

RESEARCH ARTICLE

Cross-Chain Protocol Lightning-C for Internet of Vehicles Environment

LILI LU¹, WEIHENG GU^{2,3}, XI XU^{2,3}, MUSEN WANG^{2,3}, SHANGDONG LIU^{2,3}, AND FEI WU³¹School of Artificial Intelligence, Nanjing Vocational College of Information Technology, Nanjing 210023, China²School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China³Institute of High Performance Computing and Big Data, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

Corresponding author: Lili Lu (lull@njcit.cn)

This work was supported in part by the Open Research Project of Zhejiang Laboratory under Grant 2021KF0AB05, and in part by the Future Network Scientific Research Fund Project under Grant FNSRFP-2021-YB-15.


ABSTRACT Blockchain is gaining popularity as a technology that provides secure data sharing and management for the Internet of vehicles (IoV). The current research on cross-chaining between blockchains is still inefficient, and how to perform efficient cross-chain data communication has not been well studied. This paper studies the cross-chain protocol in the IoV environment, and designs a lightning-c cross-chain protocol based on the lightning network. The protocol consists of three steps. The user opens the cross-chain channel through election of nodes, and conducts transactions in the cross-chain channel, then the cross-chain channel is closed and the transaction is broadcasted to the blockchain. Our proposed approach enables efficient interoperability between heterogeneous blockchains. Compared with the traditional cross-chain protocol BTCRelay, it is proved that the lightning-c cross-chain mechanism can greatly improve the throughput of cross-chain transactions. The experimental results show that the method proposed in this paper is more efficient.

INDEX TERMS Blockchain, Internet of Vehicles, cross-chain protocol, lightning network.

I. INTRODUCTION

Internet of Vehicles is an emerging technology that is being used to realize the vision of intelligent transportation systems. Due to the rapid development of vehicle technology, high-throughput satellite communication, Internet of Things and information physics systems, iot has become an important research field with influential applications. Iot is the ability to integrate smart cars with the Internet and system components such as public infrastructure, sensors, computing nodes, pedestrians and other vehicles. By developing a common platform for the exchange of information between vehicles and heterogeneous vehicle networks, this integration aims to create a better environment and public space for people and enhance safety for all road users.

However, with the rapid development of information technology, intelligent cars produce more and more exchange data, such a huge amount of data is difficult to manage. At the same time, due to the characteristics of high mobility,

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

low delay, context complexity and heterogeneity, iot faces great difficulties in directly using traditional cloud storage and other management methods. It is also difficult to ensure compatibility between service providers among different iot entities. Therefore, the data exchange and storage platform of Internet of vehicles needs to be decentralized, distributed, interoperable, flexible and extensible to adapt to the development of Internet of vehicles in the future. So blockchain technology offers great opportunities for the Internet of cars.

Blockchain technology [1] is a new information transmission technology in the Internet era. As a popular distributed ledger, blockchain technology can be considered as a powerful solution to solve the challenges of the Internet of vehicles. Provide credit support for core information management of the Internet of Vehicles at a low cost, and improve the environment of the Internet of vehicles into a transparent, unchanged and privacy-protecting environment [2]. Blockchain can solve many problems in the Internet of cars. For the authentication of vehicle access in the Internet of vehicles, the blockchain consensus mechanism can be used to guarantee the authentication of vehicle access [3].

The security of the Internet of vehicles, the use of blockchain can complete the data record tamper-proof, data theft problem [4], [5]. At the same time, blockchain also has a large number of applications in the Internet of vehicles. Javaid et al. [6] proposed a vehicle solution based on blockchain to realize the protection and management of data in the Internet of vehicles. Li et al. proposed a blockchain-based IoV framework [7], which supports secure and efficient data transactions, thus protecting user privacy to a greater extent. Brousmiche has implemented a fraud prevention system for used car transactions, which is capable of unified vehicle life cycle management and uses blockchain for accountability and traceability of used car transactions [8].

Therefore, it can be seen that there are many applications of iot based on blockchain, which also brings a problem. So many blockchains are not interworking with each other, and the data interaction between chains has become a major problem. Moreover, the throughput of cross-chain protocol must meet the requirements of the iot environment, so it is inevitable to design a suitable cross-chain protocol to solve the cross-chain problem.

In view of the above problems, there has been a lot of work to solve the appeal issue. Wang [9] et al. put forward the concept of blockchain router, which can break the current isolation between different chains, maximize the potential of blockchain, and realize cross-chain interconnection, interoperation and mutual trust. Hei [10] et al. proposed a new cross-chain system based on intelligent contract and trusted computing technology, which has the advantages of good compatibility, great flexibility and high reliability, but there is a certain time cost. Pillai [11] et al. proposed a mechanism that uses transactions to provide cross-chain interoperability, which can realize interoperability between different blockchains.

It can be seen that although existing cross-chain protocols (such as BTRelay) can solve the cross-chain problem of various blockchains in the Internet of vehicles, their characteristics of linear increase of transaction time with the increase of transaction volume obviously cannot meet the cross-chain demand in the Internet of vehicles, and the scalability and high tps of Lightning network [12], Therefore, this paper plans to design a lightning network cross-chain protocol, which is an improvement of the original lightning network off-chain protocol, to improve the off-chain protocol into a cross-chain protocol, so as to take advantage of the high throughput of lightning network to improve the throughput of cross-chain protocol.

II. RELATED WORKS

The emergence of cross-chain protocols is originally to solve the transfer of assets between blockchain. In terms of finance, it was almost impossible to interact with external data when bitcoin was first born, but since the development of blockchain, with the emergence of more and more blockchains, the disadvantage of the island effect of a single blockchain is becoming more and more obvious. Therefore,

there is an urgent need for a protocol that can connect the blockchains. Therefore, the cross-chain protocol came into being. In terms of IoV, since different cities or countries or even a small area may have one or more sets of blockchain systems, when vehicles are connected to different blockchain systems, if there is no corresponding cross-chain protocol between cross-domain blockchains, each blockchain will be isolated, and there may be a lot of redundancy in the stored IoV data, and the vehicle information recorded by other blockchains cannot be obtained. Therefore, the cross-chain protocol in IoV is an important research direction.

Jiang et al. [12] divided the blockchain in IoV into five categories, namely, the blockchain that records vehicle driving, the blockchain that records vehicle conditions, the blockchain that records user privacy, the blockchain that records vehicle insurance, and the blockchain that records data transactions. Then there must be a cross-chain protocol between these blockchains to ensure the interaction of these data. This paper investigates related cross-chain protocols. The cross-chain technology of blockchain can be mainly divided into four major categories [13]: 1) notary schemes, 2) sidechains/relay, 3) hash-locking, and 4) distributed private key control. The notary schemes represented by Corda [14] is to choose a trusted middleman. This middleman can be independent of the two chains. Both chains hand over the assets to be exchanged to the middleman for data exchange, so the middleman must be reliable and credible. Sidechain/relay [15] utilizes a relay chain to relay data sent by other chains and exchange these data. Hash-locking technology is currently mainly aimed at off-chain expansion. It is equivalent to a deposit mechanism. We need to pay a deposit to enter the trading channel. At the same time, the funds can be unlocked at the end of the transaction. Its representative technology is the Lightning Network [16]. Distributed private key control [17] refers to dividing the private key of the current ownership of the asset into several private key fragments and handing them over to several nodes in the entire network for locking. Only when these nodes show the private key fragments together, they can unlock these assets.

The nodes in the IoV have the characteristics of high dynamics. Nodes may often move significantly, frequently access, exit, etc. The characteristics of the blockchain can solve this high dynamics problem [5]. Therefore, for the cross-chain problem of different blockchains in the IoV, it is only necessary to design a cross-chain protocol for the high dynamic characteristics of the IoV, which must have the characteristics of high throughput. In the study of the Lightning Network, Robert et al. [18] used an enhanced Lightning Network in the micro-transactions of the IoV. And Fang et al. [19] used the blockchain and the Lightning.

Current cross-chain work on blockchain mainly focuses on cross-chain protocols to guarantee the security of cross-chain communication. She et al. [20] proposed a multi-energy complementary and secure transaction model on a heterogeneous energy blockchain based on a relay mode cross-chain mutual trust transaction approach. Ochôa et al. [21] proposed a

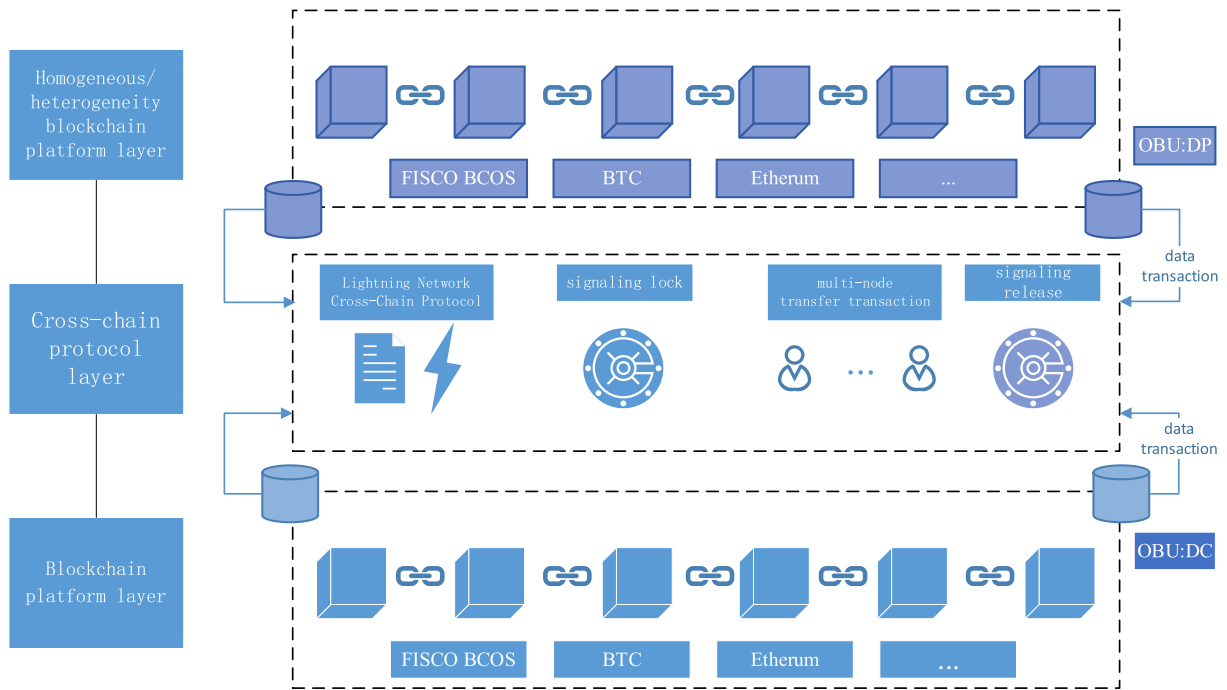


FIGURE 1. Architecture diagram of lightning network cross-chain protocol model.

cross-blockchain architecture. The proposed architecture provides users with security, by using side-chain communication techniques reliability and confidentiality. Firoozjaei et al. [22] proposed a hybrid blockchain with subnets that aims to protect the privacy of users and provide a trusted workflow for billing and charging transactions. It can be seen that most of the current work focuses on security and the issue of improving the throughput of cross-chain communication in the Telematics environment has not been well investigated.

Compared to traditional cross-chain technologies, the lightning network cross-chain protocol designed in this paper has significant differences. The lightning network cross-chain protocol designed in this paper can serve the cross-chain of various heterogeneous blockchains with high throughput at the same time.

III. METHODS

This section will introduce the model and architecture of the Lightning Network cross-chain protocol. As shown in Figure 1, the lightning network cross-chain protocol model in this paper is divided into three layers: the first layer is the blockchain platform layer, which is composed of various types of blockchains and the IoV node OBU (On Board Unit), the blockchain can be one of the commonly used FISCO BCOS [23], BTC [1], Ethereum [27] and other commonly used blockchains. OBU is a mobile node of the IoV which is connected to these blockchains. It is divided into data requester DC (Data Consumer) and data provider DP (Data Provider). As mentioned in the related works above, the blockchains in IoV are not interconnected, so when the

OBUs intend to exchange assets or data, they need to use the second layer of cross-chain protocol layer. The second layer is the cross-chain protocol layer. The Lightning Network cross-chain protocol mainly includes signaling locking, multi-node transfer transactions, signaling release operations, signaling refers to the UTXO structure in Bitcoin or a non-property token. Signaling locking refers to locking the corresponding signaling on a chain, similar to the locking process of the Lightning Network. A multi-node transfer transaction refers to a user transferring data from one chain to another. This data can be an asset or a corresponding message. In this model, both asset transfer and message transfer require two chains to ensure the reliability of message transfer, and each node in the multi-node group must have corresponding accounts on both chains to be eligible for guarantee. Signaling release refers to the process of releasing signaling and messages between multi-node group nodes and nodes on another chain. The homogeneous/heterogeneous blockchain platform layer is the target blockchain to which the data is transferred to reach, and it may be homogeneous or heterogeneous with the blockchain platform layer's blockchain.

A. DETAILED IMPLEMENTATION

The detailed implementation of the Lightning Network cross-chain protocol lightning-c will be described in detail. The execution sequence of the model in this paper is shown in Figure 2. First, user A on blockchain A calls the Lightning Network cross-chain channel contract on blockchain A, and locks the amount of signaling (hereinafter referred to as deposit) that he wants to trade with user B on chain B,

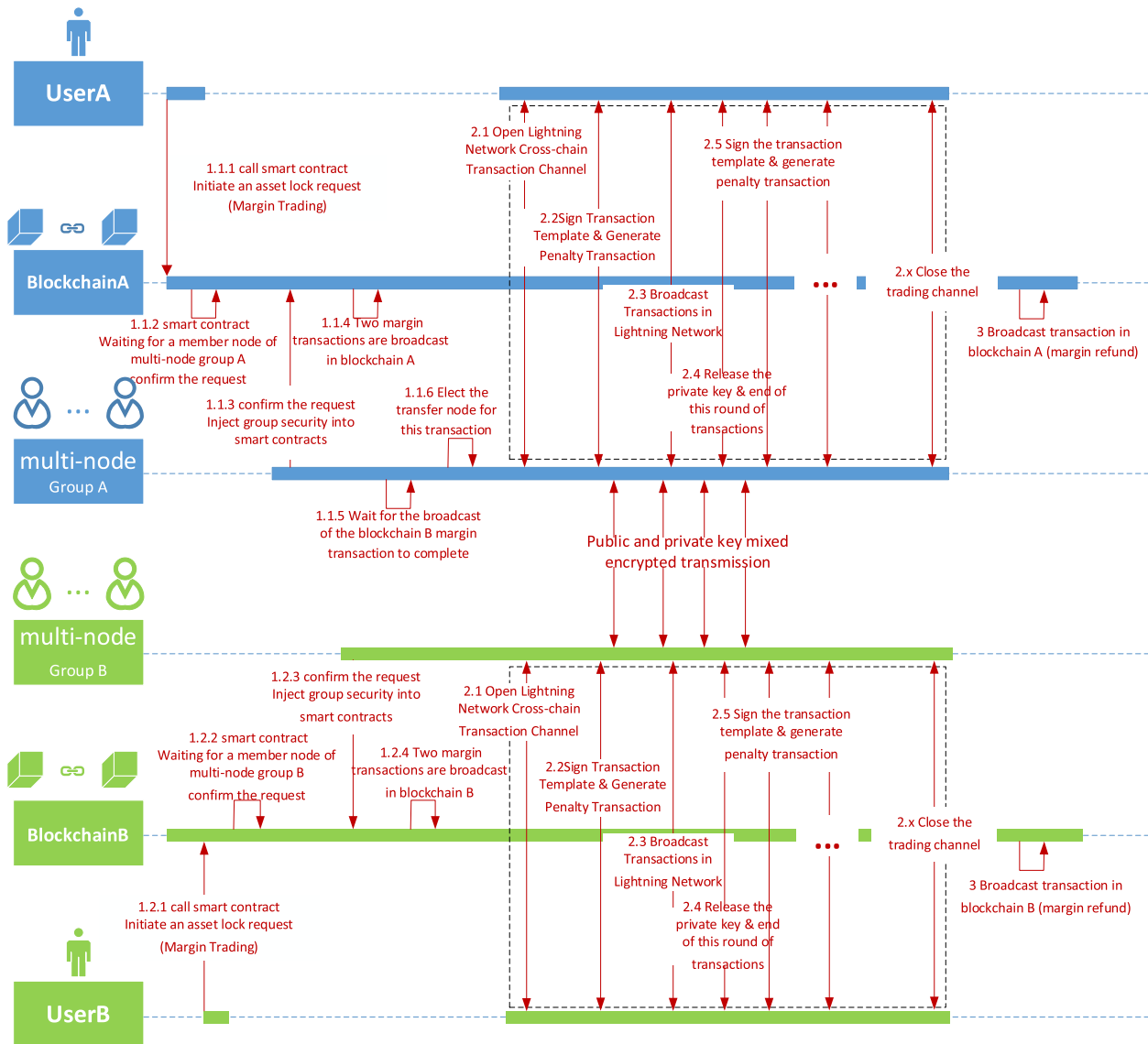


FIGURE 2. Lightning-c protocol execution process.

and waits for multi-node group A confirming the smart contract request. Here, the multi-node group A is composed of multiple full nodes on the blockchain A. It can join the group through an identity-recognized smart contract (a smart contract that verifies the credibility of the current node) if it is in the public chain, these full nodes must be verified and confirmed to join the group if it is in the alliance chain. No matter what type of chain it is, after joining the group, it will lock a part of the deposit into the blockchain, and this part of the deposit will also serve as its own position in the group. When the nodes in the group confirm the transaction, group A will lock a signaling equivalent to user A, and then broadcast the two locked transactions in the blockchain. At the same time, the same process of locking the deposit on the blockchain A is also performed on the blockchain B. Then, according to the weights of participating nodes in group A, the specific nodes

that perform transaction transmission this time are confirmed (these nodes must have their own accounts on both chains to be eligible for election). These nodes are called transaction node groups. Then the Lightning Network cross-chain transaction channel is opened. Similar to the Lightning Network, user A conducts transactions with transaction node group A, and generates corresponding transaction templates and penalty transactions. These transaction templates are accompanied by the signaling that they need to trade and the transaction data in the related car networking. Then these node groups are transmitted to their own node group B on another chain through public and private key hybrid encryption. Node group B then conducts transactions with user B. From the perspective of user A and user B, they can directly conduct off-chain transactions similar to Lightning Network, rather than cumbersome cross-chain transactions. After the

transaction template and penalty transaction are signed, their corresponding transactions are broadcasted in the channel of the Lightning Network cross-chain protocol (not in the blockchain). After user A, user B, transaction node group A, and transaction node group B complete the transaction, the transaction channel can be closed. Then, according to its specific remaining signaling, the corresponding transactions are broadcasted on blockchain A and blockchain B respectively. Since the above operations will only write the corresponding transaction into the blockchain when the transaction channel is opened and closed, the throughput of the blockchain is greatly improved, so that multiple microtransactions can be processed in Lightning Completed in the network cross-chain protocol. The following figure will describe the process of this model in detail with user A, user B, blockchain A, blockchain B, multi-node group A, and multi-node group B.

B. LIGHTNING NETWORK CROSS-CHAIN CHANNEL PREPARATION

When user A on blockchain A wants to make a transaction with user B on blockchain B, the blockchain where user A is located cannot communicate directly with the blockchain where user B is located because the chains may be heterogeneous with each other. The preparation of the Lightning Network cross-chain channel is mainly composed of two parts. The first part is the locking of the deposit, and the second part is the election of the transmission node responsible for this transaction. These two parts will be described in detail below.

When the user needs to open the Lightning Network cross-chain channel, it needs to inject all the signaling b of this transaction as a deposit in advance into the address $t'addr$ specified by the smart contract. This address can be specified by the multi-node group A to save the deposit. And no new margin can be injected into the current cross-chain channel contract again during the transaction, then a new cross-chain channel must be opened again in case of insufficient signaling. When user A has injected deposit into the contract, the contract will wait for multi-node group A to confirm the validity of the current transaction, that is, it needs to obtain the signatures of $1/3$ of all nodes

Algorithm 1 Lightning Network Cross-Chain Channel Contract

1. input: $b, t'addr, \{A'_1sig \dots A'_n sig\}, b'$
 2. transfer $(b) \rightarrow t'addr$
 3. Waiting for multi-node group confirmation
 4. require $(b' > b)$
 5. require $(n > \frac{1}{3} \times tn)$
 6. checkAndTransfer $(b', A'_1sig \dots A'_n sig) \rightarrow t'addr$
 7. Election of the node responsible for this message delivery
 8. Trading channel open
-

tn in the group to confirm the legitimacy, and injects the same amount of signaling $b', \{b' = b\}$ as the deposit paid by the user, so as to ensure that user A and multi-node group A are difficult to do evil. Algorithm 1 shows how this contract works.

Here, transfer represents the process of transferring signaling from user A to $t'addr$, checkAndTransfer represents the process of transferring signaling from a multi-node group to $t'addr$, where n is the number of nodes joining the group, $A'_1sig \dots A'_n sig$ represents the signature of each node in multi-node group A. As long as the number of signatures reaches $1/3$ of the total number of nodes in the group (the total number of nodes is n), it means that multi-node group A agrees to the transaction, and transfer the corresponding deposit b' , and then the multi-node group A will elect the node responsible for this transaction.

Before introducing the algorithm for electing nodes, the process of joining a multi-node group needs to be described in detail. First of all, to join the group, you must be a full node on the current chain A, and you must also have a corresponding full node on blockchain B. If you only have a single full node on the current chain, the node cannot be transmitted as a common node of both chains for communication. Finally, we need to pay a signaling on the current chain as the weight w on the current chain. The calculation method of w is shown in (1):

$$w = \frac{En}{Tb} \times 100 \quad (1)$$

where En is the number of signaling paid when the current full node enters the group, and Tb is the total number of signaling in the current group.

The following multi-node group A will elect a group of nodes n' participating in this transaction transmission from the n nodes that join the group. During the election, the order of being elected will be calculated according to the weight w and the time t when the current node is not elected. The election formula is as follows:

$$order_i = w_i + \frac{t_i}{100}, i \in [1, n] \quad (2)$$

t_i is the time when the current node is not elected, that is, after the node is elected, t_i becomes 0, and w_i is the weight generated by the current node paying the deposit when joining the group. This value is a dynamic value. The weight needs to be recalculated according to the number of signaling paid by the joining node. $order_i$ is the sum of the weight of the current node and the waiting time that has not been elected, which represents the order in which the current node is elected as the communication transmission node. The larger the current value, the easier it is to be elected as the node for this communication. In this paper, the number of nodes for each communication is set to 10, then the top ten $order_i$ with the largest ranking are selected as the communication nodes each time, that is, $\sum_{n-9}^n order_i$.

After user A and multi-group node A open the corresponding cross-chain channel on blockchain A, they need to wait for user B and multi-group node B to open the corresponding

cross-chain channel on blockchain B. The process of user B and multi-group node B opening channels on blockchain B is the same as that on A. In this way, the Lightning Network cross-chain channel has been opened.

C. LIGHTNING NETWORK CROSS-CHAIN TRANSACTION

The blockchain is a decentralized system, so the lightning network cross-chain protocol that relies on the blockchain must also be a decentralized system. This paper designs RSMC-C for cross-chain systems based on the RSMC [15], an off-chain protocol for lightning networks. After the channel is opened, the first step is to carry out a confirmation broadcast of the deposit, which is different from the deposit broadcast before the channel is opened. The broadcast of deposit before channel opening is used to open the cross-chain channel, and the broadcast after channel opening is used to reconfirm the deposit. As shown in Figure 3, A represents user A (hereinafter referred to as A), and A_group represents multi-node group A (hereinafter referred to as A_group). First, A_group will construct a template transaction Tx1ag and give it to A for signature, and A will also construct a template transaction Tx1au to A_group for signature. After both parties have signed, the two signed template transactions can be sent to the Lightning Network cross-chain channel of the current blockchain A.

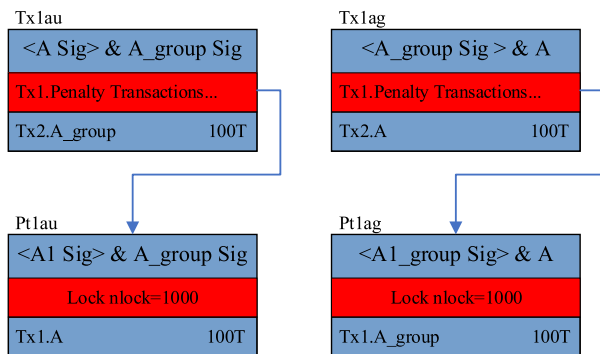


FIGURE 3. Deposit confirmation transaction in the lightning-c channel.

Assuming that the deposit paid by A is 100T, and the deposit paid by B is 1000E (the unit of signaling on chain A is T, the unit of signaling on chain B is E, and the conversion rate is T:E = 1:10). A_group pays the deposit that must be equal to that paid by A, which is also 100T. Tx1au is a transaction proposed by A, which requires A's signature and A_group's signature for approval. Tx1 is a penalty transaction. When A finishes the current transaction first, it will broadcast Tx1au and call 100T to the A_group account according to the Tx2 agreement. At the same time Pt1au is broadcasted as a penalty transaction in Tx1au, and the 100T agreed to give A for that penalty transaction needs to wait for 1000 blockchains to lock. Such an operation ensures that even in these 1000 locks, if A commits false reporting, then A_group can penalize A for the duration of the lock, with the penalties described in

detail below. Meanwhile Tx1bu and Tx1bg on blockchain B are consistent with the above principle.

When A intends to perform the next transaction, it first needs to abolish the status transactions of Tx1au and Tx1ag. A needs to hand over the private key of A1 to A_group, and A_group also needs to hand over the private key of A1_group to A, so that A_group can modify the penalty transaction for Pt1au which is done to prevent A from rolling back Tx1au in the future transaction state Tx2au and Tx3au. If A intends to roll back to Tx1au, then A_group can generate Pt1au' according to the A1 private key given by A. As shown in Figure 4, if A_group finds that A has rolled back the transaction, then A_group directly generates Pt1au' (The precondition of the transaction broadcast is that A has already broadcasted Tx1au), skips the block lock and transfer the deposit originally belonging to A to his own account as a punishment for A.

After the private key swap, when A needs to change the state of the current transaction, it will perform the following operations. For example, A wants to send a car network message to B across the chain, and it needs to attach a transaction signaling, and the number of its transaction guarantee signaling is 10T, which is consistent with the negotiation process described above, and the state update in the channel is Figure 5, which is mainly divided into 3 steps:

1) STATUS UPDATE IN A-CHAIN CHANNEL

A agrees to transfer 110T to A_group (Tx2) once Tx2au is broadcast. It generates the penalty transaction Pt2au with the agreement to lock 1000 blocks after broadcasting and then transfer 90T to A. Then A fills in the message field, which is the IoV message that A will send to B. If it only involves the transfer of property, we can omit this field, and finally hand Tx2au and Pt2au to A_group for signature. Tx2ag and Pt2ag are basically the same, the difference is that the message field in Tx2ag is the message sent by user B to A.

2) CROSS-CHAIN MESSAGE TRANSFER

The cross-chain transfer of messages is undertaken by A_i , $i \in [1, n], n \geq 10$, that is, several nodes selected above. These nodes will generate a set of public and private key pairs A_i^{pri} and A_i^{pub} when they join a multi-node group. In this paper, we adopt the public-private key asymmetric hybrid encryption method. This is because if all the encryption is done purely with private keys, then it is easy to use the public keys of these nodes to crack if the transmission channel is intercepted; if only public keys are used to encrypt, then it will result in user A being able to forge A_group encrypted messages for communication. Therefore, the use of public-private key asymmetric hybrid encryption communication can solve the above problems. The data transmitted is a two-tuple $data=(token, message)$, where token is the number of signaling guaranteed this time, and message is the message to be transmitted. The encryption process is

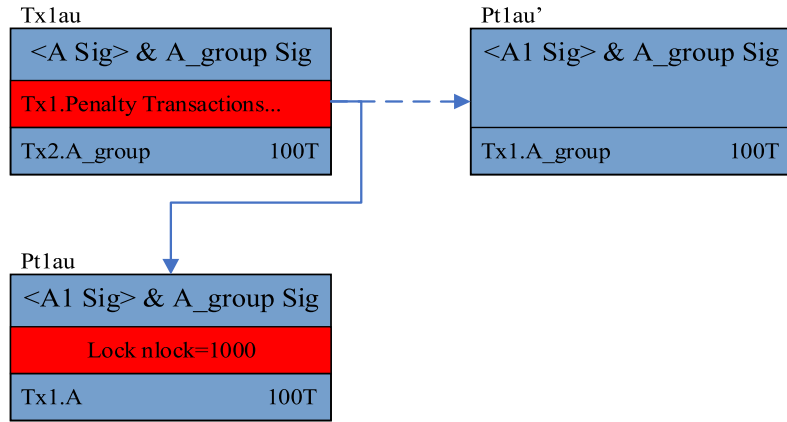


FIGURE 4. Penalty transaction Pt1au'.

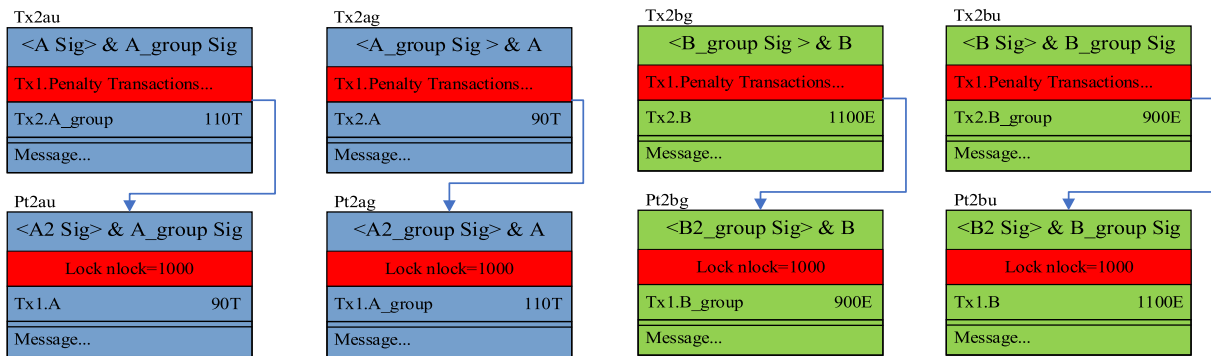


FIGURE 5. Transaction update in the lightning-c channel.

as follows:

CrossChainMessage

$$= \begin{cases} \text{Enc} \left(A_1^{pri}, \text{Enc} \left(A_2^{pub}, \text{Enc} \left(A_3^{pri}, \dots \text{Enc} \left(A_n^{pri}, \text{data} \right) \right) \right) \right), \\ \quad n \bmod 2 \neq 0 \\ \text{Enc} \left(A_1^{pri}, \text{Enc} \left(A_2^{pub}, \text{Enc} \left(A_3^{pri}, \dots \text{Enc} \left(A_n^{pub}, \text{data} \right) \right) \right) \right), \\ \quad n \bmod 2 = 0 \end{cases} \quad (3)$$

where Enc is the encryption algorithm, which can be any kind of asymmetric encryption algorithm, such as SHA256, etc., mod is the modulo, and crossChainMessage is the message sent by A_i , $i \in [1, n]$, $n \geq 10$.

B_i is the full node corresponding to A_i on blockchain B. When A_i receives the crossChainMessage, it will decrypt the current message. The decryption process is as follows:

DecData

$$= \begin{cases} \text{Dec} \left(A_n^{pri}, \dots \text{Dec} \left(A_3^{pri}, \text{Dec} \left(A_2^{pub}, \text{Dec} \left(A_1^{pri}, \text{data} \right) \right) \right) \right), \\ \quad n \bmod 2 \neq 0 \\ \text{Dec} \left(A_n^{pub}, \dots \text{Dec} \left(A_3^{pri}, \text{Dec} \left(A_2^{pub}, \text{Dec} \left(A_1^{pri}, \text{data} \right) \right) \right) \right), \\ \quad n \bmod 2 = 0 \end{cases} \quad (4)$$

where Dec is the decryption algorithm, which corresponds to the encryption algorithm of asymmetric encryption.

According to the message data obtained from decryption, B_group converts 10T to 100E according to the exchange rate, and at the same time initiates the status update of Tx2bg and Pt2bg, agrees to give B user 1100E after the transaction broadcast, and at the same time agrees to lock 1000 blocks after the penalty transaction broadcast, and then gives B_group group 900E, message fills the message message in data, and B also draws up the corresponding Tx2bu and Pt2bu for mutual signature verification, since then a cross-chain process is complete.

After performing the above process several times, if A and B intend to end the current transaction channel, they need to construct a transaction to close the channel. This transaction is the same as the deposit transaction after opening the channel, and only necessary to confirm the proportion of the amount of the two parties who finally exit the channel. When the trading channel is closed, the settled deposit is returned to A and A_group through the smart contract, thus completing the entire Lightning Network cross-chain process.

IV. RESULTS

A. EXPERIMENTAL ENVIRONMENT

In this experiment, multiple nodes are set up in two virtual machines to build two blockchains. The open-source platform

FISCO BCOS is used to build the blockchain platform, and the Java SDK interface provided by FISCO BCOS is used to build the background system locally. The corresponding mysql is deployed in the docker in the VIRTUAL machine to store and transmit data in the Internet of vehicles. The specific experimental environment is shown in Table 1:

TABLE 1. Experimental environment.

Equipment	Configuration
virtual machine	Centos:7.2
FISCO BCOS	FISCO BCOS v2.2.0
Java	jdk1.8
Mysql	mysql:5.7

B. EXPERIMENTAL RESULTS AND ANALYSIS

In experiment, we simulate the process of data interaction between Vehicles in the IoV environment, that is, the vehicle A connected to the blockchain A intends to obtain some traffic information of the location provided by the vehicle B connected to the blockchain B. This paper stores some traffic information in the database of the vehicle B itself. Vehicle A first needs to reach a consensus with vehicle B to use the Lightning Network cross-chain protocol. Then vehicle A and vehicle B open the corresponding cross-chain channel on the corresponding blockchain (the first broadcasted transaction to the blockchain). Then Vehicle A and Vehicle B open the corresponding cross-chain channel on the corresponding blockchain (first broadcast transaction to the blockchain), and then Vehicle A and Vehicle B communicate messages between Chain A and Chain B with the help of a cluster of nodes, and close the channel after completion (second broadcast transaction to the blockchain), final write to their corresponding blockchain.

At the same time, in order to highlight the advantages of the Lightning Network cross-chain protocol, we compare it with the cross-chain protocol BTCRelay [24]. Since the BTCRelay cross-chain protocol is a dedicated cross-chain protocol for both Bitcoin and Ethereum, in order to control the corresponding variables, we transplant its core part to FISCO BCOS for a control experiment. The core part of the principle is that when vehicle A connected to blockchain A needs to connect to some traffic information on the location of vehicle B on blockchain B. Vehicle A first needs to reach a consensus with vehicle B to use the Lightning Network cross-chain protocol, and then vehicle A locks the asset, and uses the locked asset to use the smart contract to send signaling and messages to the smart contract on the blockchain B through the elected node. And then the smart contract sends the signaling and message to B. B sends the messages and signaling needed by A to the smart contract on blockchain B after receiving it, and the smart contract replies to the smart contract on blockchain A through the elected node, and the smart contract on blockchain A then returns data and signaling to vehicle A, and so on and so forth, completing

several communication transactions (n times broadcast to the blockchain).

We assume that the number of transactions required is n. It is well known that the throughput of the blockchain can be improved by increasing the block size and shortening the block interval [25]. If a block contains more transactions, the current blockchain throughput must be higher. From the perspective of reducing the number of transactions, if a cross-chain transaction is broadcasted less frequently on the blockchains of both parties, it must save corresponding space for other transactions, just like the Lightning Network cross-chain protocol. No matter the number of intermediate transactions, it only needs to be broadcasted once on both blockchains when the channel is opened and closed, and there are only four broadcasted transactions in total, that is, 4 times. BTCRelay will broadcast once on different chains after each cross-chain interaction, and perform several intermediate transactions, then there will be several intermediate broadcasts, that is, n times. At the same time, the time of cross-chain transactions is also a measure of the cross-chain protocol [26]. For the same number of transactions, if the cross-chain time is less, it will definitely occupy less resources on the two blocks. The lightning network cross-connection protocol only needs to broadcast four times, so the overhead of using the blockchain is extremely low. Assuming that the time for broadcasting a transaction using the blockchain is $t_{broadcast}$, and the time for each transaction to be transmitted between the two blockchain nodes is $t_{between}$, then the overhead of the Lightning Network protocol is $4 \times t_{broadcast} + n \times t_{between}$. Every time BTCRelay crosses the chains, it will broadcast twice on both chains, and the total overhead time is $2n \times t_{broadcast} + n \times t_{between}$. According to the cost, the following cost comparison can be clearly obtained:

$$cost = \begin{cases} 4 \times t_{broadcast} + n \times t_{between}, \\ LightningNetworkCross - ChainProtocol \\ 2n \times t_{broadcast} + n \times t_{between}, \\ BTCRelay \end{cases} \quad (5)$$

Obviously, when the number of transactions is less than 2, the overhead of BTCRelay should be less than that of the Lightning Network cross-chain protocol. When the number of transactions is equal to two, the Lightning Network cross-chain protocol overhead is similar to BTCRelay. When the number is more than two, the overhead of the Lightning Network cross-chain protocol is definitely less than that of BTCRelay. And as the transaction volume increases, the overhead gap will further be widen.

We set a group of control experiments to compare BTCRelay and Lightning Network cross-chain protocol lightning-c from the time of cross-chain transactions and the specific number of transactions occupied in the block. We set the number of transactions as 1, 2, 4, 8, 16, and 32 as the number of transactions. Table 2 lists the time required for cross-chain transactions corresponding to different transaction volumes.

TABLE 2. Comparison of the time required for different transaction volumes of BTCRelay and lightning-c.

Number of transactions	1	2	4	8	16	32
BTCRelay	970	1981	4007	8068	16181	32381
lightning-c	1977	1980	2006	2032	2180	2366

It can be seen from Table 2 that when the number of transactions is 1, the time required for BTCRelay cross-chain transactions is lower than that of lightning-c. When the number of transactions is 2, the time required for BTCRelay and lightning-c is almost the same. When the number of transactions is greater than 2, the cost of cross-chain time required by BTCRelay increases with the number of transactions, and the gap between lightning-c becomes larger.

As shown in Figure 6, both the total time of BTCRelay and lightning-c cross-chain transactions show a linear growth trend, but the growth trend of BTCRelay is faster than that of lightning-c, and when the transaction volume is small, the time spent by BTCRelay is slightly lower than lightning-c. Thus, when the transaction volume is small, choosing BTCRelay is a good choice, but when the number of transactions is huge, for example, all of them are small micro-transactions, then using lightning-c will be a good choice.

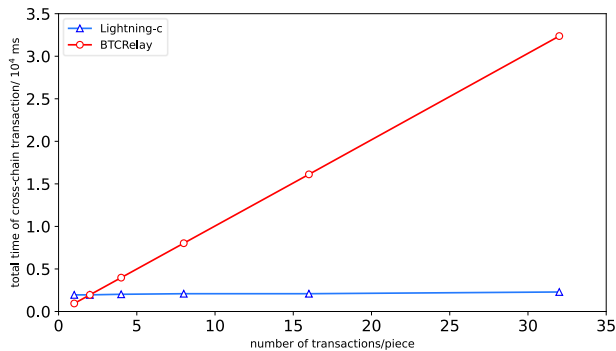


FIGURE 6. Comparison of the total time of BTCRelay and lightning-c cross-chain transactions.

As shown in Table 3, for BTCRelay, as the number of transactions grows, the number of broadcasted transactions for the cross-chain interaction increases. However, for lightning-c, the number of broadcasted transactions for the cross-chain interaction does not increase with the increasing of the number of transactions, and is fixed to 4.

TABLE 3. Comparison of the total number of broadcasted transactions between BTCRelay and lightning-c cross-chain transactions.

Number of transactions	1	2	4	8	16	32	64
BTCRelay	2	4	8	16	32	64	128
lightning-c	4	4	4	4	4	4	4

As shown in Figure 7, for BTCRelay, the total number of broadcasted transactions increases linearly with the increase

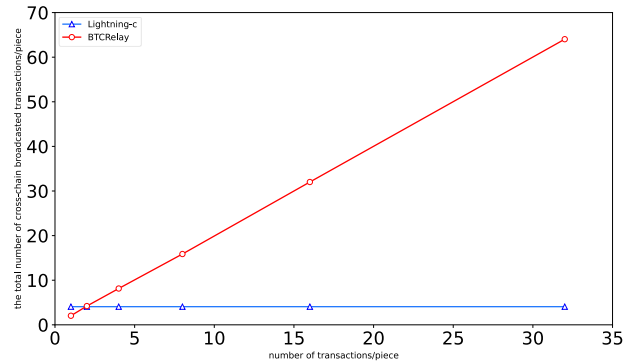


FIGURE 7. Comparison of the total number of cross-chain broadcasted transactions between BTCRelay and lightning-c.

of the number of transactions, and for lightning-c, it shows a constant trend. When the transaction volume is low, the number of BTCRelay broadcasts on the blockchain is small, the performance loss of the blockchain is also less than lightning-c. However, as the transaction volume increases, the number of BTCRelay broadcasts on the blockchain will also increase linearly, and at this time, the number of transactions broadcasted by lightning-c is fixed to 4. Thus, the more transactions, the better the performance of lightning-c.

V. CONCLUSION

This paper designs a lightning network cross-chain protocol lightning-c, which is aimed at the cross-chain scenario of a large number of micro-transactions. It has a three-layer model, and the interaction between the three-layer models is divided into three steps: the opening of the cross-chain channel, the transaction in the cross-chain channel, and the closing of the cross-chain channel. Considering the blockchain cross-chain protocol in the Internet of Vehicles environment, there are generally a large number of communication requirements between vehicles, so there must be a large number of micro-transactions. According to the discussion of the above experiments, lightning-c is superior to traditional cross-chain protocols such as BTCRelay in terms of a large number of transactions, both in terms of cross-chain communication time and blockchain resource consumption. Therefore, lightning-c can greatly improve the cross-chain throughput in the case of cross-chain interactions with a large number of transactions.

REFERENCES

- [1] S. Nakamoto, and A. Bitcoin. (2008). *A Peer-to-Peer Electronic Cash System*. [Online]. Available: <https://bitcoin.pdf>
- [2] Y. Pu, T. Xiang, C. Hu, A. Alrawais, and H. Yan, "An efficient blockchain-based privacy preserving scheme for vehicular social networks," *Inf. Sci.*, vol. 540, pp. 308–324, Nov. 2020.
- [3] X. Wang, P. Zeng, N. Patterson, F. Jiang, and R. Doss, "An improved authentication scheme for Internet of Vehicles based on blockchain technology," *IEEE Access*, vol. 7, pp. 45061–45072, 2019.
- [4] F. Ahmad, C. A. Kerrache, F. Kurugollu, and R. Hussain, "Realization of blockchain in named data networking-based Internet-of-Vehicles," *IT Prof.*, vol. 21, no. 4, pp. 41–47, Jul. 2019.

- [5] 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.
- [6] U. Javaid, M. N. Aman, and B. Sikdar, "DrivMan: Driving trust management and data sharing in VANETs with blockchain and smart contracts," in *Proc. IEEE 89th Veh. Technol. Conf.*, Apr. 2019, pp. 1–5.
- [7] H. Liu, Y. Zhang, and T. Yang, "Blockchain-enabled security in electric vehicles cloud and edge computing," *IEEE Netw.*, vol. 32, no. 3, pp. 78–83, May 2018.
- [8] K. L. Brousmiche, T. Heno, C. Poulain, A. Dalmieres, and E. Ben Hamida, "Digitizing, securing and sharing vehicles life-cycle over a consortium blockchain: Lessons learned," in *Proc. 9th IFIP Int. Conf. New Technol., Mobility Secur. (NTMS)*, Feb. 2018, pp. 1–5.
- [9] H. Wang, Y. Cen, and X. Li, "Blockchain router: A cross-chain communication protocol," in *Proc. 6th Int. Conf. Informat., Environ., Energy Appl.*, Mar. 2017, pp. 94–97.
- [10] Y. Hei, D. Li, C. Zhang, J. Liu, Y. Liu, and Q. Wu, "Practical AgentChain: A compatible cross-chain exchange system," *Future Gener. Comput. Syst.*, vol. 130, pp. 207–218, May 2022.
- [11] B. Pillai, K. Biswas, and V. Muthukumarasamy, "Cross-chain interoperability among blockchain-based systems using transactions," *Knowl. Eng. Rev.*, vol. 35, p. e23, 2020, doi: [10.1017/S0269888920000314](https://doi.org/10.1017/S0269888920000314).
- [12] T. Jiang, H. Fang, and H. Wang, "Blockchain-based Internet of Vehicles: Distributed network architecture and performance analysis," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4640–4649, Jun. 2019.
- [13] R. Belchior, A. Vasconcelos, S. Guerreiro, and M. Correia, "A survey on blockchain interoperability: Past, present, and future trends," *ACM Comput. Surveys*, vol. 54, no. 8, pp. 1–41, Oct. 2021.
- [14] R. G. Brown, J. Carlyle, I. Grigg, and M. Hearn, "Corda: An introduction," *R3 CEV*, vol. 1, no. 15, p. 14, Aug. 2016.
- [15] S. Schulte, M. Sigwart, P. Frauenthaler, and M. Borkowski, "Towards blockchain interoperability," in *Proc. Int. Conf. Bus. Process Manage.*, 2019, pp. 3–10.
- [16] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou, "A survey of distributed consensus protocols for blockchain networks," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1432–1465, 2nd Quart., 2020.
- [17] L. Deng, H. Chen, J. Zeng, and L. J. Zhang, "Research on cross-chain technology based on sidechain and hash-locking," in *Proc. Int. Conf. Edge Comput.*, Jun. 2018, pp. 144–151.
- [18] J. Robert, S. Kubler, and S. Ghatpande, "Enhanced lightning network (off-chain)-based micropayment in IoT ecosystems," *Future Gener. Comput. Syst.*, vol. 112, pp. 283–296, Nov. 2020.
- [19] Z. Fang, K. Gai, L. Zhu, and L. Xu, "LNBFSM: A food safety management system using blockchain and lightning network," in *Proc. Int. Conf. Algorithms Archit. Parallel Process.*, Sep. 2020, pp. 19–34.
- [20] W. She, Z. Gu, X. Yang, Z. Tian, J. Chen, and W. Liu, "Multi-energy complementary secure transaction model of heterogeneous energy blockchain," *Power Grid Technol.*, vol. 16, no. 1, pp. 1–20, 2019.
- [21] I. S. Ochoa, L. A. Silva, G. de Mello, N. M. Garcia, J. F. de Paz Santana, and V. R. Q. Leithardt, "A cost analysis of implementing a blockchain architecture in a smart grid scenario using sidechains," *Sensors*, vol. 20, no. 3, p. 843, Feb. 2020.
- [22] M. Daghmehchi Firoozjaei, A. Ghorbani, H. Kim, and J. Song, "Hy-bridge: A hybrid blockchain for privacy-preserving and trustful energy transactions in Internet-of-Things platforms," *Sensors*, vol. 20, no. 3, p. 928, Feb. 2020.
- [23] R. Wang, K. Ye, T. Meng, and C.-Z. Xu, "Performance evaluation on blockchain systems: A case study on Ethereum, fabric, sawtooth and Fisco-Bcos," in *Proc. Int. Conf. Services Comput.*, 2020, pp. 120–134.
- [24] H. Jin, X. Dai, and J. Xiao, "Towards a novel architecture for enabling interoperability amongst multiple blockchains," in *Proc. IEEE 38th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2018, pp. 1203–1211.
- [25] E. Akbari, W. Zhao, S. Yang, and X. Luo, "The impact of block parameters on the throughput and security of blockchains," in *Proc. 2nd Int. Conf. Blockchain Technol.*, May 2020, pp. 13–18.
- [26] P. Robinson and R. Ramesh, "General purpose atomic crosschain transactions," in *Proc. IEEE Int. Conf. Blockchain Cryptocurrency (ICBC)*, May 2021, pp. 1–3.
- [27] V. Buterin, "A next-generation smart contract and decentralized application platform," *White Paper*, vol. 3, no. 37, pp. 1–2, 2014.

LILI LU is currently teaching with the School of Artificial Intelligence, Nanjing Vocational College of Information Technology. Her main research interests include big data and software computing.

WEIHENG GU received the master's degree from Nanjing University of Posts and Telecommunications. His main research interest includes blockchain.

XI XU is currently pursuing the master's degree with Nanjing University of Posts and Telecommunications. His main research interest includes blockchain.

MUSEN WANG is currently pursuing the master's degree with Nanjing University of Posts and Telecommunications. His main research interest includes blockchain.

SHANGDONG LIU received the Ph.D. degree in computer system structure from Southeast University. His main research interests include artificial intelligence, big data, cloud computing, and cyberspace security.

FEI WU received the Ph.D. degree from Nanjing University of Posts and Telecommunications. His main research interests include pattern recognition, machine learning, and software engineering.

• • •