEEE Commun. Mag.*, vol. 57, no. 10, pp. 77–83, Oct. 2019.
- [10] C. Yu, C. Zhang, X. Gou, and Y. Ji, "Study on supernode election algorithm in P2P network based upon district partitioning," in *Proc. Int. Conf. Commun. Softw. Netw.*, Chengdu, China, 2009, pp. 196–199.
- [11] J. Wu, M. Dong, K. Ota, J. Li, W. Yang, and M. Wang, "Fog-computing-enabled cognitive network function virtualization for an information-centric future internet," *IEEE Commun. Mag.*, vol. 57, no. 7, pp. 48–54, Jul. 2019.

- [12] 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 Trans. Ind. Informat.*, vol. 14, no. 10, pp. 4702–4711, Oct. 2018.
- [13] A. Hassanzadeh, S. Modi, and S. Mulchandani, "Towards effective security control assignment in the industrial internet of things," in *Proc. IEEE 2nd World Forum Internet Things*, Milan, Italy, 2015, pp. 795–800.
- [14] J. Chen, X. Cao, P. Cheng, Y. Xiao, and Y. Sun, "Distributed collaborative control for industrial automation with wireless sensor and actuator networks," *IEEE Trans. Ind. Electron.*, vol. 57, no. 12, pp. 4219–4230, Dec. 2010.
- [15] P. Fraga-Lamas and T. M. Fernandez-Carames, "A review on blockchain technologies for an advanced and cyber-resilient automotive industry," *IEEE Access*, vol. 7, pp. 17578–17598, 2019.
- [16] X. Wang, Q. Feng, and J. Chai, "The research of consortium blockchain dynamic consensus based on data transaction evaluation," in *Proc. 11th Int. Symp. Comput. Intell. Des.*, Hangzhou, China, 2018, pp. 214–217.
- [17] J. Kim, J. Jin, and K. Kim, "A study on an energy-effective and secure consensus algorithm for private blockchain systems (PoM: Proof of Majority)," in *Proc. Int. Conf. Inf. Commun. Technol. Convergence*, Jeju, South Korea, 2018, pp. 932–935.
- [18] X. Lin, J. Li, J. Wu, H. Liang, and W. Yang, "Making knowledge tradable in edge-AI enabled IoT: A consortium blockchain-based efficient and incentive approach," *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6367–6378, Dec. 2019.
- [19] K. Lei, Q. Zhang, L. Xu, and Z. Qi, "Reputation-based Byzantine fault-tolerance for consortium blockchain," in *Proc. IEEE 24th Int. Conf. Parallel Distrib. Syst.*, Singapore, 2018, pp. 604–611.
- [20] P. Gazi, A. Kiayias, and A. Russell, "Stake-bleeding attacks on proof-of-stake blockchains," in *Proc. Crypto Valley Conf. Blockchain Technol.*, 2018, pp. 85–92.
- [21] J. Moubarak, E. Filiol, and M. Chamoun, "On blockchain security and relevant attacks," in *Proc. IEEE Middle East North Africa Commun. Conf.*, Jounieh, Lebanon, 2018, pp. 1–6.
- [22] M. Ulutas, V. V. Nabyev, and G. Ulutas, "A PVSS scheme based on Boolean operations with improved contrast," in *Proc. Int. Conf. Netw. Service Secur.*, 2009, pp. 1–3.
- [23] E. Ben-Sasson, A. Chiesa, M. Green, E. Tromer, and M. Virza, "Secure sampling of public parameters for succinct zero knowledge proofs," in *Proc. IEEE Symp. Secur. Privacy*, San Jose, CA, USA, 2015, pp. 287–304.
- [24] S. Jarecki, A. Kiayias, H. Krawczyk, and J. Xu, "Highly-efficient and composable password-protected secret sharing (Or: How to protect your bitcoin wallet online)," in *Proc. IEEE Eur. Symp. Secur. Privacy*, Saarbrücken, Germany, 2016, pp. 276–291.
- [25] E. Syta *et al.*, "Scalable bias-resistant distributed randomness," in *Proc. IEEE Symp. Secur. Privacy*, San Jose, CA, USA, 2017, pp. 444–460.
- [26] I. Cascudo and B. David, "SCRAPE: Scalable randomness attested by public entities," in *Proc. Int. Conf. Appl. Cryptography Netw. Secur.*, 2017, pp. 537–556.
- [27] J. Wu, M. Dong, K. Ota, J. Li, and Z. Guan, "Big data analysis-based secure cluster management for optimized control plane in software-defined networks," *IEEE Trans. Netw. Service Manage.*, vol. 15, no. 1, pp. 27–38, Mar. 2018.
- [28] E. Knapp and J. Langill, *Industrial Network Security: Securing Critical Infrastructure Networks for Smart Grid SCADA, and Other Industrial Control Systems*. Rockland, MA, USA: Syngress, 2014.