IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 12 September 2022, accepted 31 October 2022, date of publication 9 November 2022, date of current version 15 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3220852

RESEARCH ARTICLE

Secure Trust-Based Delegated Consensus for Blockchain Frameworks Using Deep Reinforcement Learning

YUNYEONG GOH[1], JUSIK YUN[1,2], DONGJUN JUNG[1], AND JONG-MOON CHUNG[1], (Senior Member, IEEE)

[1]School of Electrical and Electronic Engineering, College of Engineering, Yonsei University, Seoul 03722, South Korea
[2]Samsung Electronics, Suwon 16677, South Korea

Corresponding author: Jong-Moon Chung (jmc@yonsei.ac.kr)

This work was supported in part by the Ministry of Science and Information and Communication Technology (ICT) under Grant D0318-21-1008, and in part by the National Fire Agency of the Republic of Korea under Grant 20017102.

ABSTRACT Internet of Things (IoT) networks generate massive amounts of data while supporting various applications, where the security and protection of IoT data are very important. In particular, blockchain technology supporting IoT networks is considered as the most secure, expandable, and scalable database storage solution. However, existing blockchain systems have scalability problems due to low throughput and high resource consumption, and security problems due to malicious attacks. Several studies have proposed blockchain technologies that can improve the scalability or the security level, but there have been few studies that improve both at the same time. In addition, most existing studies do not consider malicious attack scenarios in the consensus process, which deteriorates the blockchain security level. In order to solve the scalability and security problems simultaneously, this paper proposes a Dueling Double Deep-Q-network with Prioritized experience replay (D3P) based secure trust-based delegated consensus blockchain (TDCB-D3P) scheme that optimizes the blockchain performance by applying deep reinforcement learning (DRL) technology. The TDCB-D3P scheme uses a trust system with a delegated consensus algorithm to ensure the security level and reduce computing costs. In addition, DRL is used to compute the optimum blockchain parameters under the dynamic network state and maximize the transactions per second (TPS) performance and security level. The simulation results show that the TDCB-D3P scheme can provide a superior TPS and resource consumption performance. Furthermore, in blockchain networks with malicious nodes, the simulation results show that the proposed scheme significantly improves the security level when compared to existing blockchain schemes by effectively reducing the influence of malicious nodes.

INDEX TERMS Blockchain, consensus algorithm, deep reinforcement learning (DRL), Internet of Things (IoT), trust.

I. INTRODUCTION

A. BACKGROUND AND MOTIVATION

Blockchains provide transparency and security to data management systems and can be utilized in various domains [1], [3]. When used for systems that generate massive real-time data based on various applications (e.g., social networks, extended reality (XR) services, financial systems, and autonomous control), database storage blockchain systems will need to satisfy stringent quality of service (QoS) requirements. Especially, blockchain-enabled real-time Internet of Things (IoT) networks need to address the low throughput problems arising from the consensus process. Bitcoin and Ethereum, which are representative blockchain systems, are set up to support 3 to 4, and 14 transactions per second (TPS), respectively, which are insufficient to satisfy the data generation rates of IoT networks and credit card transactions [4], [5]. Traditional blockchains use the proof of work (PoW) consensus algorithm. In PoW, miners repeat hash operations to solve mathematical puzzles, which results in massive energy

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

consumption [6]. In addition, as the number of nodes in the blockchain network increases, the computation cost and time required for block verification increase, which reduces the average throughput of the blockchain. To solve the scalability problem, a more advanced scalable blockchain technology is needed. For example, EOS [7] applies a consensus method called delegated proof of stake (DPoS), which delegates mining tasks to a small number of nodes. Zilliqa introduced the sharding approach to boost the TPS of blockchains [8].

In addition, security as well as scalability is a very important factor in blockchain systems. If a colluding group of malicious nodes occupies the hash power or voting rights of a blockchain network, then the group can take control of the blockchain consensus process through a 51% attack and manipulate the data in the blockchain [9]. Especially, in blockchain networks that use a consensus process based on delegation or sharding (in which only selected nodes participate in the consensus process), the security level is further weakened because the colluding group of malicious nodes can more easily override the consensus process with fewer malicious nodes, like in a single-shard takeover attack [10]. In addition, if a malicious block producer presents an invalid block to the consensus process, the maliciously produced block will be rejected by other honest nodes and will not be chained to the blockchain even if malicious nodes do not occupy a majority of the blockchain network. This type of malicious block generation prevents honest transaction ledgers from being chained to the blockchain, which can greatly reduce the TPS or result in a database denial of service (DoS).

However, most existing blockchain systems do not simultaneously consider scalability, security, and decentralization, which have a trilemma relationship. If one performance improvement is considered, the other blockchain performances may be degraded, so all performance parameters should be considered simultaneously. In addition, there have been very few research publications on security and blockchain performance enhancement for situations where there are nodes maliciously participating in the consensus process.

B. CONTRIBUTION

In order to solve these problems, this paper presents a Dueling Double Deep-Q-network with Prioritized experience replay (D3P) based secure trust-based delegated consensus blockchain (TDCB-D3P) scheme. The contributions of this paper are summarized as follows.

1) A secure blockchain consensus algorithm called trust-based delegated practical byzantine fault tolerance (TD-PBFT) that considers scalability and security is proposed. In order to reduce the influence of malicious nodes in the blockchain network, the TD-PBFT scheme applies a secure trust system that evaluates the reliability of blockchain nodes and delegates reliable nodes to the consensus process. A method of evaluating the trust of nodes based on the delegated consensus result is proposed.
2) The proposed TDCB-D3P scheme combines delegated consensus with deep reinforcement learning (DRL) to improve the throughput while maintaining a high security level. To improve the convergence speed and performance of the deep-Q-network (DQN) system, the proposed TDCB-D3P framework applies double DQN (DDQN), dueling network (DN), and prioritized experience replay (PER) technology to the DQN. The proposed TDCB-D3P framework optimizes the throughput of the blockchain system considering malicious attacks by reflecting the dynamic state of the network.
3) Blockchain performance metrics and the security condition that guarantees the safety of the consensus are analyzed in the TDCB-D3P scheme. In addition, simulation results show that the TDCB-D3P model improves the TPS performance while reducing computing costs compared to existing models. In addition, the simulation results (based on the ratio of malicious nodes) show that the proposed TDCB-D3P framework's security performance exceeds the existing models.

The remaining parts of this paper are organized as follows. The related works are described in Section II. Section III presents the system model of this paper. The performance of the TDCB-D3P scheme is analyzed in Section IV. The proposed DRL-based performance optimization framework and simulation results are presented in Sections V and VI, respectively. Finally, in Section VII, the conclusion of the paper is presented.

II. RELATED WORKS

In this section, related works regarding blockchain-enabled IoT schemes and blockchain consensus algorithms are presented. In addition, research on blockchain systems applying DRL is introduced.

A. BLOCKCHAIN ON INTERNET OF THINGS

Considerable research has been conducted to manage large amounts of data and transactions generated by IoT applications in various fields through blockchain systems. For example, in [1], Li et al. proposed a consortium blockchain for energy trading through a credit-based payment scheme in an industrial IoT environment. In [3], Yao et al. proposed a cloud computing service-based blockchain system for resource trading. In [11], Singh et al. proposed a decentralized healthcare management system for blockchain-enabled healthcare applications. In [12], Shen et al. proposed a blockchain-based secure device authentication mechanism to ensure the security and privacy in cross-domain industrial IoT network. In addition, in [13], Aujla et al. proposed a decoupled blockchain scheme to manage IoT health monitoring sensor data and preserve the security of the data. In [14], Saba et al. proposed a fault-tolerant routing algorithm based on machine learning for autonomous IoT security. The authors utilized a cipher block chaining mode to maintain the privacy and authentication of data transmission. In [15], Haseeb et al.

proposed a fault-tolerant supervised routing model to verify data blocks and support trust-worthy communication using a trust system against malicious threats in 6G IoT networks.

B. CONSENSUS ALGORITHMS ON BLOCKCHAIN

There are various consensus algorithms used in blockchain systems. In Bitcoin [16], a consensus algorithm based on hash computation called PoW is used. However, PoW requires numerous hash operations that results in massive resource consumption issues and has a very low TPS performance. To solve this problem, consensus algorithms such as proof of stake (PoS) and practical byzantine fault tolerance (PBFT) have been proposed. PBFT is one of the byzantine fault tolerant algorithms, which verifies blocks through a voting consensus process [17]. PBFT can reduce unnecessary hash computations, but the message complexity is high when there are a large number of blockchain nodes. In addition, Zilliqa utilizes sharding technology to increase the TPS by processing transactions in parallel [8]. However, sharding increases the latency because the consensus proceeds in two-phases, and a single shard takeover attack can weaken the security level.

In addition, to address scalability issues, a delegated consensus algorithm has been introduced in which only a few delegated nodes participate in the blockchain consensus process. For example, EOS [7] uses a DPoS method that selects a specific number of validators to conduct the blockchain consensus process. In addition, in [18], Abishu et al. proposed a PBFT-based proof of reputation consensus model for blockchain-based energy trading system in electric vehicles. The consensus model selects a set number of validators using the reputation of vehicles based on evaluations of each other. In [19], Li et al. proposed a committee consensus scheme to prevent malicious attacks in a federated learning framework. The NEO scheme presented in [20] was designed to reduce the energy consumption of the blockchain system by using a delegated PBFT method, which selects a representative for consensus through voting of NEO coin owners. In these delegation-based blockchain consensus algorithms, as the number of nodes participating in the consensus process decreases, the computing costs and communication traffic are correspondingly reduced. However, there are problems that the decentralization property and security level of the blockchain may be degraded. However, existing delegated consensus algorithms do not consider malicious attacks to occur during the consensus process, which can significantly affect the blockchain's performance. Furthermore, research on how to properly set the delegation ratio to ensure the fairness and security of the consensus process has not been conducted.

C. DEEP REINFORCEMENT LEARNING BASED BLOCKCHAINS

DRL is a machine learning technology that combines deep learning with reinforcement learning (RL). DRL helps the agent to make optimal decisions under dynamic states. For example, the DQN developed by DeepMind approximates the state-action values in Atari games through a deep neural network (DNN) [21]. In addition, methods to improve the performance of DQN were also studied. For example, in [22], DDQN technology was used to improve the overestimation problem of DQN based on a double estimator structure. In [23], DN is used to increase the performance of the DQN system by using DNN consisting of two separate streams applying an advantage function.

In addition, research has been conducted to optimize the performance of the blockchain through DRL technology. For example, in [24], Liu et al. proposed a DRL based blockchain (DRLB) framework that optimizes the blockchain parameters according to the network state in order to maximize the TPS while satisfying the constraints for security, latency, and decentralization. In [25], Yun et al. proposed a sharded blockchain framework that improves the TPS through sharding technology using DQN. In [26], Liu et al. proposed an efficient and secure DRL based data sharing scheme to achieve the maximum amount of data collection. In [27], Yang et al. proposed a DRL based energy-efficient resource allocation scheme. In [28], Dai et al. proposed a blockchain based content caching framework that conducts optimal content caching using DRL to support the security and privacy protection process. In addition, in [29], He et al. propose a blockchain system that optimizes resource allocation through DRL to ensure the security and privacy of edge computing enabled IoT networks. Furthermore, in [30], Feng et al. propose a reinforcement learning-based blockchain-enabled mobile edge computing system that jointly optimizes the cooperative offloading resource allocation problem and blockchain performance. However, these studies do not consider malicious attacks that could occur during the blockchain consensus process and do not include a defense mechanism that can efficiently protect against malicious nodes in a dynamic blockchain network. In addition, there lack studies on how to effectively apply the many benefits of DRL technology to the blockchain delegated consensus process and trust systems, which is the focus of this paper.

III. SYSTEM MODEL

In this section, the system model of the blockchain-enabled IoT network and delegated consensus process are introduced.

A. BLOCKCHAIN SYSTEM WITH DELEGATED CONSENSUS IN IoT

Fig. 1 shows a blockchain system using delegated consensus in an IoT network, where a large number of transactions are generated through various IoT applications (e.g., smart factories, smart homes, surveillance, etc.). Transactions of IoT applications such as data storing and data sharing are transmitted to the blockchain network to be recorded in a distributed ledger. In a general blockchain system, all blockchain nodes, called miners or validators, participate in the consensus process to verify the transactions. However, in a delegated consensus system, only the chosen delegated nodes participate in the consensus process.

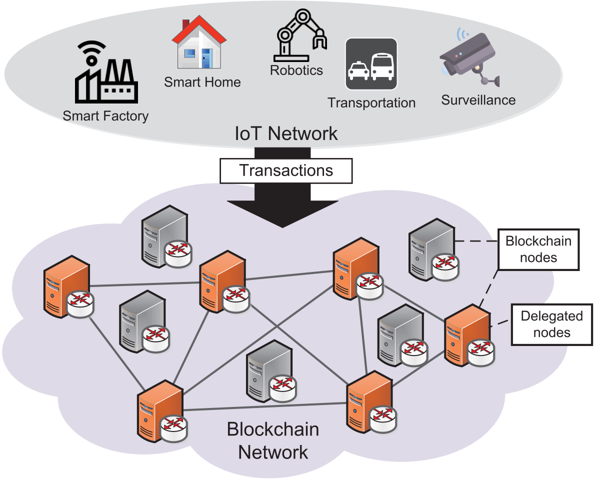

FIGURE 1. Delegated consensus based blockchain-enabled IoT system example.

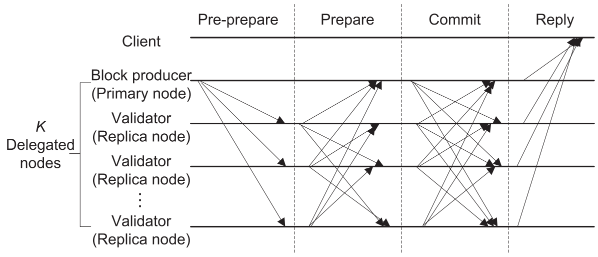

FIGURE 2. Trust-based delegated PBFT algorithm procedures.

In this paper, it is assumed that there are N blockchain nodes that process the transactions. The set of blockchain nodes is denoted as $\mathcal{N} = \{1, 2, \cdots, N\}$. Among the N nodes, a certain percentage of the nodes participate in the delegated consensus process according to the delegation ratio φ $(0 < \varphi \leq 1)$. Therefore, the $K = \lfloor \varphi N \rfloor$ delegated nodes will participate in the consensus process and become elements of the delegated node set $\mathcal{N}_D$, in which $\mathcal{N}_D \subseteq \mathcal{N}$.

B. TRUST-BASED DELEGATED CONSENSUS PROCESS

The consensus algorithm used in the proposed scheme is based on PBFT [17]. The PBFT consensus algorithm is suitable for scalable IoT networks due to the low consumption of computational resources and high consensus speed. In this paper, a TD-PBFT algorithm is presented. The TD-PBFT algorithm operates in the following order as shown in Fig. 2.

1) TRUST-BASED DELEGATION

According to the delegation ratio φ, the number of nodes to participate in the consensus process is determined. The value of φ is determined by the DRL system according to the network state, which is described in sections IV and V. Delegation is performed based on the trust values of the nodes. In this scheme, trust is a measure of how trustworthy a blockchain node is expected to be in the consensus process [31]. Nodes with high trust values have a high probability of honestly participating in the consensus process. Conversely, nodes with low trust values are more likely to behave maliciously. Therefore, to make the consensus process secure, nodes with high trust values are set to have a high probability of being selected as delegated nodes. Among the delegated nodes, one block producer (primary node) is randomly selected, and the rest of the delegated nodes participate in the consensus process as validators (replica nodes) that verify the block transmitted by the block producer.

2) DELEGATED CONSENSUS

The block producer collects transactions produced in the IoT network and creates a block according to the blockchain parameters. In the pre-prepare stage, the block producer transmits the block to the validators and requests for verification. In the prepare stage, validators that received a block from the block producer propagate a confirmation message to the other delegated nodes. During the commit stage, the delegated nodes check the validity of the block, and attempt to confirm whether it is valid or invalid. A valid block refers to a block in which its header content and transactions list have not been forged and the hash value is also correct. The delegated nodes verify that there was no transaction information forgery in the verification process and send a commit message to each other. Considering the quorum requirement for correct consensus in BFT-based algorithms, if a new block's commit result receives more than two-thirds of the valid votes, then that block is judged to be valid and chained to the blockchain [17]. The results of the blockchain consensus are reported in the reply step, and all blockchain nodes can investigate the results of the consensus.

3) SYSTEM ASSUMPTIONS

In the proposed TDCB-D3P framework, K delegated nodes out of N blockchain nodes proceed with the TD-PBFT consensus. In the consensus process, some additional assumptions are established.

a) The data transmission rate $R_{i,j}(t)$ in the process of message exchange is applied through a finite-state Markov channel model [24]. $R_{i,j}(t)$ refers to the data transmission rate of the link sent from node i to node j, quantified as $\mathbb{R} = \{\mathbb{R}_1, \mathbb{R}_2, \cdots, \mathbb{R}_r\}$. In addition, the state transition probability matrix is given as $[p_R(t)]_{r\times r}$, where $p_R(t) = Pr[R_{i,j}(t+1) = \mathbb{R}_b \mid R_{i,j}(t) = \mathbb{R}_a]$ and $\mathbb{R}_a, \mathbb{R}_b \in \mathbb{R}$. In addition, the computing capability $c_i(t)$ is applied through a finite-state Markov channel model, where $c_i(t)$ refers to the computing capability of node i, quantified as $\mathbb{C} = \{\mathbb{C}_1, \mathbb{C}_2, \cdots, \mathbb{C}_c\}$, the state transition probability matrix is given as $[p_c(t)]_c$, where $p_c(t) = Pr[c_i(t+1) = \mathbb{C}_b \mid c_i(t) = \mathbb{C}_a]$ and $\mathbb{C}_a, \mathbb{C}_b \in \mathbb{C}$.

b) The message verification process considers cryptographic operations that include verifying signatures, generating or verifying message authentication codes (MACs), requiring θ and α cycles, respectively [32]. In addition, the message verification tasks are based on a round-robin scheduling.

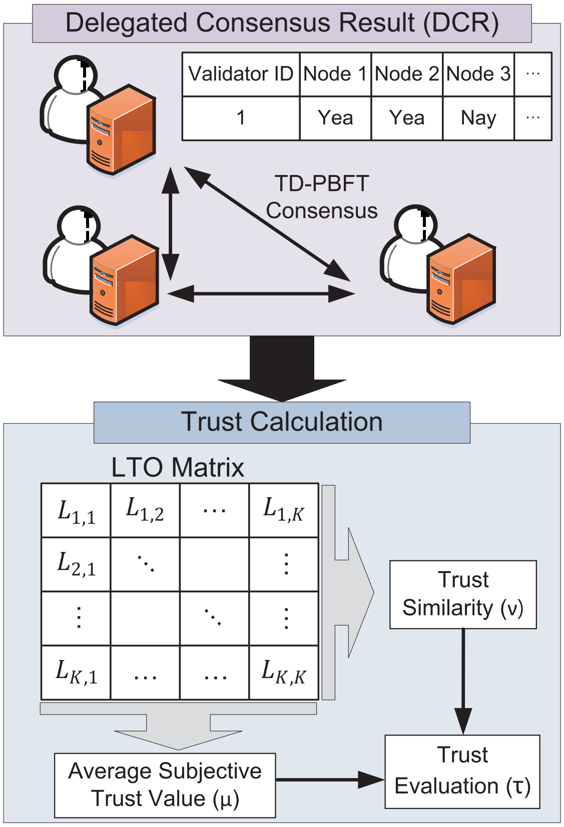

FIGURE 3. Trust evaluation system.

c) When the trust-based delegation process is completed, the delegated nodes proceed with the consensus process for a predefined period, which is defined as one epoch.

C. TRUST SYSTEM

Fig. 3 describes the process of evaluating the trust of nodes. Trust, which represents the level of reliability and security of the nodes, is calculated based on a node's reputation in the blockchain consensus process [33]. The trust authority (TA) calculates trust values based on the result of the blockchain consensus. The detailed process of evaluating trust is as follows.

1) DELEGATED CONSENSUS RESULT

The delegated nodes participating in the TD-PBFT process create a subjective array called the delegated consensus result (DCR) that stores the commit results of other delegated nodes in the consensus process. For example, if a node gives a commit message to agree on block generation, it is stored as 'Yea,' and if a node gives a commit message against block generation, it is stored as 'Nay.' It is assumed that the DCR information of the delegated nodes can be collected by the DRL agent that can play the role of the TA.

2) LOCAL TRUST OPINION MATRIX

The local trust opinion (LTO) matrix for delegated nodes is generated based on the DCR in which the delegated nodes subjectively evaluate each other. The records of commit results in the DCR are transformed into values in the LTO matrix. In the LTO matrix, $L_{i,j}$ represents the subjective trust of delegated node i for delegated node j. The $L_{i,j}$ value is computed as

$$L_{i,j} = \begin{cases} \dfrac{\Sigma_i(Yea)}{\Sigma_i(Yea) + \Sigma_i(Nay)}, & \text{if commit of } j \text{ is 'Yea'} \\ \dfrac{\Sigma_i(Nay)}{\Sigma_i(Yea) + \Sigma_i(Nay)}, & \text{if commit of } j \text{ is 'Nay'}, \end{cases} \tag{1}$$

where $\Sigma_i(Yea)$ and $\Sigma_i(Nay)$ are the number of 'Yea' and 'Nay' in the DCR of delegated node i. For example, if a delegated node i receives a 'Yea' commit message from delegated node j and the proportion of 'Yea' in the DCR of node i is 75%, $L_{i,j}$ becomes 0.75.

3) TRUST EVALUATION

Based on the LTO matrix, the trust to evaluate the reliability of the delegated nodes is calculated. Two types of information can be derived from the LTO matrix.

First, the average subjective trust value (μ) can be computed as

$$\mu_i = \frac{1}{K} \sum_{j=1}^{K} L_{j,i}, \tag{2}$$

where μ_i refers to the average subjective trust value of delegated node i, which is an indicator used to determine if the delegated node has submitted accurate comments when validating the block. If a valid block is presented, honest delegated nodes will present a 'Yea' commit message. However, if a malicious delegated node submits a commit message of 'Nay,' the μ of the malicious node will be degraded because the ratio of 'Nay' is small.

Second, the trust similarity (ν) can be evaluated through the row vectors of the LTO matrix. Trust similarity is based on the cosine similarity of the row vectors and can be obtained from

$$\nu_i = \frac{1}{K} \sum_{j=1}^{K} \frac{L_i \cdot L_j}{\|L_i\| \|L_j\|}, \tag{3}$$

where L_i and L_j respectively refer to the ith and jth row vectors of the LTO matrix and ν_i refers to the trust similarity value of delegated node i. Trust similarity can be used to penalize a node for forging a DCR report. If a malicious node generates DCR by counterfeiting commit messages reversely to degrade the trust level of a honest node, the trust similarity of the malicious node is degraded based on the similarity calculation.

Finally, based on μ and ν, the trust value τ of the corresponding consensus round is computed as

$$\tau_i = \mu_i \nu_i, \tag{4}$$

where τ_i is the trust value of delegated node i. The trust value calculated based on the LTO indicates how reliable a node can be considered in the consensus process. The trust value

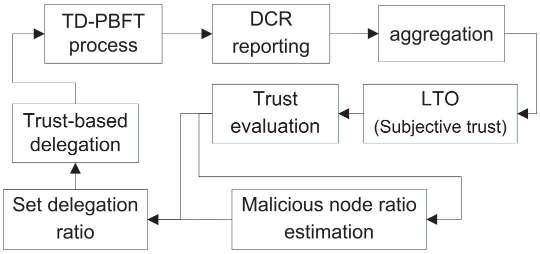

FIGURE 4. Block diagram for trust system.

is updated by averaging the trust value of the round and the previous rounds.

Therefore, if a node does malicious actions, such as not responding to block verification or reporting incorrect verification results during the TD-PBFT consensus process, the trust value of the node will be reduced. Nodes with low trust values have a low probability of being selected as delegated nodes, so malicious nodes are less likely to participate in the consensus. Therefore, nodes acting maliciously cannot serve as block producers and validators, and thus do not receive rewards such as coins or tokens provided by the blockchain network. Therefore, honest nodes of the blockchain network will be motivated to participate honestly in the TD-PBFT consensus process.

In addition, the probability that a malicious node is included in the delegated node set can be estimated based on the trust values. The initial trust value of the nodes is set to 0.5. If a node honestly participates in the blockchain consensus process, the trust value of the node will increase. In the opposite case, the trust value of the node will be degraded. Therefore, nodes with a trust value less than a certain threshold ψ (e.g., 0.5) can be determined to be malicious nodes, and the estimated probability of a malicious delegated node ($\widehat{p_d}$) can be computed as follows.

$$\widehat{p_d} = \frac{n\,(\,i\mid\, i \in \mathcal{N}_D,\quad \tau_i \leq \psi)}{K}. \tag{5}$$

The trust value of the nodes and the estimated probability of a malicious delegated node are used to select the delegated nodes in the subsequent consensus rounds and calculate the safe delegation ratio through DRL.

Fig. 4 shows the block diagram for the overall trust system of the proposed model. The delegated nodes participating in the TD-PBFT consensus process report the DCR, which records the results of other blockchain nodes, to the TA. The DCR of each node is aggregated to create an LTO matrix. Each value in the LTO matrix represents the subjective trust between nodes. Then, the average subjective trust value and trust similarity are calculated and multiplied to evaluate the trust values of the blockchain nodes. Based on the trust values, the estimated probability of a malicious delegated node is computed, and the trust values are used as parameters for blockchain performance optimization in the DRL process. The delegation ratio of the next round is determined by the DRL agent. The delegated nodes to participate in the blockchain consensus process in the next round are selected based on their trust values.

4) ADVERSARIAL MODEL

The adversarial models affecting the blockchain system are represented in [33], in which the Naïve Malicious Attack (NMA) and Collusive Rumor Attack (CRA) models are applied in this paper. In the two models, malicious nodes act as opposed to honest nodes in the blockchain consensus. If one of the malicious nodes is selected as a block producer, it will create an invalid block and propose it to the validators. In addition, the malicious nodes reject a valid block and accept an invalid block. Furthermore, in the CRA model, when malicious nodes create a DCR, they report the opposite of the commit results received from the other nodes. The purpose of CRA is to disrupt accurate trust evaluations as well as the consensus process.

IV. PERFORMANCE ANALYSIS

The blockchain's scalability is an indicator that determines whether the blockchain system can achieve a sufficient TPS performance as the network grows. The latency that occurs during the message exchange and validation procedures also needs to be considered. Furthermore, performance measurements should take into account the decentralization and security corresponding to the trilemma relationship of the blockchain along with scalability. These blockchain performance indicators are in a trade-off relationship. This section describes the blockchain performance parameters used in the proposed TDCB-D3P scheme.

A. SCALABILITY (THROUGHPUT)

A blockchain's TPS represents the amount of transactions per second that the blockchain network can process and confirm. A block producer makes a B byte block which includes a H byte block header in order to propose the block to the consensus round. In the blockchain system, the maximum TPS can be computed as

$$\mathcal{T}\,(B, T_I) = \frac{\lfloor (B-H)/b \rfloor}{T_I}, \tag{6}$$

where T_I is the block interval and b is the average transaction size.

B. LATENCY

Latency refers to the time spent for a transaction to get in the blockchain network and be processed through consensus and irreversibly chained. The total latency ($T_{latency}$) can be denoted as

$$\begin{aligned} T_{latency} &= T_I + T_{consensus} \\ &= T_I + T_v + T_p, \end{aligned} \tag{7}$$

where $T_{consensus}$ is the total consensus time consisting of T_v and T_p that respectively represent the message validation delay and message propagation time.

1) MESSAGE VALIDATION DELAY

In TD-PBFT, one of the K delegated nodes participates in the delegated consensus process as a block producer and the other

K-1 nodes serve as validators. The block producer generates $\mathcal{M}$ (batch size) blocks and then propagates the blocks to the validators [34]. During the consensus process, the block producer processes $\mathcal{M}$ signatures and $2\mathcal{M} + 4(K - 1)$ MAC operations while the validators perform verification of $\mathcal{M}$ signatures and $\mathcal{M} + 4(K - 1)$ MAC operations each [24]. Thus, the validation delay of the block producer can be computed as $T_{v\_bp} = \frac{\mathcal{M}\theta+[2\mathcal{M}+4(K-1)]\alpha}{c_{bp}}$ where c_{bp} refers to the computing capability of the block producer. The validator requires a validation delay of $T_{v\_val} = \frac{\mathcal{M}\theta+[\mathcal{M}+4(K-1)]\alpha}{c_i}$ where c_i is the computing capability of the validator i. The validation process of the delegated nodes is conducted in parallel. Therefore, the message validation delay can be expressed as follows.

$$T_v = \frac{1}{\mathcal{M}} \max_{i \in \mathcal{N}_D} (T_{v\_bp}, T_{v\_val}). \tag{8}$$

2) MESSAGE PROPAGATION TIME

The message propagation time refers to the delay spent for a node to transmit a message and arrive at the target node. The timeout ζ is set at each consensus phase to restrain excessive delays in the consensus process due to unresponsive nodes. Thus, the message propagation time of the request according to each consensus step can be computed as follows [24].

$$\begin{aligned} T_p &= \frac{1}{\mathcal{M}} \left(T_{Pre-prepare} + T_{Prepare} + T_{commit} + T_{reply}\right) \\ &= \frac{1}{\mathcal{M}}(min\left(\max_{i \in \mathcal{N}_D, i \neq bp} \frac{\mathcal{M}B}{R_{bp,i}}, \zeta\right) \\ &\quad +, min\left(\max_{i,j \in \mathcal{N}_D, i \neq bp, j \neq i} \frac{\mathcal{M}B}{R_{i,j}}, \zeta\right) \\ &\quad +, min\left(\max_{i,j \in \mathcal{N}_D, j \neq i} \frac{\mathcal{M}B}{R_{i,j}}, \zeta\right) \\ &\quad +, min\left(\max_{i \in \mathcal{N}_D} \frac{\mathcal{M}B}{R_{i,client}}, \zeta\right)). \end{aligned} \tag{9}$$

In addition, to satisfy the blockchain finality property, the consensus process should be completed within the consecutive block interval (u) [24]. Considering all delay factors, the constraint for the latency can be computed as follows.

$$T_{latency} = T_I + T_v + T_p \leq uT_I. \tag{10}$$

C. SECURITY ANALYSIS

Depending on the blockchain consensus algorithm, security constraints are different. For example, in the PoW consensus algorithm, if an organization of nodes occupies more than 51% of the hash power, then it can completely take over the blockchain consensus [8]. However, byzantine fault tolerance (BFT) based algorithms reduce individual centralization by using a voting scheme. PBFT can accommodate f malicious nodes out of N nodes that satisfy the condition of $(3f + 1) \leq N$ to ensure the safety of the consensus process [17].

In the delegated PBFT consensus process, the delegated nodes conduct the consensus process, in which case the following lemma is established.

Lemma 1: The range of the delegation ratio φ that guarantees safety of the consensus process in the delegated PBFT consensus process is $3Np + 1 \leq \lfloor \varphi N \rfloor \leq N$. □

Proof: In a blockchain system with N nodes, when the delegation ratio is φ, $\lfloor \varphi N \rfloor$ delegated nodes participate in the consensus. If the probability of a malicious node is p, the worst-case is that all Np malicious nodes are assigned to be delegated nodes. In this case, since $3Np + 1 \leq \lfloor \varphi N \rfloor$ must be satisfied and φ's maximum is 1, the range of the delegation ratio becomes $3Np + 1 \leq \lfloor \varphi N \rfloor \leq N$. ■

In Lemma 1, the worst-case occurs when all malicious nodes unfortunately participate in the delegated consensus, resulting in a too strict security constraint. However, since the TD-PBFT algorithm performs trust-based delegations, nodes that are considered malicious nodes are highly likely to be excluded from the consensus process. Eventually, the proportion of malicious nodes in the delegated node set becomes an important factor to guarantee the security level of the consensus. Therefore, the following lemma can be established in the TD-PBFT scheme.

Lemma 2: The range of the delegation ratio φ that guarantees the safety of the consensus result in the TD-PBFT consensus process is $3 \lfloor \varphi N \rfloor p_d + 1 \leq \lfloor \varphi N \rfloor \leq N$. □

Proof: If the probability of a malicious node in the delegated node set is p_d, the number of malicious delegated nodes that take part in the TD-PBFT consensus process is equal to $\lfloor \varphi N \rfloor p_d$. Therefore, from the safety condition of the PBFT consensus algorithm, the condition of $3 \lfloor \varphi N \rfloor p_d + 1 \leq \lfloor \varphi N \rfloor$ must be satisfied to ensure the safety of the TD-PBFT. Therefore, the range of the delegation ratio is same as $3 \lfloor \varphi N \rfloor p_d + 1 \leq \lfloor \varphi N \rfloor \leq N$. ■

The condition of $3 \lfloor \varphi N \rfloor p_d + 1 \leq \lfloor \varphi N \rfloor \leq N$ in Lemma 2 is set as a security constraint of the TDCB-D3P scheme, allowing a valid block to be generated through the consensus process even if malicious nodes participate in the consensus. The DRL agent sets the delegation ratio that satisfies the security constraint according to the state of the blockchain network.

D. DECENTRALIZATION

There are several ways to measure the level of decentralization of a blockchain, which include, fairness, entropy, and similarity [35]. To evaluate the decentralization of the TDCB-D3P scheme, this paper uses the Gini coefficient, which is used in diverse fields to measure inequality [36]. Since the TDCB-D3P scheme is based on a delegation method, if the same node is continuously assigned as the delegated node, the decentralization of the blockchain may be degraded. Therefore, this scheme focuses on decentralizing the delegated consensus. The Gini coefficient is computed as

$$G(\delta) = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} \left|\delta_i - \delta_j\right|}{2 \sum_{i=1}^{N} \sum_{j=1}^{N} \delta_i}$$

$$= \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} \left|\delta_i - \delta_j\right|}{2N \sum_{i=1}^{N} \delta_i}, \qquad (11)$$

where δ_i represents the number of times node i participated in the TD-PBFT process. The Gini coefficient is a value between 0 and 1, and if the value is close to 1, the variance of δ_i is large, resulting in a degradation in its decentralization performance. Thus, it is possible to maintain the decentralized performance by setting a decentralization threshold η as follows.

$$G(\delta) \leq \eta. \qquad (12)$$

If the Gini coefficient violates the constraint, the DRL agent will increase the delegation ratio, allowing more nodes to equally participate in the TD-PBFT process, thus lowering the Gini coefficient. In the same way, the stake or geographic location of nodes can be used as decentralization indicators [24].

V. PERFORMANCE OPTIMIZATION FRAMEWORK USING DEEP REINFORCEMENT LEARNING

To reflect the dynamic state of the blockchain-enabled IoT scheme, the proposed TDCB-D3P framework applies DRL to optimize the blockchain's performance. DRL is effective in solving high-dimensional problems in time-varying networks, which is why it is applied to blockchain-based IoT networks in this paper. The framework optimizes the block size, block interval, and delegation ratio by reflecting the state of the network. The states, actions, reward, and overall DRL framework of the proposed TDCB-D3P system are described in the following.

A. STATE SPACE

According to the decision epoch t, five kinds of variables can be denoted as

$$S^t = [R, c, \boldsymbol{\delta}, \boldsymbol{\tau}, \widehat{p_d}]^t, \qquad (13)$$

where $R = \left\{R_{i,j} \mid 1 \leq i, j \leq N\right\}$ is the data transmission rate of the link sent from node i to node j and $c = \{c_i \mid 1 \leq i \leq N\}$ is the computing capability of node i. In addition, based on the TD-PBFT consensus result, the number of times a node participates in the consensus process is recorded in $\boldsymbol{\delta} = \{\delta_i \mid 1 \leq i \leq N\}$, the trust value of the nodes is $\boldsymbol{\tau} = \{\tau_i \mid 1 \leq i \leq N\}$, and the estimated probability of the malicious delegated nodes is $\widehat{p_d}$.

B. ACTION SPACE

The DRL agent selects actions to maximize the long-term reward by considering the state of the dynamic blockchain environment. The action space for epoch t consists of a block size B, block interval time T_I, and delegation ratio φ, which can be denoted as

$$A^t = [B, T_I, \varphi]^t, \qquad (14)$$

where the block size $B \in \left\{1, 2, \cdots, \dot{B}\right\}$ has a maximum block size $\dot{B}$ and the block interval $T_I \in \left\{0.5, 1, \cdots, \dot{T_I}\right\}$ has a maximum block interval $\dot{T_I}$. In addition, according to the delegation ratio $\varphi \in \{0.1, 0.2, \cdots, 1\}$, the delegated node set $\mathcal{N}_D$ is selected based on the trust value of the nodes.

Algorithm 1 DRL Process for the TDCB-D3P Performance Optimization Framework

```
Input: minibatch size $k$, learning rate $\mathcal{L}$, priority coefficient $\varkappa$ and $\kappa$, replay period $\xi$, target-Q-network update period $\mathbb{T}$
Initialize replay memory $D$ of the size of $\dot{D}$ and $\Delta = 0$
Initialize action-value function $Q$ of random weights $\omega$
Initialize target action-value function $Q^*$ of weights $\omega^* \leftarrow \omega$
Load initial state space and input it into the actor network
Deep reinforcement learning
for each decision epoch $t$ do
  if $p_\epsilon \geq \epsilon$ then select random action with probability $\epsilon$,
  otherwise select action $A^t = \mathrm{argmax}_A\, Q\left(S^t, A^t; \omega\right)$
  Execute action $A^t$ to select the TD-PBFT consensus parameters
  Observe reward $\mathcal{R}^t$ and next state $S^{t+1}$ from the consensus result
  Store transition $\left(S^t, A^t, \mathcal{R}^t, S^{t+1}\right)$ in replay memory $D$ with maximal priority $p_t = \max_{i<t} p_i$
  if $t \equiv 0$ mod $\xi$ then
    for $j = 1$ to $k$ do
      Sample transition $j \sim P_j = \frac{{p_j}^{\varkappa}}{\sum_j {p_j}^{\varkappa}}$
      Compute importance-sampling weight
        $\varpi_j = (\dot{D}P_j)^{-\kappa} / max_i \varpi$
      Compute TD-Error $\rho_j$
      Update transition priority $p_j \leftarrow \left|\rho_j\right|$
      Accumulate $\Delta \leftarrow \Delta + \varpi_j \rho_j \nabla_\omega Q(S_{j-1}, A_{j-1})$
    end for
    Update weight $\omega \leftarrow \omega + \mathcal{L}\Delta$, reset $\Delta = 0$
    Update target-Q-network weight $\omega^* \leftarrow \omega$ for every $\mathbb{T}$
  end if
end for
```

C. REWARD FUNCTION

The reward of the DRL is set to maximize the TPS of the blockchain system while meeting the constraints, such as latency, security, and decentralization. The objective function and constraints of the reward can be summarized as

$$\begin{aligned} &Objective: \ \max_{A} Q(S, A), \\ &Constraint, 1: \ T_{latency} = T_I + T_v + T_p \leq uT_I, \\ &Constraint, 2: \ 3\lfloor \varphi N \rfloor \widehat{p_d} + 1 \leq \lfloor \varphi N \rfloor \leq N, \\ &Constraint, 3: \ G(\delta) \leq \eta, \end{aligned} \qquad (15)$$

where $Q(S, A) = V(S) + (\mathbb{A}(S, A) - \frac{1}{|A|} \sum_{a'} A(S, a'))$ is the state-action value function that includes value function $V(S)$ and advantage function $\mathbb{A}(S, A)$. The reward function $(\mathcal{R}^t)$ according to epoch t is calculated as $\mathcal{R}^t = \mathcal{R}\left(S^t, A^t\right) = \frac{\lfloor (B-H)/b \rfloor}{T_I}$ only when all constraints are satisfied. If any of the constraints are not satisfied, the reward function has a zero value. Thus, the DRL agent reflects the state of the blockchain network and selects the optimal actions that maximize the long-term reward while satisfying the constraints.

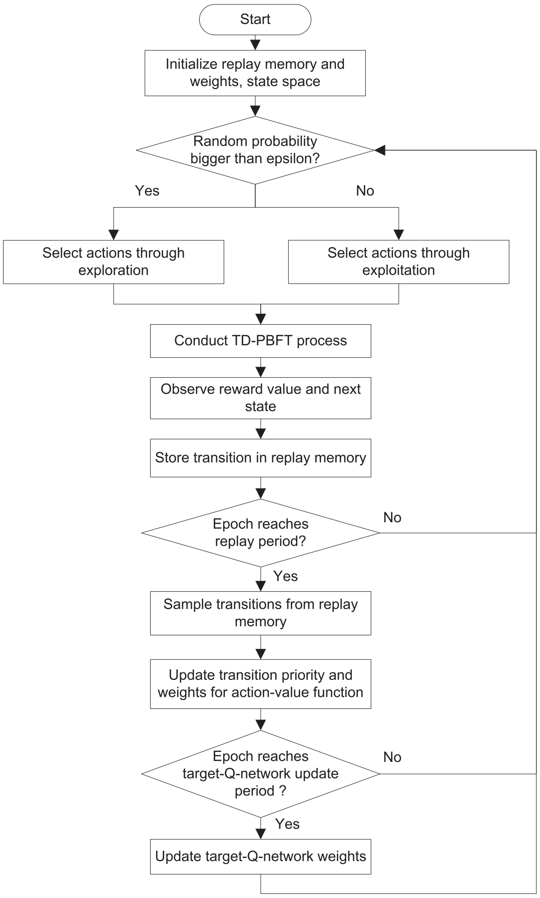
Start
Initialize replay memory and weights, state space
Random probability bigger than epsilon?
Yes
No
Select actions through exploration
Select actions through exploitation
Conduct TD-PBFT process
Observe reward value and next state
Store transition in replay memory
Epoch reaches replay period?
No
Yes
Sample transitions from replay memory
Update transition priority and weights for action-value function
Epoch reaches target-Q-network update period ?
No
Yes
Update target-Q-network weights

FIGURE 5. Flowchart of the TDCB-D3P DRL process.

D. OVERALL DRL FRAMEWORK OF TDCB-D3P

In the TDCB-D3P scheme, the purpose of the DRL system is to select blockchain parameters to maximize the reward, which will maximize the system's performance. The DRL agent reflects the states (e.g., computing capability and transmission rate of the blockchain nodes) and evaluates trust values based on the blockchain's consensus result. Then, the DRL agent finds the optimal actions to maximize the reward by evaluating the optimal state-action value function through reinforcement learning.

In this paper, the DRL framework is based on DQN, which is suitable for frameworks that use discrete action spaces such as blockchain system. DQN uses a DNN as an approximator function to find the maximum Q-value. The DNN is trained to minimize the loss function that is measured as the difference between the predicted Q-value and the target Q-value [37].

The overall DRL process and flowchart of the proposed TDCB-D3P scheme are presented in Algorithm 1 and Fig. 5. In the beginning, the replay memory D (with size $\dot{D}$), and weights for the action-value function Q, and the target action-value function Q^* are initialized. In addition, the initial state space for the blockchain network is set and entered into the actor network. The DRL agent of TDCB-D3P selects an action space consisting of block size, block interval time, and delegation ratio by exploration or exploitation through a decaying epsilon-greedy algorithm. Exploration is a non-greedy action that tries new action values to receive a larger reward than prior experiences. On the other hand, exploitation is a greedy action that selects actions that have an optimal state-action value function $Q(S, A)$ through the Bellman equation. Accordingly, action values are selected by comparing the random probability value p_e with ϵ. After the action A^t is selected, the blockchain nodes proceed with the TD-PBFT consensus process, and the reward $\mathcal{R}^t$ and next state space S^{t+1} are computed through the consensus result. The DRL agent stores the transition $\left(S^t, A^t, \mathcal{R}^t, S^{t+1}\right)$ according to epoch t in the experience replay memory. For every replay period ξ, the agent samples the transition set accumulated in the replay memory according to the minibatch size k and uses it to update the weights of the DNN. In the TDCB-D3P scheme, sampling of transitions is conducted based on the priorities of transitions. Then, the important-sampling weight ϖ and the temporal-difference (TD) error ρ are computed through the sampled transitions, and the two values and the gradient of the state-action value function are multiplied and accumulated in the weight-change Δ. Then, the weights of the action-value function (ω) are updated according to the learning rate $\mathcal{L}$, and the target-Q-network weights (ω^*) are updated every target-Q-network update period $\mathbb{T}$. In the proposed DRL framework of TDCB-D3P, additional improvements to the DQN scheme were applied as follows.

1) DOUBLE DQN

In conventional Q-learning, the overestimation problem of the Q-value occurs due to the maximization expectation for the function approximator. Because the same value is used to select and evaluate an action in the max operator, it results in overoptimistic action value estimates. The DDQN [22] solves the overestimation problem by decoupling action selection from action evaluation. The loss function of DDQN is computed by $L^t = \mathcal{R}^{t+1} + \gamma Q^* \left(S^{t+1}, \text{argmax}_{a'} Q\left(S^{t+1}, a'\right)\right) - Q\left(S^t, A^t\right)$. Thus, the DDQN can improve the DRL performance by preventing overestimation problems.

2) PRIORITIZED EXPERIENCE REPLAY

In the conventional DQN scheme, transitions are randomly sampled from an experience replay memory and used for updating weights of the DNN. But the random sampling method does not reflect the importance of transitions. The PER [38] computes the priority of transitions based on the TD error and samples the transitions by reflecting the priority. The TD error of transition j (ρ_j) is computed by $\rho_j = \mathcal{R}_j + \gamma Q^*(S_j, \text{argmax}_{a'} Q(S_j, a')) - Q(S_j, A_j)$. Then, the sampling probability of transition j (P_j) is set as $P_j = \frac{p_j{}^{\varkappa}}{\sum_j p_j{}^{\varkappa}}$, where $\varkappa$ is a priority coefficient, and $p_j = \left|\rho_j\right|$ is the transition priority. If the $\varkappa$ value is 0, uniform random sampling is conducted. The PER method has a higher learning convergence speed than the random sampling method of the conventional DQN

TABLE 1. Simulation Parameters.

Symbol	Parameters	Value
N	Number of blockchain nodes	100-400
b	Average size of transactions	100-500 Bytes
H	Size of block header [16]	80 Bytes
$\dot{B}$	Maximum size of block	8 MB
$\dot{T}_I$	Maximum block interval	10 s
$R_{i,j}$	Data transmission rate of link sent from node i to node j [24]	10-100 Mbps
c_i	Computing resource of node i [24]	10-30 GHz
θ	Computing cost for the signature verification [32]	2 MHz
α	Computing cost for generating and verifying the MAC [32]	1 MHz
ψ	Trust threshold to judge malicious nodes	0.5
u	Consecutive block interval that satisfies the blockchain finality [39]	6
$\mathcal{M}$	Batch size [34]	3

TABLE 2. DRL Parameters.

Parameters	Value
Learning rate	0.0001
Discount rate	0.99
Explore decay rate	0.0001
Replay memory size	1000
Multi-step returns	3
Priority coefficients $\varkappa$ and κ	0.6, 0.4
Type of DNN	CNN
Neurons in convolution layer 1	32
Neurons in convolution layer 2	64
Neurons in convolution layer 3	64
Neurons in FC layers	512
Activation function of convolution layers	ReLu
Activation function of FC layer 1	ReLu
Activation function of FC layer 2	Linear
Optimizer	RMSProp
Loss function	RMSE
Target-Q-network update period	10

by allowing transitions with high TD-error values to be frequently sampled when updating the weights of the DNN.

3) DUELING NETWORK

DN [23] uses a network structure consisting of two separate streams. One stream is used to approximate the state-value function $V(S)$, and the other stream is used to approximate the state-dependent action advantage function $\mathbb{A}(S, A)$, which has not been used in conventional DQN systems. Then, the two streams are combined to evaluate the state-action value function $Q(S, A)$, which can result in a faster convergence speed.

VI. SIMULATION RESULTS

The DRL framework is implemented in a PyTorch environment which is utilized to optimize the proposed blockchain system. Table 1 and Table 2 summarize the parameters used in the simulation and the DRL parameters, respectively. The performance of TPS, which denotes the blockchain throughput, was mainly analyzed under the constraints of the latency, decentralization, and security.

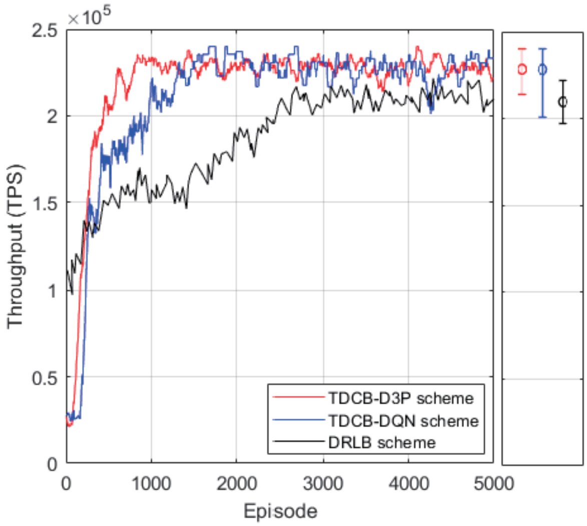

FIGURE 6. TPS performance and convergence trend analysis.

A. SCALABILITY ANALYSIS

Fig. 6 shows the simulation results of TPS. The simulation environment is based on a total of 100 blockchain nodes and an average transaction size of 200 bytes. The TPS performance analysis of the proposed TDCB-D3P scheme is compared to the following schemes.

a) TDCB-D3P scheme: Optimizes the blockchain parameters in TD-PBFT through DRL. To improve the convergence speed, DDQN, DN, and PER are applied to the DQN.

b) TDCB-DQN scheme: Instead of D3P, conventional DQN is used as the DRL framework.

c) DRLB scheme: Applies the DRL based blockchain optimization framework of [24], which is based on DQN and does not use a trust system or delegation consensus.

The simulation results show that the TPS is low at the starting of the reinforcement learning episode, but gradually increases as the DRL agent starts to find the optimum blockchain parameters. DDQN, PER, and DN technologies were applied to the TDCB-D3P scheme to improve the convergence rate and learning stability. When compared to the TDCB-DQN scheme, the TDCB-D3P scheme shows a faster convergence rate which results in a higher TPS by finding the optimal action even in the initial episode of reinforcement learning where transitions are not accumulated much. In addition, after convergence, the TDCB-D3P scheme has a higher average throughput and higher minimum TPS than the DRLB scheme. The TDCB-D3P scheme has an average throughput that is about 9% higher than the DRLB scheme. The proposed TDCB-D3P scheme can satisfy the latency constraint more easily because fewer nodes participate in the consensus process through trust-based delegation.

Fig. 7 provides a comparison between TDCB-D3P (with several delegation ratios) and other blockchain schemes based on the total computing costs. The PBFT scheme

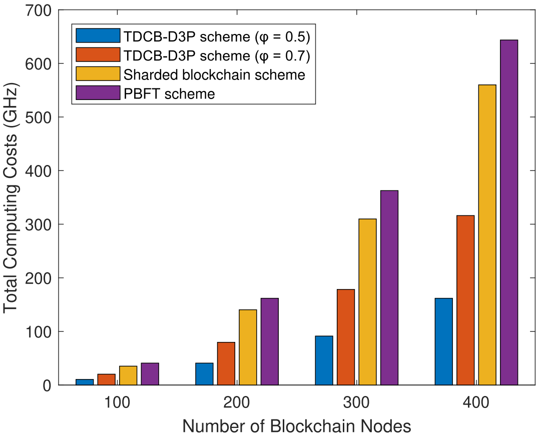

FIGURE 7. Total computing costs according to number of nodes.

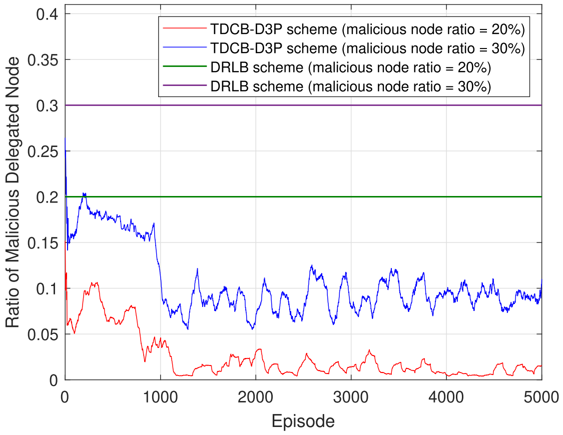

FIGURE 8. Malicious node ratio according to the episodes.

represents the blockchain schemes in which the entire blockchain nodes participate in the PBFT consensus process. The sharded blockchain scheme refers to blockchain schemes such as Zilliqa [8] in which blockchain nodes are divided into shards, and nodes in each shard proceed with a PBFT-based consensus process [25]. The total computing cost is measured as the sum of the computing costs consumed by blockchain nodes during the consensus process. As blockchain nodes exchange blocks and messages in the consensus process, computing costs are required for the signature verification and MAC operation. Since the PBFT algorithm has a message complexity of $O\left(n^2\right)$, the total computing cost increases exponentially as the number of nodes increases. The PBFT algorithm consumes computing costs of about 41 GHz cycle when the number of nodes is 100, and about 644 GHz cycle when the number of nodes is 400. The sharded blockchain scheme first proceeds with consensus within each shard, and then proceeds with a final consensus that verifies the intra-shard consensus. In the intra-shard consensus, nodes can consume a lesser computing cost, but overall, the total computing cost is slightly less compared to the PBFT scheme because the blockchain consensus process is executed in two-phases. But, in the TDCB-D3P scheme, fewer nodes participate in the consensus process than PBFT and sharded blockchain schemes through delegation. When the number of nodes is 400 and the delegation ratio φ is 0.5, TDCB-D3P requires a computing cost of about 162 GHz, which is only about 25.2% of the PBFT scheme and 28.9% of the sharded blockchain scheme. As presented, the TDCB-D3P scheme consumes fewer computing resources and reduces the message complexity of the consensus process, making it more suitable for larger blockchains where scalability is necessary.

B. SECURITY ANALYSIS

Fig. 8 shows the change in the ratio of malicious nodes participating in the consensus process according to episodes under situations where malicious nodes performing NMA are injected into the blockchain network. Simulations are performed for two blockchain network situations with a 20% malicious node ratio and a 30% malicious node ratio, and the consensus process for block generation proceeds successively according to the episode (consensus round). The DRLB scheme where all nodes participate in the consensus process shows the constant malicious node ratio for all consensus rounds. In contrast, in the TDCB-D3P scheme, the ratio of malicious nodes participating in the consensus process changes because some representative nodes participate in the consensus process through trust-based delegation. At the beginning of the consensus episodes, the TDCB-D3P scheme has a malicious delegated node ratio similar to the overall malicious node ratio. However, as the consensus episodes progress, the trust values of the blockchain nodes are evaluated and accumulated through the trust system. In addition, blockchain parameters according to trust values and estimated malicious delegated node ratios are trained and optimized by the DRL agent. The ratio of malicious delegated nodes gradually decreases as the episodes progress, and the performance converges after about 1,000 episodes. In the blockchain network with 20% malicious nodes, the TDCB-D3P shows an average 1% ratio of malicious delegated nodes. In addition, in the blockchain network where 30% malicious nodes exist, the TDCB-D3P shows an average 9% ratio of malicious delegated nodes. This shows that the TDCB-D3P scheme (which combines trust-based delegated consensus and DRL technology) can significantly reduce the influence of malicious nodes in the consensus process, resulting in a superior security performance.

Fig. 9 shows the simulation results for NMA and CRA according to the ratio of malicious nodes from 0% to 40%. Since the difference between NMA and CRA is to counterfeit the delegated consensus result, both attack types apply equally to DRLB, which does not use a trust system. The ratio of malicious delegated nodes indicated by the dashed lines represents the proportion of malicious nodes in the delegated node set. In the DRLB scheme, all malicious nodes take part in the consensus process. Thus, as the ratio of malicious nodes increases, the possibility that an invalid block of a malicious node may be selected increases. Such invalid blocks will be voted against (i.e., 'Nay') by the honest nodes, and the probability of arriving at a consensus for block chaining will decrease, resulting in the TPS performance dropping sharply.

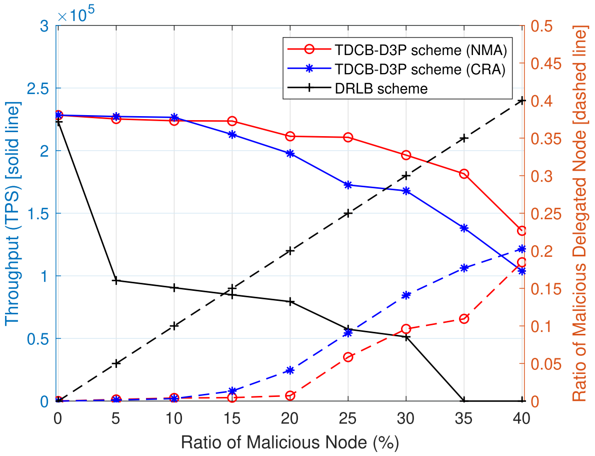

FIGURE 9. TPS performance according to the ratio of malicious node.

However, since the TDCB-D3P scheme can exclude some malicious nodes by performing a trust-based delegation, the results show that the ratio of malicious delegated nodes is consistently lower than the ratio of malicious nodes and thus a higher TPS can be maintained compared to the DRLB scheme. When the malicious node ratio reaches up to 30%, the performance of the DRLB scheme drops to the level of 23%, but the proposed TDCB-D3P scheme is lowered to the level of 86% and 74% for NMA and CRA, respectively. CRA degrades the performance more than NMA because it interferes with the trust evaluation system. In addition, when the ratio of malicious nodes is more than 33%, in the case of DRLB, the TPS is always 0 due to the security constraints. However, the TDCB-D3P scheme keeps the ratio of malicious delegated nodes below 33% and TPS above 100,000 through trust-based delegation. In the TDCB-D3P scheme, when the ratio of malicious nodes is 40%, the ratio of malicious delegated nodes for NMA and CRA is about 18% and 20%, respectively.

Fig. 10 shows the TPS changes according to the average transaction size in presence of malicious nodes. The transaction size varies depending on the type of transactions that are created and recorded in the blockchain-enabled IoT network. Fig. 10(a) and Fig. 10(b) present the simulation results when the malicious node ratio is 20% and 30%, respectively. The simulation results show that as the average transaction size increases, the TPS decreases in all blockchain schemes. This is because when the transaction size increases, the number of transactions that can be stored in one block decreases. As shown in the simulation results of Fig. 9, when there are malicious nodes in the blockchain network, the TPS of the TDCB-D3P scheme is always higher than that of the DRLB scheme under the same conditions. In addition, the adversarial model of CRA shows a lower TPS than NMA. When the average transaction size is 500 Bytes, the TPS difference between the TDCB-D3P scheme and the DRLB scheme is the smallest because the maximum possible TPS is low due to the small number of transactions that can be contained in one block. However, in the case of a small transaction size of 100 Bytes, when the malicious ratio is 20% and 30%, the TDCB-D3P scheme shows a TPS performance of about 3.01 times and 4.43 times higher than the DRLB scheme, respectively. This is because the performance of DRLB is greatly reduced due to the cumulative consensus failure caused by the malicious nodes, but the TDCB-D3P scheme prevents malicious nodes from interfering and generates blocks of the maximum size.

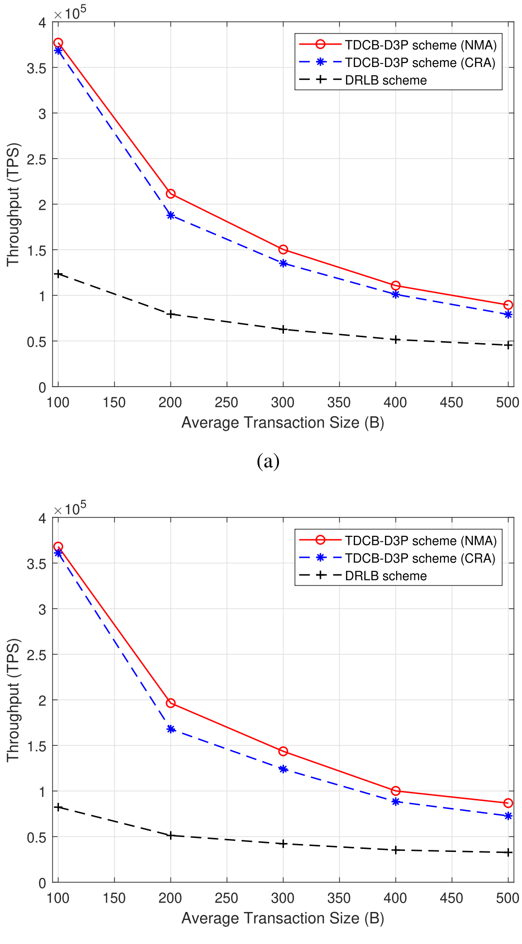

FIGURE 10. TPS performance according to the average transaction size (a) when the malicious node ratio is 20% (b) when the malicious node ratio is 30%.

Fig. 11 shows the security performance in reference to the ratio of malicious nodes. The performance of the proposed TDCB-D3P scheme is compared to the following schemes.

a) PBFT scheme: Does not have a trust system and does not conduct delegation. Thus, all blockchain nodes participate in the consensus process [17].
b) Delegated PBFT (DPBFT) scheme: Delegated PBFT consensus is conducted without a trust system. Since there is no security constraint, the delegation ratio is set randomly, and randomly delegated nodes conduct the PBFT consensus.

The consensus success probability (CSP) represents the probability that a valid block is presented from a block producer and is approved through the consensus process and connected to the chain. Malicious block producers present

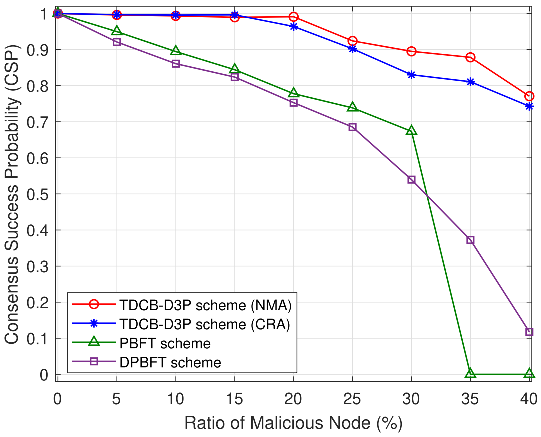

FIGURE 11. **Security performance according to the ratio of malicious node.**

invalid blocks and malicious validators prevent valid blocks from being approved by the consensus process, so the CSP is an indicator of a security level indicating the influence of honest nodes in the consensus process. In the PBFT and DPBFT schemes, as the number of malicious nodes increases, the CSP decreases proportionally because the probability of malicious nodes being selected as block producers increases. In addition, in the DPBFT scheme that uses random delegation (i.e., does not use any trust system), some randomly selected nodes take part in the consensus process. Therefore, even if the ratio of malicious nodes is less than 33%, the malicious nodes may occupy more than 33% of the delegated consensus process and prevent a valid block from being connected to the blockchain, thus resulting in the lowest CSP performance. But the proposed TDCB-D3P scheme evaluates the trust of the blockchain nodes and selects a delegation ratio that does not violate the security constraint, and delegates blockchain nodes to the consensus process based on the trust value. The trust system increases the probability of honest nodes being selected as block producers that present valid blocks, and increases the ratio of honest nodes in delegated node sets, resulting in the highest CSP performance. When the malicious node ratio is 20%, the CSP for NMA and CRA is about 0.99 and 0.96, respectively. Even when the malicious node ratio is 40%, the proposed TDCB-D3P maintains a CSP performance above approximately 0.7, whereas the DPBFT scheme shows a sharp CSP drop to approximately 0.12.

VII. CONCLUSION

This paper proposes a TDCB-D3P framework that integrates the delegated consensus method with DRL to maximize the blockchain throughput and security performance in blockchain-enabled IoT networks. The TD-PBFT blockchain consensus algorithm was developed to cope with malicious attacks more effectively and improve the scalability. The scheme delegates nodes using a trust system that evaluates the reliability of blockchain nodes participating in the consensus process. In the proposed consensus algorithm, blockchain performance metrics are analyzed, and constraints are established to consider the blockchain trilemma properties such as latency, decentralization, and security. The DRL agent of the TDCB-D3P framework reflects the state of the dynamic blockchain network and selects the optimal actions that maximize the TPS while satisfying the constraints. The simulation results show that the proposed TDCB-D3P model can provide a higher TPS throughput as well as enhanced scalability and security performance against malicious attack scenarios compared to the existing blockchain schemes. A study on how to manage trust in a more decentral method by having multiple DRL agents will be considered in future work.

REFERENCES

[1] Z. Li, J. Kang, R. Yu, D. Ye, Q. Deng, and Y. Zhang, "Consortium blockchain for secure energy trading in industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3690–3700, Aug. 2018.

[2] J. Wan, J. Li, M. Imran, and D. Li, "A blockchain-based solution for enhancing security and privacy in smart factory," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3652–3660, Jun. 2019.

[3] H. Yao, T. Mai, J. Wang, Z. Ji, C. Jiang, and Y. Qian, "Resource trading in blockchain-based industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3602–3609, Jun. 2019.

[4] *Bitcoin Daily Transactions*. Accessed: Jul. 26, 2021. [Online]. Available: https://www.blockchain.com/explorer?view=btc_txperday

[5] *Ethereum Project*. Accessed: Jul. 26, 2021. [Online]. Available: https://www.ethereum.org/

[6] U. Mir, "Bitcoin and its energy usage: Existing approaches, important opinions, current trends, and future challenges," *KSII Trans. Internet Inf. Syst.*, vol. 14, no. 8, pp. 3243–3256, Aug. 2020.

[7] I. Grigg. (Jul. 2017). *EOS an Introduction*. [Online]. Available: https://eos.io/documents/EOS-An-Introduction.pdf

[8] ZILLIQA. (Aug. 2017). *The Zilliqa Technical Whitepaper V0.1*. [Online]. Available: https://docs.zilliqa.com/whitepaper.pdf

[9] R. Shrestha and S. Y. Nam, "Regional blockchain for vehicular networks to prevent 51% attacks," *IEEE Access*, vol. 7, pp. 95033–95045, 2019.

[10] Ethereum. *Ethereum/Sharding*. Accessed: Jul. 26, 2021. [Online]. Available: https://github.com/ethereum/sharding/blob/develop/docs/doc.md

[11] A. P. Singh, N. R. Pradhan, A. K. Luhach, S. Agnihotri, N. Z. Jhanjhi, S. Verma, Kavita, U. Ghosh, and D. S. Roy, "A novel patient-centric architectural framework for blockchain-enabled healthcare applications," *IEEE Trans. Ind. Informat.*, vol. 17, no. 8, pp. 5779–5789, Aug. 2021.

[12] M. Shen, H. Liu, L. Zhu, K. Xu, H. Yu, and X. Du, "Blockchain-assisted secure device authentication for cross-domain industrial IoT," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 942–954, May 2020.

[13] G. S. Aujla and A. Jindal, "A decoupled blockchain approach for edge-envisioned IoT-based healthcare monitoring," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 2, pp. 491–499, Feb. 2021.

[14] T. Saba, K. Haseeb, A. A. Shah, A. Rehman, U. Tariq, and Z. Mehmood, "A machine-learning-based approach for autonomous IoT security," *IT Prof.*, vol. 23, no. 3, pp. 69–75, May 2021.

[15] K. Haseeb, T. Saba, A. Rehman, Z. Ahmed, H. H. Song, and H. H. Wang, "Trust management with fault-tolerant supervised routing for smart cities using Internet of Things," *IEEE Internet Things J.*, vol. 9, no. 22, pp. 22608–22617, Nov. 2022, doi: 10.1109/JIOT.2022.3184632.

[16] S. Nakamoto. (2009). *Bitcoin: A Peer-to-Peer Electronic Cash System*. [Online]. Available: http://www.bitcoin.org/bitcoin.pdf

[17] M. Castro and B. Liskov, "Practical Byzantine fault tolerance," in *Proc. 3rd Symp. Oper. Syst. Design Implement.*, New Orleans, LA, USA, Feb. 1999, pp. 173–186.

[18] H. N. Abishu, A. M. Seid, Y. H. Yacob, T. Ayall, G. Sun, and G. Liu, "Consensus mechanism for blockchain-enabled vehicle-to-vehicle energy trading in the internet of electric vehicles," *IEEE Trans. Veh. Technol.*, vol. 71, no. 1, pp. 946–960, Jan. 2022.

[19] Y. Li, C. Chen, N. Liu, H. Huang, Z. Zheng, and Q. Yan, "A blockchain-based decentralized federated learning framework with committee consensus," *IEEE Netw.*, vol. 35, no. 1, pp. 234–241, Jan. 2021.

[20] E. Elrom, "NEO blockchain and smart contracts," in *The Blockchain Developer*. Berkeley, CA, USA: Apress, Jul. 2019, pp. 257–298.

[21] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari with deep reinforcement learning," in *Proc. NIPS Deep Learn. Workshop*, Carson City, NV, USA, 2013, pp. 1–9.

[22] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Proc. AAAI*, Phoenix, AZ, USA, vol. 2, Feb. 2016, pp. 2094–2100.

[23] C. Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, and N. De Freitas, "Dueling network architecture for deep reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, New York, NY, USA, Jun. 2016, pp. 1995–2003.

[24] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Performance optimization for blockchain-enabled industrial Internet of Things (IIoT) systems: A deep reinforcement learning approach," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3559–3570, Jun. 2019.

[25] J. Yun, Y. Goh, and J.-M. Chung, "DQN-based optimization framework for secure sharded blockchain systems," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 708–722, Jan. 2021.

[26] C. H. Liu, Q. Lin, and S. Wen, "Blockchain-enabled data collection and sharing for industrial IoT with deep reinforcement learning," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3516–3526, Jun. 2019.

[27] L. Yang, M. Li, P. Si, R. Yang, E. Sun, and Y. Zhang, "Energy-efficient resource allocation for blockchain-enabled industrial Internet of Things with deep reinforcement learning," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2318–2329, Feb. 2021.

[28] Y. Dai, D. Xu, K. Zhang, S. Maharjan, and Y. Zhang, "Deep reinforcement learning and permissioned blockchain for content caching in vehicular edge computing and networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4312–4324, Apr. 2020.

[29] Y. He, Y. Wang, C. Qiu, Q. Lin, J. Li, and Z. Ming, "Blockchain-based edge computing resource allocation in IoT: A deep reinforcement learning approach," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2226–2237, Feb. 2021.

[30] J. Feng, F. R. Yu, Q. Pei, X. Chu, J. Du, and L. Zhu, "Cooperative computation offloading and resource allocation for blockchain-enabled mobile-edge computing: A deep reinforcement learning approach," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6214–6228, Jul. 2020.

[31] X. Chen, J.-H. Cho, and S. Zhu, "GlobalTrust: An attack-resilient reputation system for tactical networks," in *Proc. 11th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, Singapore, Jun./Jul. 2014, pp. 275–283.

[32] A. Clement, E. Wong, L. Alvisi, and M. Dahlin, "Making Byzantine fault tolerant systems tolerate Byzantine faults," in *Proc. 6th USENIX Symp. Netw. Syst. Des. Implement.*, Boston, MA, USA, Apr. 2009, pp. 153–168.

[33] J. Yun, Y. Goh, and J.-M. Chung, "Trust-based shard distribution scheme for fault-tolerant shard blockchain networks," *IEEE Access*, vol. 7, pp. 135164–135175, 2019.

[34] A. Singh, T. Das, P. Maniatis, P. Druschel, and T. Roscoe, "BFT protocols under fire," in *Proc. 5th USENIX Symp. Netw. Syst. Des. Implement.*, San Francisco, CA, USA, Jan. 2008, pp. 189–204.

[35] S. P. Gochhayat, S. Shetty, R. Mukkamala, P. Foytik, G. A. Kamhoua, and L. Njilla, "Measuring decentrality in blockchain based systems," *IEEE Access*, vol. 8, pp. 178372–178390, 2020.

[36] L. Ceriani and P. Verme, "The origins of the Gini index: Extracts from variabilità e mutabilità (1912) by corrado Gini," *J. Econ. Inequality*, vol. 10, no. 3, pp. 421–443, Sep. 2012.

[37] D. Seng, J. Zhang, and X. Shi, "Visual analysis of deep Q-network," *KSII Trans. Internet Inf. Syst.*, vol. 15, no. 3, pp. 853–873, Mar. 2021.

[38] E. T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," in *Proc. Int. Conf. Learn. Represent.*, San Juan, Puerto Rico, Feb. 2016, pp. 1–21.

[39] *Blockchain Confirmation*. Accessed: Mar. 16, 2018. [Online]. Available: https://en.bitcoin.it/wiki/Confirmation

YUNYEONG GOH received the B.S. degree in electrical and electronic engineering from Yonsei University, Seoul, Republic of Korea, in 2019, where he is currently pursuing the combined M.S. and Ph.D. degrees in electrical and electronic engineering. He is also a Researcher of the Communications and Networking Laboratory (CNL), Yonsei University. His current research interests include blockchain, deep reinforcement learning, XR, AI, trust networks, and security systems.

JUSIK YUN received the B.S. degree in electrical and electronic engineering and the combined M.S. and Ph.D. degrees in electrical and electronic engineering from Yonsei University, Seoul, Republic of Korea, in 2016. He is currently a Senior Engineer at Samsung Electronics. His current research interests include deep reinforcement learning, 5G mobile edge computing, network intelligence, trust management network security systems, UWB, and blockchain.

DONGJUN JUNG received the B.S. degree in electrical and electronic engineering from Yonsei University, Seoul, Republic of Korea, in 2019, where he is currently pursuing the combined M.S. and Ph.D. degrees in electrical and electronic engineering. He is also a Researcher of the CNL, Yonsei University. His current research interests include deep reinforcement learning, MECs, the IoT, 5G mobile systems, augmented reality (AR) systems, XR, military networks, game theory, and deep learning-based optimization.

JONG-MOON CHUNG (Senior Member, IEEE) received the B.S. and M.S. degrees in electronic engineering from Yonsei University and the Ph.D. degree in electrical engineering from The Pennsylvania State University. From 1997 to 1999, he was an Assistant Professor and an Instructor at the Department of Electrical Engineering, The Pennsylvania State University. From 2000 to 2005, he was a Tenured Associate Professor with the School of Electrical and Computer Engineering, Oklahoma State University (OSU). Since 2005, he has been a Professor with the School of Electrical and Electronic Engineering, Yonsei University, where he is currently an Associate Dean of the College of Engineering and a Professor with the Department of Emergency Medicine, College of Medicine, Yonsei University. In 2019 and 2021, he received the Minister Award from the Ministry of the Interior and Safety of the Republic of Korea. In 2008, 2018, and 2019, he received the Outstanding Accomplishment Faculty Awards, and in 2007, 2009, 2014, 2019, and 2021, he received the Outstanding Teaching Awards from Yonsei University. In 2012, he received the Republic of Korea Government's Defense Acquisition Program Administration Award. As a Tenured Associate Professor at OSU, in 2005, he received the Regents Distinguished Research Award and the Halliburton Outstanding Young Faculty Award. He received the Technology Innovator Award and the Distinguished Faculty Award from OSU, in 2003 and 2004, respectively, and the First Place Outstanding Paper Award at the IEEE EIT 2000 Conference held in Chicago, USA, in 2000. He is also a Pledge Book Award Winning Member of the Eta Kappa Nu (HKN) Epsilon Chapter. He has served as the General Co-Chair for IEEE ICCE 2022, and was the General Chair of IEEE ICCE-Asia 2020 and IEEE MWSCAS 2011. He also serves as the Vice President for the IEEE Consumer Technology Society and the IEEE Product Safety Engineering Society. He is a Senior Editor of the IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, a Section Editor of the *ETRI Journal* (Wiley), and a Co-Editor-in-Chief of the *KSII Transactions on Internet and Information Systems*.