CLAIMS

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:

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, an outgoing third message, to the other 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 1, wherein:

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

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 blocks; and/or

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.

3. The method of claim 1, wherein the composite hash value of the plurality of EC blocks is generated using a hash tree; optionally wherein the hash tree comprises a Merkle tree, and wherein the composite hash value is a root hash value of the Merkle tree.

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

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 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, an outgoing second message to the other 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 blocks; and/or

- 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.

8. The method of claim 6, wherein the composite hash value of the plurality of EC blocks is generated using a hash tree; optionally wherein the hash tree comprises a Merkle tree, and wherein the composite hash value is a root hash value of the Merkle tree.

9. The method of claim 8, further 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.

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

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.

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.