

CONSENSUS METHOD AND APPARATUS FOR BLOCKCHAIN SYSTEM

[0001] This application claims priority to Chinese patent application No. 202410844017.9, titled “CONSENSUS METHOD AND APPARATUS FOR
5 BLOCKCHAIN SYSTEM”, and filed to China National Intellectual Property Administration on June 27, 2024, the entire content of which is incorporated herein by reference.

TECHNICAL FIELD

[0002] The present invention relates to the technical field of block chains, and in particular, to a consensus method and an apparatus for a blockchain system.

10 BACKGROUND

[0003] In a blockchain, a consensus mechanism is used to ensure that all nodes in a network agree on a current state of the network and authenticity of a transaction, which is critical for maintaining security and integrity of the blockchain. A validation node participating in consensus needs to validate blocks and perform signature voting on the blocks.
15 According to a principle of byzantine fault tolerance, blocks with a quorum of $2f+1$ signature votes can pass validation, and can be correctly submitted and recorded in the blockchain. The quorum of $2f+1$ signatures is proportional to the total quantity of $3f+1$ validation nodes in a system. When the number of validation nodes in the system increases drastically, the number of signatures required for each block also increases drastically. At this point, the overhead of
20 transmitting and validating signature messages becomes extremely high and difficult to control. Therefore, how to reduce cost of transmitting and validating signature messages in the consensus process has become a technical problem that urgently needs to be solved.

SUMMARY

Technical Problem

[0004] A consensus method and apparatus for a blockchain system are provided in embodiments of the present invention, which can solve the problem of high cost of transmitting and validating signature messages in the consensus process.

Technical Solution

[0005] In a first aspect, a consensus method for a blockchain system is provided in an embodiment of the present invention. The blockchain system includes a plurality of validation nodes. The consensus method is performed by a validation node, and the consensus method includes:

[0006] generating a subsequent proposal block by referencing a plurality of proposal blocks and combining current transactions from clients in an $(n+x_2)$ -th consensus round, in which the plurality of proposal blocks are correspondingly generated by the plurality of validation nodes referencing a preceding proposal block in an n -th consensus round, and the preceding proposal block is generated in an $(n-x_1)$ -th consensus round; and

[0007] achieving consensus on the preceding proposal block by using the subsequent proposal block as a certificate for the preceding proposal block.

[0008] In a possible implementation, x_1 and x_2 are identical or different, and x_1 and x_2 are each equal to 1 or each not equal to 1.

[0009] In a possible implementation, the corresponding generation of the plurality of proposal blocks by the plurality of validation nodes referencing the preceding proposal block includes:

[0010] correspondingly generating the plurality of proposal blocks by the validation node, together with others of the plurality of validation nodes, referencing the preceding proposal block.

[0011] In a possible implementation, a step of referencing any proposal block includes:
5 referencing a hash index of the any proposal block. The any proposal block is one of the plurality of proposal blocks or the preceding proposal block.

[0012] In a possible implementation, the preceding proposal block is generated by the validation node, or the preceding proposal block is generated by other validation nodes.

[0013] In a possible implementation, using the subsequent proposal block as the
10 certificate for the preceding proposal block includes:

[0014] using the subsequent proposal block together with subsequent proposal blocks generated by at least one other validation node independently as certificates for the preceding proposal block, in which when the quantity of the certificates is greater than a preset threshold, the preceding proposal block is validated.

15 [0015] In a possible implementation, the quantity of the plurality of proposal blocks is not less than $2f+1$, f is the quantity of dishonest validation nodes in the n -th consensus round, and $3f+1$ is the total quantity of validation nodes in the n -th consensus round.

[0016] In a second aspect, a consensus apparatus for a blockchain system is provided in an embodiment of the present invention, including a subsequent proposal block generating
20 module and a consensus module.

[0017] The subsequent proposal block generating module is configured to generate a subsequent proposal block by referencing a plurality of proposal blocks and combining

current transactions from clients in an $(n+x_2)$ -th consensus round. The plurality of proposal blocks are correspondingly generated by the plurality of validation nodes referencing a preceding proposal block in an n -th consensus round, and the preceding proposal block is generated in an $(n-x_1)$ -th consensus round.

5 [0018] The consensus module is configured to achieve consensus on the preceding proposal block by using the subsequent proposal block as a certificate for the preceding proposal block.

[0019] In a possible implementation, x_1 and x_2 are identical or different, and x_1 and x_2 are each equal to 1 or each not equal to 1.

10 [0020] In a possible implementation, the validation node is configured to correspondingly generate the plurality of proposal blocks by referring to the preceding proposal block together with other validation nodes.

[0021] In a possible implementation, the subsequent proposal block generating module is specifically configured to reference a hash index of any proposal block. The any proposal
15 block is one of the plurality of proposal blocks or the preceding proposal block.

[0022] In a possible implementation, the preceding proposal block is generated by the validation node, or the preceding proposal block is generated by other validation nodes.

[0023] In a possible implementation, the consensus module is specifically configured to use the subsequent proposal block together with subsequent proposal blocks generated by at
20 least one other validation node independently as certificates for the preceding proposal block. When the quantity of the certificates is greater than a preset threshold, the preceding proposal block is validated.

[0024] In a possible implementation, the quantity of the plurality of proposal blocks is not less than $2f+1$, f is the quantity of dishonest validation nodes in the n -th consensus round, and $3f+1$ is the total quantity of validation nodes in the n -th consensus round.

[0025] In a third aspect, a terminal device is provided in an embodiment of the present invention, including a memory, a processor, and a computer program stored in the memory and executable on the processor. When executing the computer program, the processor implements the method described above.

[0026] In a fourth aspect, a computer-readable storage medium is provided in an embodiment of the present invention, on which a computer program is stored. The computer program is executed by a processor to implement the method described above.

Beneficial Effects

[0027] A consensus method, a consensus apparatus, a terminal device, and a storage medium are provided in the present invention. Signature messages in conventional consensus methods are replaced by reference relationships, where a validation node referencing a certain proposal block indicates that the validation node has validated the proposal block. Therefore, the consensus method of the present invention eliminates the need to transmit a large amount of signature messages in the consensus process. Instead, consensus on a specific block is achieved by establishing reference relationships between blocks generated in different consensus rounds. In this way, the overhead of transmitting and validating signature messages in the consensus process is reduced.

BRIEF DESCRIPTION OF THE DRAWINGS

[0028] To more clearly illustrate the technical solutions in the embodiments of the present

invention, the accompanying drawings used in the embodiments or the description of the related art will be briefly introduced below. It is apparent that the drawings in the following description are only some embodiments of the present invention. For those skilled in the art, other drawings may also be derived from these drawings without creative efforts.

5 [0029] FIG. 1 is a schematic flowchart of a consensus method in an embodiment of the present invention.

[0030] FIG. 2 is a schematic diagram of a first embodiment of the present invention.

[0031] FIG. 3 is a schematic diagram of a second embodiment of the present invention.

[0032] FIG. 4 is a schematic diagram of a third embodiment of the present invention.

10 [0033] FIG. 5 is a schematic diagram of reference relationships in an embodiment of the present invention.

[0034] FIG. 6 is a schematic diagram of a fourth embodiment of the present invention.

[0035] FIG. 7 is a schematic diagram of a fifth embodiment of the present invention.

[0036] FIG. 8 is a schematic diagram of a sixth embodiment of the present invention.

15 [0037] FIG. 9 is a schematic diagram of a seventh embodiment of the present invention.

[0038] FIG. 10 is a schematic diagram of an eighth embodiment of the present invention.

[0039] FIG. 11 is a schematic diagram of a ninth embodiment of the present invention.

[0040] FIG. 12 is a schematic diagram of a tenth embodiment of the present invention.

[0041] FIG. 13 is a schematic diagram of an eleventh embodiment of the present invention.

20 [0042] FIG. 14 is a schematic structural diagram of a consensus apparatus in an embodiment of the present invention.

[0043] FIG. 15 is a schematic structural diagram of a terminal device in an embodiment

of the present invention.

DETAILED DESCRIPTION

[0044] In the following description, specific details such as specific system structures and technologies are set forth for illustrative purposes rather than limitation, so as to provide a thorough understanding of the embodiments of the present invention. However, it should be clear to those skilled in the art that the present invention may be implemented in other embodiments without these specific details. In other instances, detailed descriptions of well-known systems, apparatus, circuits, and methods are omitted to avoid unnecessary details from obscuring the description of the present invention.

[0045] It should be understood that when used in the description and appended claims of the present invention, the term “comprise” or “include” indicates the presence of described features, integers, steps, operations, elements, and/or components, but does not exclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or combinations thereof.

[0046] It should also be understood that the term “and/or” as used in the description and the appended claims of the present invention refers to any combination of one or more of the associated listed items and all possible combinations, and includes such combinations.

[0047] As used in the description and the appended claims of the present invention, the term “if” may be interpreted contextually as “when”, “once”, “in response to determining”, or “in response to detecting”. Similarly, the phrase “if it is determined” or “if [a described condition or event] is detected” may be interpreted contextually to mean “once it is determined”, “in response to determining”, “once [the described condition or event] is

detected”, or “in response to detecting [the described condition or event]”.

[0048] In addition, in the explanation of the description and the appended claims of the present invention, the terms “first”, “second”, “third”, and the like are merely used to distinguish explanations, and should not be construed as indicating or implying relative importance.

[0049] In the description of the present invention, references to “one embodiment” or “some embodiments” and the like in the description of the present invention mean that specific features, structures, or characteristic described in connection with the embodiment are included in one or more embodiments of the present invention. Thus, appearance of the phrases “in one embodiment”, “in some embodiments”, “in some other embodiments”, “in yet other embodiments”, and the like in various places throughout this description are not necessarily all referring to the same embodiment, but rather mean “one or more but not all embodiments”, unless otherwise specifically emphasized. The terms “include”, “comprise”, “having”, and variations thereof are mean “include but are not limited to”, unless otherwise specifically emphasized.

[0050] In a blockchain system, a consensus protocol must be employed to ensure consistency among distributed nodes in the system. Each validation node (also referred to as validator) in the consensus protocol is also referred to as a consensus node, and is required to validate blocks and perform signature voting on the blocks. According to a principle of byzantine fault tolerance, blocks with quorum of $2f+1$ signature votes can pass validation, and can be correctly submitted and recorded in the blockchain. The quorum of $2f+1$ signatures is proportional to the total quantity of $3f+1$ validation nodes in the system. When the number

of the validation nodes in the system increases, the number of signatures required for each block also increases drastically. At this point, the overhead of transmitting and validating signature messages becomes extremely high and difficult to control. Therefore, how to reduce cost of transmitting and validating of signature messages in the consensus process has become
5 a technical problem that urgently needs to be solved.

[0051] A consensus method, a consensus apparatus, a terminal device, and a storage medium are provided in the present invention. Signature messages in conventional consensus methods are replaced by reference relationships, where a validation node referencing a certain proposal block indicates that the validation node has validated the proposal block. Therefore,
10 in the consensus method of the present invention, the need to transmit a large amount of signature messages in the consensus process is eliminated, and it is determined whether a proposal block has been validated through a sufficient number of validation nodes by examining these reference relationships. In this way, the overhead of transmitting and validating signature messages in the consensus process is reduced.

15 [0052] Specific embodiments are provided below to illustrate the technical solutions of the present invention.

[0053] Referring to a flowchart of an embodiment of a consensus method for a blockchain system shown in FIG. 1, which is provided by way of example and is not limited thereto. The consensus method includes the following steps:

20 [0054] Step 100, generating a subsequent proposal block by referencing a plurality of proposal blocks and combining current transactions from clients in an $(n+x_2)$ -th consensus round. The plurality of proposal blocks are correspondingly generated by the plurality of

validation nodes referencing a preceding proposal block in an n-th consensus round, and the preceding proposal block is generated in an (n-x₁)-th consensus round.

[0055] Step 200, achieving consensus on the preceding proposal block by using the subsequent proposal block as a certificate for the preceding proposal block.

5 [0056] It should be noted that the blockchain system of the present invention includes a plurality of validation nodes, and the consensus method in the embodiments of the present invention is performed by a validation node. Validation nodes, which are specifically authenticated and authorized nodes in a blockchain network, have permission to participate in consensus processes, validate transactions, and package blocks. Specifically, tasks under
10 the responsibility of the validation nodes include executing a consensus algorithm. As described in the above solution, the validation nodes reference proposal blocks in specific consensus rounds, generate the subsequent proposal block, and participate in consensus on the preceding proposal block. The validation nodes further validate transactions, to ensure legality, validity, and compliance with protocol rules of the blockchain for the transactions.
15 In addition, based on results of the consensus algorithm, the validation nodes package legitimate transactions to generate new blocks and attempt to add the blocks to the blockchain.

[0057] It may be understood that a threshold is usually set in the blockchain system, and a specific quantity of validation nodes are required to validate a same proposal block, so that the proposal block can be added to the blockchain. This process ensures that most validation
20 nodes in the network have reached consensus on new block content. In a conventional consensus process, if a validation node validates a proposal block, the proposal block needs to be signed and broadcast to other validation nodes in the system, and the other validation

nodes need to validate the signature, resulting in the transmission and verification of a large amount of signature messages in the consensus process.

[0058] In the consensus method according to the present invention, when a validation node references a proposal block generated in a previous consensus round, it indicates that the validation node has validated the proposal block. That is, signature messages in conventional consensus methods are replaced by reference relationships in the present invention. Therefore, the transmission of the large amount of signature messages is no longer required in the consensus process. Instead, it is determined whether a proposal block is validated through a sufficient number of validation nodes by examining the reference relationships.

[0059] It should be noted that, in the present invention, the establishment of a reference relationship is based on successful validation. That is, a block needs first to be validated to ensure that the content of the proposal block complies with rules of the blockchain and security requirements of the network. When the validation is passed, a reference relationship with the block can be established.

[0060] Exemplarily, a validation node may validate the correctness of a proposal block by examining signature, block data content, block hash, and block timestamp of the proposal block. For example, the validation node may examine whether transaction data in the block is valid and legitimate, including examining whether a field of the transaction is complete, whether a format of the transaction complies with the protocol rules of the blockchain, and the like. For each transaction, the validation node may also need to examine whether input and output of the transaction are balanced. For another example, the validation node may

further calculate a hash value of the proposal block, and compare the hash value with a hash value stored in a block header, to ensure that the block is not tampered with during transmission. Alternatively, the validation node may examine whether a timestamp in the proposal block is reasonable, and the timestamp is configured to record a time at which the block is created, and may help the validation node detect potential double-spending or duplicate transactions.

[0061] Therefore, it may also be understood that referencing a proposal block indicates approval of the proposal block. In a subsequent consensus process, it may be determined whether a proposal block is referenced by a sufficient number of validation nodes via examining reference relationships between proposal blocks. When the quantity of validation nodes referencing the proposal block meets certain requirements, it may be considered that the proposal block has obtained sufficient approval, meaning the proposal block is trustworthy.

[0062] In a conventional consensus method, each proposal block needs to carry at least $2f+1$ signatures, and the quantity of signatures required by each block increases drastically when the quantity of validation nodes in the system increases drastically. In contrast, the consensus method provided in present invention uses reference relationships to indicate the approval of a certain proposal block, rather than relying on signatures to achieve consensus on the proposal block. Consequently, the transmission of a large amount of signature messages is no longer required in the consensus process. Thus, the overhead of transmitting and validating the signature messages in the consensus process is reduced.

[0063] It should be noted that at step 100, when generating the subsequent proposal block, the validation node may further need to integrate the current transactions from the clients.

Transactions may refer to a series of operations or requests initiated by users (i.e., clients) on a blockchain network, which alter a state or data on the blockchain. These transactions may include invocation of smart contracts, writing or updating of data, and the like.

[0064] It should be noted that the implementation process of referencing the proposal
5 blocks is based on a directed acyclic graph (DAG). This means the blockchain system is not only a simple linear structure, but uses a more complex graph structure to organize and reference the proposal blocks.

[0065] For better understanding of the present invention, a brief introduction to the DAG is provided first. The DAG is a graph that includes directional edges and no cycles. In a
10 blockchain system. The DAG may be used to represent reference relationships between proposal blocks, and each proposal block may point to one or more preceding proposal blocks to form a complex network structure.

[0066] Specifically, the blockchain system may maintain a pre-stored DAG that records all proposal blocks and reference relationships among the proposal blocks. When a validation
15 node intends to reference a preceding proposal block, the validation node may query the pre-stored DAG to find a suitable block to reference. The basis for referencing may include age of the block (i.e., generation time), the quantity of involved transactions, quality of the block, or other customized strategies.

[0067] When constructing a new proposal block, the validation node may select one or
20 more preceding proposal blocks to reference based on current transaction data and reference relationships in the DAG. When a new proposal block is successfully added to the DAG and confirmed by other validation nodes in the network, the structure of the DAG may be updated

accordingly to reflect the new reference relationships and addition of the block.

[0068] The consensus method that references proposal blocks via the DAG may enable the blockchain system to be more flexible and efficient when processing a large amount of transactions and rapidly generating blocks. The validation nodes may achieve consensus by processing multiple proposal blocks in parallel and finding suitable paths in the DAG, thereby enhancing throughput and performance of the entire system.

[0069] In a possible implementation, x_1 and x_2 may be identical or different, and x_1 and x_2 may be each equal to 1 or each not equal to 1.

[0070] It should be noted that x_1 may be 1 or a positive integer greater than 1. When x_1 is 1, it indicates that the round in which a proposal block is generated is consecutive to the round in which a preceding proposal block is generated. When x_1 is not 1, it indicates that the round in which a proposal block is generated is not consecutive to the round in which a preceding proposal block is generated. Similarly, x_2 may also be 1 or a positive integer greater than 1. When x_2 is equal to 1, it indicates that the round in which a proposal block is generated is consecutive to the round in which a subsequent proposal block is generated. When x_2 is not equal to 1, it indicates that the round in which a proposal block is generated is not consecutive to the round in which a subsequent proposal block is generated.

[0071] The above content will be explained below with specific embodiments.

[0072] A first embodiment: referring to FIG. 2, in the present embodiment, $x_1 = 1$ and $x_2 = 1$. That is, in an $(n+1)$ -th consensus round, proposal blocks A_n , B_n , and C_n generated in an n -th consensus round may be referenced to generate a subsequent proposal block B_{n+1} . Each of the three proposal blocks A_n , B_n , and C_n may reference a preceding proposal block B_{n-1}

generated in an $(n-1)$ -th consensus round. Therefore, the subsequent proposal block B_{n+1} may be used as a certificate of the preceding proposal block B_{n-1} for achieving consensus on the preceding proposal block B_{n-1} .

[0073] A second embodiment: referring to FIG. 3, in the present embodiment, $x_1 = 1$ and
5 $x_2 \neq 1$. Specifically, taking x_2 equal to 2 as an example, this means in an $(n+2)$ -th consensus round, proposal blocks A_n , B_n , and C_n generated in an n -th consensus round may be referenced in an indirect referencing manner to generate a subsequent proposal block B_{n+2} . Each of the three proposal blocks A_n , B_n , and C_n may reference a preceding proposal block B_{n-1} generated in an $(n-1)$ -th consensus round. Therefore, the subsequent proposal block B_{n+2} may be used
10 as a certificate of the preceding proposal block B_{n-1} for achieving consensus on the preceding proposal block B_{n-1} .

[0074] It should be noted that since blocks generated in the $(n+1)$ -th consensus round have reference relationships with the proposal blocks generated in the n -th consensus round, the proposal blocks generated in the n -th consensus round may be referenced by the
15 subsequent proposal block B_{n+2} in an indirect referencing manner by referencing the blocks generated in the $(n+1)$ -th consensus round.

[0075] Exemplarily, since a block A_{n+1} has a reference relationship with a proposal block A_n , the proposal block A_n may be referenced by the subsequent proposal block B_{n+2} by referencing the block A_{n+1} . Similarly, since a block C_{n+1} has a reference relationship with a
20 proposal block B_n , the proposal block B_n may be referenced by a subsequent proposal block B_{n+2} by referencing the block C_{n+1} . Similarly, since a block D_{n+1} has a reference relationship with a proposal block C_n , the proposal block C_n may be referenced by the subsequent proposal

block B_{n+2} by referencing the block D_{n+1} .

[0076] A third embodiment: referring to FIG. 4, in the present embodiment, $x_1 \neq 1$ and $x_2 = 1$. Specifically, taking x_1 equal to 2 as an example, this means in an $(n+1)$ -th consensus round, proposal blocks A_n , B_n , and C_n generated in an n -th consensus round may be referenced to generate a subsequent proposal block B_{n+1} . Each of the three proposal blocks A_n , B_n , and C_n may reference a preceding proposal block B_{n-2} generated in an $(n-2)$ -th consensus round in an indirect referencing manner. Therefore, the subsequent proposal block B_{n+1} may be used as a certificate of the preceding proposal block B_{n-2} for achieving consensus on the preceding proposal block B_{n-2} .

[0077] It should be noted that since blocks generated in the $(n-1)$ -th consensus round have reference relationships with preceding proposal blocks generated in the $(n-2)$ -th consensus round, the preceding proposal blocks generated in the $(n-2)$ -th consensus round may be referenced in an indirect referencing manner by a proposal block by referencing the blocks generated in the $(n-1)$ -th consensus round.

[0078] Exemplarily, since a block A_{n-1} has a reference relationship with a proposal block B_{n-2} , a proposal block A_n may reference the preceding proposal block B_{n-2} by referencing the block A_{n-1} . Similarly, since a block B_{n-1} has a reference relationship with a preceding proposal block B_{n-2} , a proposal block C_n may reference the preceding proposal block B_{n-2} by referencing the block B_{n-1} . Similarly, since a block C_{n-1} has a reference relationship with a preceding proposal block B_{n-2} , a proposal block D_n may reference the preceding proposal block B_{n-2} by referencing the block C_{n-1} .

[0079] It should be noted that, as demonstrated by the above embodiments, referencing

may include direct referencing and indirect referencing. The direct referencing means referencing blocks generated in the immediately preceding round. Exemplarily, referring to FIG. 3, a block A_n references a block B_{n-1} , a block B_{n+2} references a block C_{n+1} , or a block A_{n+1} references a block A_n . The indirect referencing means referencing proposal blocks generated in a non-consecutive round. Exemplarily, referring to FIG. 3, a block B_{n+2} references a block A_n , which is achieved by the block B_{n+2} referencing a block A_n through a block A_{n+1} in an indirect referencing manner.

[0080] In a possible implementation, the corresponding generation of the plurality of proposal blocks by the plurality of validation nodes referencing the preceding proposal block may include:

[0081] correspondingly generating the plurality of proposal blocks by the validation node, together with others of the plurality of validation nodes, referencing the preceding proposal block.

[0082] It should be noted that in the n -th consensus round, the process of referencing a preceding proposal block to generate a proposal block is involved. An entity executing this process may or may not include the validation node.

[0083] This will be explained by using the first embodiment and the third embodiment described above. Referring to the schematic diagram of the first embodiment shown in FIG. 2, the validation node represented as Validator B participates in the generation process of the proposal block in the n -th consensus round. That is, the Validator B references the preceding proposal block B_{n-1} to generate the proposal block B_n in the n -th consensus round. However, in the third embodiment shown in FIG. 4, the validation node represented as Validator B does

not participate in the generation process of the proposal block in the n-th consensus round.

That is, the Validator B does not reference a preceding proposal block and fail to generate the proposal block B_n in the n-th consensus round.

[0084] Therefore, in embodiments of the present invention, the entity executing the
5 process of referencing a preceding proposal block to generate a proposal block may include the validation node, or may not include the validation node.

[0085] In a possible implementation, a step of referencing any proposal block may include referencing a hash index of the any proposal block. The any proposal block may be one of the plurality of proposal blocks or the preceding proposal block.

10 [0086] It should be noted that the proposal block may include a transaction list, a current consensus round, a proposal signer, a proposal signature, and reference relationships with proposal blocks generated in previous rounds. Referring to FIG. 5, each proposal block may include a transaction list (represented as transactions), a current consensus round (represented as round), a proposal signer (represented as author), a proposal signature (represented as
15 signature), and reference relationships (represented as parents) to previous-round proposal blocks.

[0087] In a possible implementation, the hash index may be generated based on the current consensus round, the proposal signer, the transaction list, and the reference relationships. Referring to FIG. 5, in the illustrated case, hash indices may include h1, h2, and
20 h3. Here, h1 is calculated by a hash function according to the current consensus round r, the proposal signer v1, the transaction list txs1, and the reference relationship p1; h2 is calculated by a hash function according to the current consensus round r, the proposal signer v2, the

transaction list txs2, and the reference relationship p2; and h3 is calculated by a hash function according to the current consensus round r, the proposal signer v3, the transaction list txs3, and the reference relationship p3.

[0088] It should be noted that referencing is based on the proposal block having been validated. The validation may include validation of correspondence between the signer and the signature of the proposal block, as well as validation of correctness of the reference relationship.

[0089] Specifically, it is necessary to validate the correspondence between the signer and the signature of the proposal block. According to the present invention, consensus is achieved through a plurality of consensus rounds. Each honest validation node may sign unique proposal block thereof in each consensus round and then broadcast the signed proposal block to other validation nodes, to prove that the block is indeed generated by the honest validation node. The validation node may examine whether the signature matches the signer and whether the signature is valid (not tampered with or counterfeit), thus preventing malicious nodes from counterfeiting blocks or tampering with transactions.

[0090] In addition, it is necessary to validate the correctness of the reference relationship. Since each proposal block references several (not less than a quorum of) proposal blocks, a validation node needs to examine whether a hash value of a referenced proposal block is correct to ensure integrity and continuity of the blockchain.

[0091] Furthermore, when generating the subsequent proposal block, current transactions from clients may also need to be incorporated, and these transactions may need to be validated as well. Specifically, a transaction list of the current transactions may be validated, including

validating validity of each transaction, identities of a sender and a receiver, whether a transaction amount is correct, and the like, to ensure that legitimate and valid transactions can be added to the blockchain.

[0092] In a possible implementation, the preceding proposal block may be generated by the validation node, or the preceding proposal block may be generated by other validation nodes.

[0093] It should be noted that, in the first embodiment, the second embodiment, and the third embodiment, each of the preceding proposal block and the subsequent proposal block may be generated by the same validator. However, in some possible implementations, corresponding to the above three cases, the preceding proposal block and the subsequent proposal block may also be generated by different validators. This will be explained below with specific embodiments.

[0094] A fourth embodiment: referring to FIG. 6, in the present embodiment, $x_1 = 1$, $x_2 = 1$, and the preceding proposal block and the subsequent proposal block are generated by different validation nodes.

[0095] It should be noted that the present embodiment differs from the first embodiment in that the preceding proposal block and the subsequent proposal block are generated by different validation nodes. In the first embodiment, each of the preceding proposal block B_{n-1} and the subsequent proposal block B_{n+1} may be generated by the validation node represented as Validator B. However, in the fourth embodiment, the preceding proposal block D_{n-1} may be generated by the validation node represented as Validator D, and the subsequent proposal block B_{n+1} may be generated by the validation node represented as Validator B.

[0096] Specifically, in an $(n+1)$ -th consensus round, a subsequent proposal block B_{n+1} may be generated by referencing proposal blocks A_n , B_n , and C_n generated in an n -th consensus round. Since each of the three proposal blocks A_n , B_n , and C_n references the preceding proposal block D_{n-1} generated in the $(n-1)$ -th consensus round, the subsequent
5 proposal block B_{n+1} may be used as a certificate of the preceding proposal block D_{n-1} for achieving consensus on the preceding proposal block D_{n-1} .

[0097] A fifth embodiment: referring to FIG. 7, in the present embodiment, $x_1 = 1$, $x_2 \neq 1$, and the preceding proposal block and the subsequent proposal block are generated by different validation nodes.

10 [0098] It should be noted that the present embodiment differs from the second embodiment in that the preceding proposal block and the subsequent proposal block are generated by different validation nodes. In the second embodiment, each of the preceding proposal block B_{n-1} and the subsequent proposal block B_{n+1} is generated by the validation node represented as Validator B. However, in the fifth embodiment, the preceding proposal
15 block C_{n-1} is generated by the validation node represented as Validator C, and the subsequent proposal block B_{n+2} is generated by the validation node represented as Validator B.

[0099] Specifically, the subsequent proposal block B_{n+2} may be generated in an $(n+2)$ -th consensus round by referencing the proposal blocks A_n , B_n , and C_n generated in the n -th consensus round in an indirect referencing manner. Since each of the three proposal blocks
20 A_n , B_n , and C_n references the preceding proposal block C_{n-1} generated in the $(n-1)$ -th consensus round, the subsequent proposal block B_{n+2} may be used as a certificate of the preceding proposal block C_{n-1} for achieving consensus on the preceding proposal block C_{n-1} .

[0100] A sixth embodiment: referring to FIG. 8, in the present embodiment, $x_1 \neq 1$, $x_2 = 1$, and the preceding proposal block and the subsequent proposal block are generated by different validation nodes.

[0101] It should be noted that the present embodiment differs from the third embodiment
5 in that the preceding proposal block and the subsequent proposal block are generated by different validation nodes. In the third embodiment, each of the preceding proposal block B_{n-2} and the subsequent proposal block B_{n+1} is generated by the validation node represented as Validator B. However, in the sixth embodiment, the preceding proposal block D_{n-2} is generated by the validation node represented as Validator D, and the subsequent proposal block B_{n+1} is
10 generated by the validation node represented as Validator B.

[0102] Specifically, the subsequent proposal block B_{n+1} may be generated in an $(n+1)$ -th consensus round by referencing the proposal blocks A_n , B_n , and C_n generated in the n -th consensus round. Since each of the three proposal blocks A_n , B_n , and C_n references the preceding proposal block D_{n-2} generated in the $(n-2)$ -th consensus round in an indirect
15 referencing manner, the subsequent proposal block B_{n+1} may be used as a certificate of the preceding proposal block D_{n-2} for achieving consensus on the preceding proposal block D_{n-2} .

[0103] In a possible implementation, using the subsequent proposal block as the certificate for the preceding proposal block may include: using the subsequent proposal block together with subsequent proposal blocks generated by at least one other validation node
20 independently as certificates for the preceding proposal block. When the quantity of the certificates is greater than a preset threshold, the preceding proposal block may be validated.

[0104] It should be noted that a certificate represents confirmation and acceptance of a

to-be-submitted proposal block by a validation node. Each validation node may generate certificates for one or more to-be-submitted proposal blocks. These certificates accumulate to form a quantitative indicator of extent to which one proposal block is accepted. In order to ensure the stability and security of the blockchain, when the quantity of certificates for the proposal block to be submitted reaches or exceeds the preset threshold, the proposal block is considered to have sufficient support and thus submitted to the target blockchain.

[0105] It should be further noted that the preset threshold for the quantity of certificates may be adjusted according to a specific circumstance of the blockchain network. For example, when the blockchain network has a relatively small scale or the quantity of validation nodes is relatively small, the preset threshold for the quantity of certificates may be appropriately decreased. Conversely, when the blockchain network has a relatively large scale or the quantity of validation nodes is relatively large, the preset threshold for the quantity of certificates may be appropriately increased.

[0106] In the present embodiment of the present invention, for a same preceding proposal block, a plurality of subsequent proposal blocks may be used as certificates for the preceding proposal block. The plurality of certificates may be generated by different validation nodes in a same consensus round, or may be generated by a same validation node in different rounds. This will be explained below through specific embodiments.

[0107] A seventh embodiment: referring to FIG. 9, in the present embodiment, a preceding proposal block B_{n-1} generated by Validator B in an $(n-1)$ -th consensus round has two certificates, which may be respectively generated by Validator B and Validator D in an $(n+1)$ -th consensus round.

[0108] Specifically, in the $(n+1)$ -th consensus round, the validation node Validator B may generate a subsequent proposal block B_{n+1} by referencing proposal blocks A_n , B_n , and C_n generated in an n -th consensus round. Since each of the three proposal blocks A_n , B_n , and C_n references a preceding proposal block B_{n-1} generated in the $(n-1)$ -th consensus round, the subsequent proposal block B_{n+1} may be used as a certificate of the preceding proposal block B_{n-1} for achieving consensus on the preceding proposal block B_{n-1} . Meanwhile, in the $(n+1)$ -th consensus round, the validation node Validator D may also generate a subsequent proposal block D_{n+1} by referencing the proposal blocks A_n , B_n , and C_n generated in the n -th consensus round. Since each of the three proposal blocks A_n , B_n , and C_n references the preceding proposal block B_{n-1} generated in the $(n-1)$ -th consensus round, the subsequent proposal block D_{n+1} may be used as a certificate of the preceding proposal block B_{n-1} for achieving consensus on the preceding proposal block B_{n-1} .

[0109] It should be noted that, in the seventh embodiment, both the subsequent proposal block D_{n+1} generated by the Validator D and the subsequent proposal block B_{n+1} generated by the Validator B may be used as certificates of the preceding proposal block B_{n-1} for achieving consensus on the preceding proposal block B_{n-1} .

[0110] In the above embodiment, a single preceding proposal block may have multiple subsequent proposal blocks used as certificates thereof, and each of the multiple subsequent proposal blocks used as certificates is generated in the same consensus round. It should be noted that, in some possible embodiments, the multiple subsequent proposal blocks used as certificates may be generated across different consensus rounds, and this case will be explained below by an eighth embodiment.

[0111] The eighth embodiment: referring to FIG. 10, in the present embodiment, the validation node Validator B may generate a subsequent proposal block B_{n+1} in the $(n+1)$ -th consensus round by referencing proposal blocks A_n , B_n , and C_n generated in an n -th consensus round. Since each of the three proposal blocks A_n , B_n , and C_n references a preceding proposal block C_{n-1} generated in an $(n-1)$ -th consensus round, the subsequent proposal block B_{n+1} may be used as a certificate of the preceding proposal block C_{n-1} for achieving consensus on the preceding proposal block C_{n-1} .

[0112] Meanwhile, the validation node Validator B may generate a subsequent proposal block B_{n+2} in the $(n+2)$ -th consensus round by referencing the proposal blocks A_n , B_n , and C_n generated in the n -th consensus round in an indirect reference manner. Since each of the three proposal blocks A_n , B_n , and C_n references the preceding proposal block C_{n-1} generated in the $(n-1)$ -th consensus round, the subsequent proposal block B_{n+2} may also be used as a certificate of the preceding proposal block C_{n-1} for achieving consensus on the preceding proposal block C_{n-1} .

[0113] Therefore, in the present embodiment, each of the subsequent proposal block B_{n+1} and the subsequent proposal block B_{n+2} may be used as a certificate for the preceding proposal block C_{n-1} , and the subsequent proposal block B_{n+1} and the subsequent proposal block B_{n+2} are generated in different consensus rounds.

[0114] In the above eighth embodiment, the two subsequent proposal blocks used as certificates are generated in different rounds, but both are generated by the same validation node represented as Validator B. It should be noted that in some possible embodiments, multiple subsequent proposal blocks used as certificates may also be generated by different

validation nodes in different consensus rounds. This will be explained below by a ninth embodiment.

[0115] Referring to FIG. 11, in the ninth embodiment, the validation node represented as Validator D may generate a subsequent proposal block D_{n+1} in an $(n+1)$ -th consensus round by referencing proposal blocks A_n , B_n , and C_n generated in an n -th consensus round. Since each of the three proposal blocks A_n , B_n , and C_n references a preceding proposal block C_{n-1} generated in an $(n-1)$ -th consensus round, the subsequent proposal block D_{n+1} may be used as a certificate of the preceding proposal block C_{n-1} for achieving consensus on the preceding proposal block C_{n-1} .

[0116] Meanwhile, in the $(n+2)$ -th consensus round, the validation node represented as Validator B may generate a subsequent proposal block B_{n+2} in the n -th consensus round by referencing the proposal blocks A_n , B_n , and C_n generated in an indirect referencing manner. Since each of the three proposal blocks A_n , B_n , and C_n references the preceding proposal block C_{n-1} generated in the $(n-1)$ -th consensus round, the subsequent proposal block B_{n+2} may also be used as a certificate of the preceding proposal block C_{n-1} for achieving consensus on the preceding proposal block C_{n-1} .

[0117] Therefore, in the present embodiment, each of the subsequent proposal block D_{n+1} and the subsequent proposal block B_{n+2} may be used as a certificate for the preceding proposal block C_{n-1} , and the subsequent proposal block D_{n+1} and the subsequent proposal block B_{n+2} may be generated by different validation nodes in different consensus rounds.

[0118] It should also be noted that in some possible embodiments, the same subsequent proposal block may also be used as a certificate for multiple preceding proposal blocks. This

case will be explained by using a tenth embodiment.

[0119] Referring to FIG. 12, in the tenth embodiment, the preceding proposal blocks include B_{n-1} and D_{n-1} . Based on the reference relationships as shown, the subsequent proposal block B_{n+1} may be used as a certificate for either or both of the preceding proposal block B_{n-1} and the preceding proposal block D_{n-1} .

[0120] Specifically, the validation node represented as Validator B may generate the subsequent proposal block B_{n+1} in an $(n+1)$ -th consensus round by referencing proposal blocks A_n , B_n , and C_n generated in an n -th consensus round. Since each of the three proposal blocks A_n , B_n , and C_n references the preceding proposal block D_{n-1} generated in an $(n-1)$ -th consensus round, the subsequent proposal block B_{n+1} may be used as a certificate of the preceding proposal block D_{n-1} for achieving consensus on the preceding proposal block D_{n-1} . In addition, since each of the proposal blocks A_n , B_n , and C_n references the preceding proposal block B_{n-1} generated in the $(n-1)$ -th consensus round, the subsequent proposal block B_{n+1} may also be used as a certificate of the preceding proposal block B_{n-1} for achieving consensus on the preceding proposal block B_{n-1} . Therefore, in the present embodiment, the subsequent proposal block B_{n+1} may be used as a certificate for both the preceding proposal block B_{n-1} and the preceding proposal block D_{n-1} .

[0121] It should be noted that, when the same subsequent proposal block is used as a plurality of preceding proposal blocks, it is still not limited that the preceding proposal block and the subsequent proposal block must be generated by the same validation node. This case will be explained below by using an eleventh embodiment.

[0122] Referring to FIG. 13, the validation node represented as Validator B may generate

a subsequent proposal block B_{n+1} in an $(n+1)$ -th consensus round by referencing proposal blocks A_n , B_n , and C_n generated in an n -th consensus round. Since each of the three proposal blocks A_n , B_n , and C_n references a preceding proposal block D_{n-1} generated in an $(n-1)$ -th consensus round, the subsequent proposal block B_{n+1} may be used as a certificate of the preceding proposal block D_{n-1} for achieving consensus on the preceding proposal block D_{n-1} . In addition, since each of the proposal blocks A_n , B_n , and C_n references a preceding proposal block A_{n-1} generated in the $(n-1)$ -th consensus round, the subsequent proposal block B_{n+1} may also be used as a certificate of the preceding proposal block A_{n-1} for achieving consensus on the preceding proposal block A_{n-1} . Therefore, in the present embodiment, the subsequent proposal block B_{n+1} may be used as a certificate for both the preceding proposal block A_{n-1} and the preceding proposal block D_{n-1} .

[0123] In a possible implementation, the quantity of proposal blocks is not less than $2f + 1$, f is the quantity of dishonest validation nodes in the n -th consensus round, and $3f + 1$ is the total quantity of validation nodes in the n -th consensus round.

[0124] It should be noted that using a subsequent proposal block as a certificate for a preceding proposal block needs to satisfy a certain condition. For example, the quantity of proposal blocks referenced by the subsequent proposal block needs to satisfy a condition of $2f + 1$. This condition may be based on a byzantine fault tolerant algorithm, and the byzantine fault tolerant algorithm is a fault tolerant algorithm oriented to a Byzantine problem, to resolve a problem of how to achieve consensus in a network where communication is reliable but nodes may fail or act maliciously. The core theory is that the total quantity of nodes in the system is greater than or equal to $3f + 1$, and f is the quantity of allowed faulty nodes. In other

words, for a certain proposal block, only a quorum (not less than $2f + 1$) of validation nodes validating the proposal block can ensure the correctness of the proposal block. Therefore, it may be required for a preceding proposal block that the quantity of proposal blocks generated by the preceding proposal block is not less than $2f + 1$ in the embodiments of the present invention.

[0125] In addition, for the examination of the reference relationships, a depth-first search (DFS) manner may be adopted. A certain subsequent proposal block generated in an $(n+x_2)$ -th consensus round may be used as a starting point, and proposal blocks having reference relationships with the subsequent proposal block may be traversed preferentially. In this way, it may be determined whether a specific preceding proposal block is supported by enough validation nodes through the reference relationships.

[0126] The DFS is an algorithm used for traversing or searching a tree/graph. In the embodiments of the present invention, the traversal may be performed based on a directed acyclic graph. Specifically, when a subsequent proposal block references proposal blocks from previous rounds, the referenced blocks may be tracked and validated in the DFS manner by using the subsequent proposal block as a starting point.

[0127] The traversal process will be explained below in combination with the principle of the DFS. A validation node may take a subsequent proposal block generated in an $(n+x_2)$ -th consensus round as a starting point, and then need to select first neighbor nodes. Since the current block references one or more previous blocks, the validation node will first select these referenced blocks as the first neighbor nodes for traversal. Then, for each of the referenced blocks (neighbor nodes), the validation node will examine whether the block in

turn references other blocks. If yes, the validation node may recursively continue to traverse these referenced blocks until there are no more referenced blocks. Subsequently, after traversing all blocks of one branch, the validation node may trace back to a previous node and continue to traverse other untraversed neighbor nodes of that node. The traversal process may
5 conclude when all reachable blocks are traversed by the validation node or when a certain preset termination condition (such as traversal depth limitation and time limitation) is satisfied. Thus, it can be determined whether a certain preceding proposal block is supported by enough validation nodes.

[0128] Referring to FIG. 14, a consensus apparatus 400 for a blockchain system according
10 to an embodiment of the present invention includes a subsequent proposal block generating module 401 and a consensus module 402.

[0129] The subsequent proposal block generating module 401 is configured to generate a subsequent proposal block by referencing a plurality of proposal blocks and combining current transactions from clients in an $(n+x_2)$ -th consensus round. The plurality of proposal
15 blocks are correspondingly generated by the plurality of validation nodes referencing a preceding proposal block in an n -th consensus round, and the preceding proposal block is generated in an $(n-x_1)$ -th consensus round.

[0130] The consensus module 402 is configured to achieve consensus on the preceding proposal block by using the subsequent proposal block as a certificate for the preceding
20 proposal block.

[0131] It should be noted that in the consensus apparatus 400 according to the present invention, signature messages in the conventional consensus method are replaced by reference

relationships. That is, when a subsequent proposal block references a plurality of proposal blocks, and each of the plurality of proposal blocks references a certain preceding proposal block, the subsequent proposal block is used as a certificate for the preceding proposal block, and the consensus process may be completed according to the certificate. Therefore, when
5 using the consensus apparatus 400 according to the present invention to achieve consensus, there is no longer a need to transmit a large amount of signature messages. Instead, it is possible to determine whether a preceding proposal block is validated by a sufficient number of validation nodes through the reference relationships. In this way, the overhead of transmitting and validating signature messages in the consensus process is reduced.

10 [0132] Based on the same inventive concept, in a possible implementation, x_1 and x_2 may be identical or different, and x_1 and x_2 may be each equal to 1 or each not equal to 1.

[0133] It should be noted that x_1 may be 1 or a positive integer greater than 1. When x_1 is 1, it indicates that the round in which a proposal block is generated is consecutive to the round in which a preceding proposal block is generated. When x_1 is not 1, it indicates that the round
15 in which a proposal block is generated is not consecutive to the round in which a preceding proposal block is generated. Similarly, x_2 may also be 1 or a positive integer greater than 1. When x_2 is equal to 1, it indicates that the round in which a proposal block is generated is consecutive to the round in which a subsequent proposal block is generated. When x_2 is not equal to 1, it indicates that the round in which a proposal block is generated is not consecutive
20 to the round in which a subsequent proposal block is generated.

[0134] Based on the same inventive concept, a validation node may be configured to generate a corresponding proposal block by referring to a preceding proposal block together

with other validation nodes.

[0135] It should be noted that, in the n-th consensus round, the process of referencing a preceding proposal block to generate a proposal block is involved. An entity executing this process may or may not include the validation node.

5 [0136] Based on the same inventive concept, the subsequent proposal block generation module may be specifically configured to reference a hash index of the any proposal block. The any proposal block may be one of the plurality of proposal blocks or the preceding proposal block.

[0137] It should be noted that the proposal block may include a transaction list, a current
10 consensus round, a proposal signer, a proposal signature, and reference relationships with proposal blocks generated in previous rounds. Referring to FIG. 5, each proposal block may include a transaction list (represented as transactions), a current consensus round (represented as round), a proposal signer (represented as author), a proposal signature (represented as signature), and reference relationships (represented as parents) to previous-round proposal
15 blocks.

[0138] In a possible implementation, the hash index may be generated based on the current consensus round, the proposal signer, the transaction list, and the reference relationships. Referring to FIG. 5, in the illustrated case, hash indices may include h1, h2, and h3. Here, h1 is calculated by a hash function according to the current consensus round r, the
20 proposal signer v1, the transaction list txs1, and the reference relationship p1; h2 is calculated by a hash function according to the current consensus round r, the proposal signer v2, the transaction list txs2, and the reference relationship p2; and h3 is calculated by a hash function

according to the current consensus round r , the proposal signer v_3 , the transaction list txs_3 , and the reference relationship p_3 .

[0139] Based on the same inventive concept, the preceding proposal block may be generated by the validation node, or the preceding proposal block may be generated by other validation nodes.

[0140] It should be noted that, in the first embodiment, the second embodiment, and the third embodiment, each of the preceding proposal block and the subsequent proposal block may be generated by the same validator. However, in some possible implementations, corresponding to the above three cases, the preceding proposal block and the subsequent proposal block may also be generated by different validators. This will be explained below with specific embodiments.

[0141] Based on the same inventive concept, the consensus module is specifically configured for using the subsequent proposal block together with subsequent proposal blocks generated by at least one other validation node independently as certificates for the preceding proposal block. When the quantity of the certificates may be greater than a preset threshold, the preceding proposal block may be validated.

[0142] It should be noted that a certificate represents confirmation and acceptance of a to-be-submitted proposal block by a validation node. Each validation node may generate certificates for one or more to-be-submitted proposal blocks. These certificates accumulate to form a quantitative indicator of extent to which one proposal block is accepted. In order to ensure the stability and security of the blockchain, when the quantity of certificates for the proposal block to be submitted reaches or exceeds the preset threshold, the proposal block is

considered to have sufficient support and thus submitted to the target blockchain.

[0143] It should be further noted that the preset threshold for the quantity of certificates may be adjusted according to a specific circumstance of the blockchain network. For example, when the blockchain network has a relatively small scale or the quantity of validation nodes is relatively small, the preset threshold for the quantity of certificates may be appropriately decreased. Conversely, when the blockchain network has a relatively large scale or the quantity of validation nodes is relatively large, the preset threshold for the quantity of certificates may be appropriately increased.

[0144] In the present embodiment of the present invention, for a same preceding proposal block, a plurality of subsequent proposal blocks may be used as certificates for the preceding proposal block. The plurality of certificates may be generated by different validation nodes in a same consensus round, or may be generated by a same validation node in different rounds. This will be explained below through specific embodiments.

[0145] Based on the same inventive concept, the quantity of proposal blocks is not less than $2f + 1$, f is the quantity of dishonest validation nodes in the n -th consensus round, and $3f + 1$ is the total quantity of validation nodes in the n -th consensus round.

[0146] It should be noted that using a subsequent proposal block as a certificate for a preceding proposal block needs to satisfy a certain condition. For example, the quantity of proposal blocks referenced by the subsequent proposal block needs to satisfy a condition of $2f + 1$. This condition may be based on a byzantine fault tolerant algorithm, and the byzantine fault tolerant algorithm is a fault tolerant algorithm oriented to a Byzantine problem, to resolve a problem of how to achieve consensus in a network where communication is reliable but

nodes may fail or act maliciously. The core theory is that the total quantity of nodes in the system is greater than or equal to $3f + 1$, and f is the quantity of allowed faulty nodes. In other words, for a certain proposal block, only a quorum (not less than $2f + 1$) of validation nodes validating the proposal block can ensure the correctness of the proposal block. Therefore, it may be required for a preceding proposal block that the quantity of proposal blocks generated by the preceding proposal block is not less than $2f + 1$ in the embodiments of the present invention.

[0147] In addition, for the examination of the reference relationships, a depth-first search (DFS) manner may be adopted. A certain subsequent proposal block generated in an $(n+x_2)$ -th consensus round may be used as a starting point, and proposal blocks having reference relationships with the subsequent proposal block may be traversed preferentially. In this way, it may be determined whether a specific preceding proposal block is supported by enough validation nodes through the reference relationships.

[0148] FIG. 15 is a schematic structural diagram of a terminal device according to an embodiment of the present invention. The terminal device 1000 includes at least one processor 1001 (only one processor is shown in FIG. 15), a memory 1002, and a computer program 1003 stored in the memory 1002 and executable on the at least one processor 1001. When executing the computer program 1003, the processor 1001 implements the steps in the method of the above embodiments.

[0149] The terminal device 1000 may be a computing device such as a desktop computer, a notebook computer, a palmtop computer, and a cloud server. The terminal device may include, but not be limited to, a processor 1001 and a memory 1002. Those skilled in the art

may understand that FIG. 15 is merely an example of the terminal device 1000, and does not constitute a limitation on the terminal device 1000. The terminal device may include more or fewer components than those shown in the figure, or some components may be combined, or different components may be included. For example, the terminal device may further include
5 an input/output device, a network access device, and the like.

[0150] The processor 1001 may be a central processing unit (CPU). Alternatively, the processor 1001 may be another general-purpose processor, a digital signal processor (DSP), an application-specific integrated circuit (ASIC), a field-programmable gate array (FPGA) or other programmable logic devices, a discrete gate or transistor logic device, a discrete
10 hardware component, or the like. The general-purpose processor may be a microprocessor or the processor may be any conventional processor or the like.

[0151] In some embodiments, the memory 1002 may be an internal storage unit of the terminal device 1000, for example, a hard disk drive or an internal storage of the terminal device 1000. In some other embodiments, the memory 1002 may be an external storage device
15 of the terminal device 1000, for example, a removable hard disk, a smart media card (SMC), a secure digital (SD) card, a flash card, or the like that is provided on the terminal device 1000. Furthermore, the memory 1002 may include both the internal storage unit and the external storage device of the terminal device 1000. The memory 1002 is configured to store an operating system, an application program, a boot loader, data, other programs, and the like,
20 for example, program code of the computer program. The memory 1002 may be further configured to temporarily store output data or to-be-output data.

[0152] It can be clearly understood by those skilled in the art that, for describing

conveniently and concisely, dividing of the above various functional units and functional modules is described exemplarily merely. In an actual application, the above functions can be assigned to different functional units and functional modules to be accomplished. That is, an inner structure of the apparatus is divided into different functional units or modules so as to
5 accomplish the whole or a part of functionalities described above. The various functional units, modules in the embodiments can be integrated into a processing unit, or each of the units exists independently and physically, or two or more than two of the units are integrated into a single unit. The above integrated unit can be either implemented in the form of hardware or in the form of software functional units. In addition, specific names of the various functional
10 units and modules are only used for distinguishing from each other conveniently, but not intended to limit the protection scope of the present invention. Regarding a specific operating process of the units and modules in the above system, reference can be made to a corresponding process in the above method embodiments, which is not repeated herein.

[0153] An embodiment of the present application further provides a computer-readable
15 storage medium, on which a computer program is stored. When the computer program is executed by a processor, steps in the above method embodiments may be implemented.

[0154] Embodiments of the present invention provide a computer program product. When the computer program product runs on a mobile terminal, the mobile terminal is enabled to implement the steps in the above method embodiments.

20 [0155] When the integrated unit is implemented in the form of a software functional unit and sold or used as an independent product, the integrated unit may be stored in a computer-readable storage medium. Based on such understanding, all or part of the processes of the

method in the above embodiments may be implemented by instructing related hardware by using a computer program, and the computer program may be stored in a computer-readable storage medium. When the computer program is executed by a processor, the steps of the above method embodiments may be implemented. The computer program includes computer
5 program code, which may be in the form of source code, object code, an executable file, some intermediate forms, or the like. The computer-readable medium may include at least any entity or apparatus capable of carrying computer program code to a photographing apparatus/terminal device, a recording medium, a computer memory, a read-only memory (ROM), a random access memory (RAM), an electrical carrier signal, a telecommunication
10 signal, or a software distribution medium, such as a USB flash drive, a mobile hard disk, a magnetic disk or an optical disk. In some jurisdictions, computer-readable media may not include the electrical carrier signal and the telecommunications signal according to legislation and patent practices.

[0156] In the above embodiments, the description of each embodiment has its own
15 emphasis. For parts not detailed or described in a certain embodiment, reference may be made to relevant descriptions in other embodiments.

[0157] Those skilled in the art will appreciate that the units and algorithm steps of the examples described in connection with the embodiments disclosed herein can be implemented in electronic hardware, or a combination of computer software and electronic hardware. It
20 depends on the specific application and design constraints of the technical solution whether these functions are performed in hardware or software. Professionals may use different methods for each specific application to implement the described functionality, but such

implementation should not be considered beyond the scope of the present invention.

[0158] In the embodiments provided in the present invention, it should be understood that the disclosed apparatus/network device and method may be implemented in other ways. For example, the apparatus/network device embodiments described above are merely illustrative.

5 For example, the division of the modules or units is based on logical functionality. In practical implementation, there may be other divisions. For example, multiple units or components may be combined or may be integrated into another system, or some features may be omitted or not executed. Alternatively, the couplings, direct couplings, or communicating connections between the shown or discussed components may be indirect couplings or communicating
10 connections through some interfaces, apparatus, or units, and may be electrical, mechanical, or of other forms.

[0159] The units described as separate components may or may not be physically separate. The components displayed as units may or may not be physical units. That is, the components may be located in one place, or may be distributed across multiple network units. Some or all
15 of the units may be selected as required to achieve the objective of the solution in the embodiments.

[0160] The above embodiments are merely intended to illustrate the technical solutions of the present invention, but are not intended to limit the present invention. Although the present invention has been described in detail with reference to the above embodiments, those
20 skilled in the art should understand that the technical solutions described in the above embodiments may still be modified, or some technical features may be equivalently replaced. Such modifications or substitutions do not cause the essence of the corresponding technical

solutions to depart from the spirit and scope of the technical solutions of the embodiments of the present invention and shall fall within the protection scope of the present invention.

CLAIMS

What is claimed is:

1. A consensus method for a blockchain system, characterized in that the blockchain system comprises a plurality of validation nodes, the consensus method is executed by a validation
5 node, and the consensus method comprises:

generating a subsequent proposal block by referencing a plurality of proposal blocks and combining current transactions from clients in an $(n+x_2)$ -th consensus round, wherein the plurality of proposal blocks are correspondingly generated by the plurality of validation nodes referencing a preceding proposal block in an n -th consensus round, and the preceding proposal
10 block is generated in an $(n-x_1)$ -th consensus round; and

achieving consensus on the preceding proposal block by using the subsequent proposal block as a certificate for the preceding proposal block.

2. The consensus method of claim 1, wherein x_1 and x_2 are identical or different, and x_1 and
15 x_2 are each equal to 1 or each not equal to 1.

3. The consensus method of claim 1, wherein the corresponding generation of the plurality of proposal blocks by the plurality of validation nodes referencing the preceding proposal block comprises:

20 correspondingly generating the plurality of proposal blocks by the validation node, together with others of the plurality of validation nodes, referencing the preceding proposal block.

4. The consensus method of claim 1, wherein a step of referencing any proposal block comprises: referencing a hash index of the any proposal block, wherein the any proposal block is one of the plurality of proposal blocks or the preceding proposal block.

5

5. The consensus method of claim 1, wherein the preceding proposal block is generated by the validation node, or the preceding proposal block is generated by other validation nodes.

6. The consensus method of claim 1, wherein using the subsequent proposal block as the certificate for the preceding proposal block comprises:

using the subsequent proposal block together with subsequent proposal blocks generated by at least one other validation node independently as certificates for the preceding proposal block, and when the quantity of the certificates is greater than a preset threshold, the preceding proposal block is validated.

15

7. The consensus method of claim 1, wherein the quantity of the plurality of proposal blocks is not less than $2f+1$, where f is the quantity of dishonest validation nodes in the n -th consensus round, and $3f+1$ is the total quantity of validation nodes in the n -th consensus round.

20 8. A consensus apparatus for a blockchain system, characterized by comprising:

a subsequent proposal block generating module, configured to generate a subsequent

proposal block by referencing a plurality of proposal blocks and combining current transactions from clients in an $(n+x_2)$ -th consensus round, wherein the plurality of proposal blocks are correspondingly generated by the plurality of validation nodes referencing a preceding proposal block in an n -th consensus round, and the preceding proposal block is generated in an $(n-x_1)$ -th consensus round; and

a consensus module, configured to achieve consensus on the preceding proposal block by using the subsequent proposal block as a certificate for the preceding proposal block.

9. The consensus apparatus of claim 8, wherein x_1 and x_2 are identical or different, and x_1 and x_2 are each equal to 1 or each not equal to 1.

10. The consensus apparatus of claim 8, wherein a step of referencing any proposal block comprises: referencing a hash index of the any proposal block, wherein the any proposal block is one of the plurality of proposal blocks or the preceding proposal block.

15

11. The consensus apparatus of claim 8, wherein the preceding proposal block is generated by the validation node, or the preceding proposal block is generated by other validation nodes.

12. The consensus apparatus of claim 8, wherein the quantity of the plurality of proposal blocks is not less than $2f+1$, where f is the quantity of dishonest validation nodes in the n -th consensus round, and $3f+1$ is the total quantity of validation nodes in the n -th consensus round.

13. A terminal device, comprising a memory, a processor, and a computer program stored in the memory and executable on the processor, characterized in that the terminal device is any one of a plurality of validation nodes included in a blockchain system; and when executing the computer program, the processor implements the following steps:

- 5 generating a subsequent proposal block by referencing a plurality of proposal blocks and combining current transactions from clients in an $(n+x_2)$ -th consensus round, wherein the plurality of proposal blocks are correspondingly generated by the plurality of validation nodes referencing a preceding proposal block in an n -th consensus round, and the preceding proposal block is generated in an $(n-x_1)$ -th consensus round; and
- 10 achieving consensus on the preceding proposal block by using the subsequent proposal block as a certificate for the preceding proposal block.

14. The terminal device of claim 13, wherein x_1 and x_2 are identical or different, and x_1 and x_2 are each equal to 1 or each not equal to 1.

15

15. The terminal device of claim 13, wherein the corresponding generation of the plurality of proposal blocks by the plurality of validation nodes referencing the preceding proposal block comprises:

- correspondingly generating the plurality of proposal blocks by the validation node,
- 20 together with others of the plurality of validation nodes, referencing the preceding proposal block.

16. The terminal device of claim 13, wherein a step of referencing any proposal block comprises: referencing a hash index of the any proposal block, wherein the any proposal block is one of the plurality of proposal blocks or the preceding proposal block.
- 5 17. The terminal device of claim 13, wherein the preceding proposal block is generated by the validation node, or the preceding proposal block is generated by other validation nodes.
18. The terminal device of claim 13, wherein using the subsequent proposal block as the certificate for the preceding proposal block comprises:
- 10 using the subsequent proposal block together with subsequent proposal blocks generated by at least one other validation node independently as certificates for the preceding proposal block, and when the quantity of the certificates is greater than a preset threshold, the preceding proposal block is validated.
- 15 19. The terminal device of claim 13, wherein the quantity of the plurality of proposal blocks is not less than $2f+1$, where f is the quantity of dishonest validation nodes in the n -th consensus round, and $3f+1$ is the total quantity of validation nodes in the n -th consensus round.
- 20 20. A computer-readable storage medium, on which a computer program is stored, characterized in that the computer program is executed by a processor to implement the steps of the consensus method of any one of claims 1 to 7.