

METHOD, APPARATUS AND ELECTRONIC DEVICE FOR BLOCKCHAIN TRANSACTIONS

CROSS REFERENCE TO RELATED APPLICATION

[0001] The present application is based on and claims priority to Chinese Patent Application No. 201811258194.X, filed on October 26, 2018, which is incorporated herein by reference in its entirety.

TECHNICAL FIELD

[0002] One or more embodiments of the present specification relate to the technical field of blockchain, particularly to a method, apparatus, and electronic device for implementing blockchain-based transactions.

BACKGROUND

[0003] Blockchain nodes of a blockchain network may jointly maintain a unified blockchain ledger through a consensus reached among the blockchain nodes, to record data of transactions having occurred on the blockchain network. The blockchain ledger can be made public for viewing and verification of historical data of the occurred transactions at any time.

SUMMARY

[0004] One or more embodiments of the present specification provide a method, apparatus, non-transitory computer-readable storage medium, and electronic device for implementing blockchain-based transactions.

[0005] According to a first aspect of the one or more embodiments of the present specification, provided is a method for implementing blockchain-based transactions, implementable by a computing device. The method may comprise: determining a to-be-remitted amount for each of a plurality of remitters participating in a transaction and a to-be-received amount for each of a plurality of receivers participating in the transaction, wherein: the plurality of remitters include one or more real remitters, the plurality of receivers include one or more real receivers, and the plurality of remitters include one or more cover-up remitters and/or the plurality of receivers include one or more cover-up receivers; generating a commitment of the to-be-remitted amount corresponding to the each remitter and a commitment of the to-be-received amount corresponding to the each receiver;

and submitting to a blockchain the transaction comprising a blockchain account address of the each remitter, a blockchain account address of the each receiver, the each commitment of the to-be-remitted amount, and the each commitment of the to-be-received amount, for a corresponding commitment of the to-be-remitted amount to be deducted from a commitment of the each remitter's balance recorded in the blockchain in association with the blockchain account address of the each remitter, and for a corresponding commitment of the to-be-received amount to be added to a commitment of the each receiver's balance recorded in the blockchain in association with the blockchain account address of the each receiver, after the transaction is implemented by the blockchain.

[0006] In some embodiments, the to-be-remitted amount corresponding to each of the one or more real remitters and the to-be-received amount corresponding to each of the one or more real receivers are preset; and after the transaction is implemented by the blockchain, the to-be-remitted amount is deducted from the each real remitter's balance corresponding to the blockchain account address of the real remitter, and the to-be-received amount is added to the each real receiver's balance corresponding to the blockchain account address of the real receiver.

[0007] In some embodiments, after the transaction is implemented by the blockchain, the each cover-up remitter's balance in association with a corresponding blockchain account address and the each cover-up receiver's balance in association with a corresponding blockchain account address remain unchanged.

[0008] In some embodiments, the commitment of the each receiver's balance comprises a homomorphic encryption of the each receiver's balance based on a first random number; and the commitment of the to-be-received amount corresponding to the each receiver comprises a homomorphic encryption of the to-be-received amount based on a second random number. In some embodiments, the homomorphic encryption is Pedersen Commitment.

[0009] In some embodiments, before submitting to the blockchain the transaction, the method may further comprise: sending the second random number to a second computing device associated with the each receiver via an off-chain channel for the second computing device to update the each receiver's balance according to an updated commitment of the each receiver's balance and an updated first random number.

[0010] In some embodiments, the updated commitment of the each receiver's balance is a

sum of the commitment of the to-be-received amount and the commitment of the receiver's balance; the updated first random number is a sum of the second random number and the first random number; and the each receiver's balance is updated to be a sum of the each receiver's balance and the to-be-received amount corresponding to the each receiver.

[0011] In some embodiments, before submitting to the blockchain the transaction, the method may further comprise sending the to-be-received amount and the commitment of the to-be-received amount to the second computing device associated with the each receiver via the off-chain channel for the second computing device associated with the each receiver to verify an association among the commitment of the to-be-received amount, the second random number, and the to-be-received amount.

[0012] In some embodiments, the method may further comprise obtaining a receiver signature endorsing the commitment of the to-be-received amount, wherein the receiver signature is generated by the second computing device associated with the each receiver based on a receiver private key, in response to successful verification of the association; and adding the receiver signature to the transaction for a plurality of blockchain nodes maintaining the blockchain to verify the receiver signature.

[0013] In some embodiments, the method may further comprise obtaining a remitter signature generated by a third computing device associated with a remitter based on a remitter private key; and adding the remitter signature to the transaction for a plurality of blockchain nodes maintaining the blockchain to verify the remitter signature.

[0014] In some embodiments, the method may further comprise obtaining a remitter range proof and a receiver range proof; and adding the remitter range proof and the receiver range proof to the transaction for a plurality of blockchain nodes maintaining the blockchain to: verify that the to-be-remitted amount corresponding to the each remitter is not less than zero, verify based on the remitter range proof that the each remitter's balance is not less than the to-be-remitted amount corresponding to the each remitter, and verify based on the receiver range proof that the to-be-received amount corresponding to the each receiver is not less than zero.

[0015] In some embodiments, the commitment of the each to-be-remitted amount corresponding to the each remitter comprises a homomorphic encryption of the to-be-remitted amount based on a third random number corresponding to the remitter; and the commitment of the each to-be-received amount corresponding to the each receiver

comprises a homomorphic encryption of the to-be-received amount based on a fourth random number corresponding to the receiver.

[0016] In some embodiments, the method may further comprise generating a verification random number by summing the third random number corresponding to the each remitter and subtracting the fourth random number corresponding to the each receiver; and adding the verification random number into the transaction for a plurality of blockchain nodes maintaining the blockchain to verify that an input of the transaction is equal to an output of the transaction.

[0017] According to a second aspect of the one or more embodiments of the present specification, an apparatus for implementing blockchain-based transactions is provided. The apparatus may comprise: one or more processors and one or more non-transitory computer-readable memories coupled to the one or more processors and configured with instructions executable by the one or more processors to cause the apparatus to perform operations comprising: determining a to-be-remitted amount for each of a plurality of remitters participating in a transaction and a to-be-received amount for each of a plurality of receivers participating in the transaction, wherein: the plurality of remitters include one or more real remitters, the plurality of receivers include one or more real receivers, and the plurality of remitters include one or more cover-up remitters and/or the plurality of receivers include one or more cover-up receivers; generating a commitment of the to-be-remitted amount corresponding to the each remitter and a commitment of the to-be-received amount corresponding to the each receiver; and submitting to a blockchain the transaction comprising a blockchain account address of the each remitter, a blockchain account address of the each receiver, the each commitment of the to-be-remitted amount, and the each commitment of the to-be-received amount, for a corresponding commitment of the to-be-remitted amount to be deducted from a commitment of the each remitter's balance recorded in the blockchain in association with the blockchain account address of the each remitter, and for a corresponding commitment of the to-be-received amount to be added to a commitment of the each receiver's balance recorded in the blockchain in association with the blockchain account address of the each receiver, after the transaction is implemented by the blockchain.

[0018] According to a third aspect of the one or more embodiments of the present specification, a non-transitory computer-readable storage medium is provided. The non-transitory computer-readable storage medium may be configured with instructions

executable by one or more processors to cause the one or more processors to perform operations comprising: determining a to-be-remitted amount for each of a plurality of remitters participating in a transaction and a to-be-received amount for each of a plurality of receivers participating in the transaction, wherein: the plurality of remitters include one or more real remitters, the plurality of receivers include one or more real receivers, and the plurality of remitters include one or more cover-up remitters and/or the plurality of receivers include one or more cover-up receivers; generating a commitment of the to-be-remitted amount corresponding to the each remitter and a commitment of the to-be-received amount corresponding to the each receiver; and submitting to a blockchain the transaction comprising a blockchain account address of the each remitter, a blockchain account address of the each receiver, the each commitment of the to-be-remitted amount, and the each commitment of the to-be-received amount, for a corresponding commitment of the to-be-remitted amount to be deducted from a commitment of the each remitter's balance recorded in the blockchain in association with the blockchain account address of the each remitter, and for a corresponding commitment of the to-be-received amount to be added to a commitment of the each receiver's balance recorded in the blockchain in association with the blockchain account address of the each receiver, after the transaction is implemented by the blockchain.

[0019] According to a fourth aspect of the one or more embodiments of the present specification, an electronic device is provided. The electronic device may comprise: a processor, and a memory for storing instructions executable by the processor, wherein the processor executes the instructions to achieve the methods in any of the foregoing embodiments.

BRIEF DESCRIPTION OF THE DRAWING

[0020] FIG. 1 is a flow chart of a method for implementing blockchain-based transactions according to some embodiments.

[0021] FIG. 2 is a schematic diagram of implementation of a remittance transaction in a blockchain network according to some embodiments.

[0022] FIG. 3 is a schematic diagram of achievement of anonymous remittance in a blockchain according to some embodiments.

[0023] FIG. 4 is a flow chart of implementation of a remittance transaction in a blockchain network according to some embodiments.

[0024] FIG. 5 is a structure diagram of a device according to some embodiments.

[0025] FIG. 6 is a block diagram of an apparatus for implementing blockchain-based transactions according to some embodiments.

DETAILED DESCRIPTION

[0026] Embodiments will be described in detail, with examples shown in the accompanying drawings. When the description below involves the accompanying drawings, unless otherwise indicated, the same numeral in different accompanying drawings stands for the same or similar element. The implementation manners described in the following embodiments do not represent all the implementation manners consistent with the present specification. Conversely, they are only examples of the apparatuses and methods described in detail in the attached claims and consistent with some aspects of one or more embodiments of the present specification.

[0027] In some embodiments, it is not necessary to perform steps in the methods according to the sequence illustrated and described in the present specification. In some other embodiments, the steps of the methods may be more or less than those described in the present specification. Further, a single step described in the present specification may be split into multiple steps for description in other embodiments, while multiple steps described in the present specification may be merged into one single step for description in yet other embodiments.

[0028] In the specification, a remitter, receiver, and organizer may refer to an individual or entity participating the blockchain-based transactions. For example, a remitter may be an individual or entity remitting money to a receiver through a transaction. A receiver may be an individual or entity receiving money from a remitter through the transaction. An organizer may be an individual or entity organizing or facilitating the transaction. For convenience and conciseness, a remitter, receiver, and organizer described in the specification may also refer to a computing device associated with the human or entity remitter, receiver, and organizer respectively. The computing device may be a personal computer, a laptop computer, a cell phone, a camera phone, a smart phone, a PDA (personal digital assistant), a media player, a navigation device, an email sending and receiving device, a game console, a tablet computer, a wearable device or any combination of a few of these devices.

[0029] FIG. 1 is a flow chart of a method for implementing blockchain-based transactions

according to one embodiment. As shown in FIG. 1, this method may be implemented by one or more computing devices as blockchain nodes, and may comprise the following steps:

[0030] Step 102, determining a to-be-remitted amount for each of multiple remitters participating in a transaction and a to-be-received amount for each of multiple receivers participating in the transaction, wherein the multiple remitters include one or more real remitters, the multiple receivers include one or more real receivers, the multiple remitters include one or more cover-up remitters and/or the multiple receivers include one or more cover-up receivers.

[0031] In some embodiments, the plurality of remitters may include one or more real remitters and one or more cover-up remitters. A blockchain transaction solution provided by the present specification may be applied to a remitter device corresponding to a real or cover-up remitter, such as a remitter device corresponding to either a real remitter or a cover-up remitter. Alternatively, the plurality of remitters may only include one or more real remitters, and a blockchain transaction solution provided by the present specification may be applied to remitter devices corresponding to real remitters.

[0032] In some embodiments, a blockchain transaction solution provided by the present specification may be applied to an organizer device corresponding to an organizer. The organizer may be any of the foregoing remitters. Alternatively, the organizer may not be a remitter. This is not limited by the embodiments of the present specification.

[0033] In still some embodiments, by including one or more cover-up remitters in the all remitters, the one or more real remitters' identities may be hidden. In other words, it is indeterminable whether each remitter is a real remitter or a cover-up remitter. When there are a large number of cover-up remitters, the one or more real remitters can be covered even better. Similarly, by including one or more cover-up receivers in the all receivers, the one or more real receivers' identities may be hidden. In other words, it is indeterminable whether each receiver is a real receiver or a cover-up receiver. When there are a large number of cover-up receivers, real receivers can be covered even better.

[0034] In still other embodiments, the remitters may negotiate with each other to determine to-be-remitted amounts and to-be-received amounts. The to-be-remitted amount of one or more real remitters and the to-be-received amount of one or more real receivers match the remittance amount of the transaction between them. The to-be-remitted amount of one or more cover-up remitters and the to-be-received amount of one or more cover-up

receivers may use any value determined through negotiation. For example, the to-be-remitted amount of a cover-up remitter and the to-be-received amount of a cover-up receiver may both be zero, while the to-be-remitted amount of a real remitter and the to-be-received amount of a real receiver may be both the remittance amount. This method may avoid incurring changes to blockchain accounts of cover-up remitters and cover-up receivers and thus help to reduce the complexity of subsequent processing. In addition, the to-be-remitted amounts and to-be-received amounts may be recorded in the blockchain accounts in the form of commitments of the to-be-remitted amounts and commitments of the to-be-received amounts respectively. The commitments of zero values and non-zero values are indistinguishable from one another. Therefore, the actual values of the to-be-remitted amounts and to-be-received amounts cannot be recognized directly and identities of the real remitters and real receivers will be protected.

[0035] In some embodiments, by remitting an asset certificate corresponding to a to-be-remitted amount from a remitter's blockchain account and remitting an asset certificate corresponding to a to-be-received amount into a receiver's blockchain account, the remittance between the remitter and the receiver may be achieved. The asset certificate may correspond to a token, a digital asset, or any other intelligent asset in a blockchain. Alternatively, the asset certificate may also correspond to cash, security, discount coupon, real estate, and any other off-chain asset. This is not limited by embodiments of the present specification.

[0036] In some embodiments, cover-up remitters and cover-up receivers may correspond to real individuals or enterprises. For example, blockchain accounts held by these individuals or enterprises may be borrowed to be used as cover-ups, provided that these blockchain accounts will not suffer loss or privacy leak.

[0037] In still some embodiments, cover-up remitters and cover-up receivers may be used to achieve remittance cover-up exclusively, but do not correspond to real individuals or enterprises. In other words, cover-up remitters and cover-up receivers may correspond to virtual objects. In some embodiments, cover-up remitters and cover-up receivers may exist for a long time, and are used to achieve cover-up in the process of remittance. Alternatively, cover-up remitters and cover-up receivers may be created temporarily each time when remittance is to be implemented, and be written off after the remittance is completed.

[0038] Step 104, generating a commitment of the to-be-remitted amount corresponding to

the each remitter and generating a commitment of the to-be-received amount corresponding to the each receiver.

[0039] In some embodiments, the blockchain nodes in a blockchain respectively maintain a unified blockchain ledger based on consensus. By registering commitments of to-be-remitted amounts and commitments of to-be-received amounts in the blockchain, the blockchain ledger maintained by the blockchain nodes records the commitments of to-be-remitted amounts and commitments of to-be-received amounts, and may not directly record to-be-remitted amounts and to-be-received amounts. As a result, the to-be-remitted amounts and the to-be-received amounts are concealed as private data. Similarly, a remitter's balance may be recorded as a commitment of the remitter's balance in the blockchain ledger, and a receiver's balance may be recorded as a commitment of the receiver's balance in the blockchain ledger. Any type of balance associated with a remitter or a receiver may be recorded as a corresponding commitment of the balance in a blockchain ledger to protect the privacy of the owner. The relation between any balance and a corresponding commitment of the balance, the relation between any to-be-remitted amount and a corresponding commitment of the to-be-remitted amount, and the relation between any to-be-received amount and a corresponding commitment of the to-be-received amount described herein will not be elaborated again below. For conciseness, a commitment of a to-be-remitted amount may be referred to as a to-be-remitted amount commitment, a commitment of a to-be-received amount may be referred to as a to-be-received amount commitment, a commitment of the remitter's balance may be referred to as a remitter's balance commitment, and a commitment of the receiver's balance may be referred to as a receiver's balance commitment, hereinafter.

[0040] Further, a balance may be converted into a corresponding balance commitment through a homomorphic encryption algorithm. The balance commitment includes a homomorphic encryption of the balance based on a random number. Similarly, the foregoing to-be-remitted amount may be converted into a corresponding to-be-remitted amount commitment. Any type of homomorphic encryption algorithm may be adopted, such as a Pedersen commitment mechanism. A homomorphic encryption may be Pederson Commitment. This is not limited by the embodiments of the present specification.

[0041] Step 106, submitting to a blockchain the transaction comprising a blockchain account address of the each remitter, a blockchain account address of the each receiver, the each commitment of the to-be-remitted amount, and the each commitment of the to-be-

received amount, for a corresponding commitment of the to-be-remitted amount to be deducted from a commitment of the each remitter's balance recorded in the blockchain in association with the blockchain account address of the each remitter, and for a corresponding commitment of the to-be-received amount to be added to a commitment of the each receiver's balance recorded in the blockchain in association with the blockchain account address of the each receiver, after the transaction is implemented by the blockchain.

[0042] In some embodiments, the transaction amounts between the real remitters and the real receivers are preset amounts. After completion of the transaction, the preset amounts are deducted respectively from the real remitter's balances corresponding to the blockchain account addresses of the real remitters, the preset amounts are added respectively to the real receiver's balances corresponding to the blockchain account addresses of the real receivers, and the cover-up remitter's balances corresponding to the blockchain account addresses of the cover-up remitters and the cover-up receiver's balances corresponding to the blockchain account addresses of the cover-up receivers remain unchanged.

[0043] For example, to-be-remitted amounts of real remitters and to-be-received amounts of real receivers are both preset amounts, while to-be-remitted amounts of cover-up remitters and to-be-received amounts of cover-up receivers are both zero, such that remittance operations occur from the real remitters to the real receivers in the foregoing transaction process, while no funds flow between the cover-up remitters and the cover-up receivers.

[0044] In another example, real remitters, real receivers, cover-up remitters, and cover-up receivers all have a flow of funds in the foregoing interaction process. But cover-up remitters may also play a role of cover-up receivers, and cover-up receivers may also play a role of cover-up remitters. For example, a blockchain account functions as a cover-up remitter with a corresponding to-be-remitted amount of $E1$, and simultaneously functions as a cover-up receiver with corresponding to-be-received amounts of $E2$ and $E3$. In addition, $E1 = E2 + E3$. Therefore, the balance corresponding to this blockchain account's address is not changed after completion of the transaction (break-even is achieved between the to-be-remitted amount $E1$ and the to-be-received amounts $E2$ and $E3$).

[0045] In some embodiments, a commitment of a receiver's balance SB_j (also referred to as a receiver's balance commitment SB_j , hereinafter) corresponding to a receiver B_j is calculated by using a homomorphic encryption algorithm according to a receiver's balance

s_{B_j} and a receiver's balance random number r_{B_j} . A commitment T_{B_j} of a to-be-received amount t_{B_j} (also referred to as a to-be-received amount commitment T_{B_j} , hereinafter) corresponding to the receiver B_j is calculated by using the homomorphic encryption algorithm according to a to-be-received amount t_{B_j} and a to-be-received amount random number b_j . Blockchain nodes may send the to-be-received amount random number b_j to the receiver B_j via an off-chain channel, so that the receiver B_j determines an updated receiver's balance s_{B_j}' according to an updated receiver's balance commitment S_{B_j}' and an updated receiver's balance random number r_{B_j}' . For example, the updated receiver's balance commitment S_{B_j}' is obtained by adding the to-be-received amount commitment T_{B_j} to the receiver's balance commitment S_{B_j} , the updated receiver's balance random number r_{B_j}' is obtained by adding the to-be-received amount random number b_j to the receiver's balance random number r_{B_j} , and the updated receiver's balance s_{B_j}' is the sum of the receiver's balance s_{B_j} and the to-be-received amount t_{B_j} . The receiver B_j stands for the j^{th} receiver and when the number of receivers is " n ," $1 \leq j \leq n$.

[0046] In some embodiments, a receiver's balance random number r_{B_j} may refer to a random number used for generating the commitment of the receiver's balance. For example, the receiver's balance random number may be determined by an organizer or remitter. Similarly, a to-be-received amount random number b_j may refer to a random number used for generating the commitment of the to-be-received amount. For example, the to-be-received amount random number may be determined by an organizer or remitter.

[0047] In other embodiments, before the transaction is submitted, blockchain nodes may send a to-be-received amount t_{B_j} and a to-be-received amount commitment T_{B_j} to a receiver B_j via an off-chain channel for the receiver B_j to verify the association among the to-be-received amount commitment T_{B_j} , the to-be-received amount random number b_j , and the to-be-received amount t_{B_j} corresponding to the receiver B_j . The receiver B_j may authorize implementation of the transaction after the verification succeeds, otherwise may reject the implementation of the transaction. For example, the receiver B_j may provide a receiver signature described below after the verification succeeds, otherwise may reject to provide the receiver signature.

[0048] In some embodiments, blockchain nodes may obtain a receiver signature SIG_{B_j} that is generated by the receiver B_j via a receiver private key held by the receiver and is

related to the to-be-received amount commitment T_{B_j} . The receiver signature SIG_{B_j} is generated by the receiver B_j after the association has passed verification. Then the receiver signature SIG_{B_j} is added to the transaction for the blockchain nodes in the blockchain to conduct signature verification. For example, when a transaction does not contain this receiver signature SIG_{B_j} , the blockchain nodes receiving this transaction may determine that consensus failed, and thereby reject to proceed with this transaction.

[0049] In some embodiments, blockchain nodes may obtain a remitter signature SIG_{A_i} that is generated by the remitter A_i via a remitter private key held by the remitter, and add the remitter signature SIG_{A_i} to the transaction for the blockchain nodes in the blockchain to conduct signature verification. In other embodiments, the remitter signature may also be related to a range proof. For example, when a transaction does not contain this remitter signature, the blockchain nodes receiving this transaction may determine that consensus failed, and thereby reject to proceed with this transaction. The remitter A_i stands for the i^{th} remitter and when the number of remitters is “m,” $1 \leq i \leq m$.

[0050] In other embodiments, blockchain nodes may obtain a remitter range proof Pf_{A_i} and a receiver range proof Pf_{B_j} , and add the remitter range proof Pf_{A_i} and the receiver range proof Pf_{B_j} to the transaction for the blockchain nodes in the blockchain to conduct value verification. In some embodiments, the remitter range proof Pf_{A_i} is used to prove that a to-be-remitted amount t_{A_i} corresponding to a remitter A_i is not less than zero, and a remitter’s balance s_{A_i} corresponding to the remitter A_i is not less than the to-be-remitted amount t_{A_i} . The receiver range proof Pf_{B_j} is used to prove that a to-be-received amount t_{B_j} corresponding to a receiver B_j is not less than zero.

[0051] In some embodiments, a remitter A_1 , a remitter A_2 , ..., and a remitter A_m generate a remitter range proof Pf_{A_1} , a remitter range proof Pf_{A_2} , ..., and a remitter range proof Pf_{A_m} , respectively. A receiver B_1 , a receiver B_2 , ..., and a receiver B_n generate a receiver range proof Pf_{B_1} , a receiver range proof Pf_{B_2} , ..., and a receiver range proof Pf_{B_n} , respectively. Then, a blockchain node that has applied a blockchain transaction solution provided by the present specification may respectively receive the remitter range proof Pf_{A_i} and the receiver range proof Pf_{B_j} generated by the each remitter and receiver respectively. For example, the blockchain node may be a remitter A_x . If so, the remitter A_x may receive the remitter range proofs generated by the other remitters as well as the receiver range proofs generated by the receivers.

[0052] In other embodiments, a blockchain node that has applied a blockchain transaction solution provided by the present specification may generate the foregoing remitter range proof Pf_A_i and receiver range proof Pf_B_j respectively.

[0053] In some embodiments, a commitment T_A_i of the to-be-remitted amount t_A_i (also referred to as a to-be-remitted amount commitment T_A_j , hereinafter) is calculated by using a homomorphic encryption algorithm based on the to-be-remitted amount t_A_i and a to-be-remitted amount random number a_i corresponding to the remitter A_i , where $0 < i \leq m$ and m is the number of the remitters. The to-be-received amount commitment T_B_j is calculated by using a homomorphic encryption algorithm based on the to-be-received amount t_B_j and the to-be-received amount random number b_j corresponding to the receiver B_j , where $0 < j \leq n$ and n is the number of the receivers. The blockchain node may generate a verification random number $f = a_1 + \dots + a_m - b_1 - \dots - b_n$, and add the verification random number “f” to the transaction for the blockchain nodes in the blockchain to verify that the input of the transaction is equal to the output of the transaction. If the input is not equal to the output, execution of this transaction may be rejected. Similar to the to-be-received amount random number, a to-be-remitted amount random number a_i may refer to a random number used for generating the commitment of the to-be-remitted amount. For example, the to-be-received amount random number may be determined by an organizer or remitter.

[0054] A remittance transaction in a blockchain network is taken as an example below to describe the technical solution of the present specification in detail. FIG. 2 is a schematic diagram of implementation of a remittance transaction in a blockchain network according to various embodiments. As shown in FIG. 2, it is assumed that a user A conducts remittance to a user B in a blockchain. A “user” in the present specification may be manifested as a login user account, while this user account may belong to an individual or an organization. This is not limited by the embodiments of the present specification.

[0055] It is assumed that a remitter device used by the user A is a user device 1. For instance, a user account corresponding to the user A is logged in on this user device 1. Similarly, a receiver device used by the user B is a user device 2. An off-chain channel may be established between the user device 1 and user device 2 to achieve corresponding off-chain communication.

[0056] A client program of blockchain may be run on the user device 1 so that the user

device 1 runs a corresponding blockchain node in the blockchain network, such as the node 1 shown in FIG. 2. Similarly, a client program of blockchain may be run on the user device 2 so that the user device 2 also runs a corresponding blockchain node in the blockchain network, such as the node 2 shown in FIG. 2. The blockchain network also includes other blockchain nodes, such as the node k shown in FIG. 2. The blockchain nodes are not enumerated here. Through the foregoing node 1, node 2, etc., the remittance transaction between the user A and the user B may be implemented via the blockchain network, and related transaction information may be recorded in the blockchain ledger maintained by the blockchain nodes respectively, thereby preventing the recorded transaction information from being tampered with, and also facilitating subsequent inspection.

[0057] FIG. 3 is a schematic diagram of achievement of anonymous remittance in a blockchain according to various embodiments. In respect to the foregoing remittance transaction from the user A to the user B, the user A is a real remitter of the remittance transaction, and the user B is a real receiver of the remittance transaction. Based on a technical solution of the present specification, cover-up remitters other than the user A and cover-up receivers other than the user B may be added, shown as a remitter A_1, a remitter A_2, ..., a remitter A_m and a receiver B_1, a receiver B_2, ..., a receiver B_n in FIG. 3, to conceal the user A among the remitters A_1 ~ A_m and the user B among the receivers B_1 ~ B_n. For example, the user A may be any one of the remitters A_1 ~ A_m, and the user B may be any one of the receivers B_1 ~ B_n. In addition, the to-be-remitted amounts and to-be-received amounts may be also concealed through homomorphic encryption. This approach may hide the real identities of the each remitter and receiver in the public record of the blockchain transaction, and make it difficult to determine whether the remittance is made from the user A to the user B, or to determine the actual amount of the remittance, thereby achieving anonymous remittance in the blockchain.

[0058] It is assumed that the node 1 is an organizer among the remitters A_1 ~ A_m in FIG. 3. Through a process of interaction among the organizer, the receiver B_j (the processing methods of the receivers B_1 ~ B_n are the same and the “receiver B_j” is taken as an example) and the blockchain nodes as shown in FIG. 4, a remittance transaction between the user A and the user B is achieved. FIG. 4 is a flow chart of implementation of a remittance transaction in a blockchain network according to various embodiments.

[0059] As shown in FIG. 4, the process of interaction among the organizer, the receiver B_j and the blockchain nodes may comprise the following steps.

[0060] At step 401, the organizer negotiates with other remitters and drafts a transaction.

[0061] In some embodiments, it is assumed that the organizer is the remitter A_1 and the user B corresponds to the receiver B_1 , while the remitters $A_2 \sim A_m$ are cover-up remitters, and the receivers $B_2 \sim B_n$ are cover-up receivers. After it is determined that remittance is to be made from the user A to the user B, the remitter A_1 may select the foregoing remitters $A_2 \sim A_m$ and the receivers $B_2 \sim B_n$ by a preset method, or the remitters $A_2 \sim A_m$ and the receivers $B_2 \sim B_n$ are determined by any other method so that the remitters $A_2 \sim A_m$ and the receivers $B_2 \sim B_n$ may cooperate with the remitter A_1 and the receiver B_1 to achieve anonymous remittance in the blockchain.

[0062] In other embodiments, negotiations may be conducted among remitters $A_1 \sim A_m$ to determine the to-be-remitted amount t_{A_i} to which the each remitter A_i (where, $1 \leq i \leq m$) corresponds, as well as the to-be-received amount t_{B_j} to which the each receiver B_j (where, $1 \leq j \leq n$) corresponds.

[0063] In some embodiments, a corresponding to-be-remitted amount random number a_i may be determined for the each to-be-remitted amount t_{A_i} , and a corresponding to-be-received amount random number b_j may be determined for the each to-be-received amount t_{B_j} . Then through a Pedersen commitment (PC) mechanism, a commitment T_{A_i} of the to-be-remitted amount t_{A_i} and a commitment T_{B_j} of the to-be-received amount t_{B_j} corresponding are calculated respectively, where the to-be-remitted amount commitment $T_{A_i} = PC(a_i, t_{A_i}) = a_i \times G + t_{A_i} \times H$, the to-be-received amount commitment $T_{B_j} = PC(b_j, t_{B_j}) = b_j \times G + t_{B_j} \times H$, and G and H are known functions (such as generators used to generate an elliptic curve).

[0064] At step 402, the organizer sends $(A_1, \dots, A_m; b_j, t_{B_j}, T_{B_j})$ to a receiver B_j via an off-chain channel.

[0065] In some embodiments, sending $(A_1, \dots, A_m; b_j, t_{B_j}, T_{B_j})$ via an off-chain channel rather than a blockchain network may avoid recording the to-be-received amount random number b_j and to-be-received amount t_{B_j} in the blockchain ledger, and ensure the received remittance amount t_{B_j} is unknown except to remitter A_i and receiver B_j .

[0066] In other embodiments, the organizer sends $(A_1, \dots, A_m; b_1, t_{B_1}, T_{B_1})$ to the receiver B_1 , sends $(A_1, \dots, A_m; b_2, t_{B_2}, T_{B_2})$ to the receiver B_2 , ..., and sends $(A_1, \dots, A_m; b_j, t_{B_j}, T_{B_j})$ to a receiver B_j , and every receiver verifies the

receiver's corresponding to-be-received amount, to-be-received amount random number and commitment of the to-be-received amount.

[0067] In some embodiments, $A_1 \sim A_m$ in $(A_1, \dots, A_m; b_j, t_{B_j}, T_{B_j})$ represent the foregoing remitter $A_1 \sim$ remitter A_m .

[0068] At step 403, the receiver B_j verifies the received $(A_1, \dots, A_m; b_j, t_{B_j}, T_{B_j})$.

[0069] In some embodiments, the receiver B_j may verify the to-be-received amount commitment T_{B_j} . In other words, the receiver B_j may calculate the to-be-received amount random number b_j and the received remittance amount t_{B_j} through a Pedersen commitment mechanism to verify whether the commitment $T_{B_j} = PC(b_j, t_{B_j})$ is correct. If yes, it means that the verification is successful. If not, it means that the verification is not successful.

[0070] At step 404, after the verification is successful, the receiver B_j generates a signature and returns the signature to the organizer.

[0071] In some embodiments, after the verification is successful, the receiver B_j may use a receiver private key to sign $(A_1, \dots, A_m; b_j, t_{B_j}, T_{B_j})$, generate a signature SIG_{B_j} and return the signature to a remitter. This signature SIG_{B_j} indicates that the receiver B_j agrees to the remittance transaction participated by remitters $A_1 \sim A_m$, and that the receiver B_j will obtain the to-be-received amount corresponding to the commitment T_{B_j} from the remittance transaction.

[0072] At step 405, after receiving the signature SIG_{B_j} , the organizer generates range proofs Pf_{A_i} and Pf_{B_j} .

[0073] In some embodiments, after the organizer receives a signature SIG_{B_1} returned by the receiver B_1 , a signature SIG_{B_2} returned by the receiver B_2 , ...a signature SIG_{B_j} returned by the receiver B_j , ..., a signature SIG_{B_n} returned by the receiver B_n , the organizer determines that the receivers $B_1 \sim B_n$ permit implementation of the remittance transaction.

[0074] In other embodiments, the blockchain nodes may determine that the to-be-remitted amount t_{A_i} corresponding to the remitter A_i meets the following conditions: $t_{A_i} \geq 0$ and $s_{A_i} \geq t_{A_i}$, where s_{A_i} is a remitter's balance corresponding to the remitter A_i . This remitter's balance s_{A_i} is recorded as a corresponding remitter's balance

commitment S_{A_i} in a blockchain ledger. The range proof technology enables blockchain nodes to verify whether a transaction meets the foregoing conditions in a ciphertext state. For example, the present specification may be achieved by using a Bulletproofs solution, a Borromean ring signature solution, etc., which is not limited by the embodiments of the present specification. Each remitter A_i may use range proof technology to generate a corresponding range proof Pf_{A_i} , and provide the range proof to the organizer. Here, the organizer, as the remitter A_1 , may generate a range proof Pf_{A_1} by itself to prove $t_{A_1} \geq 0$ and $s_{A_1} \geq t_{A_1}$, while the other remitters $A_2 \sim A_m$ may generate range proofs $Pf_{A_2} \sim Pf_{A_m}$ respectively and send the range proofs to this organizer respectively.

[0075] In some embodiments, the blockchain nodes may determine that the to-be-received amount t_{B_j} corresponding to the receiver B_j meets the following conditions: $t_{B_j} \geq 0$. Similar to the foregoing range proof Pf_{A_i} , the range proof technology may also be used to generate a range proof Pf_{B_j} corresponding to the to-be-received amount t_{B_j} . If a remitter A_x makes a transfer to a receiver B_y , the remitter A_x may generate a range proof Pf_{B_y} corresponding to a to-be-received amount t_{B_y} for this receiver B_y , and send this range proof Pf_{B_y} to the organizer; alternatively, the organizer may generate a range proof Pf_{B_j} corresponding to the to-be-received amount t_{B_j} .

[0076] At step 406, the organizer calculates a verification random number f .

[0077] In some embodiments, based on the remitter random numbers $a_1 \sim a_m$ corresponding to the remitters $A_1 \sim A_m$, and the receiver random numbers $b_1 \sim b_n$ corresponding to the receivers $B_1 \sim B_n$, a corresponding verification random number $f = a_1 + \dots + a_m - b_1 - \dots - b_n$ may be calculated. Accordingly, based on the characteristics of the Pedersen commitment mechanism, this verification random number may be used in combination with the to-be-remitted amounts commitments $T_{A_1} \sim T_{A_m}$ and the to-be-received amounts commitments $T_{B_1} \sim T_{B_n}$ to verify $t_{a_1} + \dots + t_{a_m} = t_{b_1} - \dots - t_{b_n}$. In other words, the total to-be-remitted amount is equal to the total to-be-received amount, that is, the input of the remittance transaction is equal to the output of the remittance transaction.

[0078] At step 407, the organizer obtains a signature of each remitter.

[0079] In some embodiments, the remitters $A_1 \sim A_m$ sign ($A_1: T_{A_1}, Pf_{A_1}; \dots; A_m: T_{A_m}, Pf_{A_m}; B_1: T_{B_1}, Pf_{B_1}; \dots; B_n: T_{B_n}, Pf_{B_n}; f$) respectively to

obtain the corresponding remitter signatures SIG_A_1 ~ SIG_A_m, and provide the remitter signatures to the organizer.

[0080] At step 408, the organizer submits a transaction to the blockchain.

[0081] In some embodiments, the organizer submits a corresponding remittance transaction (A_1:T_A_1, Pf_A_1;...; A_m:T_A_m, Pf_A_m; B_1:T_B_1, Pf_B_1; ... ; B_n:T_B_n, Pf_B_n; f; SIG_A_1, ..., SIG_A_m; SIG_B_1, ..., SIG_B_n) to the blockchain via the node 1 for execution of this remittance transaction. The remittance transaction may be transmitted to all the blockchain nodes in the blockchain network and all the blockchain nodes may verify this remittance transaction respectively to execute a remittance operation when the verification is successful, or to reject the remittance transaction when the verification is not successful.

[0082] At step 409, a blockchain node checks whether the transaction has been executed.

[0083] In some embodiments, the blockchain node may be any blockchain node in the blockchain network. In other words, every blockchain node in the blockchain network may receive the foregoing remittance transaction, and implement verification and other operations through Steps 409 ~ 412.

[0084] In other embodiments, after the blockchain node receives the foregoing remittance transaction, the blockchain node may use double-spending prevention mechanism or replay attack prevention mechanism to verify whether this remittance transaction has been executed. If this remittance transaction has been executed, the execution of this remittance transaction may be rejected, otherwise the process proceeds to Step 410.

[0085] At step 410, the blockchain node validates the signatures.

[0086] In some embodiments, the blockchain node may check whether the signatures SIG_A_i and SIG_B_j contained in the foregoing remittance transaction are correct or not. If the signatures are not correct, the execution of this remittance transaction may be rejected, otherwise the process proceeds to Step 411.

[0087] At step 411, the blockchain node validates the range proofs Pf_A_i and Pf_B_j.

[0088] In some embodiment, the blockchain node may, based on range proof technology, check the range proof Pf_A_i contained in the foregoing remittance transaction to determine whether the conditions of $t_{A_i} \geq 0$ and $s_{A_i} \geq t_{A_i}$ are met or not. Similarly, the blockchain node may, based on range proof technology, check range proof Pf_B_j

contained in the foregoing remittance transaction to determine whether the condition of $t_{B_j} \geq 0$ is met or not. If any of the conditions is not met, the execution of this remittance transaction may be rejected, otherwise the process proceeds to Step 412.

[0089] At step 412, the blockchain node checks whether the total to-be-remitted amount is equal to the total to-be-received amount.

[0090] In some embodiments, for the verification random number f contained in the foregoing remittance transaction, the blockchain node may verify whether the equation $T_{A_1} + \dots + T_{A_m} - T_{B_1} - \dots - T_{B_n} = PC(f, 0) = f \times G + 0 \times H = f \times G$ holds. If yes, it means that in the foregoing remittance transaction, the total to-be-remitted amount is equal to the total to-be-received amount, and the process proceeds to Step 413. If not, the execution of this remittance transaction may be rejected.

[0091] At step 413, in the maintained blockchain ledger, the blockchain nodes update the blockchain accounts corresponding to the remitters $A_1 \sim A_m$ and receivers $B_1 \sim B_n$ respectively.

[0092] In some embodiments, after the verification at Steps 409 ~ 412 succeeds, the blockchain nodes may update the blockchain accounts corresponding to the remitters $A_1 \sim A_m$ and receivers $B_1 \sim B_n$ respectively: the remitter's balance commitments $T_{A_1} \sim T_{A_m}$ are added to the blockchain accounts corresponding to the remitters $A_1 \sim A_m$, while the receiver's balance commitments $T_{B_1} \sim T_{B_n}$ are added to the blockchain accounts corresponding to the receivers $B_1 \sim B_n$.

[0093] For example, the added remitter's balance commitment T_{A_1} corresponding to the blockchain account of the remitter A_1 (i.e., the blockchain account 1 of the user A), and the added receiver's balance commitment T_{B_1} corresponding to the blockchain account of the receiver B_1 (i.e., the blockchain account 2 of the user B), respectively correspond to the actual remittance amount. Therefore, the remittance operation from the user A to the user B is ensured.

[0094] With respect to the other remitters $A_2 \sim A_m$ and receivers $B_2 \sim B_n$, the corresponding to-be-remitted amounts $t_{A_2} \sim t_{A_m}$ and to-be-received amounts $t_{B_2} \sim t_{B_n}$ may all be zero, making the blockchain accounts corresponding to these cover-up remitters and cover-up receivers remain unchanged.

[0095] As the remitter's balance commitments $T_{A_1} \sim T_{A_m}$ and the receiver's

balances commitments $T_{B_1} \sim T_{B_n}$ are recorded in the blockchain ledger, the actual values corresponding to-be-remitted amounts $t_{A_1} \sim t_{A_m}$ and to-be-received amounts $t_{B_1} \sim t_{B_n}$ cannot be obtained from a query of remittance transaction records in the blockchain ledger. Therefore, privacy of the remittance amount can be protected, and real identities of the remitters and receivers will not be exposed to public, thereby achieving anonymous remittance in blockchain and significantly improving the security and privacy of blockchain remittance.

[0096] FIG. 5 is a structure diagram of a device according to various embodiments. At the level of hardware, the device comprises a processor 502, an internal bus 504, a network interface 506, an internal memory 508 and a nonvolatile memory (NVM) 510. The device may further comprise other hardware. In some embodiments, the processor 502 reads corresponding computer programs from the NVM 510 to the internal memory 508 and then executes the computer programs to form a blockchain transaction apparatus at a logical level. In addition to the software implementation manner, one or more embodiments of the present specification do not exclude other implementation manners, such as a logic device or a combination of software and hardware. That is, the executive bodies of the following processing flow are not limited to logic units and may be hardware or logic devices.

[0097] Referring to FIG. 6, a block diagram of an apparatus for implementing blockchain-based transactions is provided according to various embodiments. In the software implementation manner, the apparatus for implementing blockchain-based transactions is applied to blockchain nodes and may comprise: a determination unit 601 configured to determine a to-be-remitted amount for each of remitters participating in a transaction and a to-be-received amount for each of receivers participating in the transaction, where the remitters include real remitters, the receivers include real receivers, the remitters include one or more cover-up remitters and/or the receivers include one or more cover-up receivers; a first generation unit 602 configured to generate a commitment of the to-be-remitted amount corresponding to the each remitter and generating a commitment of the to-be-received amount corresponding to the each receiver; and a submission unit 603 configured to submit the transaction to a blockchain, where the transaction comprises a blockchain account address of the each remitter, a blockchain account address of the each receiver, the each commitment of the to-be-remitted amount and the each commitment of the to-be-received amount, and where after the transaction is completed a corresponding commitment of the to-be-remitted amount is deducted from a commitment of the each remitter's balance

recorded in the blockchain with the blockchain account address of the remitter and a corresponding commitment of the to-be-received amount is added to a commitment of the each receiver's balance recorded in the blockchain with the blockchain account address of the receiver.

[0098] In some embodiments, the transaction amounts between the real remitters and the real receivers are preset amounts. For example, after completion of the transaction, the preset amounts are deducted respectively from the real remitter's balances corresponding to the blockchain account addresses of the real remitters, the preset amounts are added respectively to the real receiver's balances corresponding to the blockchain account addresses of the real receivers, and the cover-up remitter's balances corresponding to the blockchain account addresses of the cover-up remitters and the cover-up receiver's balances corresponding to the blockchain account addresses of the cover-up receivers remain unchanged.

[0099] In other embodiments, the to-be-remitted amounts of the real remitters and the to-be-received amounts of the real receivers are both preset amounts, and the to-be-remitted amounts of the cover-up remitters and the to-be-received amounts of the cover-up receivers are both zero.

[00100] In still other embodiments, a receiver's balance commitment SB_j corresponding to a receiver B_j is calculated by using a homomorphic encryption algorithm according to a receiver's balance s_{B_j} and a receiver's balance random number r_{B_j} , and a to-be-received amount commitment T_{B_j} corresponding to a receiver B_j is calculated by using the homomorphic encryption algorithm according to a to-be-received amount t_{B_j} and a to-be-received amount random number b_j . The apparatus further comprises:

[00101] a first sending unit 604 configured to send a to-be-received amount random number b_j to a receiver B_j via an off-chain channel for the receiver B_j to determine an updated receiver's balance s_{B_j}' according to an updated receiver's balance commitment S_{B_j}' and an updated receiver's balance random number r_{B_j}' ;

[00102] For example, the updated receiver's balance commitment S_{B_j}' is obtained by adding the to-be-received amount commitment T_{B_j} to the receiver's balance commitment S_{B_j} , the updated receiver's balance random number r_{B_j}' is obtained by adding the to-be-received amount random number b_j to the receiver's balance random number r_{B_j} , and the updated receiver's balance s_{B_j}' is the sum of the receiver's balance s_{B_j} and the

to-be-received amount t_{B_j} .

[00103] In yet other embodiments, the apparatus further comprises:

[00104] a second sending unit 605 configured to send a to-be-received amount t_{B_j} and a to-be-received amount commitment T_{B_j} to a receiver B_j via an off-chain channel before submission of the transaction for the receiver B_j to verify the association among the to-be-received amount commitment T_{B_j} , the to-be-received amount random number b_j and the to-be-received amount t_{B_j} corresponding to the receiver B_j .

[00105] In other embodiments, the apparatus further comprises:

[00106] a first obtaining unit 606 configured to obtain a receiver signature SIG_{B_j} that is generated by the receiver B_j via a receiver private key held by the receiver and is related to the to-be-received amount commitment T_{B_j} , where the receiver signature SIG_{B_j} is generated by the receiver B_j after the association has passed verification,

[00107] a first addition unit 607 configured to add the receiver signature SIG_{B_j} to the transaction for the blockchain nodes in the blockchain to conduct signature verification.

[00108] In still other embodiments, the apparatus further comprises:

[00109] a second obtaining unit 608 configured to obtain a remitter signature SIG_{A_i} that is generated by a remitter A_i via a remitter private key held by the remitter,

[00110] a second addition unit 609 configured to add the remitter signature SIG_{A_i} to the transaction for the blockchain nodes in the blockchain to conduct signature verification.

[00111] In yet other embodiments, the apparatus further comprises:

[00112] a third obtaining unit 610 configured to obtain a remitter range proof Pf_{A_i} and a receiver range proof Pf_{B_j} ,

[00113] a third addition unit 611 configured to add the remitter range proof Pf_{A_i} and the receiver range proof Pf_{B_j} to the transaction for the blockchain nodes in the blockchain to conduct value verification, where the remitter range proof Pf_{A_i} is used to prove that a to-be-remitted amount t_{A_i} corresponding to a remitter A_i is not less than zero, and a remitter's balance s_{A_i} corresponding to the remitter A_i is not less than the to-be-remitted amount t_{A_i} , and the receiver range proof Pf_{B_j} is used to prove that a to-be-received amount t_{B_j} corresponding to a receiver B_j is not less than zero.

[00114] In other embodiments, a to-be-remitted amount commitment T_{A_i} is calculated

by using a homomorphic encryption algorithm based on the to-be-remitted amount t_{A_i} and a to-be-remitted amount random number a_i corresponding to the remitter A_i , where $0 < i \leq m$ and m is the number of the remitters, and a to-be-received amount commitment T_{B_j} is calculated by using a homomorphic encryption algorithm based on the to-be-received amount t_{B_j} and the to-be-received amount random number b_j corresponding to the receiver B_j , where $0 < j \leq n$ and n is the number of the receivers. The apparatus further comprises:

[00115] a second generation unit 612 configured to generate a verification random number $f = a_1 + \dots + a_m - b_1 - \dots - b_n$;

[00116] a fourth addition unit 613 configured to add the verification random number f in the transaction for the blockchain nodes in the blockchain to verify that the input of the transaction is equal to the output of the transaction.

[00117] The systems, apparatuses, modules or units illustrated in the foregoing embodiments may be achieved by computer chips or entities, or by products with certain functions. A typical device is computer. An example of the computer may be a personal computer, laptop computer, cell phone, camera phone, smart phone, PDA (personal digital assistant), media player, navigation device, email sending and receiving device, game console, tablet computer, wearable device or any combination of a few of these devices.

[00118] In a typical configuration, the computer comprises one or more processors (CPU), I/O interfaces, network interfaces and internal memories. The internal memory may be in a form of volatile memory, random access memory (RAM) and/or non-volatile memory such as read-only memory (ROM) or flash memory (flash RAM) in the computer readable media. Internal memory is an example of computer readable media.

[00119] Computer readable media include non-volatile and volatile, movable and non-movable media and may achieve information storage by any method or technology. Information may be computer readable instructions, data structures, program modules or other data. Examples of computer storage media include, but are not limited to, phase change random access memory (PRAM), static random access memory (SRAM), dynamic random access memory (DRAM) and other types of random access memory (RAM), read only memory (ROM), electrically erasable programmable read only memory (EEPROM), flash memory or other memory technologies, compact disc-read only memory (CD-ROM), digital versatile disc (DVD) or other optical memory, cassette type magnetic tape,

magnetic disk memory, quantum memory, graphene-based storage media, or other magnetic storage devices or any other non-transfer media, which may be used to store information that is accessible to computer devices. According to the specification, computer readable media do not include transitory media, such as modulated data signals and carriers.

[00120] It should be further noted that terms “include,” “contain” or any variants of these terms are intended to cover non-exclusive inclusion so that a process, method, commodity or equipment including a series of elements not only includes these elements but also includes other elements not clearly set out, or also include the elements inherent to such process, method, commodity or equipment. Unless otherwise with more limitations, the elements defined by a sentence “include one.....” do not exclude that the process, method, commodity or equipment including the elements also have other same elements.

[00121] Embodiments of the present specification are described above. Other embodiments are in the scope of the attached claims. In some embodiments, the actions or steps recorded in the claims may be executed in a sequence different from that given in the embodiments and can still achieve the expected results. Further, it is not necessary for the process described in the accompanying drawings to require the given specific sequence or a continuous sequence in order to achieve the expected results. In some embodiments, multitasking processing and parallel processing are also acceptable or may be favorable.

[00122] The terms used in one or a plurality of embodiments of the present specification are for the purpose of describing embodiments and not intended to limit one or more embodiments of the present specification. The singular forms “one,” “the” and “this” used in one or more embodiments of the present specification and in the attached claims also are intended to cover plural forms unless their meanings are otherwise clearly indicated in the context. It should also be understood that the term “and/or” used in the text refers to any or all possible combinations containing one or more associated listed items.

[00123] It should be understood that although one or more embodiments of the present specification may use terms such as first, second and third to describe various kinds of information, the information should not be limited to these terms. These terms are intended to differentiate information of the same type. For example, without departing from the scope of one or more embodiments of the present specification, first information may also be referred to as second information, and similarly, second information may also be referred to as first information. Subject to the context, term “if” used here may be interpreted as “at

the time of ...,” “when,” or “in response to determination.”

[00124] The foregoing description is not intended to limit one or more embodiments of the present specification. Any modifications, equivalent replacements and improvements made without departing from the spirit and principle of one or more embodiments of the present specification shall fall within the scope of one or more embodiments of the present specification.

Claims

1. A method for implementing blockchain-based transactions, implementable by a computing device, the method comprising:

determining a to-be-remitted amount for each of a plurality of remitters participating in a transaction and a to-be-received amount for each of a plurality of receivers participating in the transaction, wherein:

the plurality of remitters include one or more real remitters,

the plurality of receivers include one or more real receivers, and

the plurality of remitters include one or more cover-up remitters and/or the plurality of receivers include one or more cover-up receivers;

generating a commitment of the to-be-remitted amount corresponding to the each remitter and a commitment of the to-be-received amount corresponding to the each receiver; and

submitting to a blockchain the transaction comprising a blockchain account address of the each remitter, a blockchain account address of the each receiver, the each commitment of the to-be-remitted amount, and the each commitment of the to-be-received amount, for a corresponding commitment of the to-be-remitted amount to be deducted from a commitment of the each remitter's balance recorded in the blockchain in association with the blockchain account address of the each remitter, and for a corresponding commitment of the to-be-received amount to be added to a commitment of the each receiver's balance recorded in the blockchain in association with the blockchain account address of the each receiver, after the transaction is implemented by the blockchain,

wherein:

the commitment of the each receiver's balance comprises a homomorphic encryption of the each receiver's balance based on a first random number; and

the commitment of the to-be-received amount corresponding to the each receiver comprises a homomorphic encryption of the to-be-received amount based on a second random number,

wherein before submitting to the blockchain the transaction, the method further comprising:

sending the second random number to a second computing device associated with the each receiver via an off-chain channel for the second computing device to update the each receiver's balance according to an updated commitment of the each receiver's balance and an updated first random number.

2. The method of claim 1, wherein:

the to-be-remitted amount corresponding to each of the one or more real remitters and the to-be-received amount corresponding to each of the one or more real receivers are preset; and

after the transaction is implemented by the blockchain, the to-be-remitted amount is deducted from the each real remitter's balance corresponding to the blockchain account address of the real remitter, and the to-be-received amount is added to the each real receiver's balance corresponding to the blockchain account address of the real receiver.

3. The method of any of the claims 1-2, wherein, after the transaction is implemented by the blockchain, the each cover-up remitter's balance in association with a corresponding blockchain account address and the each cover-up receiver's balance in association with a corresponding blockchain account address remain unchanged.

4. The method of claim 1, wherein:

the updated commitment of the each receiver's balance is a sum of the commitment of the to-be-received amount and the commitment of the receiver's balance;

the updated first random number is a sum of the second random number and the first random number; and

the each receiver's balance is updated to be a sum of the each receiver's balance and the to-be-received amount corresponding to the each receiver.

5. The method of any of the claims 1 or 4, before submitting to the blockchain the transaction, further comprising:

sending the to-be-received amount and the commitment of the to-be-received amount to the second computing device associated with the each receiver via the off-chain channel for the second computing device associated with the each receiver to verify an association among the commitment of the to-be-received amount, the second random number, and the to-be-received amount.

6. The method of claim 5, further comprising:

obtaining a receiver signature endorsing the commitment of the to-be-received amount, wherein the receiver signature is generated by the second computing device associated with the each receiver based on a receiver private key, in response to successful verification of the association; and

adding the receiver signature to the transaction for a plurality of blockchain nodes maintaining the blockchain to verify the receiver signature.

7. The method of any preceding claim, further comprising:

obtaining a remitter signature generated by a third computing device associated with a remitter based on a remitter private key; and

adding the remitter signature to the transaction for a plurality of blockchain nodes maintaining the blockchain to verify the remitter signature.

8. The method of any preceding claim, further comprising:

obtaining a remitter range proof and a receiver range proof; and

adding the remitter range proof and the receiver range proof to the transaction for a plurality of blockchain nodes maintaining the blockchain to:

verify that the to-be-remitted amount corresponding to the each remitter is not less than zero,

verify based on the remitter range proof that the each remitter's balance is not less than the to-be-remitted amount corresponding to the each remitter, and

verify based on the receiver range proof that the to-be-received amount corresponding to the each receiver is not less than zero.

9. The method of any preceding claim, wherein:

the commitment of the each to-be-remitted amount corresponding to the each remitter comprises a homomorphic encryption of the to-be-remitted amount based on a third random number corresponding to the remitter; and

the commitment of the each to-be-received amount corresponding to the each receiver comprises a homomorphic encryption of the to-be-received amount based on a fourth random number corresponding to the receiver.

10. The method of claim 9, further comprising:

generating a verification random number by summing the third random number corresponding to the each remitter and subtracting the fourth random number corresponding to the each receiver; and

adding the verification random number into the transaction for a plurality of blockchain nodes maintaining the blockchain to verify that an input of the transaction is equal to an output of the transaction.

11. A system for implementing blockchain-based transactions, comprising:

one or more processors; and

one or more computer-readable memories coupled to the one or more processors and having instructions stored thereon that are executable by the one or more processors to perform the method of any of claims 1 to 10.

12. An apparatus for implementing blockchain-based transactions, comprising a plurality of modules for performing the method of any of claims 1 to 10.

13. A non-transitory computer-readable storage medium configured with instructions executable by one or more processors to cause the one or more processors to perform the method of any of claims 1 to 10.