~~WHAT IS CLAIMED IS:~~

CLAIMS

~~1.~~ 1. A computer-implemented method for achieving a consensus among a plurality of network nodes of a blockchain network (102) comprising at least a primary node (404) and one or more backup nodes~~,~~ (406), the method comprising:

receiving~~,~~ (410), by the primary node, a transaction request;

generating~~,~~ (412), by the primary node, a plurality of erasure code~~(~~, EC~~)~~, blocks according to an EC code using the transaction request;

sending~~,~~ (416), by the primary node, a plurality of first messages to the one or more backup nodes, respectively, wherein each of the plurality of first messages comprises a composite hash value associated with the plurality of EC blocks, at least one of the plurality of EC blocks, and respective hash values of the plurality of EC blocks different from the at least one EC block, and wherein the composite hash value is based on the respective hash values;

receiving~~,~~ (420), by the primary node, at least one second message from at least one of the backup nodes, wherein the at least one second message comprises one of the plurality of first messages and a signature of the at least one of the backup nodes associated with the one of the plurality of first messages~~;~~, and wherein the at least one second message further comprises at least one of the plurality of EC blocks, the composite hash value, and respective hash values of the plurality of EC blocks different from the at least one EC block;

in response to receiving the at least one second message from the at least one of the backup node, verifying~~,~~ (422), by the primary node, whether the at least one second message is valid~~:~~, wherein verifying comprises:

~~determining,~~generating, by the primary node, a reconstructed hash tree using the at least one of the plurality of EC blocks in the at least one second message,

determining, by the primary node, a reconstructed composite hash value of the reconstructed hash tree,

comparing, by the primary node, the reconstructed composite hash value to the composite hash value in the at least one second message, and

determining, by the primary node, whether the reconstructed composite hash value matches the composite hash value in the at least one second message;

determining (424), by the primary node, whether a number of valid second messages exceeds a pre-determined threshold;

in response to determining that the number of valid second messages exceeds the pre-determined threshold, reconstructing~~,~~ (426), by the primary node, the transaction request based on a subset of the number of valid second messages according to the EC code;

in response to determining that the transaction request has been successfully reconstructed, sending~~,~~ (430), by the primary node, ~~a~~an outgoing third message, to the other ~~network~~backup nodes, wherein the outgoing third message comprises a set of signatures that are in the valid second messages;

receiving~~,~~ (434), by the primary node, at least one incoming third message from at least one of the backup nodes; and

in response to receiving a pre-determined number of incoming third messages that are identical, executing~~,~~ (436), by the primary node, the transaction request.

2. The method of ~~Claim~~claim 1, wherein~~:~~

~~2.~~ the transaction request is associated with a sequence number~~.~~; and/or

~~3. The method of Claim 1, wherein the~~ generating (412) the plurality of EC blocks according to an EC code comprises~~:~~

transforming the transaction request into an EC message using the EC code~~;~~, and

dividing the EC message into the plurality of EC ~~block.~~blocks; and/or

~~4. The method of Claim 1, wherein the composite hash value of the plurality of EC block is generated using a hash tree.~~

~~5. The method of Claim 4, wherein the hash tree comprises a Merkle tree, and wherein the composite hash value is a root hash value of the Merkle tree.~~

~~6. The method of Claim 1, wherein~~ the signature of the at least one of the backup nodes associated with the one of the plurality of first messages comprises a private key signature of the at least one of the backup nodes associated with the one of the plurality of first messages.

~~7.~~ 3. The method of ~~Claim~~claim 1, wherein the ~~at least one second message further comprises at least one of the plurality of EC blocks.~~

~~8. The method of Claim 7, wherein the verifying, by the primary node, whether the at least one second message is valid comprises:~~

~~generating, by the primary node, a reconstructed hash tree using the at least one of the plurality of EC blocks in the at least one second message;~~

~~determining, by the primary node, a reconstructed~~ composite hash value of the ~~reconstructed~~ plurality of EC blocks is generated using a hash tree; optionally wherein the hash tree~~;~~

~~comparing, by the primary node, the reconstructed~~ comprises a Merkle tree, and wherein the composite hash value ~~to~~is a ~~composite~~root hash value ~~in~~of the ~~at least one second message; and~~Merkle tree.

~~determining, by the primary node, whether the reconstructed composite hash value matches the composite hash values in the at least one second message.~~

~~9.~~ ~~4.~~ The method of ~~Claim 7, wherein the method~~claim 1, further ~~comprises:~~ comprising, in response to determining that the reconstructed composite hash value matches the composite hash values in the second messages, determining, by the primary node, that the at least one second message is valid.

5. The method of ~~Claim~~claim 1, wherein the pre-determined number of incoming third messages that are identical comprise ~~the~~a pre-determined number of ~~the~~incoming third messages having an identical set of signatures.

~~11~~

6. A computer-implemented method for achieving a consensus among a plurality of network nodes of a blockchain (102) comprising at least a primary node (404) and one or more backup nodes, the method comprising:

~~—~~receiving~~,~~ (416), by a backup node~~,~~ (406), a first message from the primary node, wherein the first message comprises a composite hash value associated with a plurality of ~~EC blocks,~~erasure code, EC, blocks, at least one of the plurality of EC blocks, and respective hash values of the plurality of EC blocks different from the at least one EC block, wherein the composite hash value is based on the respective hash values, and wherein the plurality of EC blocks are generated by the primary node according to an EC code using a transaction request;

~~—~~in response to receiving the first message, sending~~,~~ (421), by the backup node, ~~a~~an outgoing second message to the other ~~network~~backup nodes, wherein the outgoing second message comprises the first message and a signature of the backup node associated with the first message;

———receiving, (423), by the backup node, at least one incoming second message from at least one backup node other than the backup node, the at least one incoming second message further comprising at least one of the plurality of EC blocks and respective hash values of the plurality of EC blocks different from the at least one EC block;

in response to receiving the at least one incoming second message from the at least one backup node, verifying, by the backup node, whether the at least one incoming second message is valid, the verifying comprising:

generating, by the backup node, a reconstructed hash tree using the at least one of the plurality of EC blocks and the respective hash values in the at least one second message,

determining, by the backup node, a reconstructed composite hash value of the reconstructed hash tree,

comparing, by the backup node, the reconstructed composite hash value to the composite hash value in the at least one second message, and

determining, by the backup node, whether the reconstructed composite hash value matches the composite hash value in the at least one second message;

determining, by the backup node, whether a number of valid incoming second messages exceeds a pre-determined threshold;

in response to determining that the number of valid incoming second messages exceeds the pre-determined threshold, reconstructing, by the backup node, the transaction request based on a subset of the number of valid incoming second messages according to the EC code;

in response to determining that the transaction request has been successfully reconstructed, sending, (434), by the backup node, an outgoing third message, to the other backup nodes, wherein the outgoing third message comprises a set of signatures that are in the valid incoming second messages;

receiving, by the backup node, at least one incoming third message from at least one of the backup nodes; and

———in response to receiving a pre-determined number of incoming third messages that are identical, executing, by the backup node, the transaction request.

7. The method of claim 6, wherein:

- the generating the plurality of EC blocks according to an EC code comprises:

transforming the transaction request into an EC message using the EC code; and

———dividing the EC message into the plurality of EC block.

~~13. The method of claim 11, wherein the composite hash value of the plurality of EC block is generated using a hash tree.~~

~~14. The method of claim 13, wherein the hash tree comprises a Merkle tree,~~blocks; and ~~wherein the composite hash value is a root hash value of the Merkle tree.~~/or

~~15. The method of Claim 11, wherein~~ the signature of the backup node associated with the first message comprises a private key signature of the backup node associated with the first message~~.~~; and/or

- the pre-determined number of third messages that are identical comprise the pre-determined number of the incoming third messages having an identical set of signatures.

~~16.~~ 8. The method of ~~Claim 11~~claim 6, wherein the ~~at least one second message further comprises at least one of the plurality of EC blocks.~~

~~17. The method of Claim 16, wherein the verifying, by the backup node, whether the at least one second message is valid comprises:~~

~~generating, by the backup node, a reconstructed hash tree using the at least one of the plurality of EC blocks in the at least one second message;~~

~~determining, by the backup node, a reconstructed~~ composite hash value of the ~~reconstructed~~ plurality of EC blocks is generated using a hash tree; optionally wherein the hash tree~~;~~

~~comparing, by the backup node, the reconstructed~~ comprises a Merkle tree, and wherein the composite hash value ~~to~~is a ~~composite~~root hash value ~~in~~of the ~~at least one second message; and~~Merkle tree.

~~determining, by the backup node, whether the reconstructed composite hash value matches the composite hash values in the at least one second message.~~

~~18.~~ 9. The method of ~~Claim 17, wherein the method~~claim 8, further ~~comprises:~~ comprising, in response to determining that the reconstructed composite hash value matches the composite hash values in the incoming second messages, determining, by the backup node, that the at least one incoming second message is valid.

~~19.~~ 10. The method of ~~Claim 11~~claim 6, wherein the pre-determined number of incoming third messages that are identical comprise the pre-determined number of ~~the~~incoming third messages having an identical set of signatures.

~~20.~~11. A non-transitory computer-readable storage medium coupled to one or more computers and configured with instructions executable by the one or more computers to~~:~~ perform the method of any one of claims 1 to 5 or to perform the method of any one of claims 6 to 10.

~~receive, by a primary node of a blockchain network, a transaction request, wherein the blockchain network further comprises one or more backup nodes;~~

~~generate, by the primary node, a plurality of erasure code (EC) blocks according to an EC code using the transaction request;~~

~~send, by the primary node, a plurality of first messages to the one or more backup nodes, respectively, wherein each of the plurality of first messages comprises a composite hash value associated with the plurality of EC blocks;~~

~~receive, by the primary node, at least one second message from at least one of the backup nodes, wherein the at least one second message comprises one of the plurality of first messages and a signature of the at least one of the backup nodes associated with the one of the plurality of first messages;~~

~~in response to receiving the at least one second message from the at least one of the backup node, verify, by the primary node, whether the at least one second message is valid;~~

~~determine, by the primary node, whether a number of valid second messages exceeds a pre-determined threshold;~~

~~in response to determining that the number of valid second messages exceeds the pre-determined threshold, reconstruct, by the primary node, the transaction request based on a subset of the number of valid second messages according to the EC code;~~

~~in response to determining that the transaction request has been successfully reconstructed, send, by the primary node, a third message, to the other network nodes, wherein the third message comprises a set of signatures that are in the valid second messages;~~

~~receive, by the primary node, at least one third message from at least one of the backup nodes; and~~

~~in response to receiving a pre-determined number of third messages that are identical, execute, by the primary node, the transaction request.~~

~~21. The non-transitory computer-readable storage medium of Claim 20, wherein the transaction request is associated with a sequence number.~~

~~22. The non-transitory computer-readable storage medium of Claim 20, wherein the generating the plurality of EC blocks according to an EC code comprises:~~
~~transforming the transaction request into an EC message using the EC code; and~~
~~dividing the EC message into the plurality of EC block.~~

~~23. The non-transitory computer-readable storage medium of Claim 20, wherein the composite hash value of the plurality of EC block is generated using a hash tree.~~

~~24. The non-transitory computer-readable storage medium of Claim 23, wherein the hash tree comprises a Merkle tree, and wherein the composite hash value is a root hash value of the Merkle tree.~~

~~25. The non-transitory computer-readable storage medium of Claim 20, wherein the signature of the at least one of the backup nodes associated with the one of the plurality of first messages comprises a private key signature of the at least one of the backup nodes associated with the one of the plurality of first messages.~~

~~26. The non-transitory computer-readable storage medium of Claim 20, wherein the at least one second message further comprises at least one of the plurality of EC blocks.~~

~~27. The non-transitory computer-readable storage medium of Claim 26, wherein the verifying, by the primary node, whether the at least one second message is valid comprises:~~
~~generating, by the primary node, a reconstructed hash tree using the at least one of the plurality of EC blocks in the at least one second message;~~
~~determining, by the primary node, a reconstructed composite hash value of the reconstructed hash tree;~~
~~comparing, by the primary node, the reconstructed composite hash value to a composite hash value in the at least one second message; and~~
~~determining, by the primary node, whether the reconstructed composite hash value matches the composite hash values in the at least one second message.~~

~~28. The non-transitory computer-readable storage medium of Claim 27, further configured with instructions executable by the one or more computers to:~~

~~in response to determining that the reconstructed composite hash value matches the composite hash values in the second messages, determine, by the primary node, that the at least one second message is valid.~~

~~29. The non-transitory computer-readable storage medium of Claim 20, wherein the pre-determined number of third messages that are identical comprise the pre-determined number of the third messages having an identical set of signatures.~~

~~30.~~

12. A system, including:

~~———————~~one or more computers; and

~~———————~~one or more computer-readable memories coupled to the one or more computers and configured with instructions executable by the one or more computers to~~:~~ perform the method of any one of claims 1 to 5 or to perform the method of any one of claims 6 to 10.

~~receive, by a primary node of a blockchain network, a transaction request, wherein the blockchain network further comprises one or more backup nodes;~~

~~generate, by the primary node, a plurality of erasure code (EC) blocks according to an EC code using the transaction request;~~

~~send, by the primary node, a plurality of first messages to the one or more backup nodes, respectively, wherein each of the plurality of first messages comprises a composite hash value associated with the plurality of EC blocks;~~

~~receive, by the primary node, at least one second message from at least one of the backup nodes, wherein the at least one second message comprises one of the plurality of first messages and a signature of the at least one of the backup nodes associated with the one of the plurality of first messages;~~

~~in response to receiving the at least one second message from the at least one of the backup node, verify, by the primary node, whether the at least one second message is valid;~~

~~determine, by the primary node, whether a number of valid second messages exceeds a pre-determined threshold;~~

~~in response to determining that the number of valid second messages exceeds the pre-determined threshold, reconstruct, by the primary node, the transaction request based on a subset of the number of valid second messages according to the EC code;~~

~~in response to determining that the transaction request has been successfully reconstructed, send, by the primary node, a third message, to the other network nodes, wherein the third message comprises a set of signatures that are in the valid second messages;~~

~~receive, by the primary node, at least one third message from at least one of the backup nodes; and~~

~~in response to receiving a pre-determined number of third messages that are identical, execute, by the primary node, the transaction request.~~

~~31. The system of Claim 30, wherein the transaction request is associated with a sequence number.~~

~~32. The system of Claim 30, wherein the generating the plurality of EC blocks according to an EC code comprises:~~
~~transforming the transaction request into an EC message using the EC code; and~~
~~dividing the EC message into the plurality of EC block.~~

~~33. The system of Claim 30, wherein the composite hash value of the plurality of EC block is generated using a hash tree.~~

~~34. The system of Claim 33, wherein the hash tree comprises a Merkle tree, and wherein the composite hash value is a root hash value of the Merkle tree.~~

~~35. The system of Claim 30, wherein the signature of the at least one of the backup nodes associated with the one of the plurality of first messages comprises a private key signature of the at least one of the backup nodes associated with the one of the plurality of first messages.~~

~~36. The system of Claim 30, wherein the at least one second message further comprises at least one of the plurality of EC blocks.~~

~~37. The system of Claim 36, wherein the verifying, by the primary node, whether the at least one second message is valid comprises:~~

generating, by the primary node, a reconstructed hash tree using the at least one of the plurality of EC blocks in the at least one second message;

determining, by the primary node, a reconstructed composite hash value of the reconstructed hash tree;

comparing, by the primary node, the reconstructed composite hash value to a composite hash value in the at least one second message; and

determining, by the primary node, whether the reconstructed composite hash value matches the composite hash values in the at least one second message.

38. The system of Claim 37, wherein the computer memories are further configured with instructions executable by the one or more computers to:

in response to determining that the reconstructed composite hash value matches the composite hash values in the second messages, determine, by the primary node, that the at least one second message is valid.

39. The system of Claim 30, wherein the pre-determined number of third messages that are identical comprise the pre-determined number of the third messages having an identical set of signatures.

40. A non-transitory computer-readable storage medium coupled to one or more computers and configured with instructions executable by the one or more computers to:

receive, by a backup node of a blockchain network, a first message from a primary node of the blockchain network, wherein the first message comprises a composite hash value associated with a plurality of EC blocks, wherein the plurality of EC blocks are generated by the primary node according to an EC code using a transaction request;

in response to receiving the first message, send, by the backup node, a second message to the other network nodes, wherein the second message comprises the first message and a signature of the backup node associated with the first message;

receive, by the backup node, at least one second message from at least one backup node other than the backup node;

in response to receiving the at least one second message from the at least one backup node, verify, by the backup node, whether the at least one second message is valid;

determine, by the backup node, whether a number of valid second messages exceeds a pre-determined threshold;

in response to determining that the number of valid second messages exceeds the pre-determined threshold, reconstruct, by the backup node, the transaction request based on a subset of the number of valid second messages according to the EC code;

in response to determining that the transaction request has been successfully reconstructed, send, by the backup node, a third message, to the other network nodes, wherein the third message comprises a set of signatures that are in the valid second messages;

receive, by the backup node, at least one third message from at least one of the backup nodes; and

in response to receiving a pre-determined number of third messages that are identical, execute, by the backup node, the transaction request.

41. The non-transitory computer readable storage medium of claim 40, wherein the generating the plurality of EC blocks according to an EC code comprises:

transforming the transaction request into an EC message using the EC code; and

dividing the EC message into the plurality of EC block.

42. The non-transitory computer readable storage medium of claim 40, wherein the composite hash value of the plurality of EC block is generated using a hash tree.

43. The non-transitory computer readable storage medium of claim 42, wherein the hash tree comprises a Merkle tree, and wherein the composite hash value is a root hash value of the Merkle tree.

44. The non-transitory computer readable storage medium of Claim 40, wherein the signature of the backup node associated with the first message comprises a private key signature of the backup node associated with the first message.

45. The non-transitory computer readable storage medium of Claim 40, wherein the at least one second message further comprises at least one of the plurality of EC blocks.

46. The non-transitory computer readable storage medium of Claim 45, wherein the verifying, by the backup node, whether the at least one second message is valid comprises:

generating, by the backup node, a reconstructed hash tree using the at least one of the plurality of EC blocks in the at least one second message;

determining, by the backup node, a reconstructed composite hash value of the reconstructed hash tree;

comparing, by the backup node, the reconstructed composite hash value to a composite hash value in the at least one second message; and

determining, by the backup node, whether the reconstructed composite hash value matches the composite hash values in the at least one second message.

47. The non-transitory computer-readable storage medium of Claim 46, further configured with instructions executable by the one or more computers to:

in response to determining that the reconstructed composite hash value matches the composite hash values in the second messages, determine, by the backup node, that the at least one second message is valid.

48. The non-transitory computer readable storage medium of Claim 40, wherein the pre-determined number of third messages that are identical comprise the pre-determined number of the third messages having an identical set of signatures.

49. A system, including:

one or more computers; and

one or more computer-readable memories coupled to the one or more computers and configured with instructions executable by the one or more computers to:

receive, by a backup node of a blockchain network, a first message from a primary node of the blockchain network, wherein the first message comprises a composite hash value associated with a plurality of EC blocks, wherein the plurality of EC blocks are generated by the primary node according to an EC code using a transaction request;

in response to receiving the first message, send, by the backup node, a second message to the other network nodes, wherein the second message comprises the first message and a signature of the backup node associated with the first message;

receive, by the backup node, at least one second message from at least one backup node other than the backup node;

in response to receiving the at least one second message from the at least one backup node, verify, by the backup node, whether the at least one second message is valid;

determine, by the backup node, whether a number of valid second messages exceeds a pre-determined threshold;

in response to determining that the number of valid second messages exceeds the pre-determined threshold, reconstruct, by the backup node, the transaction request based on a subset of the number of valid second messages according to the EC code;

in response to determining that the transaction request has been successfully reconstructed, send, by the backup node, a third message, to the other network nodes, wherein the third message comprises a set of signatures that are in the valid second messages;

receive, by the backup node, at least one third message from at least one of the backup nodes; and

in response to receiving a pre-determined number of third messages that are identical, execute, by the backup node, the transaction request.

50. The system of claim 49, wherein the generating the plurality of EC blocks according to an EC code comprises:

transforming the transaction request into an EC message using the EC code; and

dividing the EC message into the plurality of EC block.

51. The system of claim 49, wherein the composite hash value of the plurality of EC block is generated using a hash tree.

52. The system of claim 51, wherein the hash tree comprises a Merkle tree, and wherein the composite hash value is a root hash value of the Merkle tree.

53. The system of Claim 49, wherein the signature of the backup node associated with the first message comprises a private key signature of the backup node associated with the first message.

54. The system of Claim 49, wherein the at least one second message further comprises at least one of the plurality of EC blocks.

55. The system of Claim 54, wherein the verifying, by the backup node, whether the at least one second message is valid comprises:

generating, by the backup node, a reconstructed hash tree using the at least one of the plurality of EC blocks in the at least one second message;

~~determining, by the backup node, a reconstructed composite hash value of the reconstructed hash tree;~~

~~comparing, by the backup node, the reconstructed composite hash value to a composite hash value in the at least one second message; and~~

~~determining, by the backup node, whether the reconstructed composite hash value matches the composite hash values in the at least one second message.~~

~~56. The system of Claim 55, wherein the computer memories are further configured with instructions executable by the one or more computers to:~~

~~in response to determining that the reconstructed composite hash value matches the composite hash values in the second messages, determine, by the backup node, that the at least one second message is valid.~~

~~57. The system of Claim 49, wherein the pre-determined number of third messages that are identical comprise the pre-determined number of the third messages having an identical set of signatures.~~