Reduction methods of the amount of data in blockchain node

Masaki OBAYASHI[†a)], ***Student Member***, **Takeshi OGAWA**[†b)], ***Member***

SUMMARY The amount of data that a node in a blockchain needs to store is increasing rapidly in proportion to the total number of transactions processed by the blockchain. As the number of data increases, it may be difficult to keep participating in the blockchain because of its storage capacity. If the number of nodes in a blockchain network decrease, serious problems such as unstable consensus formation in the blockchain will arise. This paper proposes novel reduction methods of the amount of data in blockchain nodes. In the proposed methods, a blockchain is divided into block groups and block group header for each block group which forms a block group chain are created. Each node stores only the latest blocks and block group headers and some block groups instead of whole blockchain and every block headers. As the result, the amount of data on the node can be significantly reduced compared to the conventional methods. It is also shown that the proposed method can significantly reduce the amount of data for both full nodes and light nodes.

keywords: *Blockchain Ethereum Markle Tree*

1. Introduction

Public blockchains such as Bitcoin and Ethereum generally consist of two types of nodes: full nodes and light nodes. Full nodes are required to store all transaction data processed on the blockchain. A light node does not need to store transaction data, but it must store the block headers of all blocks to verify that any given transaction has been processed. On the other hand, the number of transactions that need to be processed tends to increase as applications using blockchain gain attention from around 2021. In addition, the processing performance is expected to improve significantly in the future due to efforts such as Ethereum 2.0. Therefore, the amount of data that both nodes need to store is expected to increase significantly in the future.

There have been studies to reduce the amount of data per full node by distributing the blockchain among nodes [1][2], but there were two issues: support for blockchain branching (Challenge 1), and further data reduction for resource-constrained nodes such as smartphones (Challenge 2).

In this paper, we propose methods to solve both problems. The first is solved by shifting to a distributed data store after the main chain has been extended to the length that it can be finalized. The second is solved by extending the conventional Markle tree and applying it hierarchically to a fixed number of consecutive blocks (hereinafter "block groups"). The proposed method significantly reduces the amount of data for both full nodes and light nodes. We also report that the amount of data required to be retained by full nodes and light nodes can be reduced to less than 1GB using the proposed method.

In the remaining chapters of this paper, related technical terms are explained in Chapters 2 through 4, the issues of this study are presented in detail in Chapter 5, related studies are shown in Chapter 6, and requirements for data volume reduction methods are shown in Chapter 7, proposed methods are explained in detail in Chapter 8, evaluation and discussion are shown in Chapter 9, and summarized in Chapter 10.

2. Blockchain

A blockchain is a data structure in which transactions received by full nodes are organized into blocks, and the hash values of the blocks are chained together.

Each block can only be created by a full node that has been determined by an extremely difficult-to-fraudulent consensus algorithm such as PoW or PoS. If multiple full nodes are determined at almost the same time, blocks connected to the same previous block may be created and the chain may diverge. The next block is connected to the longer chain (the main chain), but immediately after the split, the shorter chain may be extended due to the propagation delay of the block, and the main chain may switch. However, as the difference in chain length increases, the probability of a subsequent length reversal decreases significantly. In the case of bitcoin, when the difference in length exceeds 6 to 8 blocks, the main chain can be almost finalized [3].

When a new node to be a full node joins a blockchain, it downloads the entire blockchain from an existing full node. The hash values of the blockchain from the genesis block to the most recent block are consistent, and the end of the chain continues to grow, so that the blockchain can be judged to be a legitimate main chain.

Since a full node retains all transaction data, it can determine whether any given transaction has been processed (i.e., whether it is part of the main chain). A light node is a node that stores only the header information of each block (hereinafter referred to as "block header chain") but not whole blocks.

The block header records a value of a Markle root of a Markle tree constructed by the transactions stored in the

[†]The author is with Tokyo Denki University at the Graduate School of Information System Engineering

[a)] E-mail: 21amj03@ms.dendai.ac.jp

[b)] E-mail: t.ogawa@mail.dendai.ac.jp

block. A Markle tree is a data structure in which *K* pieces of data are aggregated into a single hash value (Markle root) by aggregating every two pieces of data at a time (*Figure 1*).

When a light node wants to determine whether a transaction (*e.g., Transaction 0 in Figure 1*) is included in the main chain, it sends the hash value of the transaction to one of the full nodes. If the transaction is in the main chain, the full node sends back only the $\log_2 K$ hash values (*Hash0-1 and Hash1 in Figure 1*) needed to compute the Markle root in combination with the transaction hash values.

The light node calculates the Markle root value from the received hash value, compares it with the Markle root value in the block header it has retained, and if they are the same, it can determine that the transaction was included in the block.

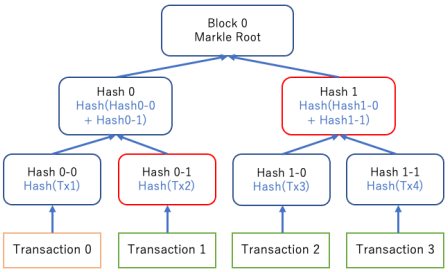

Figure 1. Markle Tree

3. Full Node and Light Node

In addition to all the data in the blockchain, each full node stores states of all accounts in the blockchain network, called world state, including bytecodes of all smart contracts. When it receives a block, it processes transactions in the block and updates the world state. By participating as a full node, a node becomes a block creator and is rewarded for block creation. The blockchain network guarantees consensus on which transactions were processed and in what order by having many full nodes creating and verifying blocks.

Light nodes store only the header information of the blockchain and do not process transactions or create blocks. Since it stores block headers but not entire blocks, it can operate with less data than a full node. However, it cannot create blocks and thus cannot earn rewards. Transactions can be created by both full and light nodes. Other devices can also send transactions to the blockchain network through these nodes. However, if a device wishes to verify that the transaction is part of the main chain, it must become a full node or a light node.

4. Blockchain data size problem

Comparing the data volume of full and light nodes of Ethereum [4] in 2021 and 2022 as compiled by EtherScan, the full node increased by 209.17 GB and the light node increased by 1.12 GB [5]. It should be noted that a full node stores world state data as well as blockchain data. It was about 35GB in 2018[6].

We expect that the amount of data is likely to increase even more rapidly as the number of users increases. The increase in node size raises the economic hurdle to become a full node and reduces the incentive to build nodes. Reduced node decentralization increases the risk of cracking due to unstable blockchain consensus, and the concentration of nodes that can send transactions to the blockchain network results in a single point of failure. Therefore, it is necessary to reduce the amount of data held by full and light nodes.

5. Related work

In Kaneko et al.'s research, nodes belonging to a network were randomly clustered based on their IP addresses. Each cluster creates blocks in turn and stores only blocks created by the cluster to reduce the storage load of each node [1]. However, it does not take blockchain branching into account and does not apply to major blockchain technologies such as Ethereum and Bitcoin. Yibin Xu et al. reduced storage load by dividing the blockchain into segments and holding them in a distributed manner [2]. However, this was insufficient as a capacity reduction for lightweight devices.

6. Requirements for reduction methods of the amo unt of data in blockchain node

In addition to solving challenges 1 and 2 above, the data volume reduction method must satisfy the following requirements.

① The validity of transferred data can be verified when a new node joins the blockchain network.
② Every node can verify with as little data as possible that a given transaction has been included in the blockchain or not.
③ No block data must be lost from within the blockchain network.

7. Proposal

7.1 System overview

The latest M to M+N blocks are maintained by all full nodes as before, and the validity of the connection between blocks is verified by the same method as before (chain of hash values of the previous blocks). M is a sufficient length to prevent forking and N is the number of blocks in the block group, both of which are fixed values (Figure 2).

When the number of blocks managed in the chain between blocks reaches M+N, the management of the older N blocks is switched to management by block groups. By delaying the transition to management by block groups, the effect of the main chain switch is avoided.

A block group is a group of N consecutive blocks in the main chain (Figure 3). The block group header contains the block group Markle root value, which is an aggregate of the Markle root values of the blocks in the block group, the hash

value of the previous block group header, and the hash value of the world state of the block immediately before the next checkpoint. The hash value is used to construct a new chain of block group headers (hereafter referred to as the block group header chain). The first block group header contains the hash value of the genesis block, and the header of the first block in each block group contains the hash value of the previous block group header (Figure 4).

Each block group and block group headers are generated independently by each full node and each light node from the blocks they have received, but they are identical because they are generated after the blocks in the block group are finalized. Then, the other full nodes, except for some full nodes, discard the blocks that make up the block group, thereby reducing the data volume of the full nodes. Light nodes store received block headers temporarily. Then if the number reached M+N, the old N block headers are switched to block group management. They discard the old block headers and store only the block group headers instead, thereby reducing the data volume of light nodes as well. Full nodes also retain the entire block group header chain.

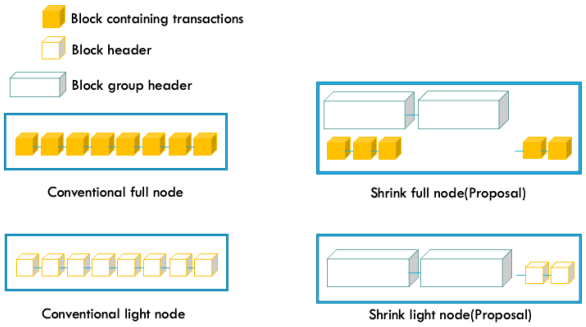

Figure 2. Proposed method

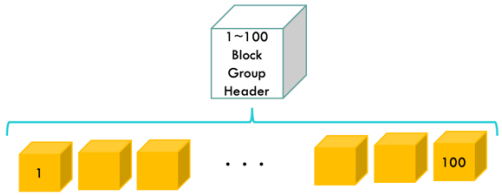

Figure 3. Block group header (Block1~100)

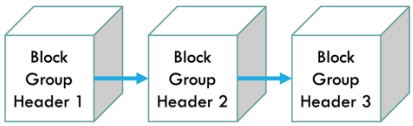

Figure 4. Block group header chain

7.2 Method of verifying transfer data at the time of new participation

To become a full node, a node downloads three types of block data from other full nodes: the entire block group header chain, data for a particular number of block groups greater than or equal to zero, and the blocks from the latest checkpoint to the latest block; and world state data.

By verifying the hash chain in the downloaded data, the node to be a full node can check for inconsistencies in the hash values of the blockchain from the genesis block to the latest block. The chain's continued growth allows the node to determine that the block group header chain and the blockchain that follows it are recognized by many other full nodes as the legitimate main chain. The block group Markle root is calculated from the Markle root value of each block in the received block group and compared with the block group Markle root value in the block group header chain to confirm that it is a legitimate block group. The world state can also be verified that it has not been tampered with by comparing the hash value of the world state of the block group with the hash value of the received world state. Similarly, for a node to become a light node, it must download the entire block group header chain and the block headers from the latest checkpoint to the latest block from another node and confirm that it is a legitimate main chain using the same method described above. After confirmation, the light node can discard the received block header data.

7.3 Method of verifying that a transaction has been in cluded in the blockchain

In the proposed method, the Markle root of the Markle tree (block group Markle root) created from the Markle root of each block in the block group is stored in the block group header (Figure 5). Let K as the average number of transactions in a block and N as the number of blocks in a block group, a light node can verify that a particular transaction is part of a certain block group by getting only $\log_2(K \cdot N)$ hash values from full nodes. If K and n is 100 and 1000 each, the number of the hash values is 17. Hash calculation is not a heavy process, so we do not consider it to be a significant burden.

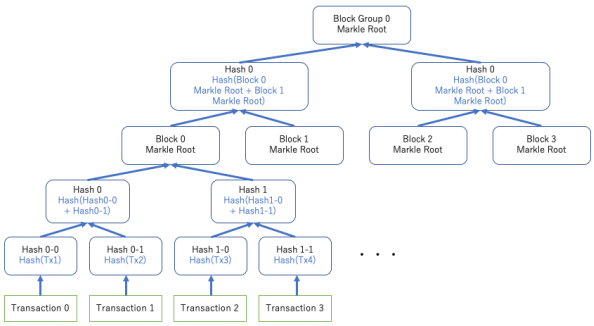

Figure 5. Extends the previous Markle tree

7.4 Prevent block lost

A mechanism is needed to ensure that all blocks continue to exist in the blockchain network, and in the Yibin Xu study, the more data a full node retains at the time of block generation, the easier it is to obtain the right to generate blocks, which motivates data retention. The amount of data held by a full node at the time of block creation is used as a motivation to retain data. In addition, proof of retention of a randomly specified transaction at the time of block

generation is required to prevent unauthorized discards of blocks. We consider that the scheme proposed in this paper can be combined with similar mechanisms, but the details are a subject for future work.

8. Evaluation and discussion

Let B be the average data volume of one block, T be the data volume of one block header, H be the number of blocks, and let N be the number of blocks in one block group, R is the number of block groups that a node has, G be the data volume of one block group header, and M is the minimum number of recent blocks that every full node stores. The equations for calculating the amount of data of shrunk full node and shrunk light node with the proposed method are as follows.

$$Shrunk\ Full\ Node\ Storage = R \cdot B \cdot N + \frac{G \cdot H}{N} + B \cdot M \quad (1)$$

$$Shrunk\ Light\ Node\ Storage = \frac{G \cdot H}{N} + T \cdot M \quad (2)$$

Table 1 shows the results of calculating the amount of data for conventional techniques, related research [2], and the proposed method based on Equations (1) and (2).

The values of each parameter in Equations (1) and (2) are calculated based on the following assumptions.

- The data of OpenEthereum of April 5, 2022 [5] was used, H was 14.5M (pieces) and B was 42.5KB.
- R varies depending on the motivation of the node that stores the data, here the minimum value of 1 was used.
- Here N was 1000, that was one block group consisting of 1000 blocks. (The blockchain was divided into 14.5K block groups.)
- G was set 96bytes because a block group header consists of 3 data; the first is 32bytes of the hash number of the previous block group header, the second is 32bytes of the Markle root of the blocks consisting of the group, and the last is 32bytes of the hash value of the world state of the block just before the next checkpoint.
- T was set to 516bytes regarding Ethereum.
- M was set to 1000 pieces.

The calculation example in Table 1 shows that the total amount of data for the smallest full node of the proposed method is about 86.4 MB, which is significantly less than that of related studies. The data volume for the light node was even smaller at about 2.4 MB. We consider that this is due to the introduction of block group headers.

In the case of full nodes, the amount of data on a node increase with R (the number of block groups held), but this amount can be changed dynamically. Therefore, we consider that even lightweight devices can be adapted.

It must be guaranteed that no blocks are lost from the network, and we will study the details of a distribution method to prevent block loss.

9. Conclusion

In this paper, we proposed methods that can significantly reduce the amount of data that needs to be stored in the blockchain node by dividing the blockchain to block groups and introducing a new data structure, the block group header.

References

[1] Yudai Kaneko, Takuya Asaka, “DHT Load balancing and clustering for storage size in blockchain,” IEICE Technical Report, vol. 118, no. 371, NS2018-161, pp. 29-34, 1.7.2021

[2] Yibin Xu, Yangyu Huan, “Segment Blockchain: A Size Reduced Storage Mechanism for Blockchain,” IEEE, 13.1.2020

[3] Satoshi Nakamoto” Bitcoin: A Peer-to-Peer Electronic Cash System” Bitcoin.org,

[4] Ethereum.org, ”Ethereum Whitepaper,” (https://ethereum.org/en/whitepaper/)

[5] Ethereum, ”Ethereum Full Node Sync (Default) Chart,” (https://etherscan.io/chartsync/chaindefault), 2021.6.11.

[6] Vitalik Buterin, “A state expiry and statelessness roadmap”, (https://notes.ethereum.org/@vbuterin/verkle_and_state_expiry_proposal)

Table 1. Comparison of data volume between conventional nodes and proposed method

	Conventional technology		Related work[2]		Proposal method	
	Full node	Light node	Full node	Light node	Full node	Light node
Block (Amount of data)	14.5M (616.2GB)	-	2K (85.0MB)	-	2K (84.0MB)	-
Block header (Amount of data)	14.5M (7.5GB)	Same as left	Same as left	Same as left	2K (1.0MB)	Same as left
Block group header (Amount of data)	-	-	-	-	14.5K (1.4MB)	Same as left
Total amount of data	616.2GB	7.5GB	7.6GB	7.5GB	86.4MB	2.4MB