Efficient Data Trading and Placement in Blockchain-Based Edge Computing Systems

SONG YANG[1] (Senior Member, IEEE), KEMING QIU[1], FEI ZHANG[2], LU CAO[2],
FAN LI[1] (Member, IEEE), MIN TANG[2], AND LIEHUANG ZHU[3] (Senior Member, IEEE)

(Special Issue on Trustworthy AI-Enabled Edge Computing in Next-Generation Wireless Networks)

[1]Beijing Institute of Technology, Beijing 100081, China
[2]National Innovation Institute of Defense Technology, Academy of Military Science, Beijing 100071, China
[3]School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing, China

CORRESPONDING AUTHOR: F. ZHANG (e-mail: feizhang100@sina.com)

ABSTRACT Edge devices (e.g., smartphones, tablelet PC, IoT devices) are becoming more prevalent in people's daily lives. With advanced sensors and processors, these devices can create massive data. These data can be used for predictive maintenance, enhancing user experience, increasing productivity, etc. These valuable data allow data producers to sell to consumers directly to generate income. Blockchain and smart-contract technology can be used to ensure transactions to be unmodifiable and undeniable. This paper first proposes a blockchain-based data relay and transaction model for the data producer, relay and consumer in edge computing. We then present a new consensus mechanism Proof-of-Data-Trading (PoDT) by combining Proof-of-Work (PoW) mechanism with Proof-of-Stake (PoS) consensus mechanism, which enables the proposed blockchain system to reach consensus with low energy consumption for edge devices. Moreover, we develop an approximation algorithm to store encrypted copies of data items on relays with smaller costs. Extensive simulations show that our proposed blockchain system works efficiently in edge computing. It achieves up to 8.19% higher profit for the data producer with the help of relays and consumers using 84.6% less time to get the data item. In addition, the new consensus mechanism consumes 87% less time when compared with the traditional PoW consensus mechanism.

INDEX TERMS Blockchain, edge computing, data trading, data placement.

I. INTRODUCTION

WITH the arrival of 5G networks and the rapid development of the Internet of Things (IoT) [1], several innovative edge devices, such as drones, smartphones, smart cars and smart homes, have entered our daily lives and changed our lives. These edge devices are equipped with advanced sensors and processors that can create a massive amount of data. With the increasing volume of data generated at the edge devices, most of these data will be analyzed and processed locally and traded between peer edge devices without the involvement of remote clouds.

Traditionally, producers have used third-party platforms to sell their profitable content to consumers. For example, "we media" content creators can create videos, pictures, audio and other digital content and sell them to consumers through third-party platforms such as Shutterstock [2]. Under this transaction mechanism, a producer can collect and process the data generated by the IoT devices it owns and provide valuable data to consumers for income. The transaction mechanism relies on third-party platforms to manage users, data and payments. However, it is not practical to use third-party platforms for data storage and transaction in a distributed environment. First, there may be usability, reliability, security, and privacy issues with third-party cloud platforms [3]. Second, in a peer-to-peer edge computing, third-party platforms are not applicable. Even if we find

a trusted third-party entity, edge nodes can hardly afford the huge computing, storage, and communication overhead. Finally, the pricing of data traded on the third-party platform is decided by the third-party platform, and each transaction requires a certain percentage of income from the third-party platform, which greatly reduces the interests of the finder.

A blockchain is a distributed database or ledger that is shared and managed among the parties/nodes in a network. Compared with the traditional centralized ledger system, all nodes in the blockchain system jointly maintain the distributed ledger through a consensus mechanism, thus constructing an immutable and consistent state record across the network [4]. Blockchain can be an effective solution to facilitate data sharing and transactions in edge computing. First, the natural decentralized architecture of blockchain meets the requirements of many heterogeneous, decentralized edge nodes in edge computing scenarios. Second, the smart contract mechanism in blockchain helps to build data trading markets [5]. Third, blockchain-based cryptocurrencies can be an effective incentive to facilitate the discovery, storage and transaction of data at various edge nodes. Fourth, all transactions will be permanently stored in the network and need to be verified by a consensus mechanism, effectively preventing double payment and other problems.

Despite the advantages of blockchain technology for data trading, there still exist challenges in the process of data relaying and trading. On the one hand, in order for consumers to efficiently obtain data items, producers need to place encrypted copies of data items on a certain number of relays. Therefore, it is necessary to design an efficient transaction mechanism for data producers, relayers and consumers. On the other hand, the heterogeneity of devices in edge computing brings challenges to the relaying and trading of data items. The cost of relaying data items varies widely due to differences in bandwidth, storage, and so on. If the producer randomly chooses to place certain data items on relays, it will consume extra cost. Therefore, it is important to transmit the data to some cost-effective relay by using an efficient data placement algorithm.

This paper proposes a blockchain-based data relay and transaction model that reduces costs and delays by introducing relays and ensures reasonable profit for the data producers and relays. The data relaying and trading between different nodes are protected and enforced by the smart contract and blockchain. To further reduce transaction costs, we propose the Rounding-based Data Placement Algorithm (RDPA) for data producers to store data items on relays with low costs. Finally, we implement a blockchain system, including a green consensus mechanism, transactions and verification mechanisms. The Proof-of-Data-Trading (PoDT) consensus mechanism combines the advantages of Proof-of-Work (PoW) and Proof-of-Stake (PoS) consensus, and can publish blocks safely, stably, and with low energy consumption in edge computing. In addition, PoDT consensus can encourage the release and transaction of data items by reducing the difficulty of publishing blocks with

high contribution and good reputation nodes. In all, our contributions in this paper are as follows:

- We propose a blockchain-based data relay and transaction model for producers to sell data items and share the income with relay. We accordingly develop a smart contract-based protocol to ensure that the data selling and relaying are efficient and trackable.
- We propose a new green and stable blockchain consensus mechanism Proof-of-Data-Trading (PoDT) combining Proof-of-Work (PoW) and Proof-of-Stake (PoS) mechanism, which is suitable for edge devices with limited resource to generate new blocks with low energy consumption on edge devices and encourage producers to publish data items and facilitate transactions.
- We have devised a rounding-based data placement approximation algorithm for data producers to store data items on relays with low costs as much as possible.
- We have conduct simulations to verify the performance of the proposed blockchain-based data trading mechanism and data placement approximation algorithm.

The rest of this paper is organized as follows. In Section II we discuss some related work on edge networks, blockchain, consensus mechanism, and smart contracts. In Section III, we present the blockchain-based data relay and transaction model. Section IV develops our proposed blockchain system for data trading. In Section V, we present the Rounding-based Data Placement Algorithm (RDPA) and analyze its approximation ratio. In Section VI, we evaluate the proposed data trading mechanism and data placement algorithm. Finally, we conclude the paper in Section VII.

II. RELATED WORK
A. BLOCKCHAIN
As the core component of the blockchain, the consensus mechanism ensures the stable operation of the blockchain system. The mainstream blockchain consensus mechanisms mainly include PoW [6], PoS [7], Delegated-Proof-of-Stake (DPoS) [8] and Practical-Byzantine-Fault-Tolerance (PBFT) [9], etc. In [10], a PoS-PoW hybrid blockchain system based on edge computing is proposed to improve the security of transactions. The blockchain system takes into account the heterogeneity of edge devices and encourages the participation of resource-limited devices and resource-rich devices. Liu et al. [11] propose a blockchain-enabled decentralized information-sharing protocol in the zero-trust Internet of Things (IoT) environment, as well as designing an effective voting strategy, to guarantee entity authentication with privacy protection for their identity. Lin et al. [12] present a peer-to-peer (P2P) knowledge market based on a consortium blockchain with the Proof of Trading (PoT) consensus mechanism, in order to efficiently make knowledge tradable in the edge-AI enabled IoT environment.

The concept of smart contracts was proposed by Szabo [13] and implemented in Ethereum [14]. The realization of the smart contract is essentially by programming and deploying the digital object on the blockchain and then

triggering the automatic generation and execution of the contract, thereby changing the state of the digital object in the blockchain network [15]. In particular, Huang et al. [5] propose a smart contract-based protocol to protect the profits of data producers while allowing consumers to resell data legally, but the blockchain system design, consensus proof design and data cost of item transshipment are not taken into account. Recently, the application of smart contracts is becoming more and more pervasive, such as in the area of medical [16], [17], intellectual property [18] and IoT [19].

B. DATA TRADING AND PLACEMENT IN BLOCKCHAIN-BASED EDGE COMPUTING SYSTEM

Huang et al. [20] propose a blockchain system for edge computing, storing metadata items on blocks, and distributed storage of corresponding data items and blocks to achieve fast data access. Yuan et al. [21] present an blockchain-based data/task offloading system, where a reputation-based incentive mechanism and consensus mechanism are proposed. Zhang et al. [22] develop a reputation-based mechanism of modifying the verification strategy and optimizing the transaction forwarding and data request protocols in the blockchain environment to accelerate the speed of propagating transactions. Zhou et al. [23] first develop a blockchain-enabled image sharing and caching scheme to obtain fast image fetching in MEC. They later leverage Lyapunov optimization and convex optimization for a two-layer approach to solve the service entity deployment and User Equipment (UE) problem. Okegbile et al. [24] present an integrated blockchain-enabled cloud-edge computing platform, where data sharing is achieved between data owners and data requesters. Based on it, they characterize and investigate the effects of communication constraints on the validation process' performance including transaction success rate, transaction latency, and so on. Huang et al. [25] study the incentive assignment problem in a PoW and PoS hybrid consensus blockchain system. They present an iterative algorithm based on a two-stage Stackelberg game to solve this problem. Fan et al. [26] first present a two-layer blockchain-based framework to provide data integrity in edge computing, and then devise a Dynamic Random Byzantine Fault Tolerance (DR-BFT) consensus algorithm. Chen et al. [27] propose a blockchain-based decentralized storage network to achieve both scalability (by using dynamic replication) and reliability (by storing file as replicas in multiple sectors) of file storage. Danish et al. [28] address the storage selection problem for blockchain-based data application. They formulate this problem to a similar facility location problem and respectively present dynamic programming-based heuristic and a greedy algorithm to solve it. A neural network-based maintenance reconfiguration mechanism is also proposed.

Different from all above work, this paper is about designing an efficient and trackable blockchain-based data relay and transaction system together with a new green and stable consensus mechanism, in order to minimize
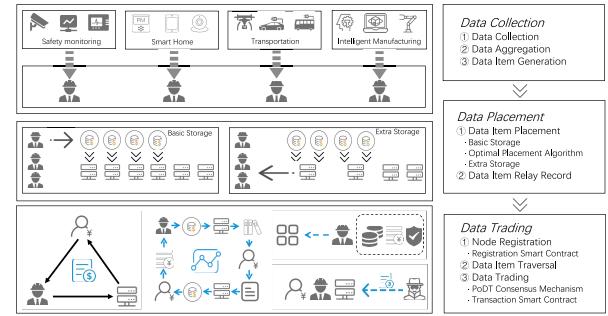

FIGURE 1. An overview of blockchain-based data relay and transaction model.

cost and delay consumption. Moreover, there are some works dealing with service/data placement in edge computing [29], [30], job/task allocation and placement in machine learning [31], [32], and virtual network function placement [33]. However, none of them can appropriately solve the data placement problem in blockchain-based edge computing environment as addressed.

III. BLOCKCHAIN-BASED DATA RELAY AND TRANSACTION MODEL

In this section, we introduce the blockchain-based data relay and transaction model. This model ensures the immutability and traceability of data item storage and transactions through blockchain technology. And the smart contract is used in the model to ensure the profit of the relay and producer and the credit of the node. We discuss the details of the blockchain system and consensus mechanism in Section IV. We discuss the storing, relaying and trading process in our proposed model. We then demonstrate the details for the "Basic Storage" part and the trading processes.

A. OVERVIEW

As shown in Fig. 1, the process of our proposed model can be divided into three parts, i.e., data collection, data placement and data relaying and trading. The storing, placing, relaying and trading between different nodes are protected by the smart contract and blockchain. In the data collection part, data producers collect massive sensory data from IoT devices and package the valuable data into data items. In the data placement part, producers upload data items into our proposed blockchain system. In the "Basic Storage", the system will allocate data items to a lowest-cost relay based on the storage and latency conditions based on our proposed data placement algorithm (see Section V). Meanwhile, conditional and willing relays can actively store data items. In the data trading part, producers, relays and consumers will jointly sign smart contracts, which will ensure the security of transactions, profit sharing between producers and relays, and the credit of nodes within the model.

1) ROLES OF NODES

In our proposed distributed system, we assume that all nodes are selfish and rational. Each node can select three roles.

The producer is the node that accepts and processes the massive perceptual information and data within its range, produces the data items, and broadcasts the data-info of the data item in the system. The producer then forwards the newly generated data item to the assigned relay according to the data placement algorithm, and concludes transactions with relays and consumers who wish to actively relay the data item. The relay is responsible for storing and relaying encrypted data items and delivers data items. Consumers are nodes that are willing to buy data items.

2) DISTRIBUTED STORAGE PROCESS

Basic Storing: As mentioned earlier, edge devices tend to have limited storage and bandwidth. In addition, the instability of edge nodes may lead to disconnection and data item loss. To ensure data items' basic availability and security, the producer should store an encrypted copy of the data item on certain cost-effective relays. The storage behaviour of the storing is compulsory. Therefore, the producer takes a cut of the sales revenue to give relays as an incentive to store and transfer data items. We call this "Basic Storage". Therefore, when a producer generates a data item, it encrypts it and generates data-info. The producer will optimally forward the encrypted data item copy to the optimal relay according to the resource placement algorithm we proposed to ensure the basic availability of the system and the security of the data items. After the data item is placed, the producer will broadcast the data-info in the network, and then other nodes in the system will wrap the received data-info items into a block. The optimal data placement algorithm we proposed can simultaneously find the best relay for storing encrypted copies of multiple data items produced over a while. The encrypted copies will be stored in relays with more resources left and easy to access instead of storing them randomly. The algorithm will also comprehensively consider the cost-effectiveness of transmission, the distance of the relay to potential consumers, and the storage willingness of the relay.

Extra Storing: The more data items a relay has, the more likely it is to participate in trade and relay activities and the higher the profit it will get. If relays have additional storage space and bandwidth, they can profit by actively storing and relaying additional data items. This also helps consumers get data items faster. The relay will traverse the data-info in the blockchain and then request the producer of the interested data-info to store the data item. To prevent malicious requests, the relay first needs to pay a margin to the producer in the form of a transaction. After the margin transaction takes effect, the producer will transmit the data item to the relay, and the relay will generate a digital signature for the received encrypted data item copy and send it to the producer. After the producer verifies the digital signature, a certain amount of reward will be sent to the relay, and the reward transaction will also be broadcast in the network in the form of a transaction and packaged in the block.

B. RELAYING PROCESS

In edge computing, storing data on multiple nodes can ensure the security of data items and make data items closer to potential consumers, improving availability and security and reducing the latency and cost for data relaying. Relays are nodes that have available storage and bandwidth and want to profit by storing and relaying data items. Producers and consumers want to complete transactions and transfer data items as soon as possible. After the consumer finds the data item it needs in the blockchain and sends a transaction request to the producer of the data item, the producer negotiates revenue sharing with relays that have encrypted copies of the data item and are willing to relay, and finally selects a relay to participate in the transaction. The three parties then sign the contract using their private keys.

The relays involved in the transaction send the encrypted copy of the data item to the consumer, who then obtains from the producer the key used to decrypt the encrypted copy of the data item. Consumers then pay for the data items. We assume that consumers will check the integrity and authenticity of our data items by comparing their digital signatures. The act of reselling data items and the circumstances of offline transactions are beyond the scope of this article.

C. TRADING PROCESS

The relay is designed to improve the availability and security of data items. Relays have storage space and bandwidth. They can profit by storing and relaying encrypted copies of data items. In the edge computing environment, the presence of relays can make it easier and more efficient for consumers to obtain data items and alleviate producers' storage and bandwidth pressure. In addition, the cache can generate many copies of data items, making the edge network more robust. When consumers traverse the data-info collection in the blockchain and find a data item they want to buy, they will send a transaction request to the producer of the data item. Then, the corresponding producer finds relays that have stored the data item and are near the consumer based on the transaction information in the blockchain and negotiates with them the revenue share, and finally chooses one of the relays to forward the data to the consumer based on the negotiation result item. It should be noted that to ensure the benefits of the relay in the system and the fairness of the transaction, the system does not allow producers to enter into transactions with consumers alone.

The producer will generate a three-way smart contract, including the producer, relay and consumer. The smart contract includes the data information of the data item, the sales price of the data item and the revenue-sharing proportion of the relay. The price of this data item is determined by the producer, and the income share ratio is negotiated between the producer and the relay. Then the three parties use their private keys to sign the contract. When the smart contract is written to the blockchain and takes effect, the relay sends an encrypted copy of the data item to the
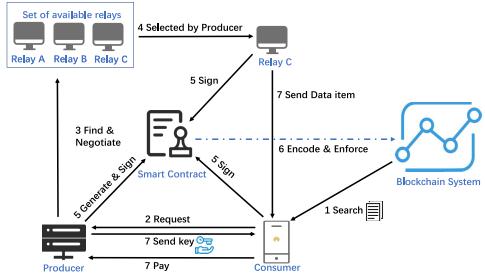
FIGURE 2. The trading process based on smart contract.

```
{
    "contentId": "11",
    "dataType": "PICTURE",
    "description": "data-info example",
    "hash": "gm2Jo5Z7EsUQmN+HS2dDi9b+JNIuBkOxlTVOTsGZhVE=",
    "location": "1GRMcgS6qjRjsyeTtJMmn5HEUaKdjMrU4J",
    "store": 626,
    "timeStamp": 1635081158470,
    "updateContentWalletAddress": "1Kq3UYmYisyXZe942PU6mnajJaGhhqxJKo",
    "validTime": 1635082750470,
    "version": 1.0,
    "signature": "9AE0BE40F138B0EDB9F55759A8E023F8"
}
```

FIGURE 3. A data-info example.

consumer, the producer sends the key to the consumer, and the consumer pays for the data item. If there are incorrect prices, wrong revenue share ratios and other misinformation in the contract, relays or consumers can directly deny signing the contract. Fig. 2 shows the data item sales and relay process.

IV. BLOCKCHAIN SYSTEM AND PODT CONSENSUS MECHANISM DESIGN

A. OVERVIEW

Our proposed system consists of the producer node, relay node and consumer node. Nodes are involved in the edge of the blockchain system device. Each node in the system has a private key and a public key to identify its identity. The private key of each node is based on elliptic curve cryptography. The principle of elliptic curve cryptography [34] can ensure that the private key of each node is unique. The public key is further used to generate the wallet address of the node. Inspired by Bitcoin [6], we use a one-way hash function to get the wallet address from the public key. The wallet address generated by the one-way hash function is very weakly readable, so we use Base58 [35] to encode the wallet address and get the final wallet address. This address encoded is a highly readable representation of the public key. This address is unique and associated with the node.

The blockchain system uses the UTXO model as the transaction model. The UTXO model [36] is a transaction model invented by Satoshi Nakamoto and is the transaction model used in Bitcoin [6]. And we use a binary Merkle tree to quickly verify the existence and integrity of the block data. Moreover, We design a energy-saving and stable consensus mechanism combining PoW [6] and PoS [7]. We discuss the details of the consensus mechanism in Section IV-D.

B. DATA-INFO DESIGN

Due to storage limitations in edge devices, storing all encrypted data items in every node is impractical. Therefore, the encrypted data item should only be stored on some nodes in the blockchain system. The data-info of the data item is stored in the blockchain. A data-info should contain the basic information of a data item. Consumers can find the producer of the data item based on the information of the data-info and verify the integrity of the data item.

The metadata is proposed for data sharing in peer edge nodes in [37]. Inspired by metadata, we design the data-info. After the producer packs the data item, he/she will generate the data-info of the data item based on the information of the data item. The data item producer then broadcasts the data-info items in the network and the node packages the received data-info items into a new block. The data-info includes the basic information of the associated data item, including the data type, size, description and the wallet address of the producer. Since the data-info item contains the information about the corresponding data item, and the data-info item is stored in the blockchain, consumers can have the information of all data items in the blockchain. The consumer can search for the content it needs in the blockchain and request data items from the producer.

To ensure the integrity of the data, the producer will generate a digital signature for the data item before encrypting it, encrypt the signature with its private key and write it into the data-info of the data. Consumers can verify data items after purchasing and decrypting. In addition, to ensure the timeliness of the data item, the producer will add the version number, timestamp, and expiration time to the data-info item. Fig. 3 shows an example of data-info.

C. CREDIT MECHANISM MODEL DESIGN

To improve the quality of data items and transactions in the system, punish producers for malicious behaviour, and allow producers to participate in transaction data items actively and maintain the blockchain system, we have introduced a credit mechanism in the blockchain system. In the credit mechanism, producers are supervised by consumers and relays, and producers with good credit will have a higher probability of obtaining accounting rights.

In our proposed blockchain network, a producer's credit is an evaluation of it by relays and consumers based on its behaviour. The credit mechanism we propose includes three types of credit, personal credit, comprehensive credit and valid historical credit. Personal credit refers to a single relay or consumer's evaluation of a certain producer that has ever interacted with. Each relay and consumer will maintain its credit list for producers based on their interaction records in the edge network. Since the personal credit list is maintained separately by a certain node, different relays and consumers may have different evaluations of a certain producer. After each transaction, the relay and the consumer will share the personal credit list in the blockchain network.

Before starting each round of consensus, producers need to obtain the latest personal credit in the current blockchain network to obtain the latest comprehensive credit. Due to the limited storage space of edge nodes, it is impossible to store all personal credit history in the block. In addition, to prevent the producer's credit value from changing too much in each consensus round. We have introduced the concept of valid historical credit. Historical valid credit aggregates all total credits in a certain historical period. The historical period is dynamically determined according to the number of registered producers in the system.

In addition, producers need to obtain valid historical credit from the latest block. After the producer's historical valid credit are obtained, the equity and mining difficulty value can be obtained according to the consensus algorithm in Section IV-D. In the consensus process, producers with higher historical valid credit will have a lower mining difficulty.

D. PROOF OF DATA TRADING CONSENSUS MECHANISM DESIGN

Proof of Work (PoW) is a traditional blockchain consensus mechanism utilized in Bitcoin [6] by Satoshi. Computationally intensive proof-of-work (PoW) is impractical and inefficient for edge devices with limited computing power and energy. Moreover, Proof-of-Stake (PoS) is another consensus mechanism, the PoS consensus essentially uses stake instead of hash power in PoW. The node with the highest stake in the system has a better chance of obtaining the block accounting right. But in a pure PoS consensus mechanism, the right to accounting comes from equity, so mining is almost cost-free (nothing-at-stake) [38]. There may be malicious nodes making fork chains, and the stability of the blockchain system cannot be guaranteed.

In the edge system, we emphasize that producers are the system's foundation. They often have sufficient computing power and resources, acting as miners in the blockchain system. We should encourage producers to continuously release many valuable data items and promote the transaction of data items as much as possible, instead of wasting computing power to solve a pure hash puzzle. In addition, to ensure the rights and interests of consumers and relays in the network and encourage producers to improve the quality of data items and transactions, we incorporate the credit mechanism into the consensus protocol. Therefore, inspired by previous work, we combine PoW and PoS mechanisms to propose a new consensus mechanism Proof-of-Data-Trading (PoDT).

In order to measure the contribution value of a particular producer p to the upload of data items in the system, we set a data share value (dataAge), and the value of dataAge is the sum of the product of a data item's value (dataValue) and its valid holding time length. For example, producer p releases a data item with data value of 5 in the system and holds it for 10 days after being valid, it will get 50

units of dataAge. In view of the infinite increase in node equity that may occur in PoS, PoDT sets a valid holding time (effTime) for every valid data-info. In order to limit the unlimited growth of the Stake, we limit the maximum value of valueTime to 90 days. If the holding time exceeds the maximum valid holding time, effTime stops increasing. In addition, the dataAge of the miner who gets the accounting right will be cleared. So, the dataAge of producer p in PoDT consensus mechanism is shown as:

$$dataAge_p = \sum_{d=1}^{|D|} dataValue_d \times effTime_d \qquad (1)$$

For producer p, $stake_p$ indicates its stake, TX_{pd} indicates the number of transactions of data item d owned by it in a period of time, and $tradCount_p$ means the total number of transactions that occurred in all data items owned by it in a period of time. The tradCount of producer p is given as:

$$tradCount_p = \sum_{d=1}^{|D|} TX_{pd} \qquad (2)$$

For producer p, $historyCredit_p$ indicates its historical valid credit. We set a certain historical period (historyPeriod), a certain historical period includes $\mathcal{T}$ consensus cycles, $t \in \mathcal{T}$. $compCredit_{pt}$ indicates the comprehensive credit of producer p in the t-th round of consensus. The comprehensive credit has a logarithmic increase in the historical effective credit, that is, the closer the comprehensive credit is to the most recently released block, the larger the proportion of the historical effective credit is calculated, λ and θ are time decay factors. The historical valid credit aggregation formula of producer p is shown as:

$$historyCredit_p = \sum_{t=1}^{\mathcal{T}} \lambda \times compCredit_{pt} \times \ln \theta t \qquad (3)$$

The stake of producer p is given as:

$$stake_p = \alpha \times dataAge_p + \beta \times tradCount_p \\ + \gamma \times historyCredit_p \qquad (4)$$

where α, β and γ are fixed parameters. Due to the openness of blockchain storing and trading records, $tradStake$ and $dataStake$ can be calculated by producer p and be verified by all nodes in the blockchain system. The stake calculation can promote producers to publish more valuable data items and accelerate the relay and transaction of data items as much as possible while maintaining credit. In the PoDT consensus mechanism, we dynamically adjust the difficulty of the hash puzzle according to producers' stakes. Therefore, the dynamic hash puzzle of PoDT consensus mechanism in the blockchain system is shown as:

$$Hash(Hash(preBlockHash, merkleRoot, timeStamp, \\ target, nonce)) \le target_{PoDT} \qquad (5)$$

$$target_p \propto stake_p = \min\{stake_p, stake_{max}\} \qquad (6)$$

TABLE 1. Notations.

Notation	Description
$\mathcal{N}$	The set of network edge nodes.
D	The set of $|D|$ data items.
D_k	The set of data items produced by producer k.
P	The set of producers.
θ_p	The accessing node of producer p.
r_p	The data transmission rate of accessing node of p.
Φ_n	The cost coefficient of placing d on node n.
S_d	The size of data file d.
$\lambda(n)$	The storage capacity of node n.
$c(u, v)$	The link capacity of (u, v).
$\mathcal{T}$	Total number of consensus cycles
$X_n^{p,d}$	A boolean value, indicating whether data producer p places data item d on node n.

$$target_{PoDT(p)} = target_{PoDT(standard)}$$
$$\times \frac{\min\{\min\{stake_p, stake_{max}\}, stake_{standard}\}}{stake_p} \quad (7)$$

where $target_{PoDT}$ and $stake_p$ have positive correlation. $stake_{max}$ is set to prevent the difficulty of mining from changing too much and causing instability. Besides, $stake_{standard}$ is set as the threshold of $stake_p$, if the value of $stake_p$ is less than the threshold, $target_{PoDT}$ will be equal to $stake_{standard}$. If the mining difficulty is the same, $target_{PoDT(standard)}$ is equivalent to the traditional $target_{PoW}$.

V. DATA PLACEMENT IN BLOCKCHAIN-BASED DATA RELAY AND TRANSACTION MODEL

After the (massive) data are collected by the data producers, valuable data are packed into data items that need to be placed on relays in order that the customers can access them reliably. However, since different relays have different costs and storage, and different node pairs have different delivering delay costs, it is non-trivial to design a cost-efficient data placement algorithm in the proposed blockchain-based edge computing system, which is the focus of this section. Table 1 provides a list of notations for ease of understanding.

A. PROBLEM DEFINITION AND FORMULATION

There is a set of edge nodes $\mathcal{N}$, a set of data items D and a set of producers P, $P \subset \mathcal{N}$. Each edge node $n \in \mathcal{N}$ which consists of a limited number of edge servers has a total capacity $\lambda(n)$. For each data item $d \in D$, we use S_d to denote its size. Each producer $p \in \mathcal{N}$ produces a variable number of data items D_k, $D_k \subset D$ and $D = D_1 \cup D_2 \cup \cdots \cup D_k \cup \cdots \cup D_P$. For each node pair (u, v) where $u, v \in \mathcal{N}$, the link capacity of (u, v) is $c(u, v)$. The accessing node of the producer is denoted as θ_p, and we use r_p to represent its data transmission rate. $t(\theta_p, n)$ denotes the time for data item d produced by producer θ_p to be routed from θ_p to node n and Φ_n^d denotes the cost coefficient of placing d on node n. Formally, the data placement problem can be defined as:

Definition 1: Given are a network $G(\mathcal{N}, \mathcal{L})$ and a set of producers P. For each data item $d \in D$, the data placement problem in the blockchain-based edge computing system is to place it on $\mathcal{N}$ with the minimum cost, such that the total load on each link must be no greater than the link capacity and the storage limit of each node cannot be exceeded.

The purpose of minimizing the placement cost $\sum_{p \in P} \sum_{d \in D} \sum_{n \in \mathcal{N}} X_n^{p,d} \cdot (t(\theta_p, n) + \Phi_n^d)$ is to save costs for the network, storage balance and avoid the network bottleneck, e.g., one or more data items may be placed on nodes with higher cost, which will undoubtedly consume a lot of resources of the producer, which is not suitable for resource-constrained edge environment. Moreover, some nodes have less free storage space and some links are highly loaded, while some other nodes have enough storage space and some other links are less loaded. In the worst case, there may be a situation where a large number of data items are congested due to a network bottleneck, or the data items are routed to a node that has no storage space and the data items cannot be placed.

We subsequently formulate this problem as an exact Integer Linear Programming (ILP) formulation, where the required notations and variable are defined in Table 1.

Objective:

$$\min \sum_{p \in P} \sum_{d \in D} \sum_{n \in \mathcal{N}} X_n^{p,d} \cdot \left(t(\theta_p, n) + \Phi_n^d \right) \quad (8)$$

Placement Constraints:

$$\sum_{n \in \mathcal{N}} X_n^{p,d} \geq 1 \quad \forall p \in \mathcal{P}, d \in \mathcal{D} \quad (9)$$

Storage Constraints:

$$\sum_{p \in \mathcal{P}} \sum_{d \in D} X_n^{p,d} \cdot S_d \leq \lambda(n) \quad \forall n \in \mathcal{N} \quad (10)$$

Link Capacity Constraints:

$$\sum_{p_i \in P_u} \sum_{d_i \in D} X_v^{p_i,d_i} \cdot r_{p_i} + \sum_{p_j \in P_v} \sum_{d_j \in D} X_u^{p_j,d_j} \cdot r_{d_j} \leq c(u, v)$$
$$\forall u, v \in \mathcal{N} \quad (11)$$

Equation (8) minimizes the sum of placing costs and delivering delay cost for all the data items. Equation (9) ensures that each produced data item must be placed on one node. Equation (10) ensures that the storage limit of each node cannot be exceeded. Equation (11) ensures that the total load on each link must be no greater than the link capacity.

B. APPROXIMATION ALGORITHM AND PERFORMANCE ANALYSIS

1) APPROXIMATION ALGORITHM

In this subsection, we present the Rounding-based Data Placement Algorithm (RDPA) in Algorithm 1 to solve the data placement problem in the blockchain-based edge computing system. The general idea of RDPA in Algorithm 1

Algorithm 1: Rounding-Based Data Placement Algorithm

Input: $\mathcal{G}(\mathcal{N}), P, D$

Output: Integral data placement solution $\overline{X_n^{p,d}}$

1 Construct an LP from the ILP in (8)–(11)

2 Solve this LP to obtain the fractional variables $\widetilde{X_n^{p,d}}$

3 **foreach** $p \in P$ **do**

4   **foreach** $d \in D$ **do**

5   **foreach** $n \in \mathcal{N}$ **do**

6   Set $\overline{X_n^{p,d}} = 1$ with probability $\widetilde{X_n^{p,d}}$.

7 Return data placement solution according to $\overline{X_n^{p,d}}$.

---

is to first construct Linear Programming (LP) from the ILP in (8)–(11) by letting $0 \leq X_n^{p,d} \leq 1$ and keep the objective and all other constraints the same. After that, RDPA solves this LP to obtain the fractional decision variables $X_n^{p,d}$ and rounds to integral solution for each producer $p \in P$, data item $d \in D$ and $n \in \mathcal{N}$ with the probability of $\widetilde{X_n^{p,d}}$.

The time complexity of RDPA is dominated by the LP. In [39], there is a proposed polynomial-time algorithm that can solve the LP with the current best worst-case complexity of $O([\frac{I^3}{\ln I}]\gamma)$ by using an interior-point method. Here, I is the number of variables and γ is the bit size of the problem (about the number of bits in its binary representation). Since there are in total $O(N|P||D| + N|P||D| + N^2|P||D|) = O(N^2|P||D|)$ variables in the LP, it leads to a total time complexity of $O(\frac{\gamma N^6 |P|^3 |D|^3}{ln(N^2|P||D|)})$ for RDPA.

2) PERFORMANCE ANALYSIS

Similar to our previous work [33], [40], we leverage the method of Upper Tail Chernoff bound [41] and Union Bound (Boole's inequality) [42, Ch. 4.7] to analyze the performance of RDPA. For completeness, we first formally define the definition of the Upper Tail Chernoff bound and Union Bound inequality:

Theorem 1: [41] Denote $x_1, x_2, \ldots, x_n$ to be n independent random variables, where $x_i \in [0, 1]$ for $1 \leq i \leq n$. By setting $\mu = E[\sum_{i=1}^{n} x_i]$, then for an arbitrary positive ϵ we have:

$$\Pr\left[\sum_{i=1}^{n} x_i \geq (1+\epsilon)\mu\right] \leq e^{\frac{-\epsilon^2 \mu}{2+\epsilon}} \qquad (12)$$

Theorem 2: Denote $A_1, A_2, \ldots, A_n$ to be n events with happening probability $\Pr[A_1], \Pr[A_2], \ldots, \Pr[A_n]$, then we have $\Pr[A_1 \cup A_2 \cup \ldots \cup A_n] \leq \sum_{i=1}^{n} \Pr[A_i]$.

Moreover, we define α to ensure that the following expected values are always fractional numbers:

$$\alpha = \min\left\{\min\left\{\frac{\min(\lambda(n))}{S_d}\right\}, \min\left\{\frac{\min c(u,v)}{r_{p_i} + r_{p_j}}\right\}\right\}$$
$$\forall n \in \mathcal{N}, u, v \in \mathcal{L}, d \in D, p_i, p_j \in P \qquad (13)$$

We first prove the node storage violating factor as follows:
Definition 2: For each node n, each producer p, and each data d, the load $z_n^{p,d}$ is defined as follows:

$$z_n^{p,d} = \begin{cases} S_d \text{ with prob. } \widetilde{X_n^{p,d}} \\ 0 \text{ otherwise} \end{cases}$$

Since $z_n^{p_1,d_2}$, $z_n^{p_2,d_2}$, ... are mutually independent based on their definitions, we can get that the expected load on node n is:

$$E\left[\sum_{p \in P} \sum_{d \in D} z_n^{p,d}\right] = \sum_{p \in P} \sum_{d \in D} S_d \cdot \widetilde{X_n^{p,d}} \leq \lambda(n) \qquad (14)$$

According to the definition of α in (13), we have $0 \leq \frac{z_n^{p,d} \cdot \alpha}{\lambda(n)} \leq 1$. Therefore, by dividing (14) with $\frac{\lambda(n)}{\alpha}$ on both sides we can get:

$$\mu_c = E\left[\sum_{p \in P} \sum_{d \in D} \frac{z_n^{p,d} \cdot \alpha}{\lambda(n)}\right] \leq \alpha \qquad (15)$$

Since $\frac{\alpha}{\mu_c} \geq 1$, we have:

$$\Pr\left[\sum_{p \in P} \sum_{d \in D} \frac{z_n^{p,d} \cdot \alpha}{\lambda(n)} \geq (1+\epsilon)\alpha\right]$$
$$\leq \Pr\left[\sum_{p \in P} \sum_{d \in D} \frac{\alpha}{\mu_c} \frac{z_n^{p,d} \cdot \alpha}{\lambda(n)} \geq (1+\epsilon)\alpha\right] \qquad (16)$$

Although we cannot directly apply Theorem 1 for $\frac{z_n^{p,d} \cdot \alpha}{\lambda(n)}$ with α, the following holds:

$$\alpha = \frac{\alpha}{\mu_c}\mu_c = \frac{\alpha}{\mu_c}E\left[\sum_{p \in P} \sum_{d \in D} \frac{z_n^{p,d} \cdot \alpha}{\lambda(n)}\right]$$
$$= E\left[\sum_{p \in P} \sum_{d \in D} \frac{\alpha}{\mu_c} \frac{z_n^{p,d} \cdot \alpha}{\lambda(n)}\right] \qquad (17)$$

By applying Theorem 1 for $\frac{\alpha}{\mu_c} \frac{z_n^{p,d} \cdot \alpha}{\lambda(n)}$, based on (17):

$$\Pr\left[\sum_{p \in P} \sum_{d \in D} \frac{\alpha}{\mu_c} \frac{z_n^{p,d} \cdot \alpha}{\lambda(n)} \geq (1+\epsilon)\alpha\right] \leq e^{\frac{-\epsilon^2 \alpha}{2+\epsilon}} \qquad (18)$$

where ϵ is an arbitrary positive value. Moreover, by applying inequalities from (16) and (18) together and introducing Δ, the following holds:

$$\Pr\left[\sum_{p \in P} \sum_{d \in D} \frac{z_n^{p,d}}{\lambda(n)} \geq (1+\epsilon)\right] \leq e^{\frac{-\epsilon^2 \alpha}{2+\epsilon}} \leq \frac{\Delta}{N} \qquad (19)$$

where Δ is a network related variables indicating that $\Delta \to 0$ when the network size grows. By solving (19), we can get

$$\epsilon \geq \frac{-\log\frac{\Delta}{N} + \sqrt{\log^2\frac{\Delta}{N^2} - 8\alpha\log\frac{\Delta}{N^2}}}{2\alpha} \qquad (20)$$

Theorem 3: RDPA can achieve a node storage violating factor of $\frac{4 \log N}{\alpha} + 3$.

Proof: By setting $\Delta = \frac{1}{N^2}$, (19) becomes:

$$\Pr\left[\sum_{p \in P}\sum_{d \in D}\frac{z_d^{p,d}}{\lambda(n)} \geq (1+\epsilon)\right] \leq \frac{1}{N^3},$$

$$\text{where } \epsilon \geq \frac{3 \log N}{\alpha} + 2. \tag{21}$$

By using Union Bound inequality for all the nodes:

$$\Pr\left[\bigcup_{n \in \mathcal{N}}\sum_{p \in P}\sum_{d \in D}\frac{z_d^{p,d}}{\lambda(n)} \geq (1+\epsilon)\right]$$

$$\leq \sum_{n \in \mathcal{N}}\Pr\left[\frac{z_n^{p,d}}{\lambda(n)} \geq (1+\epsilon)\right]$$

$$\leq N \cdot \frac{1}{N^3} = \frac{1}{N^2}, \text{ where } \epsilon \geq \frac{3 \log N}{\alpha} + 2 \tag{22}$$

Finally, we can see from (22) that the probability the expected load on any link violates $c(l)$ with a factor of $1 + \epsilon = \frac{3 \log N}{\alpha} + 3$ approaches 0 when $N \to +\infty$. ∎

Subsequently, we prove the link capacity violating factor as follows:

Theorem 4: The randomized approximation algorithm can achieve a link capacity violating factor of $\frac{3 \log N}{\alpha} + 3$.

Proof:

Definition 3: For node pair (u, v), each data producer p_i, p_j and each data d_i d_j, the traffic load $w_{u,v}^{p_i,d_i,p_j,d_j}$ is defined as follows:

$$w_{u,v}^{p_i,d_i,p_j,d_j} = \begin{cases} r_{p_i} \text{ with prob. } \widetilde{X_v^{p_i,d_i}} + r_{p_j} \text{ with prob. } \widetilde{X_u^{p_j,d_j}} \\ \forall u, v \in \mathcal{N} \\ 0 \text{ otherwise} \end{cases}$$

Since $w_{u,v}^{p_1,d_1,p_2,d_2}$, $w_{u,v}^{p_3,d_3,p_4,d_4}$... are mutually independent according to their definition, the expected load on link (u, v) is:

$$E\left[\sum_{p_i,d_i,p_j,d_j} w_{u,v}^{p_i,d_i,p_j,d_j}\right] = \sum_{p_i,d_i,p_j,d_j} E\left[w_{u,v}^{p_i,d_i,p_j,d_j}\right]$$

$$= \sum_{p_i \in P_u}\sum_{d_i \in D}\widetilde{X_v^{p_i,d_i}} \cdot r_{p_i} + \sum_{p_j \in P_v}\sum_{d_j \in D}\widetilde{X_u^{p_j,d_j}} \cdot r_{d_j} \leq c(u, v)$$

$$\forall u, v \in \mathcal{N} \tag{23}$$

According to the definition of α in (13) and (23), it holds that $0 \leq \frac{w_{u,v}^{p_i,d_i,p_j,d_j} \cdot \alpha}{c(u,v)} \leq 1$. By dividing (23) with $\frac{c(u,v)}{\alpha}$ on both sides we have:

$$\mu_\pi = E\left[\sum_{p_i,d_i,p_j,d_j}\frac{w_{u,v}^{p_i,d_i,p_j,d_j} \cdot \alpha}{c(u,v)}\right] \leq \alpha \tag{24}$$

Similarly as we proved from (15) to (19), by using $\frac{\alpha}{\mu_\pi} \geq 1$ (24) and applying Theorem 1 for $\frac{\alpha}{\mu_\pi}\frac{\beta_n^r \cdot \alpha}{\pi(n)}$ whose expectation of their sum over $r \in R$ is α, the following holds:

$$\Pr\left[\sum_{p_i,d_i,p_j,d_j}\frac{w_{u,v}^{p_i,d_i,p_j,d_j} \cdot \alpha}{c(u, v)} \geq (1+\rho)\alpha\right]$$

$$\leq \Pr\left[\sum_{p_i,d_i,p_j,d_j}\frac{\alpha}{\mu_\pi}\frac{w_{u,v}^{p_i,d_i,p_j,d_j} \cdot \alpha}{c(u, v)} \geq (1+\rho)\alpha\right] \leq e^{\frac{-\rho^2 \alpha}{2+\rho}} \tag{25}$$

where ρ is an arbitrary positive value. Further, by letting $e^{\frac{-\rho^2 \alpha}{2+\rho}}$ be less than $\frac{\Delta}{N^2}$, we arrive at:

$$\Pr\left[\sum_{p_i,d_i,p_j,d_j}\frac{w_{u,v}^{p_i,d_i,p_j,d_j} \cdot}{c(u, v)} \geq (1+\rho)\right] \leq e^{\frac{-\rho^2 \alpha}{2+\rho}} \leq \frac{\Delta}{N^2} \tag{26}$$

By solving (26), we have that

$$\rho \geq \frac{-\log\frac{\Delta}{N} + \sqrt{\log^2\frac{\Delta}{N^2} - 8\alpha\log\frac{\Delta}{N^2}}}{2\alpha} \tag{27}$$

By setting $\Delta = \frac{1}{N^2}$, (26) becomes $\Pr\left[\sum_{p_i,d_i,p_j,d_j}\frac{w_{u,v}^{p_i,d_i,p_j,d_j} \cdot}{c(u,v)} \geq (1+\rho)\right] \leq \frac{1}{N^4}$

As a result, for all the links, we have:

$$\Pr\left[\bigcup_{(u,v) \in \mathcal{L}}\sum_{p_i,d_i,p_j,d_j}\frac{w_{u,v}^{p_i,d_i,p_j,d_j}}{c(u, v)} \geq (1+\rho)\right]$$

$$\leq \sum_{(u,v) \in \mathcal{L}}\Pr\left[\sum_{p_i,d_i,p_j,d_j}\frac{w_{u,v}^{p_i,d_i,p_j,d_j}}{c(u, v)} \geq (1+\rho)\right]$$

$$\leq N^2 \cdot \frac{1}{N^4} \leq \frac{1}{N^2}, \text{ where } \rho \geq \frac{4 \log N}{\alpha} + 2 \tag{28}$$

For a network with N node, there are at most N^2 links, therefore the last inequality holds. As a result, (28) indicates that for any node in the network, the probability that its maximum storage is exceeded by a factor of $1+\rho = \frac{4 \log N}{\alpha} + 3$ will approach 0 when N grows $+\infty$. ∎

Theorem 5: RDPA can achieve a node storage violating factor of $\frac{3 \log N}{\alpha} + 3$ and link capacity violating factor of $\frac{4 \log N}{\alpha} + 3$.

Proof: The proof follows from Theorems 3 and 4. ∎

VI. PERFORMANCE EVALUATION
A. SIMULATION SETUP
We evaluate the performance of our proposed blockchain mechanism on data trading and placement. We assume that nodes in the simulation are distributed randomly in a square area, and the communication range between any two nodes is between 10m and 80m. We assume every two nodes can directly communicate and deliver data items in the blockchain network. The cost for data item delivery is proportional to the distance between two nodes and the data
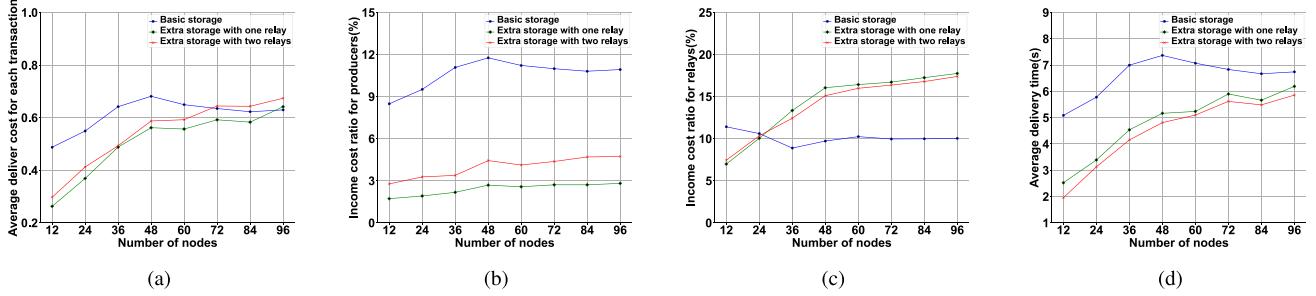
FIGURE 4. (a) The average deliver cost for each transaction, (b) Income cost ratio for producers, (c) Income cost ratio for relays and (d) The average delivery latency for consumers to receive data items under different sizes of the network.

item size. To highlight the difference in terms of the cost consumption, we let the cost for transporting the data item between two adjacent nodes vary between 0.08\$ between 0.64\$ and the cost per 10MB of data item size is 0.025\$. The data item size is set in [$10MB, 200MB$], and the node capacity is set in [$200MB, 40GB$]. We set both the producer and the relay's data item sales revenue share are 50% the producer bears the full storage cost and half of the transaction cost, and the relay bear half of the transaction cost.

We use Java and Spring boot framework to build a test blockchain and realize the process of node communication and data interaction by using socket network programming and JSON data exchange format. We deploy the PoDT mining consensus protocol and PoW mining consensus protocol using the test blockchain to test the time consumption. Since it is challenging to deploy many nodes to transact simultaneously, we do not use the test blockchain to test the delivery consumption of data relay. We implement RDPA by IBM ILOG CPLEX (CPLEX Callable Library interface) 12.6 with Java 11.0.8 and experimented with different node sizes and transaction sizes. Finally, the data placement optimization problem (8)–(11) is solved using IBM ILOG CPLEX (CPLEX Callable Library interface) 12.6 with Java 11.0.8. All the simulations[1] are implemented on a computer with an Intel Core i5-10200H processor and 16GB RAM.

B. PERFORMANCE ON DIFFERENT SIZES OF NETWORKS

First, we evaluate the average delivery cost of a transaction, the income cost ratio for producers and relays, and the average data item delivery time under different sizes of networks. We set 12 to 96 nodes in areas with the same density of nodes. To ensure the same density, 2 to 16 nodes act as producers, and the producers produce 24 to 192 data items before each round of consensus. Each node can act as a relay or consumer. The initial average delivery latency for data items is 1s, and we set the price of each data item is 6\$. Meanwhile, the delivery cost and latency are proportional to the distance between the two nodes. We assume that the number of transactions per data item is 25. A random data purchase transaction involves a random

[1]The implementation code is available at https://github.com/kismet-laoqiu/Efficient-Data-Trading-and-Placement-in-Blockchain-based-Edge-Computing-Systems.

consumer traversing the list of data items in the blockchain and purchasing random data items. After the transaction is concluded, the data item is delivered to the consumer by the relay closest.

Fig. 4(a) shows the average delivery cost for each transaction. In general, the average delivery cost for each transaction increases as more nodes is in the network. In the case of the same number of transactions per data item and the price of each data item, the nodes in the larger network scale need to bear the greater delivery cost. The relay can effectively reduce the delivery cost for each transaction. For example, when the number of transactions per data item is 50, and the node size is 24, based on "Basic storage", each data item has an active relay transfer will reduce the average delivery cost for each transaction of the network by 48.8% and with two relays actively transfer each data item, reduce the average delivery cost per transaction by 32.9%. When the transaction volume in the network is small, the transaction transfer cost saved by "Extra storage" may not cover the transfer cost caused by active storage. The average delivery cost per transaction is higher in the "Extra storage with two relays" case than in the "Extra storage with one relay" case at the current scale of transaction volume. As the scale of the network increases, the average delivery cost of each transaction will also gradually increase, and the delivery cost of the relay to actively transfer data items will also gradually increase. We find that when the number of network nodes is bigger (e.g., more than 96 nodes in the network in the current parameter setting), in the case of "Extra storage", the average delivery cost for each transaction is greater than the "Basic storage". This is because the current number of transactions cannot balance the relay's additional cost of the extra storage.

Fig. 4(b) shows the income cost ratio for producers. As the scale of the network increases, the income cost ratio for producers. At the current transaction size, the transaction transfer cost saved by "Extra storage" is not enough to cover the transfer cost caused by extra storage. Therefore, in the case of "Extra storage with two relays", the delivery cost as a percentage of the producer's revenue is higher than the case of "Extra storage with one relay". Overall, the producer receives 8.19% more profit with the help of "Extra storage with one relay" and 6.62% more profit with the help of "Extra storage with two relays".
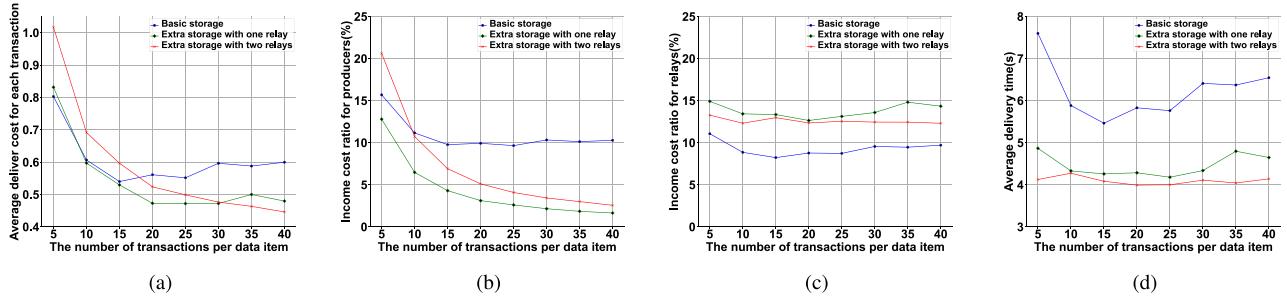
FIGURE 5. (a) The average deliver cost for each transaction, (b) Income cost ratio for producers, (c) Income cost ratio for relays and (d) The average delivery latency for consumers to receive data items under different number of transactions per data item.

Fig. 4(c) shows the income cost ratio for producers. With the scale of the network increasing, the income cost ratio for producers is stable. With the expansion of the network scale, in the case of "Basic storage", the income cost ratio for relays gradually decreases and becomes stable, and in the case of "Extra storage", the proportion of delivery cost in relay revenue gradually increased and stabilized. Due to the current number of transactions cannot balance, the relay's additional cost of the extra storage, delivery cost as a percentage of relay revenue is lower in the case of "Basic storage" than in the case of "Extra storage". Moreover, when the number of network nodes is smaller (e.g., less than 24 nodes in the network in the current parameter setting), in the case of "Basic storage", the average delivery cost for each transaction is greater than the "Extra storage", this is because the node scale is small, the cost of "Extra storage" is very low. But at the same time, in the case of "Basic storage", there is only one relay storage per data item, which leads to the consumption of relays that may face higher transaction costs when delivering data items.

Fig. 4(d) indicates that the node in our proposed mechanism can access the data within a shorter time. The time for all the mechanisms increases with more network nodes, since more hops will consume more time for data delivery. The active participation of relays based on "Basic storage" can effectively reduce the average delivery delay of data items and greatly improve consumers' purchasing experience. For instance, in the case of node size of 24, active relaying of each data item by one relay can reduce the delivery delay by 70.1%, and active relaying of each data item by two relays can reduce 84.6% delivery delay. Even without active storage of relays, in the case of "Basic storage", about 7.5 seconds in maximum are needed for a node to access the desired data.

C. PERFORMANCE ON DIFFERENT NUMBERS OF TRANSACTIONS

Next, we evaluate the average delivery cost of a transaction, the income cost ratio for producers and relays, and the average data item delivery time under the different number of transactions per data item. We set the number of transactions per data item from 5 to 40, with 36 nodes. Six nodes act as producers, and the producers produce 72 data items together.

The initial average delivery latency for data items is 1s, and we set the price of each data item is 6\$.
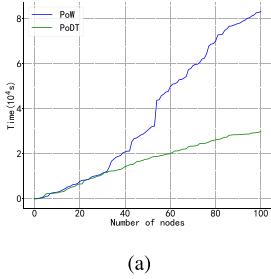
Fig. 5(a) shows the average delivery cost for each transaction. More transactions mean more delivery. When the number of random transactions of data items is relatively large, the relay can effectively reduce the average delivery cost of nodes and the average delivery cost of transactions. For example, when the number of transactions per data item is 35, active relaying each data item by one relay will reduce the average delivery cost per transaction by 17.6%. With two relays actively relaying each data item, each node can reduce the average delivery cost by 26.9% per transaction. But when the number of transactions in the network is less (e.g., less than 10 transactions per data item in the current parameter setting), the current number of transactions cannot balance the extra cost of active transfer by relays.

Fig. 5(b) and Fig. 5(c) shows respectively the income cost ratio for producers and relays. More transactions mean more profits for producers. As the number of transactions increases, the existence of relays can better reduce transaction costs for consumers, resulting in less income cost ratio for producers. For example, when the number of transactions per data item is 35, in the case of "Extra storage with one relay", the income cost ratio for producers is 1.85% and in the case of "Extra storage with two relays" is 2.99%. But in the case of "Basic storage", the income cost ratio for producers is 10.13%. As the number of transactions grows, the income cost ratio for producers for relays is stable. Overall, in the case of "Basic storage", the income cost ratio for producers is 9.3%, in the case of "Extra storage with one relay", the income cost ratio for producers is 13.4% and in the case of "Extra storage with two relays" is 12.6%.
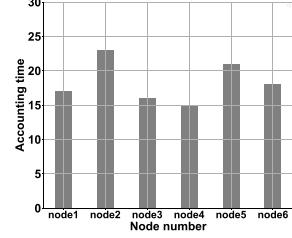
Fig. 5(d) shows the increased number of transactions has less impact on delivery delays. The active relaying of relays based on "Basic storage" can effectively reduce the average delivery delay of data items and significantly improve consumers' purchasing experience.

D. PERFORMANCE UNDER DIFFERENT CONSENSUS ALGORITHMS

We compare the consensus latency of the proposed PoDT mechanism with the traditional PoW protocol like Bitcoin and Ethereum, which are widely used in the existing
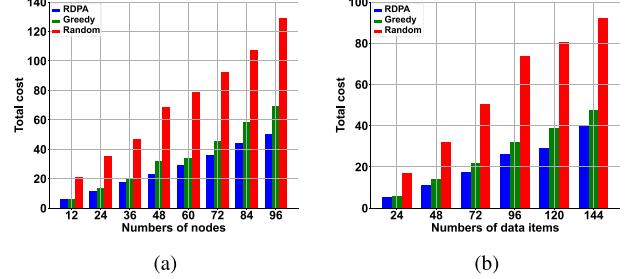
FIGURE 6. (a) The consensus latency in mining processes, (b) The number of accounting for each node participating in the consensus for the proposed PoDT mechanism and the traditional PoW protocol.

FIGURE 7. (a) The total cost under different number of nodes, (b) The total cost under different number of items.

blockchain systems. The stake for each miner can be calculated by Equation (3). There we set α is 0.0007, β is 0.007 and γ is 0.0014. After that, producers in our market attempt to act as the miner who prioritizes publishing the new blocks. Thus, the block generation time T (i.e., hash puzzle-solving time) of each producer i will be influenced by its stake $stake_i$. And we set the difficulty of PoW and PoDT to be five zeros at the beginning of the block hash. The producer making more market contributions will have a larger stake, leading to a larger hash puzzle threshold. This makes it easier to solve the hash puzzle by finding the right conditional nonce. Accordingly, we evaluate the block generation time for the PoW and the PoDT consensus mechanism under the same computing power. As shown in Fig. 6(a), the block generation time of PoW fluctuates around the mean (1550.8s), and the block generation time of PoDT fluctuates around the mean (296.6s). The random characteristic of the Hash algorithm leads to fluctuations. In addition, the block generation time of the PoDT protocol is more permanent compared with the traditional PoW protocol, and hence the longer computation time incurs more computation and energy consumption. Our proposed PoDT protocol can effectively reduce resource consumption when the block height increases.
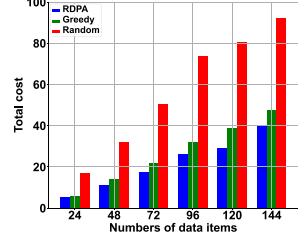
We set up six nodes with the same computing power, and let them participate in the PoDT consensus and conduct transactions, mining and publishing blocks within the network simultaneously. As shown in Fig. 6(b), the number of accounting of each node is stable. In the case of one hundred rounds of consensus, the number of accounting of a node is stable at an average of about 17 times. Therefore, our proposed PoDT consensus mechanism behaves as a green and stable consensus mechanism, which is more applicable in the edge computing environment.

E. PERFORMANCE UNDER DIFFERENT PLACEMENT ALGORITHMS
Finally, we evaluate the transmission cost of proposed RDPA, greedy, and random data placement algorithms at different network sizes and different numbers of data items. The proposed system proactively stores data items produced by producers on relays to offer quick access for all consumers who demand data. The simulation parameter setting is the same for all three placement algorithms for a fair

comparison. We test the transmission cost under different network sizes and different number of data items.

Fig. 7(a) shows the transmission cost for different size of the network. The RDPA algorithm saves much more transmission cost than greedy and random placement algorithms, especially when the size of the network is large. Overall, RDPA reduces 7.79% transmission cost versus greedy placement algorithm and 45.27% versus random placement algorithm. The results show that RDPA achieves less cost for "Basic storage" under different sizes of the network.
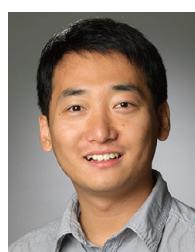
Fig. 7(b) shows the transmission cost form different number of data items in the system. The node size is 36, where 6 nodes act as producers, and the producers produce 24 to 144 data items. RDPA saves much more transmission cost than greedy and random placement algorithms, especially when the number of data items is large. Overall, RDPA strategy reduces 5.00% transmission cost versus greedy placement algorithm and 35.96% versus random placement algorithm. The results show that RDPA achieves less cost for "Basic storage" for different numbers of data items.

VII. CONCLUSION
In this paper, we first have proposed a blockchain-based data relay and transaction model for the data producer, relay and consumer in edge computing. We then have developed a new consensus mechanism called PoDT that combines PoW and PoS mechanisms in edge computing, considering the past contribution and credit of a producer. Finally, we have proposed a rounding-based data placement approximation algorithm for data producers to select the lowest cost relay to store data items. Extensive simulations have shown that our proposed mechanism works better than without relaying processes in terms of cost and delivery consensus time, and the proposed data placement approximation algorithm works better than the greedy and random data placement algorithm in achieving minimum cost consumption. With the increase of data items, the value of data items is difficult to be measured reasonably, so in our future work, we plan to design a tripartite pricing mechanism involving producers, relays and consumers. In addition, we will design a reasonable smart contract for the transmission of updated content to ensure that consumers can obtain the updated data items.

REFERENCES

[1] X. Ma et al., "Inferring hidden IoT devices and user interactions via spatial-temporal traffic fingerprinting," *IEEE/ACM Trans. Netw.*, vol. 30, no. 1, pp. 394–408, Feb. 2022.

[2] "Shutterstock." Accessed: Nov.13, 2024. [Online]. Available: https://www.shutterstock.com

[3] D. Chen and H. Zhao, "Data security and privacy protection issues in cloud computing," in *Proc. Int. Conf. Comput. Sci. Electron. Eng.*, 2012, pp. 647–651.

[4] J. J. Fang and K. Lei, "Blockchain for edge AI computing: A survey," *J. Appl. Sci.*, vol. 38, no. 1, pp. 1–21, 2020.

[5] Y. Huang, Y. Zeng, F. Ye, and Y. Yang, "Profit sharing for data producer and intermediate parties in data trading over pervasive edge computing environments," *IEEE Trans. Mobile Comput.*, vol. 22, no. 1, pp. 429–442, Jan. 2023.

[6] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," in *Proc. Decent. Bus. Rev.*, 2008, Art. no. 21260.

[7] D. Larimer, "Transactions as proof-of-stake," Bitcoin Forum, Las Vegas, NV, USA, Whitepaper, Nov. 2013.

[8] D. Larimer, "Delegated proof-of-stake white paper." 2014. [Online]. Available: http://www.bts.hk/dpos-baipishu.html

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

[10] Y. Huang, Y. Zeng, F. Ye, and Y. Yang, "Incentive assignment in PoW and PoS hybrid blockchain in pervasive edge environments," in *Proc. IEEE/ACM 28th Int. Symp. Qual. Service (IWQoS)*, 2020, pp. 1–10.

[11] Y. Liu et al., "A blockchain-based decentralized, fair and authenticated information sharing scheme in zero trust Internet-of-Things," *IEEE Trans. Comput.*, vol. 72, no. 2, pp. 501–512, Feb. 2023.

[12] X. Lin, J. Li, J. Wu, H. Liang, and W. Yang, "Making knowledge tradable in edge-AI enabled IoT: A consortium blockchain-based efficient and incentive approach," *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6367–6378, Dec. 2019.

[13] N. Szabo, *Formalizing and Securing Relationships on Public Networks*, First Monday, Tel Aviv-Yafo, Israel, 1997.

[14] V. Buterin et al., "A next-generation smart contract and decentralized application platform," Ethereum, Zug, Switzerland, White Paper, 2014.

[15] H. He, A. Yan, and Z. Chen, "Survey of smart contract technology and application based on blockckain," *J. Comput. Res. Develop.*, vol. 55, no. 11, p. 2452, 2018.

[16] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "MedRec: Using blockchain for medical data access and permission management," in *Proc. IEEE Int. Conf. Open Big Data (OBD)*, 2016, pp. 25–30.

[17] T.-T. Kuo and L. Ohno-Machado, "Modelchain: Decentralized privacy-preserving healthcare predictive modeling framework on private blockchain networks," 2018, *arXiv:1802.01746*.

[18] J.-L. de la Rosa et al., "On intellectual property in online open innovation for SME by means of blockchain and smart contracts," in *Proc. 3rd Annu. World Open Innovat. Conf. (WOIC)*, 2016, pp. 1–13.

[19] Y. Zhang, S. Kasahara, Y. Shen, X. Jiang, and J. Wan, "Smart contract-based access control for the Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1594–1605, Apr. 2019.

[20] Y. Huang, J. Zhang, J. Duan, B. Xiao, F. Ye, and Y. Yang, "Resource allocation and consensus of blockchains in pervasive edge computing environments," *IEEE Trans. Mobile Comput.*, vol. 21, no. 9, pp. 3298–3311, Sep. 2022.

[21] L. Yuan et al., "CSEdge: Enabling collaborative edge storage for multi-access edge computing based on blockchain," *IEEE Trans. Parallel Distrib. Syst.*, vol. 33, no. 8, pp. 1873–1887, Aug. 2022.

[22] J. Zhang et al., "A reputation-based mechanism for transaction processing in blockchain systems," *IEEE Trans. Comput.*, vol. 71, no. 10, pp. 2423–2434, Oct. 2022.

[23] A. Zhou, S. Li, X. Ma, and S. Wang, "Service-oriented resource allocation for blockchain-empowered mobile edge computing," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 12, pp. 3391–3404, Dec. 2022.

[24] S. D. Okegbile, J. Cai, and A. S. Alfa, "Performance analysis of blockchain-enabled data-sharing scheme in cloud-edge computing-based IoT networks," *IEEE Internet Things J.*, vol. 9, no. 21, pp. 21520–21536, Nov. 2022.

[25] Y. Huang, Y. Zeng, F. Ye, and Y. Yang, "Incentive assignment in hybrid consensus blockchain systems in pervasive edge environments," *IEEE Trans. Comput.*, vol. 71, no. 9, pp. 2102–2115, Sep. 2022.

[26] Y. Fan, H. Wu, and H.-Y. Paik, "DR-BFT: A consensus algorithm for blockchain-based multi-layer data integrity framework in dynamic edge computing system," *Future Gener. Comput. Syst.*, vol. 124, pp. 33–48, Nov. 2021.

[27] H. Chen, Y. Lu, and Y. Cheng, "FileInsurer: A scalable and reliable protocol for decentralized file storage in blockchain," in *Proc. IEEE ICDCS*, 2022, pp. 168–179.

[28] S. M. Danish, K. Zhang, and H.-A. Jacobsen, "BlockAIM: A neural network-based intelligent middleware for large-scale IoT data placement decisions," *IEEE Trans. Mobile Comput.*, vol. 22, no. 1, pp. 84–99, Jan. 2023.

[29] B. Gao, Z. Zhou, F. Liu, F. Xu, and B. Li, "An online framework for joint network selection and service placement in mobile edge computing," *IEEE Trans. Mobile Comput.*, vol. 21, no. 11, pp. 3836–3851, Nov. 2022.

[30] X. Wang, B. Veeravalli, J. Song, and H. Liu, "On the design and evaluation of an optimal security-and-time cognizant data placement for dynamic fog environments," *IEEE Trans. Parallel Distrib. Syst.*, vol. 34, no. 2, pp. 489–500, Feb. 2023.

[31] Y. Bao, Y. Peng, and C. Wu, "Deep learning-based job placement in distributed machine learning clusters with heterogeneous workloads," *IEEE/ACM Trans. Netw.*, vol. 31, no. 2, pp. 634–647, Apr. 2023.

[32] Z. Luo, Y. Bao, and C. Wu, "Optimizing task placement and online scheduling for distributed GNN training acceleration," in *Proc. IEEE INFOCOM*, 2022, pp. 890–899.

[33] S. Yang, F. Li, S. Trajanovski, X. Chen, Y. Wang, and X. Fu, "Delay-aware virtual network function placement and routing in edge clouds," *IEEE Trans. Mobile Comput.*, vol. 20, no. 2, pp. 445–459, Feb. 2021.

[34] D. Hankerson, A. J. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*. New York, NY, USA: Springer, 2004.

[35] M. R. R. Fauzi, S. M. Nasution, and M. W. Paryasto, "Implementation and analysis of the use of the blockchain transactions on the workings of the bitcoin," *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 260, no. 1, 2017, Art. no. 12003.

[36] S. Delgado-Segura, C. Pérez-Sola, G. Navarro-Arribas, and J. Herrera-Joancomartí, "Analysis of the bitcoin UTXO set," in *Proc. Int. Conf. Finan. Cryptogr. Data Secur.*, 2018, pp. 78–91.

[37] X. Song, Y. Huang, Q. Zhou, F. Ye, Y. Yang, and X. Li, "Content centric peer data sharing in pervasive edge computing environments," in *Proc. IEEE ICDCS*, 2017, pp. 287–297.

[38] D. Rose et al., "Nothing at stake in knowledge," *Noûs*, vol. 53, no. 1, pp. 224–247, 2019.

[39] K. M. Anstreicher, "Linear programming in O([n3/ln n]L) operations," *SIAM J. Optim.*, vol. 9, no. 4, pp. 803–812, 1999.

[40] S. Yang et al., "Survivable task allocation in cloud radio access networks with mobile-edge computing," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 1095–1108, Jan. 2021.

[41] R. Tarjan, *Course: Advanced Algorithm Design. Lecture: Chernoff, Probability and Computing*. Princeton, NJ, USA: Princeton Univ., 2009.

[42] L. Comtet, *Advanced Combinatorics: The Art of Finite and Infinite Expansions*. Dordrecht, The Netherlands: Springer, 1974.

SONG YANG (Senior Member, IEEE) received the Ph.D. degree from the Delft University of Technology, The Netherlands, in 2015. From August 2015 to July 2017, he worked as a Postdoctoral Researcher for the EU FP7 Marie Curie Actions CleanSky Project with Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen, Göttingen, Germany. He is currently an Associate Professor with the School of Computer Science and Technology, Beijing Institute of Technology, China. His research interests focus on data communication networks, cloud/edge computing, and network function virtualization.

KEMING QIU received the B.E. degree in software engineering from the Dalian University of Technology, in 2015, and the M.E. degree in software engineering from the Beijing Institute of Technology in 2022. His research interests include blockchain and edge computing.

FEI ZHANG received the master's degree in computer science and technology from the National University of Defense Technology, China, in 2013, and the Ph.D. degree from the University of Göttingen, Germany, in 2018. He has worked with National Supercomputer Center (NSCC), Tianjin, China, as a Researcher for one year since 2012. He currently works with the National Institute of Defense Technology Innovation, Academy of Military Science China, Beijing, China. He has published papers in several referred international conferences (such as, COMST, MASCOTS 2016, CCGrid 2017, JPDC, and SPE). His research interests include cloud computing, virtualization technology, and mobile computing.

LU CAO received the B.S., M.S., and Ph.D. degrees in navigation and control of spacecraft from the National University of Defense Technology, Changsha, China, in 2007, 2010, and 2014, respectively. He also joined McGill University as a visiting Ph.D. student from 2013 to 2014 and focused on the dynamics, navigation, and control of spacecraft. He is currently a Professor with the National Institute of Defense Technology Innovation, Academy of Military Science China, Beijing, China. In addition, His research interests include the areas of satellite navigation, control and dynamics, and estimation theory research. Moreover, he has the nano-satellite design and launch experiences, which is also another interest.

FAN LI (Member, IEEE) received the B.Eng. and first M.Eng. degrees in communications and information system from the Huazhong University of Science and Technology, China, in 1998 and 2001, respectively, the second M.Eng. degree in electrical engineering from the University of Delaware in 2004, and the Ph.D. degree in computer science from the University of North Carolina at Charlotte in 2008. She is currently a Professor with the School of Computer Science, Beijing Institute of Technology, China. Her current research focuses on wireless networks, ad hoc and sensor networks, and mobile computing. Her papers won Best Paper Awards from IEEE MASS in 2013, IEEE IPCCC 2013, ACM MobiHoc 2014, and Tsinghua Science and Technology in 2015. She is a member of ACM.

MIN TANG received the Ph.D. degree from Beijing Jiaotong University, Beijing, China, in 2021. She currently works with the National Institute of Defense Technology Innovation, Academy of Military Science China, Beijing. Her research interests include the areas of laser communication and Satellite edge computing.

LIEHUANG ZHU (Senior Member, IEEE) received the B.E. and M.E. degrees from Wuhan University, Wuhan, China, in 2001 and 1998, respectively, and the Ph.D. degree in computer science from the Beijing Institute of Technology, Beijing, China, in 2004. He is currently a Professor with the School of Cyberspace Security, Beijing Institute of Technology, Beijing, China. He has published more than 150 peer-reviewed journal or conference papers. He has been granted a number of IEEE best paper awards, including IWQoS 17', TrustCom 18', and ICA3PP 20.' His research interests include security protocol analysis and design, blockchain, wireless sensor networks, and cloud computing.