Cloud-Edge-End Collaborative Computing-Enabled Intelligent Sharding Blockchain for Industrial IoT Based on PPO Approach

Xin Xiong, Meng Li, *Senior Member, IEEE*, F. Richard Yu, *Fellow, IEEE*, Haijun Zhang, *Fellow, IEEE*, Kan Wang, *Member, IEEE*, and Pengbo Si, *Senior Member, IEEE*

***Abstract*—The security and reliability risks of industrial data have constrained the advancement of the Industrial Internet of Things (IIoT). Although blockchain can protect the security and reliability of industrial data through hash verification mechanisms, there are numerous challenges in the existing blockchain-enabled IIoT systems, such as the trilemma of scalability, decentralization and security, high computational power consumption of consensus protocols and limited computational resources of industrial devices. To address these problems, an intelligent sharding blockchain-enabled IIoT framework is proposed, in which the intelligent sharding based on the reputation mechanism and the adaptive switching for multi-consensus protocols are utilized to enhance the decentralization, security and scalability of blockchain. Considering higher requirement of computational power of the sharding blockchain, a cloud-edge-end collaborative computing framework is introduced, in which the parallel computational offloading and the Terahertz communication technology are utilized to enhance the cooperation of the cloud-edge-end networks. Furthermore, due to the highly dynamic nature of industrial devices and industrial data, we consider and design the optimization problem as a Markov decision process (MDP), which is solved via the Proximal Policy Optimization (PPO) algorithm. Simulation results show that our proposed scheme can minimize total delay and maximize transaction throughput while guaranteeing the safety as well as decentralization of blockchain-enabled IIoT systems.**

***Index Terms*—Industrial Internet of Things (IIoT), intelligent sharding blockchain, consensus protocol, cloud-edge-end collaborative computing, Terahertz communication technology, Proximal Policy Optimization (PPO).**

Received 27 October 2024; revised 17 February 2025; accepted 17 March 2025. Date of publication 25 March 2025; date of current version 6 August 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62371012, Grant 62271324, and Grant 61801379, in part by the Beijing Natural Science Foundation under Grant 4252001, and in part by Shenzhen Science and Technology Program under Grant ZDSYS20220527171400002. Recommended for acceptance by Y. Hu.
(Corresponding author: Meng Li.)

Xin Xiong, Meng Li, and Pengbo Si are with the School of Information Science and Technology, Beijing University of Technology, Beijing 100124, China (e-mail: xiongx@emails.bjut.edu.cn; limeng720@bjut.edu.cn; sipengbo@bjut.edu.cn).

F. Richard Yu is with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON K1S 5B6, Canada (e-mail: richard.yu@carleton.ca).

Haijun Zhang is with the School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China (e-mail: haijunzhang@ieee.org).

Kan Wang is with the School of Computer Science and Engineering, Xi’an University of Technology, Xi’an 710048, China (e-mail: wangkan@xaut.edu.cn).

Digital Object Identifier 10.1109/TMC.2025.3554568

I. Introduction

NOWADAYS, traditional industry has extended from digitalization to intelligence with the development of the Internet [1], thus promoting the development of the Industrial Internet of Things (IIoT) [2]. The IIoT aims to enable intelligent management and intelligent manufacturing through the comprehensive interconnection of industrial devices and operating environments [3], [4]. However, the open networks can introduce potential security vulnerabilities and reliability concerns to industrial data [5].

Blockchain is a hash proof-based storage structure that incorporates distributed storage, consensus mechanism and encryption algorithm, which allows it to effectively tackle security vulnerabilities and reliability concerns in the IIoT [6]. Moreover, blockchain technology boasts numerous advantages, including decentralization, openness, and immutability, which enable high transparency of industrial data while guaranteeing security through hash encryption [7]. However, the traditional blockchain technology suffers from a severe trilemma of decentralization, security, and scalability [8]. Therefore, traditional blockchain typically consumes abundant computational resources, thus resulting in insufficient throughput and excessive delay in blockchain-enabled IIoT systems [9].

Fortunately, sharding technology can overcome the trilemma by dramatically increasing the scalability of the blockchain without compromising its decentralization and security [10], [11]. First, sharding blockchain enhances the decentralization of blockchain by boosting its scalability, aligning seamlessly with the inherently distributed nature of the IIoT [12]. Second, by dividing the whole giant blockchain network into several smaller blockchain networks, the sharding blockchain is beneficial to effectively reduce the probability of the breakdown of the whole blockchain network caused by the single point failure [13]. Moreover, sharding technology splits the blockchain network into multiple independent sub-chains, each of which can process transactions and consensus protocols in parallel, thereby significantly enhancing the throughput of the IIoT [14]. Recently, a considerable number of scholars have devoted themselves to research in sharding blockchain-enabled IIoT and have proposed several constructive and instructive ideas and methods, which establish solid theoretical foundation for the progress of sharding blockchain-enabled IIoT [15], [16], [17]. However,

1536-1233 © 2025 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

the adaptation of sharding blockchain to dynamic environments of the IIoT are often ignored. Furthermore, to address the significant computational resources consumption for blockchain consensus protocols, researchers have proposed a variety of novel consensus protocols [18], but the existing works ignore the fact that there is no one-size-fits-all protocol considering the ever-changing environments of the IIoT.

In addition, industrial devices typically have limited computational resources, making them incapable of complex verification and computation tasks in the sharding blockchain-enabled IIoT systems [19]. Cloud-edge-end collaborative computing has the advantages of high flexibility and real-time processing capability, making it practical to process massive data tasks in real time [20]. In this case, the processing delay of the sharding blockchain-enabled IIoT can be greatly reduced. Moreover, the cloud-edge-end collaborative computing framework facilitates computational offloading from the cloud layer and task uploading from the device layer, thereby significantly reducing the computational load on industrial devices. Currently, there are numerous studies that utilize the cloud-edge-end collaborative computing to enhance the computational power of blockchain-enabled IIoT systems [21], [22].

Nevertheless, as an integrated application of emerging technologies, there are still many technical challenges that cannot be ignored. First, the application of blockchain technology into the IIoT systems enhances security but introduces limitations in scalability performance. Second, the dynamic nature and delay-sensitivity of the IIoT pose challenges for integrating it with sharding blockchain technology. Last but not least, it is difficult for industrial devices with extremely limited computational resources to execute complex blockchain consensus protocols or handle cross-shard transactions.

According to the issues mentioned above, we propose a cloud-edge-end collaborative computing-enabled intelligent sharding blockchain for the IIoT framework, which aims to jointly optimize total delay and transaction throughput of blockchain-enabled IIoT systems. Moreover, the optimization problem is formulated as a Markov Decision Process (MDP) and solved via the Proximal Policy Optimization (PPO) algorithm. The contributions in the paper are summarized below.

- A blockchain-enabled IIoT framework based on the intelligent sharding is proposed, in which the intelligent sharding blockchain based on the reputation mechanism is utilized to enhance the scalability and ensure the security and decentralization of the blockchain networks.
- A dynamic switching model for multi-blockchain consensus protocols (i.e., PBFT, Zyzzyva, and Quorum) is taken into account to adapt to the ever-changing environments of the IIoT.
- A cloud-edge-end collaborative framework is introduced to provide arithmetic support and relieve the computational load of devices, where the parallel offloading and the Terahertz communication technology are applied to enhance the cooperation of the cloud-edge-end networks.
- The optimization problem is expressed as an MDP and solved by the PPO algorithm to minimize total delay and maximize transaction throughput. Moreover, simulation results indicate that our proposed scheme surpasses other solutions.

The rest of the paper is organized into six parts. Section II gives a review of related works about sharding blockchain-enabled IIoT, consensus protocols for blockchain, cloud-edge-end collaborative computing for IIoT, and intelligent optimization methods for IIoT. In Section III, the proposed framework is formulated as mathematical models, which include network model, sharding model, consensus model, transmission model and computation model. Furthermore, Section IV presents the optimization problem obtained by formulating the proposed framework as an MDP. In Section V, the optimal policy is acquired by the application of the PPO. In addition, in Section VI, the superiority of our proposed scheme is evaluated and simulation results are discussed in detail. Eventually, in Section VII, the summary and the future works are presented, respectively.

II. Related Works

In this section, several relevant research works about the sharding blockchain-enabled IIoT, consensus protocols for blockchain, cloud-edge-end collaborative computing and intelligent optimization algorithms are reviewed.

A. Sharding Blockchain-Enabled IIoT

With the advancement and widespread application of blockchain technology, sharding has risen as a notable research field within blockchain. Recently, numerous scholars have devised a diverse array of sharding algorithms and extensively applied them to the IIoT [23]. In [24], Yang et al. proposed an overlapping self-organized sharding scheme for blockchain-enabled IIoT systems. This scheme leverages deep reinforcement learning (DRL) to facilitate self-organized sharding based on local blockchain information, thereby obtain the optimal throughput. Jiang et al. introduced a distributed digital twin-driven IIoT system in [25], which achieves efficient resource scheduling for IIoT systems by integrating joint learning with directed acyclic graph (DAG) and sharding blockchain. Furthermore, Li et al. [26] designed an access control method for sharding blockchain-enabled IIoT, which effectively addresses the scalability challenges inherent in the access control mechanisms of the IIoT. However, most of existing works have ignored the reliability and adaptability of sharding blockchain in dynamic IIoT environments.

B. Consensus Protocols for Blockchain

An increasingly number of blockchain consensus protocols has emerged and found widespread application in the IIoT with the progression of blockchain. The mainstream blockchain consensus protocols can be categorized into three primary types, namely, Proof of Work (PoW), Proof of Stake (PoS), and Byzantine Fault Tolerant (BFT) [27], [28]. In [29], Zhang et al. presented a blockchain sharding scheme that leverages certifiable stochastic functions and reputation voting, which enhancing the

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

reliability and security of the IIoT while reducing the communication overhead associated with blockchain operations. Chen et al. [30] devised a blockchain protocol for heterogeneous IIoT environments, utilizing a double-DAG structure to safeguard against cyber attacks. Meanwhile, in [31], Aljuhani et al. designed an IIoT communication framework based on private blockchain technology, employing the PoS mechanism for transaction verification. However, the dynamic switching mechanism among multiple consensus protocols which is essential to accommodate the dynamic nature of the IIoT has usually been ignored.

C. Cloud-Edge-End Collaborative Computing for IIoT

Cloud-edge-end collaborative computing is designed to facilitate flexible data sharing and unified operation of business applications, as well as meet the diverse and reliable requirements of distributed infrastructure resources [32]. In [33], Hua et al. concentrated on developing a collaborative framework for cloud-edge-end computing in the IIoT scenarios, taking into full consideration user mobility to achieve energy savings. Kim et al. [34] introduced a dynamic task scheduling framework that leverages collaborative policy learning to enhance the efficiency of task scheduling in the IoT environments. Furthermore, Gan et al. [35] presented a resource optimization framework for cloud-edge-end IoT, based on optimal transport and federated actor-critic methods, which achieves a joint optimization of delay and energy consumption. However, all of the aforementioned studies have ignored the optimization of cloud-edge computational offloading links.

D. Intelligent Optimization Methods for IIoT

Intelligent optimization constitutes a category of algorithms that tackle specific problems by emulating natural processes or human behavior. DRL is a notable example of Intelligent optimization methods, which integrates the perceptual capability of deep learning with the decision-making ability of reinforcement learning, thereby constituting a novel type of intelligent optimization algorithm. In recent years, a wide variety of DRL algorithms have emerged and applied to the IIoT [36], [37], including Deep Q-Networks (DQN), Deep Deterministic Policy Gradient (DDPG), and Asynchronous Advantage Actor-Critic (A3C), etc. Among these algorithms, Proximal Policy Optimization (PPO) is one of the most popular DRL algorithms [38], which aims to stabilize the training process by constraining the divergence between the old and new policies. To enhance topology robustness, An et al. [39] proposed a method for generating robust topologies for industrial devices of the IIoT based on PPO algorithm. This method optimizes the topology of device nodes while accounting for the constraints and limitations inherent in the deployment scenarios. Considering the energy demands of devices that continuously transmit sensory data, Lee et al. [40] introduced an autonomous control framework for transmission cycles utilizing the PPO algorithm, which adaptively regulates the transmission cycle of sensors. In [41], Mensah et al. designed an adaptive mechanism that leverages PPO and Advantage Actor-Critic (A2C) to optimize the block selection process in blockchain, achieving an adaptive and dynamic solution. However, none of these existing works have used the PPO algorithm to optimize the configuration of blockchain sharding and consensus protocols in sharding blockchain-enabled IIoT scenarios.

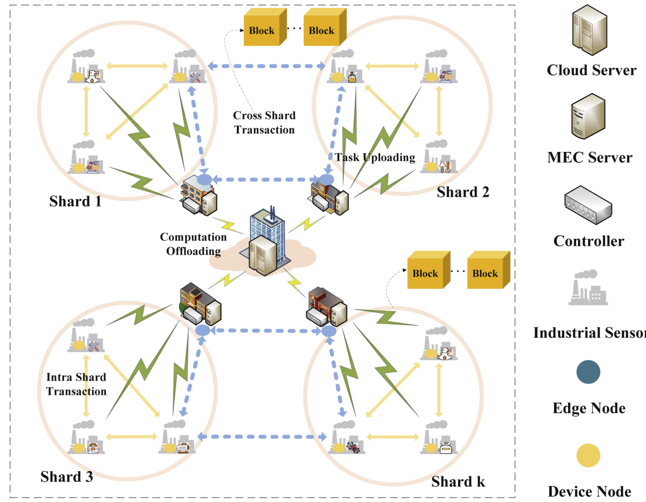


Fig. 1. System model.

In recent years, numerous optimization schemes for blockchain-enabled IIoT have been proposed. However, the majority of them can only ensure relatively adequate performance in terms of decentralization, security, and scalability, falling short of guaranteeing overall stable and superior performance across all these aspects. Consequently, the existing limitations prompt us to adopt a holistic approach to balance the performance triangle. In this paper, we introduce a reputation mechanism-based intelligent sharding strategy, an adaptive consensus protocol switching model, a cloud-edge-end collaborative computing framework, a parallel computational offloading approach, and a Terahertz communication technology to comprehensively optimize the blockchain-enabled IIoT system.

III. System Model

The proposed framework is formulated as various mathematical models in this section.

A. Network Model

The industrial network is structured as a cloud-edge-end collaborative computing framework to improve computational efficiency and relieve the computational load of industrial devices, as shown in Fig. 1.

In the cloud layer, the cloud server with powerful computing power is responsible for handling complex computational tasks and decision-making tasks involving intelligent algorithms. Therefore, the computational resources of the cloud server are considered sufficient.

There are M edge intelligent servers with limited computational resources as well as a controller in the edge layer. Considering the rational allocation of computational resources and the collaboration among the cloud, edge, and device layers, the edge intelligent servers play a dual role. On the one

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

hand, as core nodes of the blockchain network, edge intelligent servers are responsible for complex cross-shard consensus tasks of blockchain to alleviate the computational load of industrial devices. On the other hand, edge intelligent servers are also responsible for processing data computing tasks in cooperation with the cloud server. $\mathcal{M} = \{1, 2, \ldots, M\}$ indicates the collection of edge intelligent servers, and the remaining computational resources of edge intelligent servers are represented by the set $\mathcal{H} = \{h_1, h_2, \ldots, h_M\}$. In addition, the controller is responsible for assisting the cloud server to handel management and decision-making tasks.

In the device layer, there are massive amounts of smart devices, which are capable of collecting industrial data such as environmental conditions. Since industrial devices typically have extremely limited computational resources, they are only tasked with performing the simpler intra-shard consensus within the blockchain. Let $\mathcal{N} = \{1, 2, \ldots, N\}$ indicate the collection of device nodes.

The blockchain system is made up of edge intelligent servers as well as industrial devices. According to different transaction types, which are cross-shard transaction (CST) and intra-shard transaction (IST), data tasks of the IIoT can be validated by edge intelligent server nodes or device nodes and be recorded as transactions by the blockchain system. In addition, all device nodes can be divided into multiple blockchain shards based on the reputation level to handle ISTs independently. The collection of blockchain shards is $\mathcal{K} = \{1, 2, \ldots, K\}$. It should be noted that edge intelligent servers are utilized to deal with CSTs and not participate in the blockchain sharding.

B. Sharding Model

To handle the IIoT data tasks efficiently, the validators should follow the steps below: 1) Be divided into several blockchain shards. 2) Verify the transactions and package them as blocks. 3) Attach the blocks to the blockchain.

As everyone knows, sharding technology in the blockchain usually reduces the cost of malicious behavior by malicious nodes. Therefore, a sharding algorithm based on the reputation mechanism is designed, which can improve the robustness of each shard in an efficient way by evaluating the reputation of device nodes and implementing the blockchain sharding according to the node reputation score. In this case, all N device nodes with various reputation levels will be divided into K shards evenly, which prevents Byzantine nodes from clustering into one shard and leading to shard paralysis. The sharding strategy should satisfy the conditions below.

a) *All the device nodes with same reputation level will be divided into several shards evenly, and the primary nodes within each shard will be elected based on their reputation score.*
b) *Each shard has similar total reputation scores and similar ratio of honest and dishonest nodes.*
c) *Nodes with extremely low reputation scores are not allowed to participate in the verification procedure.*

Thereby, let $\mathcal{R}$ denote the reputation score of device nodes, which is formulated as a finite-state Markov procedure. Besides, we split $\mathcal{R}$ into r values, which is expressed by

$$\mathcal{X} = \{x_1, x_2, \ldots, x_r\}. \tag{1}$$

Besides, $\mathcal{R}$ evolves to next state on the basis of the state transition probability, which can be represented by

$$P_{x_i,x_j} = Pr[\mathcal{R}(t+1) = x_j \,|\mathcal{R}(t) = x_i], x_i, x_j \in \mathcal{X}. \tag{2}$$

Thus, the $r \times r$ state transition matrix of $\mathcal{R}$ is expressed as

$$\mathbb{R} = [P_{x_i,x_j}]_{r\times r}. \tag{3}$$

Before each round of transactions, a sharding permission signal is sent to the blockchain system by the intelligent agent, which is a control network composed of the cloud server and the edge controller. Then, all device nodes will be allocated to specified number of blockchain shards. It should be noted that the blockchain sharding scheme can be reconfigured as the intelligent algorithm completes iterations. During this procedure, the reputation scores of all industrial device nodes, the number of shards as well as the size of each shard are reconfigured. The specific steps are listed as follows.

1) *At first, the intelligent agent obtains the relevant information to determine the number of blockchain shards.*
2) *The reputation levels of all devices are obtained and recorded on the reputation blockchain.*
3) *Each device node will be allocated to one of the smallest shards according to the node reputation level to balance the shard size and maintain the ratio of honest and dishonest nodes within each shard.*
4) *The device nodes that are ranked in the bottom 20% in terms of reputation score will be warned, and those that are ranked in the bottom 5% will be restricted from participating in the validation process. Additionally, the blockchain system will undergo re-sharding after each round of transaction.*

C. Consensus Model

Considering the dynamic nature of the IIoT and the extremely limited computational resources of industrial devices, we have chosen three lightweight blockchain consensus protocols and implemented a flexible, intelligent switching mechanism for them. The details are illustrated as follows.

PBFT: PBFT was proposed by M. Castro et al. in 1999 and improved in 2002 [42], [43]. PBFT improves upon the Byzantine fault tolerant (BFT) by reducing the complexity to polynomial level, and thereby makes the BFT feasible for practical applications such as Hyper-ledger Fabric. In addition, PBFT is highly robust since the consensus process can be accomplished when Byzantine nodes are no more than two thirds. However, as shown in Fig. 2, PBFT consumes high delay and computational overhead because of the complex node broadcasting process. Assuming that the number of consensus nodes is N^c, the computing cycle for a signature is χ and for a message authentication code (MAC) is δ, the consensus procedure of the PBFT is listed below.

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

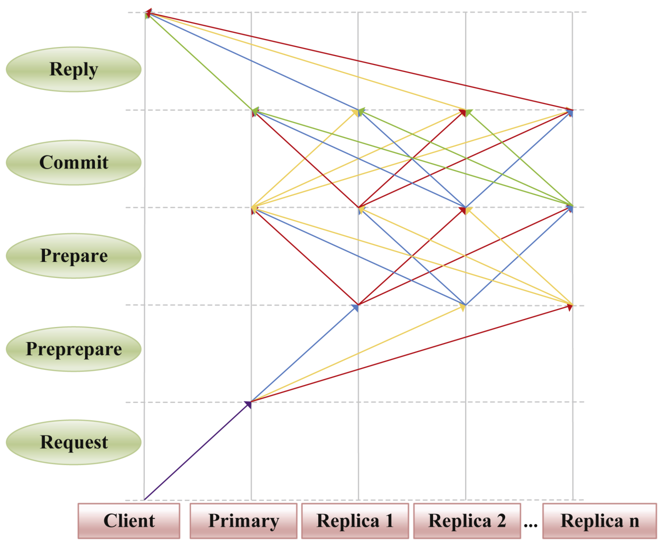


Fig. 2. Verification procedure of PBFT.

a) Request: The client submits a consensus request first, and the transaction data will be packaged into blocks and the signature and MAC will be validated by the primary. Therefore, the computing cycle is expressed by

$$c_1(t) = (\chi + \delta)\frac{\mathcal{B}(t)}{\mathcal{A}(t)}, \tag{4}$$

where $\mathcal{B}(t)$ is the transaction batch size, $\mathcal{A}(t)$ is the average transaction size.

b) Pre-prepare: Initially, a signature and $N^c - 1$ MACs are generated and forwarded to all replicas involved in the block verification. After this, each replica validates the signature and a MAC. Therefore, the computing cycle for the primary is the time taken to generate the signature and MACs and forward them to the replicas, while the computing cycle for replicas is the time taken to validate the signature and the MAC received. As described above, the computing cycle for the primary is shown as

$${c_2}^p(t) = \chi + (N^c - 1)\delta, \tag{5}$$

and the computing cycles for replicas is shown as

$${c_2}^r(t) = (\chi + \delta)\left[1 + \rho\frac{\mathcal{B}(t)}{\mathcal{A}(t)}\right], \tag{6}$$

where ρ is the probability that the transaction is correctly verified.

c) Prepare: The prepare phase involves accomplishing message broadcasting among consensus nodes. Specifically, replica nodes sign for verified blocks and generate $N^c - 1$ MACs for broadcasting to all nodes, including the primary node. It is worthy noting that PBFT can ensure the effectiveness when the number of authenticated nodes is no fewer than 2ζ (where $\zeta = (N^c - 1)/3$). As a result, the computing cycle for the primary is shown as

$${c_3}^p(t) = 2\zeta(\chi + \delta), \tag{7}$$

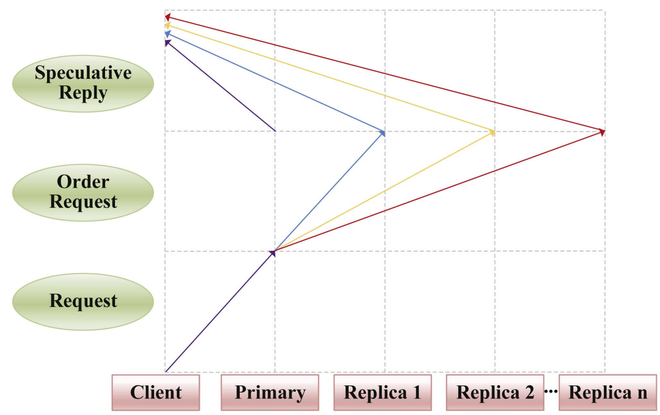


Fig. 3. Verification procedure of Zyzzyva (Fast case).

and the computing cycles for replicas is given by

$${c_3}^r(t) = 2\zeta(\chi + \delta) + \chi + (N^c - 1)\delta. \tag{8}$$

d) Commit: In the commit phase, all nodes are required to swap and verify the block information, specifically, by generating one signature and $N^c - 1$ MACs and validating the received ones. Consequently, the computing cycle for this phase can be given by

$$c_4(t) = 2\zeta(\chi + \delta) + \chi + (N^c - 1)\delta. \tag{9}$$

e) Reply: During the reply phase, consensus nodes reply with verification message to client. Additionally, the validation is deemed valid if client receives more than 2ζ reply messages. At this point, the new block is appended to the blockchain. Consequently, the computing cycle for this phase is calculated by

$$c_5(t) = 2\zeta(\chi + \delta) + (\chi + \delta). \tag{10}$$

As mentioned above, the computing cycle required for the PBFT is calculated by

$$\begin{aligned} C_P(t) = {} & \left[(1+\rho)\frac{\mathcal{B}(t)}{\mathcal{A}(t)} + 6\zeta + 5\right]\chi \\ & + \left[(1+\rho)\frac{\mathcal{B}(t)}{\mathcal{A}(t)} + 15\zeta + 2\right]\delta. \end{aligned} \tag{11}$$

Zyzzyva: The Byzantine fault is widely recognized as the most severe fault in distributed systems. The BFT algorithm provides an effective method for handling Byzantine fault, but the high redundancy verification consumes significant amounts of computational resources and affects the transaction efficiency of blockchain. Speculative BFT mechanisms like Zyzzyva have gained widespread attention because computer systems rarely encounter errors nowadays [44]. Zyzzyva can enhance the blockchain significantly by skipping the invocation of fault-tolerant mechanisms in non-fault cases. Similar to PBFT, Zyzzyva ensures the correctness of transactions when honest nodes are more than 2ζ. Zyzzyva can skip the node broadcasting and directly perform speculative execution when all nodes are honest, as shown in Fig. 3. Otherwise, Zyzzyva will initiate the

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

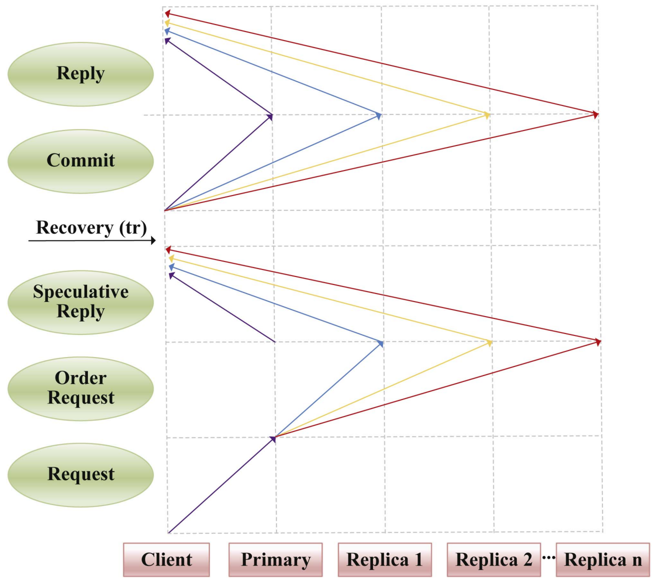


Fig. 4. Verification procedure of Zyzzyva (Two-phase case).

recovery procedure during t_r if there are any malicious nodes as in Fig. 4.

In the fast case, the authentication process is organized into three phases, that is, Request, Order Request and Speculative Reply. Hence, the computing cycle can be represented as

$$\begin{aligned} C_Z^1(t) &= c_1(t) + c_2(t) + c_5(t) \\ &= \left[(1+\rho)\frac{\mathcal{B}(t)}{\mathcal{A}(t)} + 2\zeta + 3\right]\chi \\ &\quad + \left[(1+\rho)\frac{\mathcal{B}(t)}{\mathcal{A}(t)} + 5\zeta + 2\right]\delta. \end{aligned} \tag{12}$$

In the two-phase case, the authentication process is organized into five phases, which are Request, Order Request, Speculative Reply, Commit, and Reply. Therefore, the computing cycle for the two-phase case is represented by

$$\begin{aligned} C_Z^2(t) &= 2c_1(t) + c_2(t) + 2c_5(t) \\ &= \left[(2+\rho)\frac{\mathcal{B}(t)}{\mathcal{A}(t)} + 4\zeta + 4\right]\chi \\ &\quad + \left[(2+\rho)\frac{\mathcal{B}(t)}{\mathcal{A}(t)} + 7\zeta + 3\right]\delta. \end{aligned} \tag{13}$$

Quorum: The Quorum algorithm implements a simple communication mode by applying the pigeonhole principle, specifically, it requires just one message interaction between the client and the replica to fulfill the transaction validation [45]. The consensus procedure of the Quorum consists of two steps, which are request and reply as in Fig. 5. It should be noted that the reply message comprises history digest of replicas. And the panic mechanism will be invoked and the validation process will be terminated if the client receives less than N^c replies. Hence, the computing cycle of the Quorum is represented by

$$\begin{aligned} C_Q(t) &= c_1(t) + c_5(t) \\ &= \left[\frac{\mathcal{B}(t)}{\mathcal{A}(t)} + 2\zeta + 1\right]\chi + \left[\frac{\mathcal{B}(t)}{\mathcal{A}(t)} + 7\zeta + 3\right]\delta. \end{aligned} \tag{14}$$

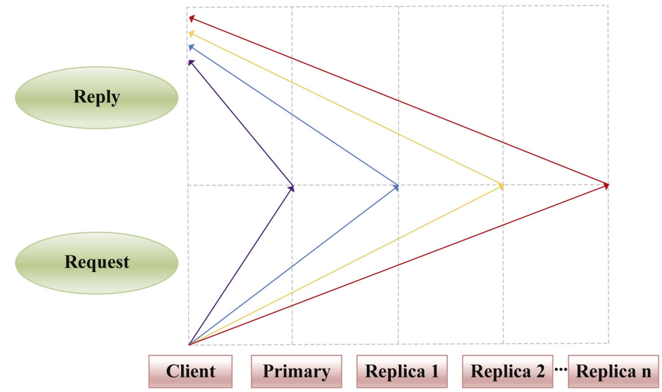


Fig. 5. Verification procedure of Quorum.

Data integrity and network robustness in blockchain require that the fraction of malicious nodes be within the security bounds of a particular consensus protocol (at most one-third malicious validators is tolerated in PBFT, Zyzzyva, and Quorum). As a result, to ensure the safety of the proposed framework, the following equation should be met [46]

$$3N^p + 1 \leq N^c, \tag{15}$$

where N^p is the number of Byzantine nodes, N^c is the number of consensus nodes.

For an IST, consensus nodes are composed of device nodes within one shard, thus $N^c = N/K$. Clearly, the number of shards K is restricted by

$$K \leq \frac{N}{3N^p + 1}. \tag{16}$$

D. Transmission Model

Terahertz communication technology is a wireless communication technology that utilizes electromagnetic waves in the Terahertz frequency band. The Terahertz frequency band is capable of supporting transmission rates over 10 Gbps and provides excellent confidentiality and anti-jamming capability [22]. Hence, the Terahertz communication technology is applied in this paper to further enhance the cloud-edge offloading link.

Assuming that the average distance of the offloading link is ℓ, as the Terahertz electromagnetic wave of frequency ω propagates in the medium, the molecular absorption attenuation resulting from the interaction of the electromagnetic wave with various molecules and atoms in the medium is shown as

$$\Lambda_{abs} = \frac{1}{\tau} = e^{\kappa(\omega)\ell}, \tag{17}$$

or

$$\Lambda_{abs}[\mathrm{dB}] = 10\kappa(\omega)\ell \lg \mathrm{e}, \tag{18}$$

where τ denotes the transmittance, and $\kappa(\omega)$ is the absorption coefficient.

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

Propagation loss is also known as the free space loss, which refers to the loss generated by the electromagnetic wave penetrating the medium. In the Terahertz communication technology, the propagation loss can be given by

$$\Lambda_{spread} = \left[\frac{4\pi\omega\ell}{c}\right]^2, \tag{19}$$

or

$$\Lambda_{spread}[\mathrm{dB}] = 20\lg\left[\frac{4\pi\omega\ell}{\mathrm{c}}\right], \tag{20}$$

where $c = 2.9979 \times 10^8\mathrm{m/s}$ is the speed of light, which refers to the propagation speed of light in the vacuum.

The total Terahertz loss is the sum of the molecular absorption loss and the free space loss, which is denoted as

$$\begin{aligned}\Lambda_{total} &= \Lambda_{abs}\Lambda_{spread}\\ &= e^{\kappa(\omega)\ell}\left[\frac{4\pi\omega\ell}{c}\right]^2,\end{aligned} \tag{21}$$

or

$$\begin{aligned}\Lambda_{total}[\mathrm{dB}] &= \Lambda_{abs}[dB] + \Lambda_{spread}[dB]\\ &= 10\kappa(\omega)\ell\lg e + 20\lg\left[\frac{4\pi\omega\ell}{c}\right].\end{aligned} \tag{22}$$

In the transmission link of Terahertz, the noise temperature of the cloud receiver mainly derives from the molecular absorption noise. Therefore, the equivalent noise temperature generated by molecular absorption is

$$N_{abs}{}^T = N_0{}^T\epsilon, \tag{23}$$

where $N_0{}^T = 296\mathrm{K}$ refers to the reference temperature, $\epsilon = 1 - \tau$ denotes the emissivity of the computation offloading channel, and τ is the transmissivity.

Hence, for a given bandwidth B, the molecular absorbed noise power is calculated as

$$P_n = \int_B N_{abs}{}^T d\omega. \tag{24}$$

Based on the analysis above, the signal to noise ratio can be shown as

$$\beta = \frac{P_t(t)\partial_t(t)\partial_r(t)\Lambda_{total}}{P_n}, \tag{25}$$

where $P_t(t)$ is the transmission power of the edge intelligent servers. Besides, $\partial_t(t)$ and $\partial_r(t)$ refer to transmit and receive antenna gains, respectively.

Consequently, the transfer rate based on the Terahertz communication technology can be calculated by

$$\begin{aligned}\lambda(t) &= B\log_2(1+\beta)\\ &= B\log_2\left(1 + \frac{P_t(t)\partial_t(t)\partial_r(t)\Lambda_{total}}{P_n}\right).\end{aligned} \tag{26}$$

E. Computation Model

In the proposed framework, the edge intelligent servers are burdened with heavy tasks of validation and computation. Therefore, to better allocate tasks to the edge intelligent servers, we need to obtain the computational resources of the edge intelligent servers at any time slot. Under this circumstance, we model the computational resources of edge intelligent servers $\mathcal{H}$ as a Markov process. Besides, $\mathcal{H}$ is split into h values, which can be expressed as

$$\mathcal{Y} = \{y_1, y_2, \ldots, y_h\}. \tag{27}$$

Besides, the computational resources of edge intelligent servers evolve to next state on the basis of the state transition probability, which is given by

$$P_{y_i,y_j} = Pr[\mathcal{H}(t+1) = y_j\,|\mathcal{H}(t) = y_i], y_i, y_j \in \mathcal{Y}. \tag{28}$$

Therefore, the state transition matrix of $\mathcal{H}$ is expressed as

$$\mathbb{H} = [P_{y_i,y_j}]_{h\times h}. \tag{29}$$

It is extremely challenging for edge intelligent servers to perform excessive tasks including cross-shard verifications and computational tasks solely due to their limited computational resources. Therefore, to alleviate the edge burden and enhance the computational efficiency of the system, a parallel cloud-edge offloading strategy is considered. Assuming that the computation offloading ratio is $\mathcal{O} \in [0, 1]$, the computing delay of edge intelligent servers and the cloud server are calculated as

$$t_1{}^m(t) = (1-\mathcal{O})\frac{\mathcal{G}(t)}{\mathcal{F}_m}, \tag{30}$$

and

$$t_1{}^c(t) = \mathcal{O}\left[\frac{\mathcal{G}(t)}{\mathcal{F}_c} + \frac{\mathcal{D}(t)}{\lambda(t)}\right], \tag{31}$$

where $\mathcal{G}(t)$ refers to the computing cycles, $\mathcal{F}_m$ and $\mathcal{F}_c$ denote the frequencies of the edge intelligent servers and the cloud server, respectively. In addition, $\mathcal{D}(t)$ represents data task size, and $\lambda(t)$ refers to the data transfer rate, as given by (26).

Therefore, the total computing delay of parallel computing can be given by the following equation according to the bucket effect

$$t_1(t) = \max\{t_1{}^m(t), t_1{}^c(t)\}. \tag{32}$$

In order to execute data tasks of the IIoT efficiently, the transaction types need to be classified clearly on the basis of blockchain sharding. It is well known that the transaction types in the sharding blockchain-enabled IIoT are categorized approximately into two types, which are IST and CST.

Therefore, the consensus delay for an IST can be shown as

$$t_2{}^d(t) = T^I + \frac{C(t)}{\mathcal{F}_d} + 4t_b, \tag{33}$$

where $C(t)$ is from the set $\{C_P(t), C_Z{}^1(t), C_Z{}^2(t), C_Q(t)\}$. Besides, T^I denotes the block interval, $\mathcal{F}_d$ refers to the frequency of smart device and t_b denotes the broadcast delay among consensus nodes.

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

In addition, the consensus delay for a CST is represented as

$$ {t_2}^m(t) = T^I + \frac{C(t)}{\mathcal{F}_m} + 4t_b. \tag{34} $$

To sum up, the total delay is calculated as

$$ \mathcal{T}(t) = t_1(t) + t_2(t), \tag{35} $$

where $t_1(t)$ represents the computing delay, which is either ${t_1}^m(t)$ or ${t_1}^c(t)$. In addition, $t_2(t)$ is the consensus delay, which hinges on the transaction type of the blockchain and is either ${t_2}^d(t)$ or ${t_2}^m(t)$.

In addition, we evaluate the transaction throughput by calculating the number of completed transactions per unit time. Moreover, since all shards validate for the IIoT data tasks in parallel, the throughput Ξ is positively correlated with the number of blockchain shards K. Specifically, the throughput Ξ is calculated by

$$ \Xi = K \frac{S^B / \mathcal{A}(t)}{T^I}. \tag{36} $$

IV. Problem Formulation

To obtain the optimal strategy, we express the framework as an MDP. In addition, the proposed system model is considered as the environment, and the control network that composed of the cloud server and the edge controller is designated as the intelligent agent.

A. State Space

The state space is a representation of the environmental information and the cornerstone for achieving long-term benefits.

The state space defined in this paper consists of five key variables, namely, task size $\mathcal{D}(t)$, batch size $\mathcal{B}(t)$, transfer rate $\lambda(t)$, computational resources of the edge intelligent servers $\mathcal{H}$ and reputation level of all device nodes $\mathcal{R}$. Therefore, the state space can be represented by

$$ S(t) = \{\mathcal{D}(t), \mathcal{B}(t), \lambda(t), \mathcal{H}, \mathcal{R}\}. \tag{37} $$

B. Action Space

To achieve better long-term rewards, we need to flexibly adjust our action strategy according to environmental changes. Thus, the action space is defined as

$$ A(t) = \{S^B, T^I, \mathcal{O}, C(t), K\}, \tag{38} $$

where $S^B = \{1, 2, \ldots, B\}$ represents the block size, $T^I = \{0.5, 1, \ldots, I\}$ denotes the block interval. In addition, $\mathcal{O} \in [0, 1]$ represents the computation offloading ratio. $C(t)$ signifies the decision factor of the consensus pattern. Furthermore, K stands for the number of shards.

C. Reward Function

It is widely recognized that IIoT is an application that highly sensitive to delay. As a result, delay performance serves as a crucial metric for evaluating IIoT systems. In addition, one of the key advantages of the sharding blockchain over traditional blockchain is the enhanced scalability, which is commonly manifested through transaction throughput. Therefore, it is essential to jointly optimize transaction throughput and delay in the proposed network scenarios. The reward function is defined as a weighted sum of the transaction throughput and the inverse of the total delay, which can be given by

$$ R(t) = \begin{cases} \alpha_1 \Xi + \alpha_2 \frac{1}{\mathcal{T}(t)} \times 10^3, & \text{if C1, C2, C3 are satisfied,} \\ \alpha_1 \Xi + \alpha_2 \frac{1}{\mathcal{T}(t)} \times 10^3 - \nu, & \text{otherwise,} \end{cases} $$

$$ \begin{aligned} s.t. \quad & C1: \mathcal{D}(t) \le S^B, \\ & C2: \mathcal{T}(t) \le T^I, \\ & C3: K \le N/(3N^p + 1), \end{aligned} \tag{39} $$

where $\alpha_1 = \alpha_2 = 0.5$, ν is the penalty index.

V. Problem Solution

In this section, the PPO-based training scheme is specifically designed to handle the formulated framework. The strengths of PPO are fully utilized to optimize policy learning while ensuring efficient adaptation to the highly dynamic characteristics of the IIoT.

A. PPO Algorithm

The PPO is a reinforcement learning algorithm proposed by Open-AI in 2017 and is considered one of the most widely applicable algorithms [40]. Considering the dynamic nature of the IIoT, the PPO algorithm is utilized to optimize long-term rewards for the reasons listed below.

First, the PPO algorithm demonstrates remarkable stability and robustness. More precisely, the PPO algorithm safeguards the stability of policy updates by imposing constraints on the disparity between the new and old policies throughout the training process, thus precluding performance deterioration that may arise from excessively large policy adjustments.

Second, the PPO algorithm boasts benefits including high sample efficiency and remarkable flexibility. Specifically, the PPO algorithm leverages the reuse of experiences for iterative training, enabling the acquisition of an optimal policy with minimal environmental interactions.

Last but not least, the PPO algorithm is highly applicable. Considering the complexity of the proposed system model, which involves both continuous and discrete variables in the state space and action space, thus the applicability of the optimization algorithm is of great importance. Fortunately, the PPO algorithm fully satisfies the requirements. On the one hand, it is well suited for solving reinforcement learning problems, both continuous and discrete variable tasks. On the other hand, the PPO algorithm has been widely applied in various fields, thus demonstrating its high maturity and effectiveness.

B. Algorithmic Flow of PPO

In the proposed system model, the intelligent agent of the PPO algorithm is a control network composed of the cloud server and the edge controller. The input of actor network includes the

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

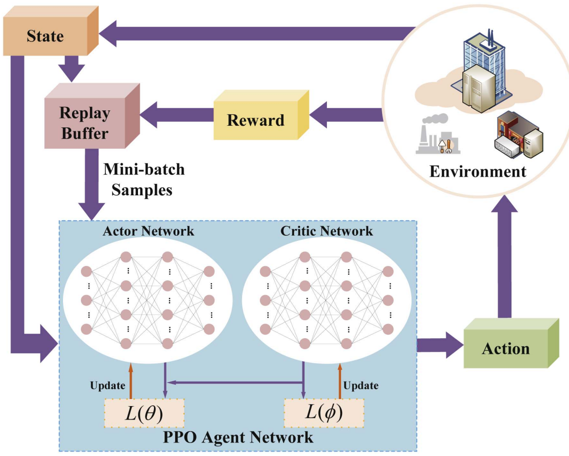


Fig. 6. Training framework of the PPO scheme.

Algorithm 1: PPO-Based Training Scheme.

Initialization:
a) Set the actor parameter θ;
b) Set the critic parameter ϕ;
c) Set the old policy parameter θ_{old}.
for episode $= 1$ to E **do**
  Interact with the environment and run policy π_θ for T timesteps to execute action.
  $\pi_{\theta_{old}} \leftarrow \pi_\theta$.
  Run policy $\pi_{\theta_{old}}$ for T timesteps and get the trajectory.
  Get the reward r_t and the next state S_{t+1}.
  Calculate the approximate value function $V_\phi(s_t)$.
  Perform the generalized advantage estimation function $\hat{A}_{\pi_{\theta_{old}}}$ using $V_\phi(s_t)$.
  Get the clipped surrogate objective function $L(\theta)$ using $\hat{A}_{\pi_{\theta_{old}}}$.
  Get the gradient descent loss function $L(\phi)$.
  for $i = 1$ to I **do**
    Update the actor parameter θ according to (44).
  end for
  for $j = 1$ to J **do**
    Update the critic parameter ϕ according to (45).
  end for
end for

---

system state given by (37). Moreover, the output of actor network is the policy π_θ obeying a normal distribution with $\mu(s_t)$ and $\sigma(s_t)$ as mean and variance, respectively. The intelligent agent samples an initial action from the normally distributed policy π_θ, which is fed into the environment to obtain information such as next state. Moreover, the PPO scheme is performed as in the Fig. 6.

On the basis of the Trust Region Policy Optimization (TRPO) algorithm [47], a clipped surrogate objective function (CSOF) is introduced to PPO to ensure consistency in policy optimization, which is then optimized using a mini-batch stochastic gradient ascent method. The CSOF can be given by

$$L(\theta) = \hat{\mathbb{E}}\left[\min\{r(\theta), \mathrm{clip}[\mathrm{r}(\theta), 1-\varepsilon, 1+\varepsilon]\}\hat{\mathrm{A}}_{\pi_{\theta_{\mathrm{old}}}}\right], \quad (40)$$

where ε denotes a clip index, and $r(\theta)$ represents the ratio of the old policy to the new policy, which is used to ensure the difference between the old and new policies, as shown by

$$r(\theta) = \frac{\pi_\theta}{\pi_{\theta_{old}}}. \quad (41)$$

Moreover, $\hat{A}_{\pi_{\theta_{old}}} \approx r(\theta) + V_\phi(s_{t+1}) - V_\phi(s_t)$ represents the generalized advantage estimation function, where $V_\phi(s_t)$ denotes the approximate value function. Furthermore, the critic network updates the policy via gradient descent on the loss function, which can be represented by

$$L(\phi) = \hat{\mathbb{E}}[(V_\phi(s_t) - r(\theta) + V_\phi(s_{t+1}))^2]. \quad (42)$$

$\hat{A}_{\pi_{\theta_{old}}}$ can quantify the merit of an action to a certain degree for a given time slot t, which can be defined by

$$\hat{A}_{\pi_{\theta_{old}}} = \hat{Q}_{\pi_{\theta_{old}}} - \hat{V}_{\pi_{\theta_{old}}}, \quad (43)$$

where $\hat{Q}_{\pi_{\theta_{old}}}$ and $\hat{V}_{\pi_{\theta_{old}}}$ represent the value functions of action and state.

Eventually, the actor and critic parameters θ and ϕ are updated by gradient ascent and gradient descent, respectively, using a mini-batch stochastic optimization algorithm. As shown in (44) and (45)

$$\theta = \arg\max_\theta L(\theta), \quad (44)$$

and

$$\phi = \arg\min_\phi L(\phi). \quad (45)$$

For clarity, the specific process of the PPO-based algorithm is outlined in Algorithm 1. It is noted that the PPO algorithm works in an on-policy manner since industrial data of the IIoT is often generated in real-time, as well as the system states and environmental conditions are constantly changing.

To enhance the clarity and readability of the article, we have summarized and comprehensively illustrated the meanings of each individual variable in Table I.

VI. Simulation Results and Analysis

The performance of our proposed scheme is evaluated by simulation experiments in terms of various aspects such as convergence and effectiveness in this section. In addition, the simulation experiments were conducted in Python 3.6 and TensorFlow 1.13.1.

A. Simulation Prameters

The reputation levels of device nodes are divided into four levels, those are, very low, low, medium and high. Additionally, the reputation score for each device node is taken from the set

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

TABLE I
VARIABLE DESCRIPTION

Parameters	Meaning	Parameters	Meaning
M	The number of edge intelligent server nodes	Ξ	The transaction throughput
N	The number of device nodes	ω	The frequency of Terahertz electromagnetic wave
N^c	The number of consensus nodes	τ	The transmittance
N^p	The number of Byzantine nodes	ν	The penalty index
K	The number of blockchain shards	ε	The clip index
$\mathcal{R}$	The reputation level of device nodes	θ	The actor parameter
$\mathcal{O}$	The computational offloading ratio	ϕ	The critic parameter
$\mathcal{H}$	The computational resources of edge intelligent servers	ϵ	The emissivity of the computation offloading channel
$\mathcal{G}$	The computing cycles	π_θ	The current policy
$\mathcal{F}_d$	The frequencies of devices	$\pi_{\theta_{old}}$	The old policy
$\mathcal{F}_m$	The frequencies of edge intelligent servers	$r(\theta)$	The ratio of the old policy to the new policy
$\mathcal{F}_c$	The frequencies of the cloud server	$L(\theta)$	The clipped surrogate objective function
$\mathcal{D}(t)$	The data task size	$L(\phi)$	The gradient descent loss function
$\mathcal{B}(t)$	The transaction batch size	$\hat{A}_{\pi_{\theta_{old}}}$	The generalized advantage estimation function
$\mathcal{A}(t)$	The average transaction size	$\hat{V}_{\pi_{\theta_{old}}}$	The state value function
$\partial_t(t)$	The transmit antenna gains	$\hat{Q}_{\pi_{\theta_{old}}}$	The action value function
$\partial_r(t)$	The receive antenna gains	T^I	The block interval
$\lambda(t)$	The transfer rate	S^B	The block size
$\mathcal{T}(t)$	The total delay	t_b	The broadcast delay among consensus nodes
$P_t(t)$	The transmission power of the edge intelligent servers	t_r	The recovery delay of Zyzzyva
P_n	The molecular absorbed noise power	${N_{abs}}^T$	The equivalent noise temperature
χ	The computing cycle for a signature	${N_0}^T$	The reference temperature
δ	The computing cycle of message authentication code	$V_\phi(s_t)$	The approximate value function
ρ	The probability that the transaction is correctly verified	$\mu(s_t)$	The mean of the normal distribution
ℓ	The average distance of the offloading link	$\sigma(s_t)$	The variance of the normal distribution

$\{0, 2, 3, 5\}$ and the transition matrix is shown as

$$\mathbb{R} = \begin{bmatrix} 0.05 & 0.15 & 0.5 & 0.3 \\ 0.3 & 0.05 & 0.15 & 0.5 \\ 0.5 & 0.3 & 0.05 & 0.15 \\ 0.15 & 0.5 & 0.3 & 0.05 \end{bmatrix}. \tag{46}$$

The remaining computational resources of edge intelligent servers can be set to four levels, which are from the set $\{1, 3, 5, 7\}$ GHz and the transition matrix of $\mathcal{H}$ can be given by

$$\mathbb{H} = \begin{bmatrix} 0.2 & 0.3 & 0.3 & 0.2 \\ 0.2 & 0.2 & 0.3 & 0.3 \\ 0.3 & 0.2 & 0.2 & 0.3 \\ 0.3 & 0.3 & 0.2 & 0.2 \end{bmatrix}. \tag{47}$$

Furthermore, other vital parameters are listed in Table II.

TABLE II
SIMULATION PARAMETERS

Parameters	Value
The amount of edge intelligent server nodes, M	16
The amount of device nodes, N	80
The amount of blockchain shards, K	$2-16$
Average transaction size, $\mathcal{A}(t)$	100B-500B [40]
Transaction batch size, $\mathcal{B}(t)$	1MB-2MB [48]
Data task size, $\mathcal{D}(t)$	4MB-8MB [36]
Ratio of Byzantine nodes	3% [49]
Learning rate of actor, η_a	0.001 [37]
Learning rate of critic, η_c	0.01 [37]
CPU cycles for an MAC, χ	1M [50]
CPU cycles for a signature, δ	2M [50]
Antenna gain to transmit and receive, $\partial_t(t)$, $\partial_r(t)$	10dBi [51]
Weight of throughput, α_1	0.5
Weight of delay, α_2	0.5

B. Convergence Comparison

In this part, the convergence of various schemes is compared under different baselines and different DRL algorithms.

Fig. 7 illustrates the convergence curves of various methods, including the approach with a fixed number of blockchain shards set to 8, the binary computation offloading scheme and the method with single consensus algorithm. We can observe that our proposed approach has excellent convergence under different baselines and achieves a higher total reward than the alternative methods. In this way, we demonstrate that the combination of intelligent sharding, parallel computation offloading, and dynamic consensus algorithm switching can effectively increase the total reward.

In Fig. 8, our proposed scheme achieves a higher total reward compared to the other two methods. This is because PPO can complete multiple policy updates with a single environmental interaction and iteratively learn from old policies, significantly enhancing sample utilization. As a result, PPO is a more suitable choice for our proposed scheme among various DRL algorithms.

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

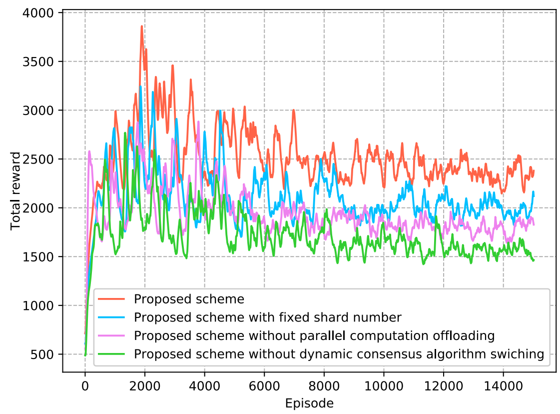


Fig. 7. Total reward under different baselines.

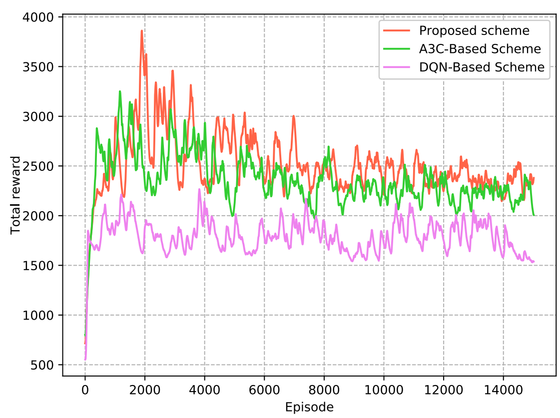


Fig. 8. Total reward with different DRL algorithms.

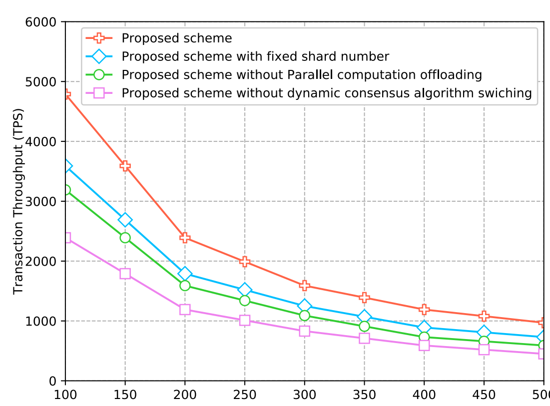


Fig. 9. Transaction throughput versus average transaction size.

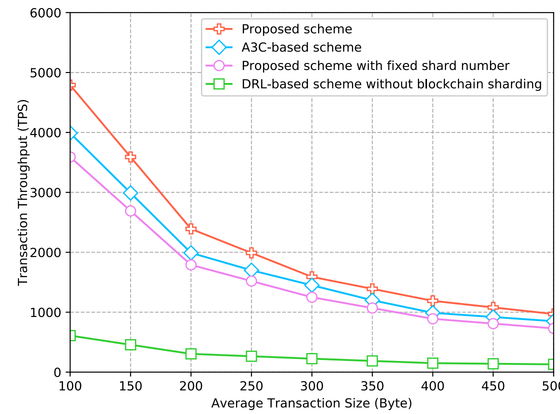


Fig. 10. Transaction throughput versus average transaction size of various existing methods.

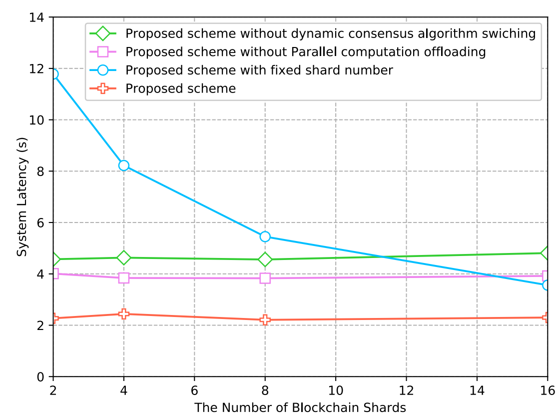


Fig. 11. System delay versus blockchain shards.

C. Effectiveness Comparison

The effectiveness of our proposed scheme from various perspectives of total delay, transaction throughput, and CST proportion is evaluated in this part. Simulation results indicate the superiority of the proposed scheme compared to other existing works.

Figs. 9 and 10 show the variation curves of transaction throughput versus average transaction size for various approaches. We can notice that the throughput gradually decreases as the average transaction size increases. In addition, when comparing the absolute values of throughput of different solutions, our proposed scheme always achieves a higher transaction throughput.

In particular, we can observe that our proposed scheme achieves higher transaction throughput than the A3C-based scheme, as well as significantly surpassing the existing schemes without blockchain sharding, as shown in Fig. 9. This is due to our comprehensive consideration of parallel computation offloading and intelligent sharding blockchain, along with the employment of the PPO algorithm, which boasts excellent training efficiency. The parallel computation offloading can significantly improve computational efficiency by legitimately deploying the computational resources of edge intelligent servers. Furthermore, the intelligent sharding blockchain method can enhance the IIoT by increasing the blockchain of blockchain and better adapting to data tasks from the IIoT.

In Figs. 11 and 12, we compare the total delay of different methods, with the number of blockchain shards as the independent variable. We can see that our proposed scheme consistently maintains a low delay. Furthermore, when using the approach of a fixed shard number, the delay decreases almost linearly as the number of shards increases. This is because the consensus delay

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

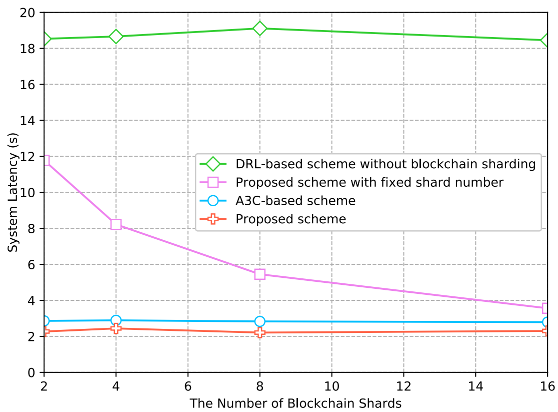


Fig. 12. System delay versus blockchain shards of various existing methods.

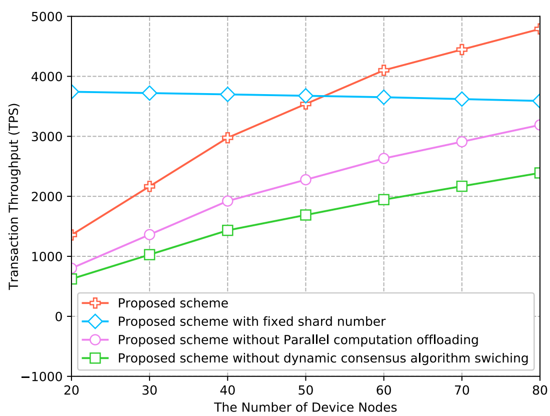


Fig. 13. Transaction throughput versus device nodes.

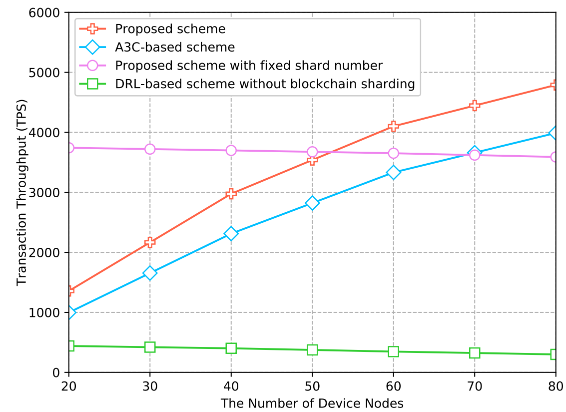


Fig. 14. Transaction throughput versus device nodes of various existing methods.

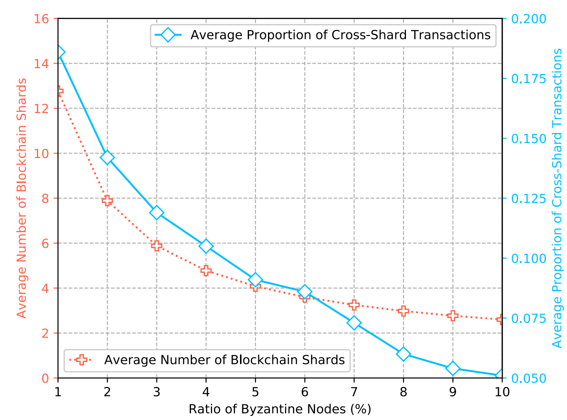


Fig. 15. Average proportion of CST and average number of blockchain shards under different ratio of Byzantine nodes.

consumed by the blockchain for transaction validation decreases in a linear fashion with the increasing number of shards.

Moreover, it can be observed that there is no significant change in total delay as the number of shards increases for all methods, except for the method with a fixed number of shards. This is because the optimal number of blockchain shards is determined by the intelligent agent, rather than a manual setting, under the approach of intelligent sharding blockchain. In this case, our proposed approach can organically integrate blockchain systems with the highly dynamic IIoT.

In Fig. 12, the A3C-based scheme is included for comparison. It can be found that our proposed scheme has a considerable advantage of system delay compared to the A3C-based scheme, which shows the algorithmic advantage of PPO for our proposed framework. In addition, the non-blockchain sharding scheme has an unbridgeable performance gap in terms of system delay when compared to our proposed scheme.

Figs. 13 and 14 indicate the impact of device nodes on the transaction throughput. We can see from the figures that the proposed scheme achieves better throughput. Moreover, the throughput grows linearly as the number of device nodes increases in all schemes, except for the fixed sharding method and the method without blockchain sharding. This is because the intelligent agent tends to create more shards as the number of device nodes increases. Furthermore, the transaction throughput of the fixed sharding approach is stable but slightly decreases as the number of device nodes increases. This is because we have fixed the number of blockchain shards to 8 in advance. This is also the reason why the fixed shards approach achieves higher transaction throughput when the number of device nodes is less than 50.

In Fig. 15, the average fraction of CSTs and the average number of blockchain shards under different Byzantine node ratios are presented. We can observe that both the average proportion of CSTs and the average number of shards decrease as Byzantine nodes increase. Actually, the average ratio of CSTs is affected by the number of blockchain shards. Specifically, the number of shards would decrease as Byzantine nodes increase to ensure the safety, that is, the number of shards is limited by Byzantine nodes. In this case, more transactions are verified and packaged into blocks within the shards, which reduces the proportion of CSTs. It is well known that CSTs consume more computational resources and communication overhead compared to ISTs. Therefore, with the proposed intelligent sharding approach, it is meaningful that the threat within the security range can be effectively reduced and transformed into lower computational overhead.

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

VII. Conclusion and Future Works

To solve the security and reliability risks and various challenges and pain points of existing blockchain-enabled IIoT systems, a cloud-edge-end collaborative computing-enabled intelligent sharding blockchain for the IIoT was proposed in this paper. In the proposed approach, the intelligent sharding based on the reputation mechanism had enhanced the scalability significantly while guaranteeing the safety and decentralization of the blockchain. Besides, the adaptive consensus protocol switching approach had substantially improved the adaptability of the blockchain to the ever-changing IIoT environments. In addition, we had introduced the cloud-edge-end collaborative computing paradigm to address the limited computational power of sharding blockchain and industrial devices, where the parallel computational offloading and the Terahertz communication technology was introduced to enhance the cooperation of the cloud-edge-end networks. Moreover, by solving the optimization problem with the PPO algorithm, our proposed scheme had achieved lower total delay and higher throughput compared to existing studies. In future works, we will further optimize the sharding scheme of the intelligent sharding blockchain. On the other hand, considering the problem of industrial pollution, we will pay more attention to energy consumption while maintaining a high productivity of the IIoT systems.

References

[1] A. Dixit, A. Singh, J. Rahulamathavan, and M. Rajarajan, "FAST DATA: A fair, secure, and trusted decentralized IIoT data marketplace enabled by blockchain," *IEEE Internet Things J.*, vol. 10, no. 4, pp. 2934–2944, Feb. 2023.

[2] L. Xu, W. He, and S. Li, "Internet of Things in industries: A survey," *IEEE Trans. Ind. Inform.*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.

[3] Y. Wu, H. Dai, and H. Wang, "Convergence of blockchain and edge computing for secure and scalable IIoT critical infrastructures in Industry 4.0," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2300–2317, Feb. 2021.

[4] G. Aceto, V. Persico, and A. Pescapé, "A survey on information and communication technologies for Industry 4.0: State-of-the-art, taxonomies, perspectives, and challenges," *IEEE Commun. Surv. Tut.*, vol. 21, no. 4, pp. 3467–3501, Fourth Quarter 2019.

[5] H. Liu, S. Li, W. Li, and W. Sun, "A distributed resource sharing mechanism in edge-enabled IIoT systems," *IEEE Internet Things J.*, vol. 11, no. 8, pp. 14296–14312, Apr. 2024.

[6] R. Huo et al., "A comprehensive survey on blockchain in Industrial Internet of Things: Motivations, research progresses, and future challenges," *IEEE Commun. Surv. Tut.*, vol. 24, no. 1, pp. 88–122, First Quarter 2022.

[7] Y. Wu, H. Dai, H. Wang, Z. Xiong, and S. Guo, "A survey of intelligent network slicing management for Industrial IoT: Integrated approaches for smart transportation, smart energy, and smart factory," *IEEE Commun. Surv. Tut.*, vol. 24, no. 2, pp. 1175–1211, Second Quarter 2022.

[8] B. Cao et al., "Blockchain systems, technologies, and applications: A methodology perspective," *IEEE Commun. Surv. Tut.*, vol. 25, no. 1, pp. 353–385, First Quarter 2023.

[9] M. Jalalzai, C. Feng, C. Busch, G. Richard, and J. Niu, "The hermes BFT for blockchains," *IEEE Trans. Dependable Secur. Comput.*, vol. 19, no. 6, pp. 3971–3986, Nov./Dec. 2022.

[10] G. Rebello et al., "A survey on blockchain scalability: From hardware to layer-two protocols," *IEEE Commun. Surv. Tut.*, vol. 26, no. 4, pp. 2411–2458, Fourth Quarter 2024.

[11] S. Khezr, A. Yassine, R. Benlamri, and M. Hossain, "An edge intelligent blockchain-based reputation system for IIoT data ecosystem," *IEEE Trans. Ind. Inform.*, vol. 18, no. 11, pp. 8346–8355, Nov. 2022.

[12] Y. Liu et al., "A flexible sharding blockchain protocol based on cross-shard Byzantine fault tolerance," *IEEE Trans. Inf. Forensic Secur.*, vol. 18, pp. 2276–2291, 2023.

[13] L. Jia, Y. Liu, K. Wang, and Y. Sun, "Estuary: A low cross-shard blockchain sharding protocol based on state splitting," *IEEE Trans. Parallel Distrib. Syst.*, vol. 35, no. 3, pp. 405–420, Mar. 2024.

[14] A. Liu et al., "CHERUBIM: A secure and highly parallel cross-shard consensus using quadruple pipelined two-phase commit for sharding blockchains," *IEEE Trans. Inf. Forensic Secur.*, vol. 19, pp. 3178–3193, 2024.

[15] X. Li et al., "Two-stage offloading for an enhancing distributed vehicular edge computing and networks: Model and algorithm," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 11, pp. 17744–17761, Nov. 2024.

[16] Q. Tang et al., "Joint service deployment and task scheduling for satellite edge computing: A two-timescale hierarchical approach," *IEEE J. Sel. Areas Commun.*, vol. 42, no. 5, pp. 1063–1079, May 2024.

[17] Q. Tang et al., "SIaTS: A service intent-aware task scheduling framework for computing power networks," *IEEE Netw.*, vol. 38, no. 4, pp. 233–240, Jul. 2024.

[18] J. Abdella, Z. Tari, R. Mahmud, N. Sohrabi, A. Anwar, and A. Mahmood, "HiCoOB: Hierarchical concurrent optimistic blockchain consensus protocol for peer-to-peer energy trading systems," *IEEE Trans. Smart Grid*, vol. 14, no. 5, pp. 3927–3943, Sep. 2023.

[19] G. Walia, M. Kumar, and S. Gill, "AI-empowered fog/edge resource management for IoT applications: A comprehensive review, research challenges, and future perspectives," *IEEE Commun. Surv. Tut.*, vol. 26, no. 1, pp. 619–669, First Quarter 2024.

[20] J. Feng, Q. Pei, F. RichardYu, X. Chu, and B. Shang, "Computation offloading and resource allocation for wireless powered mobile edge computing with latency constraint," *IEEE Wirel. Commun. Lett.*, vol. 8, no. 5, pp. 1320–1323, Oct. 2019.

[21] F. Zhang, G. Han, L. Liu, Y. Zhang, Y. Peng, and C. Li, "Cooperative partial task offloading and resource allocation for IIoT based on decentralized multiagent deep reinforcement learning," *IEEE Internet Things J.*, vol. 11, no. 3, pp. 5526–5544, Feb. 2024.

[22] J. Jornet, L. Akyildiz, Y. Li, J. Ma, and H. Dong, "Channel modeling and capacity analysis for electromagnetic wireless nanonetworks in the terahertz band," *IEEE Trans. Wirel. Commun.*, vol. 10, no. 10, pp. 3211–3221, Oct. 2011.

[23] T. Cai, W. Chen, J. Zhang, and Z. Zheng, "Smart chain: A dynamic and self-adaptive sharding framework for IoT blockchain," *IEEE Trans. Serv. Comput.*, vol. 17, no. 2, pp. 674–688, Mar./Apr. 2024.

[24] X. Yang, T. Xu, F. Zan, T. Ye, Z. Mao, and T. Qiu, "An overlapping self-organizing sharding scheme based on DRL for large-scale IIoT blockchain," *IEEE Internet Things J.*, vol. 11, no. 4, pp. 5681–5695, Feb. 2024.

[25] L. Jiang, Y. Liu, H. Tian, L. Tang, and S. Xie, "Resource efficient federated learning and DAG blockchain with sharding in digital twin driven Industrial IoT," *IEEE Internet Things J.*, vol. 11, no. 10, pp. 17113–17127, May 2024.

[26] R. Li, Y. Qin, C. Wang, M. Li, and X. Chu, "A blockchain-enabled framework for enhancing scalability and security in IIoT," *IEEE Trans. Ind. Inform.*, vol. 19, no. 6, pp. 7389–7400, Jun. 2023.

[27] S. Mighan, J. Mišić, V. Mišić, and X. Chang, "An in-depth look at forking-based attacks in ethereum with PoW consensus," *IEEE Trans. Netw. Serv. Manag.*, vol. 21, no. 1, pp. 507–516, Feb. 2024.

[28] M. Saad, Z. Qin, K. Ren, D. Nyang, and D. Mohaisen, "e-PoS: Making proof-of-stake decentralized and fair," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 8, pp. 1961–1973, Aug. 2021.

[29] P. Zhang, P. Yang, N. Kumar, C. Hsu, S. Wu, and F. Zhou, "RRV-BC: Random reputation voting mechanism and blockchain assisted access authentication for Industrial Internet of Things," *IEEE Trans. Ind. Inform.*, vol. 20, no. 1, pp. 713–722, Jan. 2024.

[30] Y. Chen, Y. Zhang, Y. Zhuang, K. Miao, S. Pouriyeh, and M. Han, "Efficient and secure blockchain consensus algorithm for heterogeneous Industrial Internet of Things nodes based on double-DAG," *IEEE Trans. Ind. Inform.*, vol. 20, no. 4, pp. 6300–6312, Apr. 2024.

[31] A. Aljuhani et al., "A deep-learning-integrated blockchain framework for securing Industrial IoT," *IEEE Internet Things J.*, vol. 11, no. 5, pp. 7817–7827, Mar. 2024.

[32] A. Alanhdi and L. Toka, "A survey on integrating edge computing with AI and blockchain in maritime domain, aerial systems, IoT, and Industry 4.0," *IEEE Access*, vol. 12, pp. 28684–28709, 2024.

[33] W. Hua, P. Liu, and L. Huang, "Energy-efficient resource allocation for heterogeneous edge ccloud computing," *IEEE Internet Things J.*, vol. 11, no. 2, pp. 2808–2818, Jan. 2024.

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.

[34] D. Kim, D. Lee, J. Kim, and H. Lee, “Collaborative policy learning for dynamic scheduling tasks in cloud-edge-terminal IoT networks using federated reinforcement learning,” *IEEE Internet Things J.*, vol. 11, no. 6, pp. 10133–10149, Mar. 2024.

[35] D. Gan, X. Ge, and Q. Li, “An optimal transport-based federated reinforcement learning approach for resource allocation in cloud-edge collaborative IoT,” *IEEE Internet Things J.*, vol. 11, no. 2, pp. 2407–2419, Jan. 2024.

[36] 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.

[37] X. Ye, M. Li, P. Si, R. Yang, Z. Wang, and Y. Zhang, “Collaborative and intelligent resource optimization for computing and caching in IoV with blockchain and MEC using A3C approach,” *IEEE Trans. Veh. Technol.*, vol. 72, no. 2, pp. 1449–1463, Feb. 2023.

[38] C. Meng, K. Xiong, W. Chen, B. Gao, P. Fan, and K. Letaief, “Sum-rate maximization in STAR-RIS-assisted RSMA networks: A PPO-based algorithm,” *IEEE Internet Things J.*, vol. 11, no. 4, pp. 5667–5680, Feb. 2024.

[39] H. An and L. Wang, “Robust topology generation of Internet of Things based on PPO algorithm using discrete action space,” *IEEE Trans. Ind. Inform.*, vol. 20, no. 4, pp. 5406–5414, Apr. 2024.

[40] G. Lee, H. Park, J. Jang, J. Han, and J. Choi, “PPO-based autonomous transmission period control system in IoT edge computing,” *IEEE Internet Things J.*, vol. 10, no. 24, pp. 21705–21720, Dec. 2023.

[41] N. Mensah et al., “Adaptive storage optimization scheme for blockchain-IIoT applications using deep reinforcement learning,” *IEEE Access*, vol. 11, pp. 1372–1385, 2023.

[42] M. Castro and B. Liskov, “Practical Byzantine fault tolerance,” in *Proc. 3rd Symp. Operating Syst. Des. Implementation*, New Orleans, LA, USA, 1999, pp. 173–186.

[43] M. Castro and B. Liskov, “Practical Byzantine fault tolerance and proactive recovery,” *ACM Trans. Comput. Syst.*, vol. 20, no. 4, pp. 398–461, Nov. 2002.

[44] R. Kotla, L. Alvisi, M. Dahlin, A. Clement, and E. Wong, “Zyzzyva: Speculative Byzantine fault tolerance,” *ACM Trans. Comput. Syst.*, vol. 27, no. 4, pp. 1–39, Jan. 2010.

[45] P. Aublin, R. Guerraoui, N. Knezevic, V. Quema, and M. Vukolic, “The next 700 BFT protocols,” *ACM Trans. Comput. Syst.*, vol. 32, no. 4, pp. 1–45, Jan. 2015.

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

[47] H. Kang, X. Chang, J. Mišić, V. Mišić, J. Fan, and Y. Liu, “Cooperative UAV resource allocation and task offloading in hierarchical aerial computing systems: A MAPPO-based approach,” *IEEE Internet Things J.*, vol. 10, no. 12, pp. 10497–10509, Jun. 2023.

[48] H. Zhou, Z. Wang, H. Zheng, S. He, and M. Dong, “Cost minimization-oriented computation offloading and service caching in mobile cloud-edge computing: An A3C-based approach,” *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 3, pp. 1326–1338, May/Jun. 2023.

[49] Z. Yang, R. Yang, F. R. Yu, M. Li, Y. Zhang, and Y. Teng, “Sharded blockchain for collaborative computing in the Internet of Things: Combined of dynamic clustering and deep reinforcement learning approach,” *IEEE Internet Things J.*, vol. 9, no. 17, pp. 16494–16509, Sep. 2022.

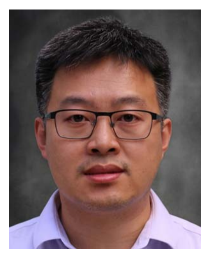
[50] F. Guo, F. R. Yu, H. Zhang, H. Ji, M. Liu, and V. Leung, “Adaptive resource allocation in future wireless networks with blockchain and mobile edge computing,” *IEEE Trans. Wirel. Commun.*, vol. 19, no. 3, pp. 1689–1703, Mar. 2020.

[51] M. Li et al., “Cloud-edge collaborative resource allocation for blockchain-enabled Internet of Things: A collective reinforcement learning approach,” *IEEE Internet Things J.*, vol. 9, no. 22, pp. 23115–23129, Nov. 2022.

Xin Xiong received the BE degree in electronic information engineering from the Beijing University of Technology, Beijing, China, in 2022. He is currently working toward the MS degree with the Beijing University of Technology. His research interests include Industrial Internet of Things, blockchain, and edge computing.

Meng Li (Senior Member, IEEE) received the PhD degree in electronic science and technology from the Beijing University of Technology, Beijing, China, in 2018. He is currently an associate professor with the School of Information Science and Technology, Beijing University of Technology. From September 2015 to 2016, he visited Carleton University, Ottawa, ON, Canada, as a visiting PhD student funded by China Scholarship Council. His research interests include M2M communications, Industrial Internet, intelligent edge computing, and blockchain. He has received the Excellent Doctoral Dissertation Award from China Education Society of Electronics in 2019 and the Best Paper Award at ICCC 2021. He has served as the technical program committee (TPC) member of several international conferences, including IEEE INFOCOM, IEEE GLOBECOM, IEEE ICC, etc.

F. Richard Yu (Fellow, IEEE) received the PhD degree in electrical engineering from the University of British Columbia (UBC) in 2003. His research interests include connected/autonomous vehicles, artificial intelligence, blockchain, and wireless systems. He has been named in the Clarivate Analytics list of “Highly Cited Researchers” since 2019, and received several Best Paper Awards from some firsttier conferences. He was a Board member the IEEE VTS and is the editor-in-chief for *IEEE VTS Mobile World newsletter*. He is a fellow of Canadian Academy of Engineering (CAE), Engineering Institute of Canada (EIC), and IET. He is a distinguished lecturer of IEEE in both VTS and ComSoc.



Haijun Zhang (Fellow, IEEE) is currently a full professor with the University of Science and Technology Beijing, China. He was a postdoctoral research fellow with the Department of Electrical and Computer Engineering, the University of British Columbia (UBC), Canada. He serves/served as track co-chair of WCNC 2020, Symposium Chair of Globecom’19, TPC co-chair of INFOCOM 2018 Workshop on Integrating Edge Computing, Caching, and Offloading in Next Generation Networks, and General Co-Chair of GameNets’16. He serves/served as an editor of *IEEE Transactions on Information Forensics and Security*, *IEEE Transactions on Communications*, *IEEE Transactions on Network Science and Engineering*, and *IEEE Transactions on Vehicular Technology*. He received the IEEE CSIM Technical Committee Best Journal Paper Award, in 2018, IEEE ComSoc Young Author Best Paper Award, in 2017, and IEEE ComSoc Asia-Pacific Best Young Researcher Award, in 2019. He is an IEEE ComSoc distinguished lecturer.

Kan Wang (Member, IEEE) received the BS degree in broadcasting and television engineering from Communications University of Zhejiang, Hangzhou, China, in 2009, and the PhD degree in military communications from the State Key Laboratory of ISN, Xidian University, Xi’an, China, in 2016. From 2014 to 2015, he was with Carleton University, Ottawa, ON, Canada, as a visiting scholar funded by the China Scholarship Council. Since 2017, he has been with the School of Computer Science and Engineering, Xi’an University of Technology, Xi’an. His current research interests include future cellular networks, wireless resource management, and wireless AI.

Pengbo Si (Senior Member, IEEE) received the BS and PhD degrees from the Beijing University of Posts and Telecommunications in 2004 and 2009, respectively. He joined Beijing University of Technology in 2009, where he is currently a professor. During 2007 and 2008, he visited Carleton University, Ottawa, Canada. During 2014 and 2015, he was a visiting scholar with the University of Florida, Gainesville FL. He serves as the associate editor of *International Journal on Ad Hoc Networking Systems*, the editorial board member of *Ad Hoc & Sensor Wireless Networks*, and the symposium chair of IEEE Globecom 2019. He also served as the guest editor of *Advances in Mobile Cloud Computing*, *IEEE Transactions on Emerging Topics in Computing* Special Issue, TPC co-chair of IEEE ICCC’13-GMCN, program vice chair of IEEE GreenCom’13, and TPC member of numerous conferences. His research interests include blockchain, SDN, resource management, cognitive radio networks, etc.

Authorized licensed use limited to: University of Regina IEL. Downloaded on November 20,2025 at 08:59:40 UTC from IEEE Xplore. Restrictions apply.