Blockchain-Enabled Secure Data Sharing Scheme in Mobile-Edge Computing: An Asynchronous Advantage Actor–Critic Learning Approach

Lei Liu, *Member, IEEE*, Jie Feng, *Member, IEEE*, Qingqi Pei, *Senior Member, IEEE*, Chen Chen, *Senior Member, IEEE*, Yang Ming, *Member, IEEE*, Bodong Shang, *Graduate Student Member, IEEE*, and Mianxiong Dong, *Member, IEEE*

***Abstract*—Mobile-edge computing (MEC) plays a significant role in enabling diverse service applications by implementing efficient data sharing. However, the unique characteristics of MEC also bring data privacy and security problem, which impedes the development of MEC. Blockchain is viewed as a promising technology to guarantee the security and traceability of data sharing. Nonetheless, how to integrate blockchain into MEC system is quite challenging because of dynamic characteristics of channel conditions and network loads. To this end, we propose a secure data sharing scheme in the blockchain-enabled MEC system using an asynchronous learning approach in this article. First, a blockchain-enabled secure data sharing framework in the MEC system is presented. Then, we present an adaptive privacy-preserving mechanism according to available system resources and privacy demands of users. Next, an optimization problem of secure data sharing is formulated in the blockchain-enabled MEC system with the aim to maximize the system performance with respect to the decreased energy consumption of MEC system and the increased throughput of blockchain system. Especially, an asynchronous learning approach is employed to solve the formulated problem. The numerical results demonstrate the superiority of our proposed secure data sharing scheme when compared with some popular benchmark algorithms in terms of average throughput, average energy consumption, and reward.**

***Index Terms*—Blockchain, data sharing, deep reinforcement learning (DRL), mobile-edge computing (MEC), security and privacy.**

I. Introduction

INTERNET of Things (IoT) has been penetrating every aspect of people'life by integrating a variety of devices and exploiting a series of advanced technologies [1]–[3]. With the remarkable increase in the number of diverse devices and emerging applications, the data in IoT are exponentially growing. Under this situation, data sharing becomes a pivotal process in IoT. In industrial IoT, data sharing can enable smart industrial decisions [4], e.g., resource scheduling and supplement. For intelligent transportation systems, data sharing helps in enhancing road safety and improving traveling quality [5]–[7].

Due to the limitation in terms of storage and computation resources, IoT devices cannot implement efficient data sharing. Given the rich resources, cloud computing is capable of improving the efficiency of data sharing between data owners and subscribers [8]. However, a large number of workloads filling into networks will result in network congestion when leveraging cloud computing. Mobile-edge computing (MEC) is proposed as a promising paradigm for handling this limitation of cloud [9], [10], by offloading computing tasks to the MEC server or base stations (BSs) [11], [12]. MEC contributes reducing the transmission delay by migrating transmitted data from the remote cloud to edge servers [13], thereby supporting latency-sensitive applications [14]. Although, MEC enables the implementation of data sharing, it also brings new security and privacy problems due to the concerns about single node failures and manipulated personal data [15]. In MEC, edge servers are vulnerable to being compromised by attackers without strong security protection [16]. Because data may contain sensitive and private information, e.g., personal behavior patterns and clinical data, careless data sharing will cause information leakage [17]. These impose great challenges for data sharing.

In order to deal with these issues above, blockchain is viewed as an appealing approach [18]. Different from

Manuscript received July 21, 2020; revised October 2, 2020; accepted December 20, 2020. Date of publication December 31, 2020; date of current version February 4, 2021. This work was supported in part by the National Key Research and Development Program of China(2020YFB1807500); in part by the National Natural Science Foundation of China (62001357,62072360,62072054); in part by the key Research and Development plan of Shaanxi province(2017ZDCXL-GY-05-01, 2019ZDLGY13-04); and in part by the Japan Society for the Promotion of Science KAKENHI Grant JP20F20080. *(Corresponding author: Jie Feng.)*

Lei Liu, Jie Feng, Qingqi Pei, and Chen Chen are with the State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China (e-mail: tianjiaoliulei@163.com; jiefengcl@163.com; qqpei@mail.xidian.edu.cn; cc2000@mail.xidian.edu.cn).

Yang Ming is with the School of Information Engineering, Chang'an University, Xi'an 710064, China (e-mail: yangming@chd.edu.cn).

Bodong Shang is with the Wireless@VT, Department of Electronics and Communication Engineering, Virginia Tech, Blacksburg, VA 24061 USA (e-mail: bdshang@vt.edu).

Mianxiong Dong is with the Department of Information and Electronic Engineering, Muroran Institute of Technology, Muroran 050-8585, Japan (e-mail: mx.dong@csse.muroran-it.ac.jp).

Digital Object Identifier 10.1109/JIOT.2020.3048345

traditional digital ledger methods, which are dependent on a trusted central authority, blockchain transfers digital assets using an entire distributed and secure manner and employing the ledger [19], [20]. To satisfy the privacy requirements of users, blockchain adopts pseudonymous techniques to disguise users' identities when sharing data in order to avoid users' privacy leakage [21]. Although utilizing pseudonymous approaches can eliminate the transactions linking to real identities, users are not entirely anonymous in their actions. That is, because all utilizations behind these pseudoanonymous may be linkable and traceable, especially when processing multiple transactions from several addresses belonging to the same user's different accounts [22].

Some existing privacy-preserving approaches have been investigated to enhance decentralized ledgers. Gao *et al.* [23] presented a blockchain-based privacy-preserving scheme for vehicle-to-grid networks, which can realize efficient data sharing while protecting sensitive user information. Kosba *et al.* [24] proposed a decentralized smart contract system called Hawk, whose financial transactions are not clearly stored on the blockchain, therefore reserving transactional privacy from the public's perspective. Alboaie and Cosovan [25] presented a distributed private data system (PDS) to achieve self-sovereign storage and sharing of private data. Miers *et al.* [26] proposed an extended cryptographic technique for Bitcoin, i.e., Zerocoin, which fully implements anonymous currency transactions. A blockchain-enabled efficient data collection and secure sharing scheme was investigated in [27], which creates a reliable and safe environment with the combination of Ethereum blockchain and deep reinforcement learning. In [28], a blockchain enabled secure data sharing architecture is designed for distributed multiple parties, and the data sharing problem is formulated into a machine-learning problem by combing privacy-preserved federated learning. An access control scheme is presented in [29] by making fair compensation for customers due to their participation in data sharing through blockchain based on the concept of differential privacy. Different users are enabled to share their encrypted data using the proposed data sharing scheme under a decentralized storage architecture based on blockchain [30]. Although the existing works on privacy-preserving approaches of blockchain in data sharing have been done, some open issues remain to be solved. For example, the blockchain-based privacy-preserving approaches for data sharing in MEC systems still remain unknown. On the other hand, due to the time-varying wireless channels and the randomness of traffic loads in the time domain, how to dynamically meet the data and privacy requirements in such practical MEC systems is worth to be investigated.

In this article, we propose a secure data sharing scheme in the blockchain-enabled MEC system using an asynchronous learning approach. A blockchain-enabled secure data sharing framework in MEC is first devised with the aim to enhance data sharing efficiency and improve network resource utilization. Then, the energy consumption of the MEC system and the throughput of the blockchain system are jointly optimized while taking into account the delay constraint. Besides, toward the dynamic characteristics in the blockchain-enabled MEC system, we adopt a deep reinforcement learning (DRL) method, i.e., asynchronous advantage actor–critic (A3C) reinforcement learning to obtain the solution to the formulated optimization problem, which is suitable to address dynamic and complex problems. We summarize the main contributions of this article as follows.

1) A secure data-sharing framework for the blockchain-enabled MEC system is presented. By integrating blockchain technology into MEC, the proposed framework ensures that users' data are shared privately and securely.
2) An adaptive privacy-preserving mechanism in the blockchain-based MEC system while considering available communication and computing resources is proposed to meet the privacy requirements of different users.
3) An optimization problem is designed with the goal of minimizing the energy consumption of MEC system and maximizing the throughput of the blockchain system. Especially, an asynchronous learning approach is employed to obtain the solution to this formulated problem, which is capable of adapting to dynamic network environments.

The remainder of this article is organized as follows. Section II introduce the system description, which elaborates the system scenario, MEC system, and blockchain system and provides the objective in the blockchain-enabled system. Section III presents the privacy-preserving solution for blockchain. An optimization problem of secure data sharing is formulated in the blockchain-enabled MEC system in Section IV. The asynchronous advantage actor–critic algorithm is adopted to obtain the solution to the formulated problem in Section V. Numerical results are given in Section VI, followed by the conclusion in Section VII. The mainly used symbols in this article is summarized in Table I for easy reference.

II. System Description

In this section, the system scenario is first introduced. Then, we elaborate the MEC system and the blockchain-enable system in detail, respectively. After that, the goal of the blockchain-enabled system is defined.

A. System Scenario

A blockchain-enabled MEC system is considered as illustrated in Fig. 1, which is an integrated system of blockchain and MEC. Since the resources of MEC servers in terms of computation and storage are sufficient, the blockchain nodes are deployed on MEC servers in the system. An MEC server can process the tasks from mobile users and the tasks in the blockchain system, simultaneously. In the MEC system, there are two data-sharing modes. One is that data can be shared with the help of MEC servers, i.e., data are uploaded to MEC servers and other users obtain data from MEC servers. The other is that data can be shared through device-to-device (D2D) communications, while MEC servers

TABLE I
MAIN SYMBOLS USED IN THIS ARTICLE

Notation	Description
τ	Time slot
$P_{n,m}^{MEC}$	The transmit power from mobile device n to MEC server m
$g_{n,m}$	The channel gain from mobile device n to MEC server m
g_n^{D2D}	The channel gain between the data provider n and its requester
$r_{n,m}^{MEC}$	The transmission rate of data uploaded to MEC server m from mobile device n
$t_{n,m}^{MEC}$	The transmit time between mobile device n and MEC server m
$E_{n,m}^{MEC}$	The sharing energy consumption from mobile device n to MEC server m
$E_{n,m}^{ser}$	The energy consumption of MEC server m for mobile device n
P_n^{D2D}	The transmit power between the data provider n and its data requester
r_n^{D2D}	The transmission rate between data provider n and its data requester
P_n^{MEC}	The transmit power of data provider n
t_n^{D2D}	The transmit time between data provider n and its data requester
E_n^{D2D}	The energy consumption of data provider n
E_{total}	The system total consumed energy
$S_m(t)$	The computing resource available to the blockchain system in slot t
T_c	The time cost of the consensus process
T_{total}	The latency time to finality of the blockchain system
Ψ	The throughput of the blockchain system
$\pi(a_t\|s_t;\theta)$	The policy with parameter θ
$V(s_t;\theta_v)$	The value function with parameter θ_v
$R_t(\theta_v)$	The discounted reward
$A(a_t,s_t;\theta,\theta_v)$	The advantage function
$f_\pi(\theta)$	The loss function of the policy
$f_v(\theta_v)$	The loss function of the value function
r_{t+i}	The immediate reward at the i^{th} step
γ	The discount factor

provide corresponding radio resources management. In the blockchain system, it mainly processes the data sharing transactions from the MEC system in a secure manner. There are two processes. The first one is the block generation process in which the transactions are packaged into a block, and the second one is the block consensus process where the block is broadcast to other consensus nodes. When a new block reaches consensus, it will be chained into the blockchain. In the system, time is slotted and the length of each time slot is denoted by τ. In the system, MEC servers can act as the blockchain nodes because of having sufficient computation and storage resources [31], [32]. These blockchain nodes play significant roles in publicly auditing and storing sharing data records in a blockchain-enabled MEC system. Blockchain nodes are selected by reputation-based voting, i.e., blockchain nodes with high reputation can provide the sharing data to prevent malicious nodes from abusing data [15].

B. MEC System

There are N mobile devices and M MEC servers in the MEC system. Similar to [33] and [34], mobile devices upload their data to MEC servers on orthogonal spectrum resources. Therefore, there is no interference among mobile devices. In the D2D data sharing model, data providers directly communicate with their requester by sharing spectrum resources of mobile devices. Let $\boldsymbol{P}^{\text{MEC}} = (P_{n,m}^{\text{MEC}}) \in \mathbb{R}^{N\times M}$ be the transmit power and $\boldsymbol{G} = (g_{n,m}) \in \mathbb{R}^{N\times M}$ denote the channel gain from mobile device n to MEC server. Let $\boldsymbol{G}^{\text{D2D}} = (g_n^{\text{D2D}}) \in \mathbb{R}^{N\times M}$ denote channel gain between the data provider n and its requester in D2D communications.

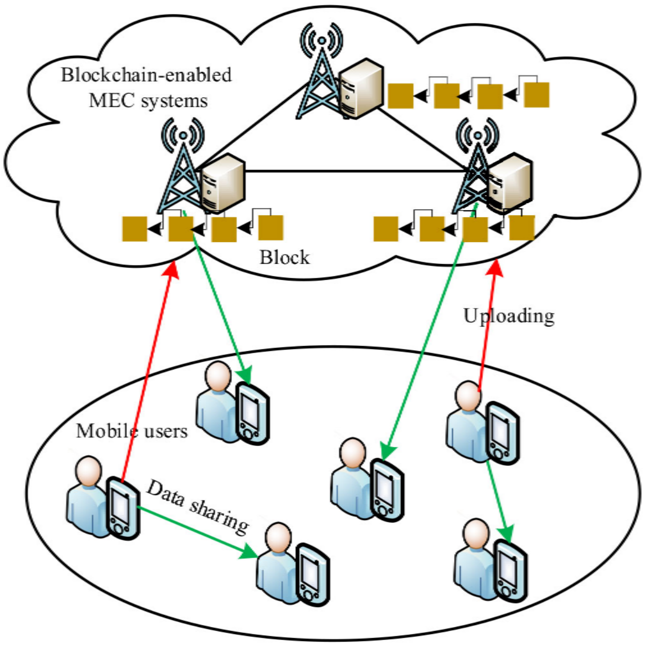


Fig. 1. System scenario.

For mobile device n, the transmission rate of data uploaded to MEC server m is given by

$$r_{n,m}^{\text{MEC}} = B\log_2\left(1 + \frac{P_{n,m}^{\text{MEC}} g_{n,m}}{\sum_{i=1}^{N} P_{\max}^{\text{D2D}} g_i^{\text{D2D}} + N_0}\right) \tag{1}$$

where $P_{\max}^{\text{D2D}}$ indicates the allowed maximum transmit power on each mobile device, N_0 is the noise power spectral density, and B stands for the bandwidth.

Accordingly, the transmit power from mobile device n to MEC server m is expressed as

$$P_{n,m}^{\text{MEC}} = \frac{A}{g_{n,m}}\left(2^{\frac{r_{n,m}^{\text{MEC}}}{B}} - 1\right) \tag{2}$$

where $A = \sum_{i=1}^{N} P_{\max}^{\text{D2D}} g_i^{\text{D2D}} + N_0$.

The transmit time is expressed as $t_{n,m}^{\text{MEC}} = D_n/r_{n,m}^{\text{MEC}}$, where D_n is the size of sharing data (in bits). Then, the sharing energy consumption from mobile device n to MEC server m is calculated as

$$E_{n,m}^{\text{MEC}} = P_{n,m}^{\text{MEC}} t_{n,m}^{\text{MEC}} = \frac{AD_n}{g_{n,m} r_{n,m}^{\text{MEC}}}\left(2^{\frac{r_{n,m}^{\text{MEC}}}{B}} - 1\right). \tag{3}$$

The energy consumption of MEC server m for mobile device n is derived as

$$E_{n,m}^{\text{ser}} = k_{\text{ser},m} s_{n,m}^2 D_n C_n \tag{4}$$

where $k_{\mathrm{ser},m}$ is the effective switched capacitance of the CPU at the mth MEC server, $s_{n,m}$ is the CPU-cycle frequency required by MEC server m, and C_n is the number of CPU cycles required for processing 1-b data.

Let $\boldsymbol{P}^{\mathrm{D2D}} = (P_n^{\mathrm{D2D}}) \in \mathbb{R}^{N\times M}$ denote the transmit power between the data provider n and its data requester. For D2D communications, the transmission rate between data provider n and its data requester is given by

$$r_n^{\mathrm{D2D}} = B\log_2\left(1+\frac{P_n^{\mathrm{D2D}} g_n^{\mathrm{D2D}}}{N_0+\sum_{j=1,j\neq n}^{N} P_{\max}^{\mathrm{D2D}} g_j^{\mathrm{D2D}}}\right). \tag{5}$$

Then, the transmit power of data provider n is given by

$$P_n^{\mathrm{D2D}} = \frac{\sum_{j=1,j\neq n}^{N} P_{\max}^{\mathrm{D2D}} g_j^{\mathrm{D2D}} + N_0}{g_n^{\mathrm{D2D}}}\left(2^{\frac{r_n^{\mathrm{D2D}}}{B}}-1\right). \tag{6}$$

The transmit time is given by

$$t_n^{\mathrm{D2D}} = \frac{D_n}{r_n^{\mathrm{D2D}}}. \tag{7}$$

Accordingly, the energy consumption of data provider n is written as

$$\begin{aligned} E_n^{\mathrm{D2D}} &= P_n^{\mathrm{D2D}} t_n^{\mathrm{D2D}} \\ &= \frac{D_n\sum_{j=1,j\neq n}^{N} P_{\max}^{\mathrm{D2D}} g_j^{\mathrm{D2D}} + N_0}{r_n^{\mathrm{D2D}} g_n^{\mathrm{D2D}}}\left(2^{\frac{r_n^{\mathrm{D2D}}}{B}}-1\right). \end{aligned} \tag{8}$$

The system total consumed energy is expressed as

$$E_{\mathrm{total}} = \sum_{n=1}^{M}\sum_{n=1}^{N} a_n\left(E_{n,m}^{\mathrm{MEC}} + E_{n,m}^{\mathrm{ser}}\right) + (1-a_n)E_n^{\mathrm{D2D}} \tag{9}$$

where a_n is the decision of data sharing. $a_n = 1$ if mobile device n shares data through MEC servers; otherwise, $a_n = 0$.

C. Blockchain-Enabled System

By employing MEC technique, mobile devices can enjoy a high-quality data sharing experience through obtaining data from the network edge. However, users' security may be compromised during data sharing, for example, malicious nodes misuse the user's data, causing user privacy to leak. The integration of blockchain into MEC system can effectively solve the problem [35]. In the system, since MEC servers have sufficient computing and storage resources, blockchain nodes are deployed on MEC servers. These blockchain nodes participate in public audits and store data sharing records in the blockchain-enabled MEC system. The main task of the blockchain system is to process data sharing transactions from the MEC system. Here, we employ public chain by which everyone is able to check and verify the transaction as well as participate in getting consensus. To address these transactions,[1] two steps need to be completed for the blockchain system, i.e., block generation process and block consensus process.

In order to make the improvement of the system performance, some nodes are selected as the delegators to participate in generating blocks. The nodes in the blockchain system are divided into two types: 1) ordinary nodes and 2) consensus nodes. Ordinary nodes are mainly used to transfer and accept ledger data and verify blocks, while consensus nodes can generate blocks and perform block verification. Let $\boldsymbol{\Upsilon}$ denote the number of stakes and $\boldsymbol{S}$ denote the available computing resources of blockchain nodes. In the blockchain nodes, it is assumed that there is a first-in–first-out (FIFO) data buffer. Then, the computing resource that is available to the blockchain system in slot t is expressed by

$$S_m(t) = \max\left\{F - \sum_{n=1}^{N} f_{n,m}(t), S_{\min}\right\} \tag{10}$$

where F is the total computing capacity of MEC servers, $f_{n,m}(t)$ represents the computing resources required to execute task n in MEC server m, and $S_{\min}$ represents the minimum computing resource required by the blockchain system.

Denote T_c as the delay in the consensus process. Similar to [36], the consensus process is composed of message propagation and message verification, which includes message authentication codes (MACs) generation, signatures verification, and MAC verification [33]. Then, the required time cost during the consensus process is written as

$$T_c = T_p + T_v \tag{11}$$

where T_p and T_v indicate the delay needed for message propagation and the validation, respectively.

The latency for the finality of the blockchain system is given by

$$T_{\mathrm{total}} = T_c + T_b \tag{12}$$

where T_b is the block interval.

For blockchain system, the throughput is given by

$$\Psi = \frac{\lfloor S_b/\chi \rfloor}{T_b} \tag{13}$$

where χ stands for the average size of transactions and S_b is the block size.

D. Definition of Objective in Blockchain-Enabled System

The goal of the system is to simultaneously minimize the energy consumption of MEC system and maximize the throughput of blockchain system. Then, the objective of the system is given by

$$O = \varpi_1 E_{\mathrm{total}} - (1-\varpi_1)\varpi_2\Psi. \tag{14}$$

$\varpi_1 (0 < \varpi_1 < 1)$ is a weight factor, which is used to combine two objective functions together into a single one, and ϖ_2 is a mapping factor that aims to ensure two objective functions at the same level.

There are four critical characteristics in blockchain, i.e., autonomous, distributed, immutability, and contractual. These characteristics make the blockchain architecture verifiable, transparent, and process integrate. Except for the above advantages of blockchain, privacy and trading account can be secured in blockchain, e.g., the leakage of users' real identity is protected. To protect users' identity during data sharing, the

[1] The transaction can be caching/computing resource allocation, radio resource allocation, and trading records.

TABLE II
COMPARISON OF DIFFERENT CRYPTOGRAPHIC SCHEMES

Cryptographic scheme	ZKP	SMPC	STARK
Succinct	×	×	Yes
Trust third party	No	No	No
Zero-Knowledge	Yes	×	Yes
Transparent	No	No	Yes
Required Resources	Medium	Low	High

blockchain hides user's real identity by using a pseudonym. Although blockchain can protect users' privacy to some extent through pseudonyms, it is not entirely anonymous for users. For example, an attacker can reveal the connection between a user's true identity and the pseudonym by analyzing the application consumption pattern, which poses a threat to users' privacy. To protect anonymity in the blockchain, it is necessary to ensure that the pseudonym is unlinkable. To ensure full anonymity in the blockchain, privacy-preserving mechanisms are required, such as zero-knowledge proofs (ZKP) [37], secure multiparty computation (SMPC) [38], and succinct, transparent arguments of knowledge (STARK) [39]. It is worth noting that there is no one-size-fits-all privacy-preserving mechanism considering various system conditions. For example, most ZPK solutions are inefficient for large-scale and responsive scenarios because they require a long computing time to generate and validate proofs. Therefore, the most suitable mechanism for protecting anonymity would be dynamically selection according to different system scenarios and application requirements. For this, a set of privacy-preserving mechanisms (i.e., ZKP, SMPC, and STARK) is considered as candidates for the system.

III. Privacy-Preserving Solution for Blockchain

In this section, we first describe the primary privacy-preserving mechanisms that can protect anonymity in the blockchain. Then, we introduce the system model of our proposed adaptive privacy-preserving scheme.

A. Main Privacy-Preserving Mechanisms in Blockchain

Table II shows the key features of these privacy-preserving approaches.

1) Zero-Knowledge Proof: ZKP is a digital protocol that allows for data to be shared between two parties without the use of a password or any other information associated with the transaction. The essence of ZKP is that it is trivial to simple reveal information to prove that someone has certain knowledge. The challenge is to prove possession of such property without revealing the information itself or any other information.

A ZKP must meet the following three requisites.

1) *Completeness:* If the statement to be proved is true, the verifier will persuade the verification to accept the fact.
2) *Soundness:* If the proven statement is false, then in addition to the minimal probability, no cheating prover can persuade the verifier that it is true.
3) *Zero-Knowledge:* If the statement to be proved is true, then no verifier will learn anything other than knowing that the statement is true. In other words, knowing only the statement, not the secret, is enough to prove that the secret is held by the proved. This is formalized by exhibiting that each verifier has some simulators that can generate a transcript that is indistinguishable from a successful proof between the honest prover and the verifier in question.

Completeness and soundness are attributes of a more general interactive proof system. The zero-knowledge is to make proof of zero-knowledge. Since, ZKP is an interactive protocol, the application is limited to cases where the prover and verifier are synchronized. ZKP is applied to the blockchain in different ways to verify transactions or provide privacy for users in the blockchain, such as ZKP-based cryptocurrencies and ZKP-based anonymous credential system. Although ZKP can protect the identity of users from being leaking, the computational cost of execution is high.

2) Secure Multiparty Computation: SMPC separates data or program states among multiple participants by using secret sharing [38]. In the secret sharing-based approach, the parties do not play a unique role. Parties are required to jointly calculate a function over their inputs and protect the privacy of the input data, the purpose of which is to generate an output value and also to reveal data. Each party does not receive all the input but only a portion of the data and keep a point on a different polynomial to characterize variables that set part of the data.

A decentralized mixing service is able to mitigate the Denial-of-Service (DoS) threats caused by centralized services. SMPC is one of the methods to achieve the decentralized mixing process without the need for a third-party anonymity proxy. Therefore, it is suitable for blockchain, because it is closer to the distributed structure of the blockchain. In the cryptographic method, it must be ensured that the majority of participants are honest to achieve secure computation. After computing, what the parties can get from it is what they learned from the output and their input. SMPC is a relatively simple encryption method used in the blockchain, which is compatible with existing blockchain networks. There are no unique consensus mechanisms in the application process, which means they need fewer resources to implement.

3) Succinct Transparent Arguments of Knowledge: STARK is similar to ZKP technology, but its structure provides more possibilities for blockchain networks. The STARK system is an automated protocol that replaces manual auditors to ensure the computational integrity of confidential data for any efficient computation, which can eliminate corruptibility and reduce costs. This type of proof not only allows one party to prove the existence of particular information but also shows that the party involved is already aware of the data. Most

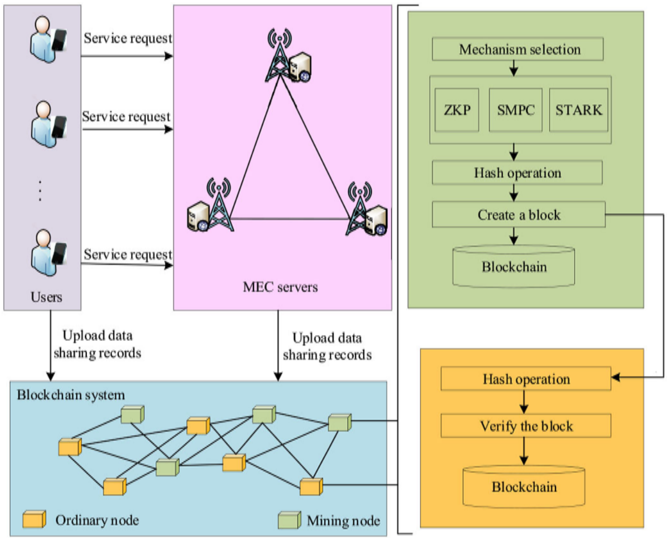


Fig. 2. Architecture of our scheme.

important of all, STARK protocols have the property of transparency, which makes the randomness used by the verifier to be public. Transparency is crucial to advance public trust because it can strictly limit the abuse of systems by the most powerful parties, so as long as there are unpredictable things in the observable environment, a transparent system is a system that the public reliably trust. STARK is an excellent method for data confidentiality. It has six core advantages: 1) confidentiality (ZK); 2) transparency; 3) postquantum security; 4) scalable verification; 5) argument of knowledge; and 6) universality. STARK can achieve privacy preservation for the blockchain, but it requires a lot of computing resources.

B. Adaptive Privacy-Preserving Scheme

Fig. 2 shows the architecture of our proposed privacy-preserving scheme. Data sharing transactions are uploaded to the blockchain system, and further handled in the blockchain.

1) Mechanism Selection: We need to select a suitable privacy-preserving mechanism to hide the connection between the user's real identity and the transaction. When users need to upload data to the blockchain, they will tell the blockchain system their level of privacy with the aim to avoid the resource waste. Anonymity shows that an adversary must not infer any private information from the link between a user's pseudonym and transactions. For mechanism selection, we only consider the computational overhead of the mechanism operation [39].

2) Block Producer Selection: After selecting a suitable mechanism, block producers need to be determined to record these transactions into the blockchain. Nodes in the blockchain periodically pack their available resources into a block for broadcast to all nodes. Each node votes to select N delegators as block producers (consensus nodes) according to its ranks of the final available resources for candidates and the number of stakes. The others act as ordinary nodes.

3) Block Generation and Verification: Assume that these N block producers take turns to generate blocks. The leader that generates a block in each round is called the speaker, and the other nodes (including ordinary nodes) are called members. The responsibility of the speaker is to broadcast the new block proposal to members. In the consensus process, the delegated Byzantine fault tolerance (dBFT) consensus mechanism is adopted to verify the block. Suppose there are K nodes, which satisfy $K \geq 3f + 1$, where denotes the maximum number of fault-tolerant nodes. The members are responsible for voting on the new block proposal. When the members exceeding $K - f$ agree to the proposal, the new block is added to the blockchain.

The privacy protection mechanism is obtained by the privacy-preserving algorithm. A optimization problem is formulated below, which includes three optimal variables, i.e., the adopted privacy-preserving algorithm denoted, the determined block interval, and the made offload policy. By using an DRL approach, the optimal solution to this formulated optimization problem can be obtained, which includes the selected privacy-preserving algorithm.

IV. Problem Formulation

In this section, an optimization problem of secure data sharing is formulated in the blockchain-enabled MEC system. Suppose that a mobile user requests data from the blockchain-enabled MEC system. The decision of data sharing should be performed at BSs, i.e., data sharing from D2D user or MEC server. It depends on two factors, i.e., the channel gain and the availability of the data stored at D2D users or MEC servers. Besides, after uploading the transactions (data sharing) to the blockchain system, the mechanism for protecting user identity privacy should be selected. In the selection process, we consider the impact of the available computing resources in MEC servers and the requirements of users' privacy. Meanwhile, we consider a dynamic scenario, where the channel conditions change over time. Therefore, we formulate the system as a discrete Markov decision process (MDP) problem, with the aim to make the maximization of the system reward. Since the state transition probability and the system reward cannot be predicted in advance, we present a model-free scheme based on DRL to address the MDP problem. We use a tuple $<\mathcal{S}, \mathcal{A}, \mathcal{P}, r>$, where $\mathcal{S}$ denotes the state space, $\mathcal{A}$ denotes the action space, $\mathcal{P}$ denotes the state transition probability, and r is the system reward.

A. System State and State Transition Probability

At the current decision epoch t, the state space is defined as a union of transmit rate $\boldsymbol{r}$, channel gain $(\boldsymbol{G}, \boldsymbol{G}^{\mathrm{D2D}})$, available computing resources of MEC servers $\boldsymbol{F}$, stake distribution $\boldsymbol{\Upsilon}$, level of user privacy $\boldsymbol{\alpha}$, and transaction size χ, which is expressed as

$$\mathcal{S}(t) \triangleq \left\{\left(\boldsymbol{G}(t), \boldsymbol{G}^{\mathrm{D2D}}(t)\right), \boldsymbol{r}, \boldsymbol{F}, \boldsymbol{\Upsilon}, \boldsymbol{\alpha}, \chi\right\}. \tag{15}$$

Since the state space is continuous, we assume that the probability of being in a certain state is zero. Then, after taking the action $a(t) \in \mathcal{A}(t)$, the probability of the current state $s(t)$ transition to the next state $s(t+1)$ is written as

$$\Pr(s(t+1)\,|\,s(t), a(t)) = \int_{\mathcal{S}^{t+1}} f\left(s(t), a(t), s'\right) ds' \tag{16}$$

where f represents the state transition probability density function.

B. Action Space

The system action consists of the privacy-preserving algorithm $\delta = \{0, 1, 2\}$, block interval $T_b = g(\delta, \boldsymbol{\alpha})$, and whether to obtain data from MEC servers or D2D users $\boldsymbol{a} = \{0, 1\}$, where $g(x, y)$ is a function indicating that its value is related to x and y. Then, the action space is expressed as

$$\mathcal{A}(t) \triangleq \{\delta, T_b, \boldsymbol{a}\}. \tag{17}$$

C. Reward Function

An optimization problem is formulated to minimize the weighted sum of the energy consumption of the MEC system and the transactional throughput of the blockchain system

$$\begin{aligned} \min_{\mathcal{A}^t} \quad & \mathbb{E}\left[\sum_{t=0}^{T-1} \varpi_1 E_{\text{total}} - (1-\varpi_1)\varpi_2 \Psi\right] \\ \text{s.t.} \quad & (C1) : \boldsymbol{\tau} \leq g(\delta, \boldsymbol{\alpha}) \\ & (C2) : T_{\text{total}} \leq \omega \times T_b \\ & (C3) : a_n \in \{0, 1\} \quad \forall n \in N. \end{aligned} \tag{18}$$

The agent gets a reward from the interactive environment based on the current state and action. Since the reward function is connected with the objective function, in the case, we employ the objective function as the reward function after meeting certain constraints. Particularly, the objective function is defined as the weighted sum of the energy consumption of MEC system and the transactional throughput of blockchain system. For the MEC system, BSs will consume energy when mobile users obtain data from MEC servers or D2D users will consume energy when sharing data via D2D communications. We use energy consumption as a performance metric for the MEC system. However, for the blockchain system, it is expected to process more transactions per second, so we adopt the transaction throughput as the performance metric.

The system reward for secure data sharing is defined as the weighted sum of the energy consumption and transactional throughput, which is a function based on the system states and actions, as follows:

$$r_t = \begin{cases} -O(t), & \text{if } C1 - C3 \text{ are satisfied} \\ 0, & \text{otherwise} \end{cases} \tag{19}$$

where $O(t) = \varpi_1 E_{\text{total}} - (1-\varpi_1)\varpi_2 \Psi$.

V. Asynchronous Advantage Actor-Critic Algorithm

Whether the reward can be optimized is determined by the system actions. In this section, the A3C algorithm in our network model is adopted to help obtain an optimization strategy that can maximize the system reward.

A3C is one of DRL algorithms, which is simpler, faster, and more robust when compared with the other DRL algorithms, e.g., actor–critic learning (AC) and deep deterministic policy gradient (DDPG). A3C takes the advantages of value-based method and policy-based method, which can work in discrete and continuous action spaces. By utilizing multiple CPU threads on a single machine, A3C is capable of efficiently obtaining optimal policies using asynchronous actor learners.

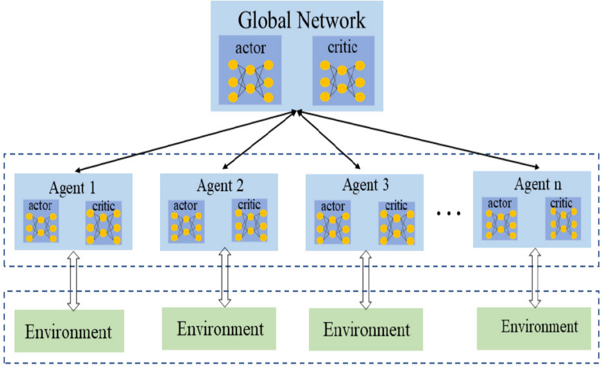


Fig. 3. Asynchronous advantage actor–critic algorithm.

The A3C algorithm consists of multiple actor learners and a global network, as shown in Fig. 3. Especially, asynchronous multithreads are utilized as multiple actors. All training weights are stored in the global network. At the beginning of the training, the global network sends parameters to these actor learners at the same time. Then, the actor learners interact with their environment parallelly to update their network parameters without interfering with each other. The trained parameters are uploaded to the global network. After certain time, the global network updates new parameters to the actor learner with the aim to guarantee that they can share a common policy.

It is needed for the A3C algorithm to maintain a policy $\pi(a_t|s_t; \theta)$ that is output by a set of action probabilities and the estimate of a value function $V(s_t; \theta_v)$ that evaluates how good a certain state is. The policy and the value function are determined by the parameter θ and the parameter θ_v, respectively. The policy $\pi(a_t|s_t; \theta)$ and the value function $V(s_t; \theta_v)$ are approximated by leveraging a single convolutional neural network. Peculiarly, the policy function and the estimate of the value function are outputted by a softmax layer and by a linear layer, respectively. The agents in the A3C algorithm update the policy (the act) and the rule by using the estimated value function (the critic) and an advantages function, respectively.

By utilizing k steps rewards, the discounted return is given by

$$R_t(\theta_v) = \sum_{i=0}^{k-1} \gamma^i r_{t+i} + \gamma^k V(s_{t+k}; \theta_v) \tag{20}$$

where k is upper bounded by a maximum value $t_{\max}$, r_{t+i} indicates the immediate reward, and γ represents the discount factor with (0, 1].

Based on (20), the advantage function can be written as

$$A(a_t, s_t; \theta, \theta_v) = R_t(\theta_v) - V(s_t; \theta_v). \tag{21}$$

The loss function of the policy is defined as

$$f_\pi(\theta) = \log \pi(a_t|s_t; \theta)(R_t - V(s_t; \theta_v)) + \beta H(\pi(s_t; \theta)). \tag{22}$$

By differentiating $f_v(\theta_v)$ with respect to θ_v, the following equation holds:

$$\nabla_\theta f_\pi(\theta) = \nabla_\theta \log \pi(a_t|s_t;\theta)(R_t - V(s_t;\theta_v)) + \beta \nabla_\theta H(\pi(s_t;\theta)). \tag{23}$$

The loss function of the estimated value function is expressed as

$$f_v(\theta_v) = (R_t - V(s_t;\theta_v))^2. \tag{24}$$

By differentiating $f_\pi(\theta)$ with respect to θ, the following equation holds:

$$\nabla_{\theta_u} f_v(\theta_v) = 2(R_t - V(s_t;\theta_v))\nabla_{\theta_v} V(s_t;\theta_v). \tag{25}$$

We minimize the loss function by using the RMSProp algorithm, by which the gradient estimate is calculated as

$$g = \alpha g + (1-a)\Delta\theta^2 \tag{26}$$

where a indicates the momentum, and $\Delta\theta$ represents the accumulated gradient of the loss function.

Then, we can update the RMSProp algorithm based on the following estimated gradient downhill:

$$\vartheta \leftarrow \theta - \eta \frac{\Delta\theta}{\sqrt{g+\varepsilon}} \tag{27}$$

where Δ indicates the learning rate, and ε represents a small positive value.

The A3C-based data sharing and resource allocation algorithm is given in Algorithm 1.

Algorithm 1 A3C-Based Data Sharing and Resource Allocation Algorithm

```
Initialization:
  • Set the parameters of the actor network and critic network in the global network to be $\theta$ and $\theta_v$, respectively.
  • Set the parameters of the actor network and critic network in the local network to be $\theta'$ and $\theta'_v$, respectively.
  • Set $T = 0$, $t = 1$, $\epsilon$, learning rate $\eta$, $T_{\max}$, and $t_{\max}$.
  • Set the number of the local networks $W$.
Iteration:
while $T < T_{\max}$ do
  for $w = 1$ to $W$ do
    Set $\theta' = \theta$, $\theta'_v = \theta_v$, and $t_0 = t$.
    Obtain the state space $\mathcal{S}(t)$.
    repeat
      Obtain the action space $\mathcal{A}(t)$ based on the policy $\pi(\mathcal{A}(t)|\mathcal{S}(t);\theta')$.
      Perform the action $\mathcal{A}(t)$, obtain reward $\mathcal{R}(t)$, and observe the next state $\mathcal{R}(t)$.
      $t = t + 1$.
    until $t - t_0 == t_{\max}$
    if $t\%t_g == 0$ then
      $R = V(\mathcal{S}(t);\theta'_v)$.
    end if
    for $i = t - 1$ to $t_0$ do
      $R = \mathcal{R}(t) + \gamma R$.
      Obtain value gradient $\nabla_{\theta'_v} f_v(\theta'_v)$ based on (23), Compute $d\theta_v = d\theta_v + \nabla_{\theta'_v} f_\pi(\theta'_v)$.
      Obtain policy gradient $\nabla_{\theta'} f_\pi(\theta')$ based on (25), and compute $d\theta = d\theta + \nabla_{\theta'} f_\pi(\theta')$.
    end for
    Asynchronously update global network parameters $\theta$ and $\theta_v$ based on (27).
  end for
end while
```

VI. Numerical Results

We evaluate the performance of our proposed data sharing scheme through extensive simulations in this section. A computer with six CPU cores is used, and the CPU is Intel Core i5-8400 with 32-GB memory. We use Tensorflow 1.10.0 with Python 3.6 on Ubuntu 18.04.2 LTS as the software environment. As for the blockchain system, virtualization is used distributed ledger technology. The simulation settings are introduced as follows. There are 20 users and 5 MEC servers randomly distributed in a 1×1 km^2 area. The transmission power is uniformly distributed in [0.1 2] W. The total computation resources of MEC servers are randomly taken from $[2, 4, 6, 8]$ GHz. The maximum block size and the maximum block interval are 8 MB and 10 s, respectively. The stake of blockchain nodes follows uniform distribution with $\Upsilon \in [5, 30]$. The level of user privacy takes form [0.2, 0.5, 0.8].

To verify the performance of the proposed scheme, the following schemes are considered as benchmark algorithms.

1) *Proposed Scheme With Fixed Privacy Scheme:* The scheme is the same as the proposed scheme except that the privacy preserve policy is fixed.
2) *MEC Scheme:* The scheme that shares data only through the MEC system.
3) *D2D Communication Scheme:* The scheme that shares data only D2D communications.

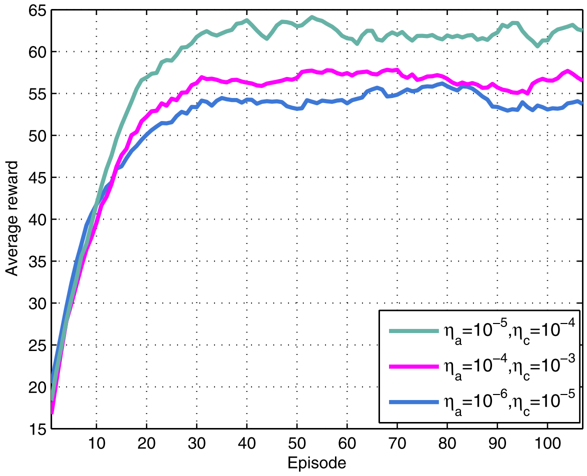


Fig. 4. Convergence performance with different learning rates.

Fig. 4 shows the convergence performance of our proposed scheme under different learning rates. From the figure, it is observed that the average reward of the proposed scheme

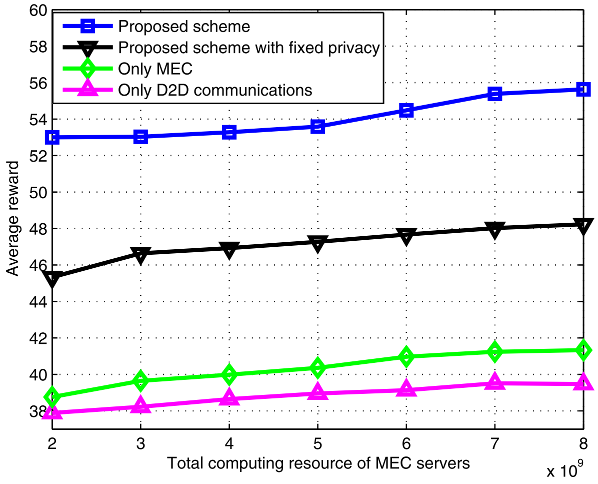


Fig. 5. Average reward versus total computing resource.

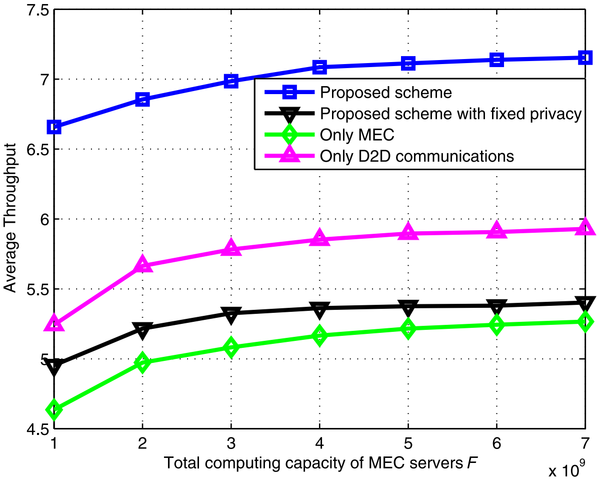


Fig. 6. Average throughput versus total computing resource.

at the beginning of the learning process is extremely low. The average reward increases with increasing the number of episodes until it converges to a relatively stable value. We can observe the proposed scheme converges around 30. Besides, it is seen that the average reward for a large learning rate (i.e., $\eta_a = 10^{-4}$ and $\eta_c = 10^{-3}$, where η_a and η_c are the actor's learning rate and the critic's learning rate, respectively) is significantly higher than the average reward for a small learning rate (i.e., $\eta_a = 10^{-6}$ and $\eta_c = 10^{-5}$). However, a large learning rate does not mean that the performance of the system is always good. For example, when the learning rate is $\eta_a = 10^{-5}$ and $\eta_c = 10^{-4}$, the performance is the best for the proposed scheme. Therefore, in the rest of simulations, we set the learning rate η_a to 10^{-5}, and η_c to 10^{-4}.

Fig. 5 shows the impact of the total computing resources of MEC servers on the average reward. From Fig. 5, we can observe that the average reward of all schemes increases slowly with the increase in the computing resources. That is because when the available computing resources of blockchain system increase, the throughput rises accordingly, as shown in Fig. 6. However, we can observe that the performance of the proposed scheme is better than the one of the other schemes.

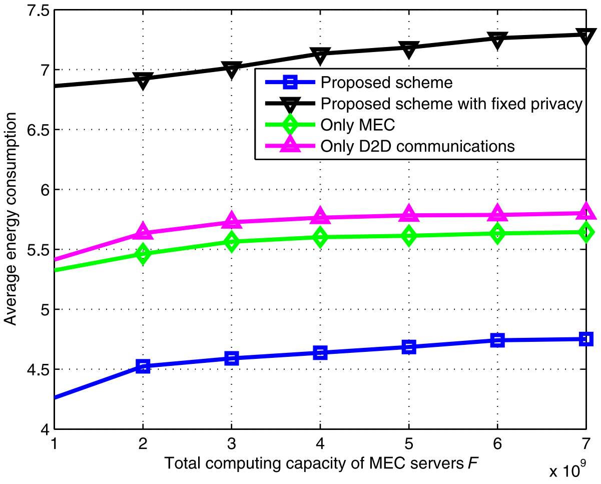


Fig. 7. Average energy consumption versus total computing resource.

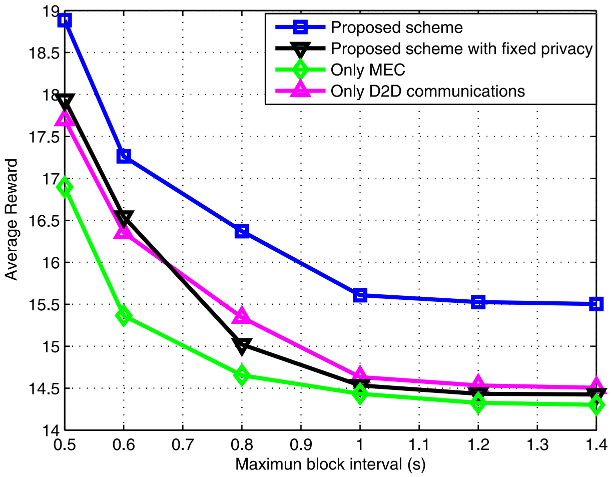


Fig. 8. Average reward versus maximum block interval.

In Fig. 7, we show the impact of the total computing resource of MEC servers on the average energy consumption. It shows that the average energy consumption of all schemes increases slowly with the increase in the computing resource. That is because the available computing resources of the MEC system increase, resulting in an increase in the energy consumed by MEC servers for task implementation.

Fig. 8 shows the impact of maximum block interval on the average reward. From the figure, we can see that the average reward of all schemes decreases with the increase in the maximum block interval. However, the performance of the proposed scheme is the best. Fig. 9 shows the impact of computing resource on the average throughput of blockchain system and the average energy consumption of the MEC system, respectively. We can find that as the computing resources change, the performance of the two subsystems hardly changes, thus achieving the tradeoff between the two subsystems.

The comparison of the objective value versus the number of mobile devices and the number of MEC servers is as shown in Fig. 10, where N and M indicate the number of devices and the one of MEC servers, respectively. As can be seen from

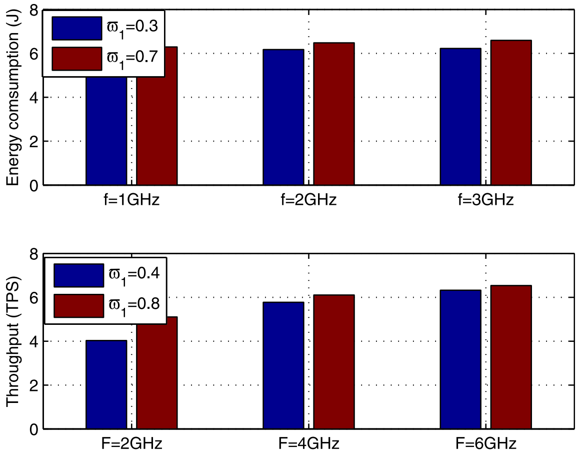


Fig. 9. Energy consumption and throughput versus total computing resource.

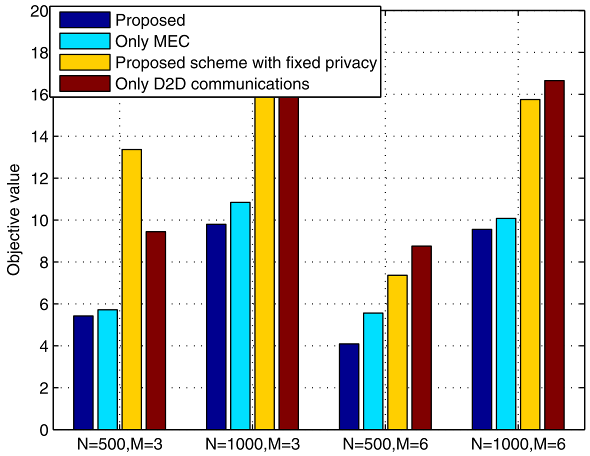


Fig. 10. Average reward versus the number of mobile devices and edge servers.

the figure, with the number of mobile devices increases, the objective value keeps increasing due to the fact that the computational resource allocation is limited. On the other hand, when the number of MEC servers increases, the objective value is reduced. This can be explained that with the number of MEC servers increasing, the speed of the blockchain system processing each transaction is increased. In addition, the available computing resources allocated to each mobile device is also increased, which helps in task processing.

VII. Conclusion

In this article, we have studied a blockchain-enabled secure data-sharing framework in MEC system. In the framework, an adaptively privacy-preserving mechanism has been proposed to protect users' identity privacy in data sharing. Besides, the performance of the MEC system and blockchain system was jointly optimized. In particular, the energy consumption and throughput were taken as the performance metrics of the MEC system and the blockchain system, respectively. Since the wireless channel fading and processing queues of MEC servers have Markov property, the optimization problem has been formulated an MDP in consideration of the dynamic environments. To handle the dynamics and complexity of the system, we have employed an A3C-based DRL algorithm to obtain the optimal solution. Simulation results have shown the performance of the proposed algorithm. Our future work will consider the privacy-preserving scheme in the blockchain-enabled MEC system.

References

[1] H. Li, K. Ota, and M. Dong, "Learning IoT in edge: Deep learning for the Internet of Things with edge computing," *IEEE Netw.*, vol. 32, no. 1, pp. 96–101, Jan./Feb. 2018.

[2] X. Wang, X. Li, S. Pack, Z. Han, and V. C. M. Leung, "STCS: Spatial–temporal collaborative sampling in flow-aware software-defined networks," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 6, pp. 999–1013, Jun. 2020.

[3] T. Yang, Z. Jiang, R. Sun, N. Cheng, and H. Feng, "Maritime search and rescue based on group mobile computing for UAVs and USVs," *IEEE Trans. Ind. Informat.*, vol. 16, no. 12, pp. 7700–7708, Dec. 2020, doi: 10.1109/TII.2020.2974047.

[4] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 968–979, May 2020, doi: 10.1109/JSAC.2020.2980802.

[5] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Blockchain empowered asynchronous federated learning for secure data sharing in Internet of Vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4298–4311, Apr. 2020.

[6] C. Chen, C. Wang, T. Qiu, M. Atiquzzaman, and D. O. Wu, "Caching in vehicular named data networking: Architecture, schemes and future directions," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 4, pp. 2378–2407, 4th Quart., 2020.

[7] L. Liu, C. Chen, T. Qiu, M. Zhang, S. Li, and B. Zhou, "A data dissemination scheme based on clustering and probabilistic broadcasting in VANETs," *Veh. Commun.*, vol. 13, pp. 78–88, Jul. 2018.

[8] R. Li, C. Shen, H. He, X. Gu, Z. Xu, and C. Xu, "A lightweight secure data sharing scheme for mobile cloud computing," *IEEE Trans. Cloud Comput.*, vol. 6, no. 2, pp. 344–357, Apr. 2018.

[9] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Commu. Surveys Tuts.*, vol. 22, no. 2, pp. 869–904, 2nd Quart., 2020, doi: 10.1109/COMST.2020.2970550.

[10] J. Du, F. R. Yu, G. Lu, J. Wang, J. Jiang, and X. Chu, "MEC-assisted immersive VR video streaming over terahertz wireless networks: A deep reinforcement learning approach," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 9517–9529, Oct. 2020.

[11] J. Du, L. Zhao, X. Chu, F. R. Yu, J. Feng, and C.-L. I, "Enabling low-latency applications in LTE-A based mixed fog/cloud computing systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1757–1771, Feb. 2019.

[12] J. Feng, F. R. Yu, Q. Pei, J. Du, and L. Zhu, "Joint optimization of radio and computational resources allocation in blockchain-enabled mobile edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 6, pp. 4321–4334, Jan. 2020, doi: 10.1109/TWC.2020.2982627.

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

[14] L. Liu, C. Chen, Q. Pei, S. Maharjan, and Y. Zhang, "Vehicular edge computing and networking: A survey," *Mobile Netw. Appl.*, to be published, doi: 10.1007/s11036-020-01624-1.

[15] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Towards secure blockchain-enabled Internet of Vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2906–2920, Mar. 2019.

[16] Y. Song, Y. Fu, F. R. Yu, and L. Zhou, "Blockchain-enabled Internet of Vehicles with cooperative positioning: A deep neural network approach," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3485–3498, Apr. 2020.

[17] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge Ai: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Netw.*, vol. 33, no. 5, pp. 156–165, Sep./Oct. 2019.

[18] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, "When mobile blockchain meets edge computing," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 33–39, Aug. 2018.

[19] Y. Dai, D. Xu, S. Maharjan, Z. Chen, Q. He, and Y. Zhang, "Blockchain and deep reinforcement learning empowered intelligent 5G beyond," *IEEE Netw.*, vol. 33, no. 3, pp. 10–17, May/Jun. 2019.

[20] Z. Zhou, B. Wang, M. Dong, and K. Ota, "Secure and efficient vehicle-to-grid energy trading in cyber physical systems: Integration of blockchain and edge computing," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 50, no. 1, pp. 43–57, Jan. 2019.

[21] Y. Zhang, Y. Dai, D. Xu, K. Zhang, and S. Maharjan, "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, doi: 10.1109/TVT.2020.2973705.

[22] J. Herrera-Joancomartí, "Research and challenges on bitcoin anonymity," in *Data Privacy Management, Autonomous Spontaneous Security, and Security Assurance*. Cham, Switzerland: Springer, 2014, pp. 3–16.

[23] F. Gao, L. Zhu, M. Shen, K. Sharif, Z. Wan, and K. Ren, "A blockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks," *IEEE Netw.*, vol. 32, no. 6, pp. 184–192, Nov. 2018.

[24] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "HAWK: The blockchain model of cryptography and privacy-preserving smart contracts," in *Proc. IEEE Symp. Security Privacy*, 2016, pp. 839–858.

[25] S. Alboaie and D. Cosovan, "Private data system enabling self-sovereign storage managed by executable choreographies," in *Proc. IFIP Int. Conf. Distrib. Appl. Interoper. Syst.*, 2017, pp. 83–98.

[26] I. Miers, C. Garman, M. Green, and A. D. Rubin, "ZeroCoin: Anonymous distributed e-Cash from bitcoin," in *Proc. IEEE Symp. Security Privacy*, May 2013, pp. 397–411.

[27] 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. 2018.

[28] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, Jun. 2020.

[29] O. Samuel, N. Javaid, M. Awais, Z. Ahmed, M. Imran, and M. Guizani, "A blockchain model for fair data sharing in deregulated smart grids," in *Proc. IEEE Glob. Commun. Conf.*, 2019, pp. 1–7.

[30] D. Li, R. Du, Y. Fu, and M. H. Au, "Meta-key: A secure data-sharing protocol under blockchain-based decentralized storage architecture," *IEEE Netw. Lett.*, vol. 1, no. 1, pp. 30–33, Mar. 2019.

[31] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1495–1505, Apr. 2019.

[32] N. Lasla, M. Younis, W. Znaidi, and D. B. Arbia, "Efficient distributed admission and revocation using blockchain for cooperative its," in *Proc. 9th IFIP Int. Conf. New Technol. Mobility Security (NTMS)*, 2018, pp. 1–5.

[33] M. S. Alam, J. W. Mark, and X. S. Shen, "Relay selection and resource allocation for multi-user cooperative OFDMA networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 5, pp. 2193–2205, May 2013.

[34] Y. Li, T. Jiang, M. Sheng, and Y. Zhu, "QoS-aware admission control and resource allocation in underlay device-to-device spectrum-sharing networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 11, pp. 2874–2886, Nov. 2016.

[35] K. Gai, Y. Wu, L. Zhu, L. Xu, and Y. Zhang, "Permissioned blockchain and edge computing empowered privacy-preserving smart grid networks," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7992–8004, Oct. 2019.

[36] 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, doi: 10.1109/JIOT.2019.2961707.

[37] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof systems," *SIAM J. Comput.*, vol. 18, no. 1, pp. 186–208, 1989.

[38] I. Damgård, M. Geisler, M. Krøigaard, and J. B. Nielsen, "Asynchronous multiparty computation: Theory and implementation," in *Proc. Int. Workshop Public Key Cryptography*, 2009, pp. 160–179.

[39] E. Ben-Sasson, I. Bentov, Y. Horesh, and M. Riabzev, "Scalable, transparent, and post-quantum secure computational integrity," IACR Cryptol. ePrint Arch., 2018, p. 46.

Lei Liu (Member, IEEE) received the B.Eng. degree in communication engineering from Zhengzhou University, Zhengzhou, China, in 2010, and the M.Sc. and Ph.D degrees in communication engineering from Xidian University, Xi'an, China, in 2013 and 2019, respectively.

From 2013 to 2015, he worked in a technology company. From 2018 to 2019, he was supported by China Scholarship Council to be a visiting Ph.D. student with the University of Oslo, Oslo, Norway. He is currently a Lecture with the Department of Electrical Engineering and Computer Science, Xidian University. His research interests include vehicular *ad hoc* networks, intelligent transportation, mobile-edge computing, and Internet of Things.

Jie Feng (Member, IEEE) received the Ph.D. degree in communication and information system from Xidian University, China, in 2020.

She is currently an Associate Professor with the Department of Electrical Engineering and Computer Science, Xidian University, Xi'an, China. From 2018 to 2019, she was with Carleton University, Ottawa, ON, Canada, as a visiting Ph.D student. Her current research interests include mobile-edge computing, blockchain, deep reinforcement learning, device to device communication, resource allocation and convex optimization, and stochastic network optimization.

Qingqi Pei (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees in computer science and cryptography from Xidian University, Xi'an, China, in 1998, 2005, and 2008, respectively.

He is currently a Professor and a member of the State Key Laboratory of Integrated Services Networks. His research interests focus on cognitive network, data security, and physical layer security.

Prof. Pei is also a Professional Member of the ACM and a Senior Member of the Chinese Institute of Electronics and China Computer Federation.

Chen Chen (Senior Member, IEEE) received the B.Eng., M.Sc., and Ph.D. degrees in electrical engineering and computer science (EECS) from Xidian University, Xi'an, China, in 2000, 2006, and 2008, respectively.

He is currently an Associate Professor with the Department of EECS, Xidian University, where he is also the Director of Edge Computing with the Engineering Technology Transfer Center, and the Associate Director of the Intelligent Transportation Research Laboratory. He was a Visiting Professor with the EECS, University of Tennesseein, Knoxville, TN, USA, from January 2013 to January 2014. He has contributed to the development of two copyrighted software systems and invented 40 patents. He has authored/coauthored two books, over 70 scientific papers in international journals and conference proceedings.

Dr. Chen serves as the general chair, the PC chair, the workshop chair or a TPC member of a number of conferences. He is a Senior Member of China Computer Federation and a member of ACM, Chinese Institute of Electronics.

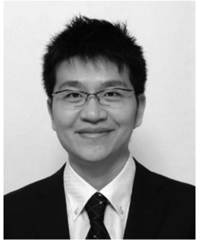
Yang Ming (Member, IEEE) received the B.S. and M.S. degrees in mathematics from Xi'an University of Technology, Xi'an, China, in 2002 and 2005, respectively, and the Ph.D. degree in cryptography from Xidian University, Xi'an, in 2008.

He is currently a Professor with Chang'an University, Xi'an. His main research interests include cryptography and wireless network security.

Bodong Shang (Graduate Student Member, IEEE) received the B.S. degree in information science and technology from Northwest University, Xi'an, China, in 2015, and the M.S. degree in communication and information system from Xidian University, Xi'an, in 2018. He is currently pursuing the Ph.D. degree with the Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA.

His recent research interests include several aspects of wireless communication, such as MIMO systems, device-to-device communication, unmanned aerial vehicle, and vehicle-to-everything, contributed to the development of two copyrighted software systems and invented 40 patents.

Mr. Shang is a Senior Member of China Computer Federation and a member of ACM, Chinese Institute of Electronics.



Mianxiong Dong (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in computer science and engineering from the University of Aizu, Aizuwakamatsu, Japan, in 2006, 2008, and 2013, respectively.

He is the youngest ever Vice President and a Professor of Muroran Institute of Technology, Muroran, Japan. He was a JSPS Research Fellow with the School of Computer Science and Engineering, University of Aizu, and was a Visiting Scholar with the BBCR Group, University of Waterloo, Waterloo, ON, Canada, supported by JSPS Excellent Young Researcher Overseas Visit Program from April 2010 to August 2011. He was selected as a Foreigner Research Fellow (a total of three recipients all over Japan) by NEC C&C Foundation in 2011.

Dr. Dong is a recipient of the IEEE TCSC Early Career Award in 2016, the IEEE SCSTC Outstanding Young Researcher Award in 2017, the 12th IEEE ComSoc Asia–Pacific (AP) Young Researcher Award in 2017, the Funai Research Award in 2018, NISTEP Researcher in 2018 (one of only 11 people in Japan) in recognition of significant contributions in science and technology, the 2019 Best Paper Award for IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTING from IEEE Computer Society, and the 9th IEEE AP Outstanding Paper Award in 2020 from Communication Society. He is Clarivate Analytics in 2019 Highly Cited Researcher (Web of Science).