2017 18th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)

Multi-Blockchain Model For Central Bank Digital Currency

He Sun[1] Hongliang MAO* Xiaomin Bai[1] Zhidong Chen[1] Kai Hu[1] Wei Yu[1]
[1](State Key Laboratory of Software Development Environment, Beihang University)
[2] (National Computer Network Emergency Response Technical Team/Coordination Center of China(CNCERT/CC))
Beijing, 100191, China
Email: mhl@cert.org.cn; sy1506405@buaa.edu.cn

***Abstract*—Digital Currency for Central Bank is becoming an important policy for country. CBDC (Central Bank Digital Currency) model should take advantages in the supervision, payment and consumption. Blockchain possesses the feature of de-centrality, tamper-resistant, and traceability. So this paper attempts to use the blockchain as the fundamental technology of CBDC. However, the challenges such as the protection for user's privacy, supervision and transaction speed should be overcome. This paper proposes a CBDC model called MBDC which is based on the permission blockchain technology. The model makes use of the multi-blockchain architecture and ChainID to improve the model's scalability and process payments more quickly. In this model, central bank and commercial banks and other agencies build and maintain the blockchain. On one hand, central bank could master the issuance of currency. On the other hand, relying on the user account address protocol, central bank could separate the user's identity and transaction information. In this way, central bank could avoid double-spending issues and protect user's privacy. In addition, the establishment of DC (Data Center) and layers of supervision provide strong supervision for the model. Finally, we also demonstrate, both theoretically and experimentally, the performance of model on the scalability and the speed of transaction execution etc.**

Keywords; multi-blockchain, digital currency, model, de-centrality.

I. INTRODUCTION

So far, the type of digital currency is close to 700, and the total amount is nearly 80 billion USD. In 2017, the Bitcoin [1], the most successful and famous digital currency, held a market capitalization of 35 billion USD and the transaction value has reached 1 billion USD. The Bitcoin has influenced many adjacent areas and has been researched by many Research institutions [2]. Blockchain[3], as the fundamental technology of bitcoin, has been paid more attention and studied because its feature of de-centrality, tamper-resistant, traceability. R.Grinberg examines a few relevant legal issues in [4]. Aitzhan N Z and Svetinovic D [5] have researched the blockchain technology in area of energy. JPMorgan Chase [6] and Nasdaq [7] have studied blockchain technology about two years.

The "first wave" of cryptocurrency research is on [8], [9].The Central Bank of Canada has revealed recently that they has developed a digital version of the Canadian dollar based on BTC, called CAD-coin. Bank of England has explored the area of digital currency and proposed their model of the digital currency with London University, called RSCoin [10]. Deutsche Bundes bank declared that his preliminary balance prototype based on blockchain has been developed in 2017 [11]. The Dutch central bank is experimenting with a bitcoin-based virtual currency called "DNB-Coin". In China, since 2014, the Central Bank has taken a prospective study for digital currency and demonstrated the feasibility for the central bank to issue legal digital currency. In 2015, the central bank made further efforts to enrich the theory of the issuance of digital currency and many related technologies, etc. Finally, the central bank published a series of research reports.

There are many digital currencies which have achieved great success. For example, Ripple Labs [12] has published their digital currency in Ripple network. Traditional international clearing system has expensive fees in cross-border deals and long processing time. The Ripple system is devoted to break the system through its liberal and global payment network. Litecoin [13] is different from Bitcoin in speed of producing block and total amount of digital currencies. If Bitcoin is the "gold" in area of digital currencies while the Litecoin is called the "silver". Peercoin [14], published in 2012, is famous for its consensus protocol which combined the POS and POW. However, these digital currencies have significant limitations in terms of computation costs, the scalability, the supervision, and the privacy. The Bitcoin could handle at most 7 transactions per second and the Paypal could handle about 400 transactions per second [10].

This paper makes the following contributions:

- We propose the CBDC model based on the multi-blockchain technology and the design of fundamental components.
- We introduce the protocol for the communication of cross-blockchain to implement the scalable of the model.
- We analysis the performance from scalability, privacy, safety and the blockchain communication traffic to prove the feasibility of the model.

II. RELATED WORK

Blockchain is a distributed ledger and each node in the blockchain network records the whole ledger. The blockchain is formed by the block according to the fixed data formats. The block is formed by a set of data including the hash of transactions and pre-block, timestamp, transactions, version of blockchain and extra data etc. After complementing the encryption algorithm for the data, the leader node would produce the block and send it to other nodes. Each block has the hash of its pre-block so that whoever wants to modify the data in the front block should modify all of hash of blocks after that block.

Currently, blockchain includes the public blockchain and generalized private blockchain. A public blockchain means that everyone could connect to the blockchain network and possess the complete blockchain ledger. So everyone could monitor and

0-7695-6330-9/17/31.00 ©2017 IEEE
DOI 10.1109/PDCAT.2017.00066

inspect the transactions' information stored in the ledger. In the public blockchain network, when a transaction comes into the nodes, nodes check the transaction and broadcast it to the whole network. Leader node has the right to record the ledger. The selection of leader node is completed by the nodes according to the consensus protocol. At regular intervals, the leader node packs the transactions into a block and broadcasts the block to other nodes. When nodes receive the block, they should check the block and add the block to their blockchain. In a word, the public blockchain means that everyone could join in and exit from the blockchain network at any time. And the consensus protocols include POW(Proof of Work)[15], POS(Proof of Stake)[14], DPOS(Delegate Proof of Stake)[16], POET(Proof of Elapsed Time), etc..

Generalized private blockchain includes consortium blockchain and private blockchain. Consortium blockchain refers to that several institutions build the blockchain network together and each institution is one of node. Other institutions which would like to join the blockchain should obtain the union's authorization. So this type of blockchain is also called permission blockchain. The consensus protocol of PBFT (Practical Byzantine Fault Tolerance) [17] is always used in this type of blockchain. Private blockchain means that all nodes should be managed by one organization. And the consensus protocol is also the PBFT.

Now the famous application platforms based on the blockchain are Bitcoin, Etherum [18] and Hyperledger. The Bitcoin and Etherum are public blockchain while the Hyperledger is a generalized private blockchain. The Bitcoin use the UTXO (Unspend Transaction Output) [19] while the other use the account.

III. ARCHITECTURE OF MBDC

MBDC use the multi-blockchain to satisfy the bank's business requirement. The permission blockchain is used to ensure that every unit of digital currency is created by the particular central bank. So that the digital currency would become significantly more manageable for government. The central bank maintains a blockchain with all of commercial banks and other agencies. The blockchain, which is called Superchain, preserves the total value in daily transactions. The central bank could make analysis of these big data which is stored in the Superchain. The commercial banks place their nodes in the Superchain so that the banks could upload the daily transaction amount to the Superchain. In this way, not only does the third party and central bank could supervise daily clearing and settlement to ensure the legitimacy of the transaction, but also central bank could make analysis of the data to make exchange rate adjustment and adopt the right monetary policy.

In addition, bank should build the DC to collect the transaction information from the branch in the regions, furthermore, the DC could supervise the transaction executed by the branch in the regions. Moreover, the DC could play the role of redundant backups. Moreover, the branch DC should upload the transaction amount to a higher DC according to the fixed format. Finally, the transaction data would be sent to their head office and the central bank. Through the layers of supervision, the branch becomes more safety. The whole MBDC is shown as Fig. 1.

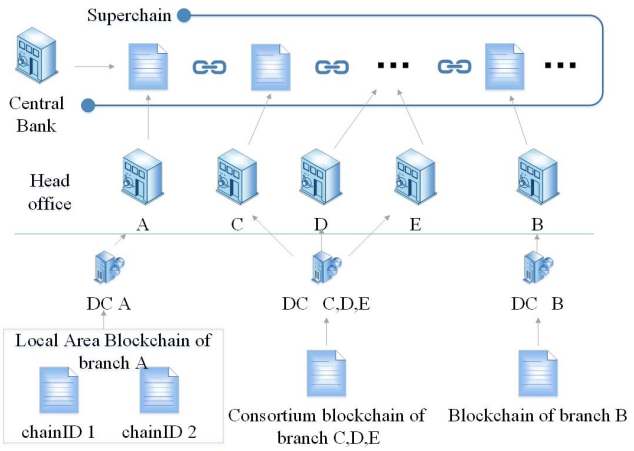


Fig. 1. Architecture of MBDC

A. Type of blockchain

Generally speaking, a branch could build several blockchains which is called the LAB (Local Area Blockchain) when the transactions can't execute in time. Several branches of bank could build a consortium blockchain if their transaction amount is very little. Normally, one branch builds one blockchain to execute transactions.

B. ChainID

Eighteen digits is used to indicate the identification of the blockchain. The first four digits are used to represent the identification of the bank, the next two digits are used to represent the provincial branches, the next two digits are used to indicate the municipal branches, and the next two digits are used for the representation of county branches, etc. And the digits left are used for extension. For example, when the bank builds a LAB, and blockchains' ID can be set through extended field. Specific designs are shown in the TABLE I.

TABLE I. CHAINID DESIGN

ID of bank	ID of provincial branches	ID of municipal branches	ID of county branches	ID of extension
4	2	2	2	8

In this way, each blockchain could be identified and the blockchain in this model becomes scalability.

C. User account address

Every bank should validate the user's identity when the user is registered, at the same time, the user's public key and private key could be generated according to the user's identity information. Users preserve their own private key and bank saves their public key. If the user's private key is lost or stolen, they can go to the bank to reset their public key and private key. Of course, in order to be more secure, users could generate their secret key through multiple signatures. If user is not convenient to reset his or her secret key, he/she could grant permissions to their families or friends to reset his/her secret key.

According to the user's public key, through a series of encryption algorithms including MD5 [20] and ECDSA [21] etc.

users could obtain their own transaction address. According to the users' transaction address and the ChainID of bank, users could obtain their wallet address. Similarly, every branch could create their own public/private key and wallet address.

MBDC adopts the method of combination of the user account and the wallet address for user to transfer accounts. When users transfer account, the value in the transaction would be subtracted from the users' account and preserved by their wallet address, so that the users could continue to transfer account. In this way, account is used to preserve the users' total money while wallet address is used to lock the transaction value. The wallet address is generated by the public key and ChainID, so that when the transaction, is sent to the nodes, the nodes would presently verify the transaction to ensure the accuracy and correctness of the transaction. The receiver could use his/her public key to compare with the wallet address of the receiver in the transaction, and if there is no problem, the transaction value will be added to receiver's account. In this way, not only the model could protect the user's private information and ensure the transparency of transactions for the branch, but also could prevent double-spending attack which is occurred in Bitcoin [22], [23].

By comparing Bitcoin and Ethernet, the design of user account address which combined the wallet address and account has advantages in privacy, safety, and supervision.

D. Transaction format

The transaction mainly includes normal transaction and other transactions. The "method" field in head of transaction is set to the "transaction" when the transaction is normal transaction, and the normal transaction includes the inter-blockchain and intra-blockchain transaction. The feedback transaction always sets the method filed to "feedback". And other types of transactions are depending on the situation. In this way, the transaction becomes scalability.

1) Intra-blockchain transaction

Intra-blockchain transaction is generated within one branch, and generally needing to be verified and executed within a branch. The method field in the transaction is set to the word "transaction". The content of intra-blockchain transaction includes the sender's signature generated by the sender's private key, the sender's public key, the sender's wallet address, the transaction value and the receiver's wallet address, the timestamp, the nonce which is used to identify a transaction and the extension field, etc.

2) Inter-blockchain transaction

Inter-blockchain transaction is executed between branches. Of course, the method field is set to the word "transaction". No matter which bank the blockchain belongs to, the branch could communicate and complete the process of transfer account through the inter-blockchain transaction execution. The content of inter-blockchain transaction includes the sign and the public key and the wallet address of the sender's branch, in addition, the transaction is also included in the inter-transaction. Then branch should use public key of transaction receiver's bank to encrypt the new inter-blockchain transaction and send it to the receiver's bank.

3) Feedback transaction

The field "transaction" is used for the reply of the inter-blockchain and intra-blockchain transaction while the method field in the transaction is set to the word "feedback" for the "feedback" transaction. The content of feedback transaction is same as inter-blockchain transaction while the extra field in the transaction would be set to the result of transaction execution.

Some letters in this paper is used to represent the field in transaction:

Pbk: is used to represent the public key which is used to verify the user's signature and wallet address.

Prk: is used to represent the private key which is unique and crucial to user.

Addr: is used to represent the transaction address which is generated by public key through a series of encryption algorithm.

Waddr: is used to represent the wallet address which is formed by the ChainID of branch's blockchain and user's Addr.

Sign: is used to represent the signature which is generated by the private key and can be verified by the public key.

V: is used to represent value in the transaction.

N: is used to represent the nonce of transaction.

Extra: is used to represent the extension field in the transaction.

Tu: is used to represent the Intra-blockchain transaction.

Tx: is used to represent the generalized transaction.

Tb: is used to represent the inter-blockchain transaction.

Tn: is used to represent the encrypted inter-blockchain transaction.

Tf: is used to represent the feedback transaction.

And the word branch is used to represent the branch of bank.

E. Timeout-retransmission

1) Tu Timeout-retransmission

When user sends a transaction to the branch, after 2^n (1<=n<=10) RTT (Round-Trip Time) time, user dose not receive the reply, then user would start the Timeout-retransmission automatically. When the branch receives the retransmission transaction, it would execute the transaction immediately and send feedback to the user. If the transaction is a normal transaction and the feedback information has been sent while the branch received the retransmission transaction again, the branch should start the 3). If the transaction is the type of Tb, the branch should start the 2) .

2) Tn Timeout-retransmission

If the transaction is a Tb, the branch of sender sends the Tb to the branch of receiver. After 2^n (1<=n<=10) RTT time the branch of sender does not receive the reply, the sender of branch start timeout-retransmission automatically. The branch of receiver receives retransmission Tb and then immediately sending the feedback to the branch of send. If the Tb has executed, and the feedback information has been sent, while the branch of receiver received the retransmission Tb, the branch should launch 3).

3) Feedback timeout-retransmission

If the transaction is a Tb, the branch of receiver receives the new transaction and the result would be packed into the feedback transaction and sent to the sender's branch, when the branch of receiver received the retransmission Tb, then immediately starting the feedback timeout-retransmission. If the transaction is a Tu, the branch of sender received the retransmission transaction while the branch of sender has sent the feedback to user, then immediately starting feedback timeout-retransmission.

IV. TRANSACTION PROTOCOL

There are many Communication protocols in area of blockchain. For example, David Mazière proposed a new consensus protocol called SCP[24](Steller Consensus Protocol), the most famous digital currency –Bitcoin- has been studied by many institutes and formed a series of protocols like[15], and our protocol is also affected by the PBFT and Bitcoin. We prepare to introduce the transaction protocol through the Tn and Tb protocol. Finally, the bank reserves protocol is proposed to improve the efficiency of the Tb protocol.

A. State transition function

From a technical point of view, the Bitcoin can be considered as a state transition system, which includes all of status of the bitcoin owners and "state transition function". The state transition function acquires a new state while the current state and transaction play the role of input. The state transition function can be defined as follows:

APPLY(S, TX) -> S' OR ERROR [25]

But in MBDC, the state transition function, APPLY(S, TX) -> S', can be defined as follows:

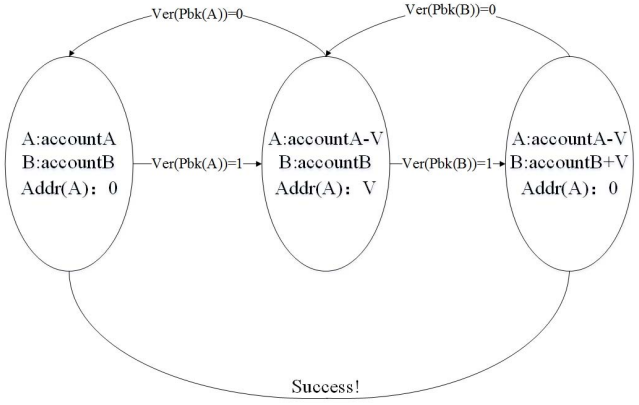


Fig. 2. State transition process

According to the Fig. 2, nodes in the system verify the transaction information when a transaction comes into the branch's blockchain. First, nodes should check the sender's Pbk and use it to verify its Sign and Waddr. If there is no problem, the nodes would lock the V in transaction into the Waddr of receiver from sender's account if the money is enough. Then nodes would check the receiver's Pbk and use it to verify Sign and Waddr of receiver. If there is no problem, the V would be added to the receiver's account. Finally, account of sender complete the process of transferring account.

This is a state transition process for user's account. Detail protocol is would be discussed as follows.

B. Tu protocol

The simple syntactic and semantic of Tu has defined in section III. 1) . So this section would introduce the process of Tu and Tb.

After sending the transaction to the branch, user begins waiting for the result of the transaction or implements the timeout-retransmission. When nodes in blockchain system receive the transaction, firstly checking the format and method field of the transaction, if the method field is set to the word "transaction" and the transaction is a Tu, the nodes begin executing the transaction according to the Tu protocol. The execution process shows as the Fig. 3.

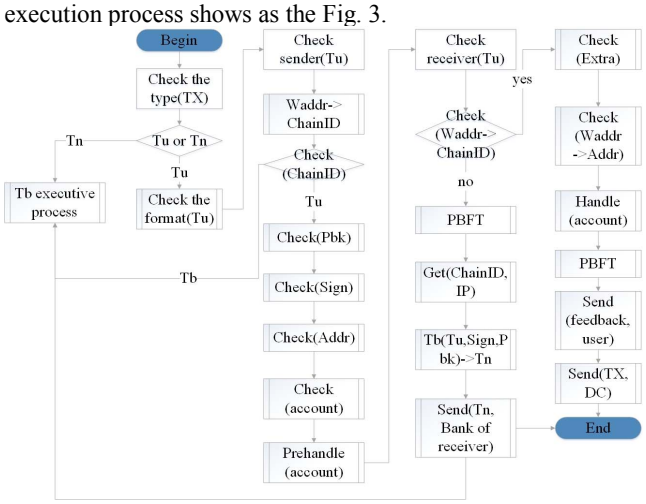


Fig. 3. Tu process

In this period, the nodes would run Algorithm IV.1. After checking the ChainID of sender, if the ChainID ∈ Branch (sender), the nodes would run Algorithm IV.2 which mainly shows the verification of the sender's Sign and Addr. If the Sign is correct, the nodes could believe that the transaction is sent by the sender. If the Addr is correct, the nodes could believe that the account belongs to the sender. In this way, the nodes could believe that the transaction is correct and safe. Then the nodes would examine the balance of the account of sender. If the balance is enough for the transaction, the nodes will continue to run Algorithm IV.1.

In this process, the nodes would lock the V in Tu into the Waddr of receiver, so that sender could continue to transfer account and the branch does not need to worry about the double-spending problem. If the transaction failed, nodes will return the V to the sender. The branch could send Tn to the branch of receiver according to the IP address stored in the branch with ChainID.

The pseudocode of the algorithm shows as follows:

Algorithm IV.1: Tu process, run by Branch

```
Input: Tu, A->B
1.    CheckFormat(Tu)
2.      GetChainID(ChainID<-Waddr, sender)
3.      If (ChainID ∈ Branch(sender))
4.          CheckTransaction(Tu, Sign, Waddr, V)
5.          GetChainID(ChainID<-Waddr, receiver)
6.            If (ChainID ∈ Branch(sender) )
7.                Handle(Account)
8.                Byzantine(Account)
9.                Return success.
10.               Send(feedback, user)
11.               Send(Tu, DC)
12.     Else   Byzantine(Account)
13.            GetChainID(ChainID<-Waddr, receiver)
14.            Send(Tu->Tb->Tn, receiver)
15.            Goto  Tb process
```

Algorithm IV.2: CheckTransaction, run by Branch

```
Input: a TX, A->B, Tx(Sign, Pbk, Waddr, V, extra)
1.    Check (Tx)
1.    GetAddr (Addr <-Waddr)
2.    If (Pbk ∈ owners && Addr ∈ Branch) then
3.       If (Pbk->Sign) then
4.          If (owner.Account>V) then
5.             PrepareHandle(Extra, Account)
6.             Return success
7.    Else return fail
```

C. Tb protocol

The simple syntactic and semantic of Tu has defined in section III. 2). So this section would introduce the process.

After receiving a transaction, the nodes in the blockchain check the format and method in the transaction. If the method field is set to the word "transaction" and the transaction is a Tn. Then the nodes will execute the transaction according to the Tb execution process as described in Fig. 4.

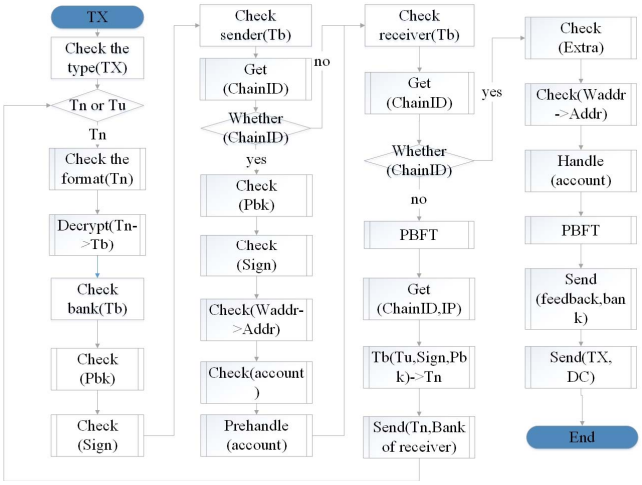


Fig. 4. Tb execution process

For the first time, nodes would decrypt the Tn to obtain the Tb, then nodes would begin running the Algorithm IV.2. If there is no problem, the nodes would run Algorithm IV.3. If the ChainID of sender in Tu belongs to branch (ChainID ∈ Branch), the nodes would think that the transaction may be an operation of withdraw from other branches. If not, the transaction maybe just an operation of remittance, and nodes could check the ChainID of receiver directly.

If the ChainID of receiver belongs to the branch, nodes would execute the transaction as the algorithm. If not, the branch would package the Tu to a Tb and encrypt the Tb to a Tn and send it to the destination. Finally, the storage is same as described in Tu protocol.

The pseudocode of the algorithm shows as follow.

Algorithm IV.3: Tb process, run by Branch

```
Input: a Tn, A->B
2.       CheckTransaction(Tb, Sign, Pbk)
3.       GetChainID(ChainID<-Waddr, sender)
4.       If (ChainID ∈ Branch)
5.             CheckTransaction(Tu, Sign, Waddr, V)
6.             GetChainID(ChainID<-Waddr, receiver)
7.              If (ChainID ∈ Branch)
8.                 Handle(Account)
9.                 Byzantine(Account)
10.                Return success
11.                Send(feedback, Branch)
12.                Send(Tu, DC)
13.             Else  Byzantine(Account)
14.               GetChainID(ChainID<-Waddr, receiver)
15.               Send(Tu->Tb->Tn, receiver)
16.               Goto Tb process
17.      Else   return 7
```

D. Tf protocol

When the method in the transaction is set to the word "feedback", the transaction is a "feedback" transaction. The branch checked the transaction's nonce and extension field. If the feedback is "success", then branch would send "success" to user. If not, the branch would lock the V into the Waddr of user and send the transaction to the branch, and send "failure" to user. If there is no problem, the branch would take a PBFT consensus to make the blockchain synchronized. Finally, the nodes send the transaction to the DC in the region.

E. Bank reserves protocol

If the number of transactions between two branches are colossal, there must be a lot of communication between two branches. Then the efficiency of transactions would be poor. In order to solve this problem, branches could create the reserve account like the cash reserves. For example, branch B could create its Pbk and Prk and Account B in branch A, when a Tb happened between branch A and branch B, branch A could use the account B in branch A to finish the transaction and does not need to transfer account through Tb process. In this way, the Tb will become a Tu, and the blockchain traffic would decrease significantly.

The branch A and branch B could implement the clearing and settlement at regular intervals. For example, through a fixed period of time, maybe 12 hours, branch A makes a clearing for all Tbs with branch B, and branch B makes a clearing for all Tbs with branch A. Finally, the branch examines and verifies the Tbs which are sent by the other branches.

V. Model Analysis

MBDC's simulate system is set up according to the architecture of model. So the performance of system could be analysed from blockchain's traffic, safety, and scalability.

A. Blockchain transaction communication

The blockchain transaction communication is concentrated on the process of consensus. The conclusion is also proved by theory analysis of Tu and Tb. We also make analysis of inter-blockchain communication with the branch accounts according to the branch reserves protocol. And the analysis is inspired by a report which is published by WeiDe Cai.

1) Tu

First, we assume that:

- Transaction is executed in one blockchain, which means that the transaction is a Tu.
- The blockchain is jointly maintained by n nodes.
- There is no delay and no retransmission.

a) When user sends transaction to the branch, the blockchain traffic is n.

b) After accepting the transaction, every node verifies and executes the transaction. This process needs to query the database and does not need to communicate with other nodes.

c) After executing the transaction, all of nodes in the blockchain system need to take a Byzantine agreement, and the blockchain traffic is $2n^2$

d) Send feedback to the user, the blockchain traffic is n.

So now we can make a conclusion that the blockchain traffic is at least $2n^2 + 2n$.

2) Tb

First, we assume that:

- Transaction is a Tb, and needs to communicate between two branches.
- Each blockchain is jointly maintained by n nodes.
- Network communication has been delayed, and there have been 4 retransmission (1 times Timeout-retransmission when user send transaction, 1 times Timeout-retransmission when branch send transaction, the branch feedback retransmission of 2 times).

a) User sends transaction to the branch, and network delay occurred, so the blockchain traffic is 2n.

b) After the nodes in the branch accepted the transaction and made sure that the transaction is a Tb, they should pre-handle the Account and take a Byzantine agreement, so the blockchain traffic is $2n^2$. The transaction needs to be sent to the receiver's branch of bank, and meanwhile the network delay occurred 1 time, so the traffic is 2n. All of the blockchain traffic in this process is $2n^2 + 2n$.

c) After the nodes in the branch of receiver received and executed the transaction, they should take a Byzantine agreement, the blockchain traffic is $2n^2$. The branch should send feedback to the branch of sender, and the network delay occurred, so the traffic is 2n. All of the blockchain traffic in the process is $2n^2 + 2n$.

d) The branch of sender received the feedback and took a Byzantine agreement, the traffic is $2n^2$, then the branch would send feedback to the user, and network delay occurred in the process, so the traffic is 2n .All of the blockchain traffic is $2n^2 + 2n$.

So in case of the Tb, the blockchain traffic is $6n^2 + 8n$.

3) The use of Bank account

When the branch adopted the bank reserves protocol, the blockchain traffic of the Tb would close to the Tu's communication. In this way, the process needs 4 times communication and 2 times Byzantine agreement based on the Tu. So the blockchain traffic in this case is $4n^2 + 6n$.

The comparison of the blockchain traffic in a transaction between the cases is shown as Fig. 5:

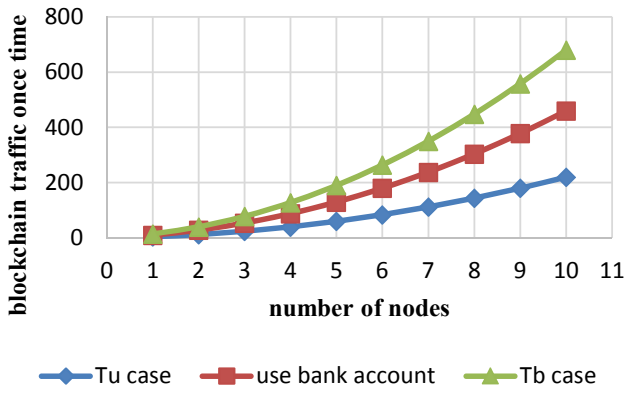


Fig. 5. Blockchain traffic in a trasaction

B. Safety

1) Tu

First, we assume that:

- Transaction is executed in one blockchain, which means that the transaction is a Tu.
- The blockchain is jointly maintained by n nodes.
- The probability of error occurred for the nodes in a transaction is P.

According to the Byzantine agreement, error occurred when more than $1/3$ nodes occurred error in the blockchain system. The probability of the error which is not occurred in the process of the execution of Tu is:

$$1 - \sum_{i=\frac{n}{3}+1}^{n} C_n^i P^i (1-P)^{n-i}$$

So when m transactions are implemented continuously, the probability of error not occurred is:

$$\left(1 - \sum_{i=\frac{n}{3}+1}^{n} C_n^i P^i (1-P)^{n-i}\right)^m$$

2) Tb

First, we assume that:

- Transaction is a Tb, and needs to communicate between branches.
- The blockchain is jointly maintained by n nodes.
- The probability of error occurred for each node in a transaction is P1.
- The probability of the error occurred in transaction transmission is P_2.

The probability of the error which is not occurred in the process of the execution of Tb is:

$$\left(1 - \sum_{i=\frac{n}{3}+1}^{n} C_n^i {P_1}^i (1-P_1)^{n-i}\right)^2 \mathrm{P}_2$$

So when m transactions are implemented continuously, the probability of an error not occurred is:

$$\left(1 - \sum_{i=\frac{n}{3}+1}^{n} C_n^i {P_1}^i (1-P_1)^{n-i}\right)^{2m} P_2^m$$

C. Scalability

Scalability for node and blockchain are very important aspects in MBDC. In term of node, the branch could add or remove node for the blockchain at any time without worrying about affecting the normal operation in MBDC model. While the other agencies need to supervise the branch's transaction or make clearing and settlement at a fixed time, they could just place a node which could connect to the branch's blockchain system in their office.

As for the blockchain, in one hand, the new branch could build its own blockchain system. Of course, the branch could participate in a consortium blockchain if its transaction amount is very little. On the other hand, when the transaction amount is too much for the branch to execute in time, the branch could build LAB. Finally, the transaction can be also considered as an innovation which shows the scalability of the system. The branch could increase the type of transaction through setting up the method field in the transaction.

So the scalability in the MBDC model mainly embodies in blockchain system and node and the transaction. And scalability for blockchain could ensure the enhancement of the speed of transaction execution with the increase number of blockchains.

VI. Experimental Analysis

A. Experimental environment

Due to the limited experimental conditions, this paper constructs a minimum Byzantine system composed of four nodes.

1) Hardware environment

Four computers are used for the experiment, two ASUS ESC1000 workstations which possess a 2.67 GHz Intel Xeon W3580 processor and 8G RAM and 8core is used. The rest two computers are the Lenovo M8000t which possess a 2.66GHz Intel Q9400 processor and 4G RAM.

2) Software environment

In order to facilitate the experiment and eliminate other interference factors, the four computers use the same software configuration.

- Operating system: CentOS 6.8
- Database: MySQL 5.1.73
- K-V database: Leveldb 0.7
- Transaction cache: Redis 3.2.1
- Java version: Java 1.8

3) Network topology

In order to eliminate the interference of other factors on the experiment, the four nodes are connected to the same switch and use the same IP subnet to build a simple private blockchain network.

B. Exeperimental analysis

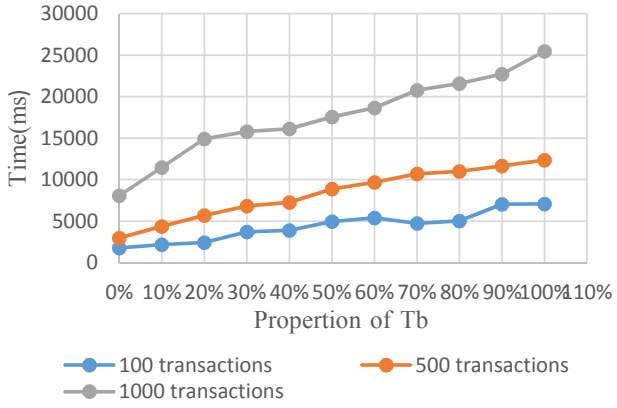
To verify the MBDC model, we build our own blockchain system which is different from Hyperledger and Etherum. The simulated transactions are designed according to the transaction format and the process of execution. The purpose of the experiment is to examine the feasibility and effective of the model. In addition, the performance of MBDC model is also an important factor for assessment of the model.

First, the constitution of time in transaction execution process is tested, and the result is shown as TABLE II.

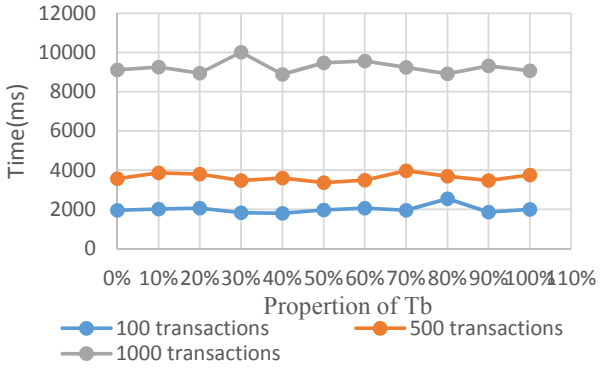
Constitution of time in Tb process (ms)	
Receive TX	150
Block Compute	400
Check TX	4
Insert into database	25
Receive Tb feedback	200
Send to user	150

TABLE II. Constitution of time in Transaction execution process (ms)

Then we tested the time of transaction execution including the Tb and Tu. The figure shows the executive time of 100, 500 and 1000 transactions in two blockchains. Especially, with the increasing number of the proportion of Tb in the transactions, the time gets longer. Meanwhile, bank account according to the bank reserves protocol is established to improve the efficiency of the process of transaction execution. And the clearing and settlement between two branches is also simulated in a fixed time. The result is shown as Fig. 6.



(a) Tu and Tb protocol



(b) Bank reserves protocol

Fig. 6. Comparison in time of 100, 500, 1000 transctions execution used three protocols. In order to verify the performance of system.

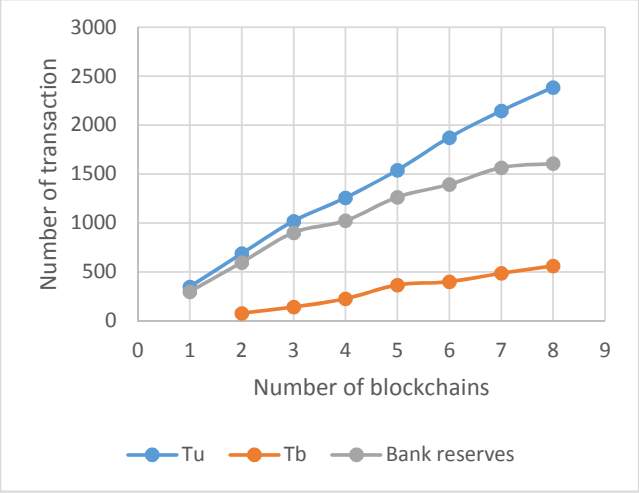


Fig. 7. Throughput in transactions per second, as a function of number of blockchains.

The result is obvious according to the figure. The transaction execution time for Tb is about four times than Tu. Because a Tb transaction process equals to the three times of Tu transaction processes and another two blockchain communications. And the result is also corresponding with the blockchain traffic which has been analysed above. When we used the bank reserves protocol to design the account and tested the transaction execution time, the blockchain traffic between two branches is sharply decreased.

Finally, we tested the throughput in transaction per second with an increasing number of blockchains. And the experimental result is shown as Fig. 7.

The transaction speed could reach about 2300tps/per second. We can see that the throughput grows near to linear. The type of Tu grows fast while the Tb grows slowest. The Tb execution process has involved two or more branches, so that the speed is seriously influenced by the communication between branches. The throughput of Tb with the usage of "bank reserves protocol" also grows fast. Of course, the transaction allocation strategy is also critical factor for the throughput of Tb. However, we haven't tested the effect of the allocation strategy for the Tb.

In addition, the result also proves that the branch can increase the transaction speed through increasing the number of blockchains. And the experimental result also proves that the MBDC is scalable.

VII. Conclusion

In this paper, we proposed the MBDC and its components and communication protocol. However, there are many problems to be solved. For example, the parallel technology used for the execution of transaction and establishment of block and consensus protocol to improve the throughput needs to research. The technology of node cluster is also used to improve the efficiency of nodes. And the allocation strategy for transaction is also an important problem to solve.

Acknowledgment

This work was partially supported by the National Natural Science Foundation of China under Grant 61672075 and 91538202, Funding of Ministry of Education and China Mo-bile MCM20160203, Project of the State Key Laboratory of Software Development Environment of China under Grant SKLSDE-2016ZX-16.

References

[1] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system[J]. Consulted, 2009.

[2] Swan M. Blockchain: Blueprint for a New Economy[M]. O'Reilly Media, Inc. 2015.

[3] Tschorsch F, Scheuermann B. Bitcoin and Beyond: A Technical Survey on Decentralized Digital Currencies[J]. IEEE Communications Surveys & Tutorials, 2016:1-1.

[4] Grinberg R. Bitcoin: An Innovative Alternative Digital Currency[J]. Social Science Electronic Publishing, 2011.

[5] Aitzhan N Z, Svetinovic D. Security and Privacy in Decentralized Energy Trading through Multi-signatures, Blockchain and Anonymous Messaging Streams[J]. 2016, PP(99):1-1.

[6] O'Leary D, D'Agostino V, Re R S, et al. Method and system for processing internet payments using the electronic funds transfer network: US, US8595083[P]. 2013.

[7] D'Antona J. Nasdaq Launches Enterprise-Wide Blockchain Technology Initiative[J]. Tradersmagazine Com, 2015.

[8] R. Parhonyi. Micropayment Systems. In Handbook of Financial Cryptography and Security. CRC, 2011.

[9] M. Belenkiy. E-Cash. In Handbook of Financial Cryptography and Security. CRC, 2011.

[10] Danezis G, Meiklejohn S. Centrally Banked Cryptocurrencies[J]. 2015.

[11] Bank of India, White Paper-Application of blockchain technology to banking and financial sector in india. January 2017.

[12] D. Schwartz, N. Youngs, and A. Britto, "The Ripple protocol consensus algorithm," 2014,ripple.com/files/ripple consensus whitepaper.pdf.

[13] Haferkorn M, Quintana Diaz J M. Seasonality and Interconnectivity within Crypto-Currencies - An Analysis on the Basis of Bitcoin, Litecoin and Namecoin[J]. 2015, 217

[14] King S, Nadal S. PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake[J]. 2012.

[15] Garay J, Kiayias A, Leonardos N. The Bitcoin Backbone Protocol: Analysis and Applications[M]// Advances in Cryptology - EUROCRYPT 2015. Springer Berlin Heidelberg, 2015:281-310.

[16] Daniel Larimer. Transactions as Proof-of-Stake. http://7fvhfe.com1.z0.glb.clouddn.com/wpcontent/uploads/2014/01/TransactionsAsProofOfStake10.pdf .

[17] Castro M, Liskov B. Practical Byzantine fault tolerance[C]// Symposium on Operating Systems Design and Implementation. USENIX Association,1999:173-186.

[18] White Paper, https://github.com/ethereum/wiki/wiki/White-Paper[R].

[19] Bonneau J, Felten E W, Kalodner H, et al. Securing Bitcoin wallets via a new DSA/ECDSA threshold signature scheme[J]. 2015.

[20] Rivest R. The MD5 Message-Digest Algorithm[M]. RFC Editor, 1992.

[21] Johnson D, Menezes A, Vanstone S. The Elliptic Curve Digital Signature Algorithm (ECDSA)[J]. International Journal of Information Security, 2001, 1(1):36-63.

[22] Karame G O, Androulaki E, Capkun S. Double-spending fast payments in bitcoin[C]// ACM Conference on Computer and Communications Security. ACM, 2012:906-917.

[23] Rosenfeld M. Analysis of Hashrate-Based Double Spending[J]. Eprint Arxiv, 2014.

[24] D. Mazi`eres, "The Stellar consensus protocol: a federated model for Internet-level consensus," 2015, www.stellar.org/papers/stellar-consensus-protocol.pdf.

[25] G Wood, Ethereum: a secure decentralised generalised transaction ledger[J].http://gavwood.com/Paper.pdf