INFOCOM 2019 WKSHPS - CryBlock 2019 1570517104

Downsampling Blockchain Algorithm

Li Quan[1], Qin Huang[1], Shengli Zhang[2], and Zulin Wang[1]
[1]School of Electronic and Information Engineering, Beihang University, Beijing, China, 100191
[2]College of Information Engineering, Shenzhen University, Shenzhen, China, 518060

***Abstract*—In blockchain, every full node has to store all history transactions in block bodies. However, with the rapid growth of transactions, the storage bloating problem has emerged, and made it challenging for a mobile system to afford the storage and synchronization overhead. This paper proposes to downsample block bodies to reduce the nodes' storage. By downsampling block bodies by M times, only about $1/M$ of the block bodies with high information entropy are downloaded and stored. Performance analysis demonstrates that the proposed downsampling nodes and full nodes have similar broadcast accuracy with appropriate M. The simulation results show that the proposed algorithm can provide a better cost-effective choice for nodes between broadcast accuracy and storage.**

***Index Terms*—Blockchain, broadcast, information entropy, downsampling, most recent state**

I. Introduction

Based on the culmination of decades of research in cryptography and distributed systems including decentralized peer-to-peer network, consensus mechanisms, and public transaction ledger [1], blockchain was invented and made value-transfer over decentralised systems possible. It has been widely used in the fields of mobile systems, finance, accountable Clouds [2] and charity with its non-falsification, non-forgery, time-series, highly-redundant storage, distributed credit and privacy protection [3]. Especially on mobile systems, some researchers have delved into blockchain-based identity management [4] and state channel [5]. However, the limited storage and bandwidth on mobile systems impose more restrictions on blockchain nodes, requiring consideration of the quality of nodes.

There are two indicators for evaluating the quality of nodes: broadcast accuracy and storage. Broadcast accuracy refers to the probability that a node can broadcast a transaction correctly. Storage refers to the total blockchain size being stored in a node.

For a maximum broadcast accuracy, nodes are required to backup all blockchain data, called full nodes [1]. Because a full node stores all blockchain history transactions, it can verify the validity of any new transactions. Full nodes have a high degree of security. However, with the widespread use of blockchain and the rapid growth of transaction volumes, the storage bloating problem has emerged, which is the lack of storage room at each node [6] and the high bandwidth requirement for initial synchronization of the new participant [7]. The history transactions of the Bitcoin are increased by about $50GB$ each year, and the accumulated block data storage requires $190GB$ hard drive now. Synchronizing these data takes about two weeks for personal computers. The storage bloating problem has become one of the major issues that restrict the blockchain efficiency and application range. In particular, it is very challenging for a mobile system to afford a full node's storage and synchronization overhead.

In the past few years, many researchers have devoted themselves to the storage bloating problem on blockchain. The studies mainly focus on pruning after synchronization and improving the external protocols of full nodes. In 2008, Nakamoto [3] proposed to prune Merkle tree. In 2014, Back et al. [8] proposed Pegged Sidechains. Bitcoin Core [9] proposed to introduce Segregated Witness to reduce storage and increase scalability. Poon et al. [10] proposed Lightning Network. Besides pruning and external protocols of full nodes, a new type of nodes, Simplified Payment Verification (SPV) nodes, has been investigated in [3]. An SPV node only stores block headers. It can verify the validity of a transaction with the support of full nodes. Because lack block bodies, SPV nodes increase the workload of full nodes, lead to the occupation of bandwidth, and reduce the security of mobile systems. The current studies of SPV nodes mainly focus on improving node security. Frey et al. [11] proposed to use the distributed hash table to improve the payment security.

As can be seen from full nodes and SPV nodes, the key point of the storage bloating problem is the contradiction between high broadcast accuracy and low storage capacity. The broadcast accuracy largely depends on the most recent state, and the size of blockchain largely depends on the number of block bodies. In this paper, we propose to reduce the storage by downsampling (DS) block bodies. In order to describe the degree of downsampling more accurately, we introduce the concept of downsampling factor in digital signal processing. The downsampling factor (M) is usually an integer or a rational fraction greater than one, and usually divides the sampling rate to reduce the sampling rate of a signal [12]. In the blockchain, the M means only about $1/M$ of the block bodies will be stored.

Initially, we proposed random sampling (RS). It has good independence, however, due to the reduction of storage, there is a great loss of information entropy, which decreases broadcast accuracy. Then, a fast algorithm, named downsampling blockchain algorithm, is proposed. The downsampling blockchain algorithm predicts block information entropy, then selects appropriate block bodies.

Although the downsampling blockchain algorithm sacrifices broadcast accuracy to reduce storage, it only excludes block

This work was supported by NSAF under Grant U1530117. (Corresponding authors: Qin Huang and Shengli Zhang. Email: qhuang.smash@gmail.com, zsl@szu.edu.cn. Quan and Huang have the same contribution in this paper.)

bodies containing less information entropy and loses very little broadcast accuracy in comparison to random sampling. According to our analysis, the downsampling blockchain algorithm provides better broadcast accuracy storage ratio than random sampling with the same downsampling factor M.

In summary, the paper makes following contributions:

- It describes a new type of nodes employing downsampling algorithm for blockchain. Different from full nodes and SPV nodes [3] widely used, the proposed nodes are based on probability model considering blockchain properties such as sequence in time. The downsampling algorithm is used to select the "useful" block bodies introduced by the probability model in block body download.
- It describes the downsampling blockchain algorithm for the proposed nodes and gives an expression of the block body usage probability. The block body usage probability is seen as the survival function of the state. It gives a new perspective on the selection of the "useful" block bodies.
- It gives an expression of the storage efficiency as a function of types of nodes and storage. Our results show that there is a fundamental trade-off in nodes: the broadcast accuracy-storage trade-off, which determines nodes' storage efficiency. It also shows that the proposed nodes can broadcast transactions and has a better storage efficiency than the full nodes, SPV nodes and RS nodes.

This paper is organized as follows. Section II gives background knowledge. Section III presents the downsampling blockchain algorithm. Section IV gives performance, complexity and security analysis. Section V concludes this paper.

II. Background

Blockchain technology is a distributed ledger that cryptographically secures records of transactions [13]. It provides a secure distributed database. This database is sequential, open, scattered and Byzantine fault tolerance [14], which describes a system's ability to tolerate the failures of the Byzantine Generals' Problem [15].

Because the first application of the blockchain is Bitcoin, we use Bitcoin as an example to illustrate the blockchain.

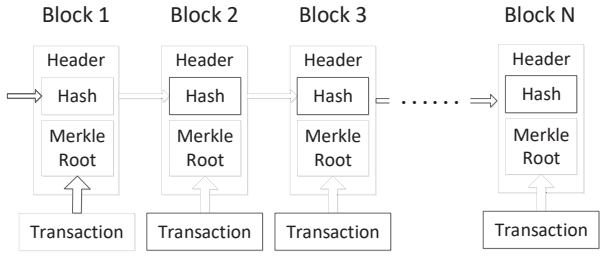

Fig. 1. Bitcoin blockchain structure.

Fig. 1 shows a typical Bitcoin blockchain structure. A complete Bitcoin blockchain consists of $N+1$ blocks sequentially stored from block 0 to block N. Each block includes a block header and a block body. The block header stores the previous block header hash, Merkle root, version, time, target threshold and nonce. With previous block header hash, different blocks are chained together sequentially. The block body contains transactions and the Merkle tree. The Merkle tree is generated by means of hash identifiers of transactions. These hash identifiers are paired, hashed, continuously repeated paired, and hashed until a unique hash is left. This hash is called the Merkle root of a Merkle tree. The Merkle root is stored in the block header. In this way, the block header and the block body are chained together to form a complete block.

A transaction includes inputs and outputs. Unspent Transaction Outputs (UTXOs) can be used as valid inputs by a new transaction. Only coinbase transaction can generate new Bitcoin without inputs. For other valid transactions, they should quote UTXOs from some existing transactions as inputs.

When the user sends a new transaction T to the network, the nodes receive T, then check it with Algorithm 1 in [1].

Algorithm 1 Transaction Verification

1: $T \notin \mathcal{S}_{processed}$.
2: T is valid, including the address is valid, the originator is the legal owner of the input address and $T_{in} \in \mathcal{S}_{UTXO}$.
3: $\sum T_{in} \geq \sum T_{out}$.

T_{in} is the input of T, T_{out} is the output of T, $\mathcal{S}_{processed}$ is the set of processed transactions and $\mathcal{S}_{UTXO}$ is the set of UTXOs. If the check is passed, T will be marked as a valid unconfirmed transaction and broadcast. In order to check all new transactions safely, nodes need to download and verify the blocks from the genesis block to the latest block.

After nodes verify and broadcast new transactions, there is a need for miners to write new transactions into the blockchain, that is, to make new transactions part of the new block. These miners generate the Merkle tree, package the transactions and Merkle tree into a block body, add the corresponding parameters of the block header, and perform calculations and other operations to make the newly packaged block satisfy the consensus mechanism and broadcast the new block.

Full nodes save the entire blockchain and can check the security of all transactions. But saving all blocks increases the amount of data the node needs to process. These data need to be downloaded, stored, verified, indexed and updated. This increases the nodes' storage overhead and reduces the efficiency of processing transactions.

Although there is another type of nodes, SPV nodes, which only save the block headers and can verify the validity and security of a transaction, its working ability depends on full nodes. SPV nodes increase the workload of full nodes, lead to the occupation of bandwidth resources, and also reduce the security of mobile systems.

Since UTXOs are very important for transaction verification and broadcast, some researchers have studied UTXOs. In 2013, Decker et al. [16] gave the information propagation in Bitcoin. Möser et al. [17] studied spend-time distributions in Bitcoin and in Monero and proposed sampling mixins. Delgado-Segura et al. [18] analyzed a variety of UTXOs

set's parameters. But these studies mainly focused on the relationship between UTXOs and time, did not establish the distribution of UTXOs on the blockchain and utilize the relationship between UTXOs and the most recent state.

In order to reduce the full nodes storage redundancy, the downsampling blockchain algorithm is proposed to obtain the most recent state as complete as possible while only storing a limited amount of data. Nodes adopting new algorithm can verify and broadcast transactions independently. Bitcoin is used as an example to illustrate the relationship between UTXOs and the most recent state, the distribution of UTXO on the blockchain, the mechanism of formation distribution and the proposed algorithm.

III. Downsampling Blockchain Algorithm

In blockchain, the most recent state is represented by a part of history transactions. And blockchain history transactions are stored in block bodies. It is important to select the block bodies to get as many parts of the most recent state as possible while only storing a limited amount of block bodies.

Definition 1. *Survival block is the number of blocks that states have been sustained.*

The survival block of the most recent state reflects the inherent rules of the most recent state. For a stable blockchain, although the distribution of the most recent state will change with the increase of the block height, the distribution of survival block is stable. In other words, the survival block is more universal significance. The distribution of the most recent state can be derived from the survival function of the most recent state's survival block.

In order to study the distribution of the most recent state's survival block, we use Bitcoin as an example. In Bitcoin, the UTXOs pool is used to verify the new transaction and represents the most recent state.

A. The survival block of UTXOs

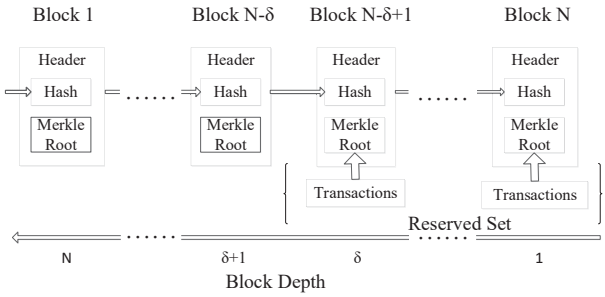

Fig. 2. Example of downsampling blockchain algorithm.

Here we use x to represent the survival block of UTXOs,

$$x = d_{produced} - d_{used}, \tag{1}$$

where d is the block depth as shown in Fig. 2. The depth of the latest block is 1. So $d_{produced}$ and d_{used} are the depth of the block where UTXOs were produced and used, respectively. The distribution of x is defined as $N(x)$.

If every UTXO is used randomly, independently and equally possible, UTXOs with less survival block will appear to be more. In this situation, the survival block of UTXOs conforms to the exponential distribution.

The probability density function of UTXOs' survival block

$$f(x) = \frac{N(x)}{\int_0^{+\infty} N(x)\mathrm{d}x}. \tag{2}$$

We count 224197 blocks and 847656 UTXOs of Bitcoin blockchain as samples on April 21, 2018, and fit the distribution of UTXOs' survival block to the function

$$N(x) = (115000e^{-2.005x}) + 38850e^{-0.1302x}, \tag{3}$$

where $R-square = 0.99$. The actual distribution of UTXOs' survival block and $N(x)$ are illustrated as Fig. 3.

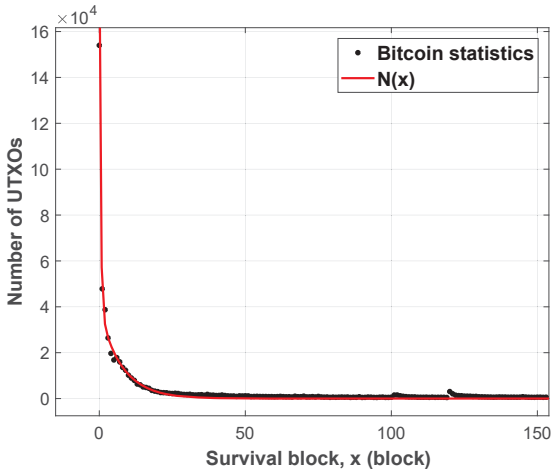

Fig. 3. UTXOs' survival block.

In Fig. 3, the horizontal coordinate is survival block, and the vertical coordinate is the number of UTXOs. It can be seen that actual UTXOs' survival block of Bitcoin and theory are consistent. UTXOs' survival block of Bitcoin is mainly distributed in less survival block areas.

According to the $N(x)$ obtained from our statistics, the probability of Bitcoin is

$$f(x) = (0.3233e^{-2.005x}) + 0.1092e^{-0.1302x}. \tag{4}$$

B. The distribution of UTXOs

From the distribution of UTXOs' survival block, we can find the newer transaction outputs are more likely to be UTXOs, and further get the distribution of UTXOs.

The cumulative function of $f(x)$ is

$$C(d) = \int_0^{d-1} f(x)\mathrm{d}x \tag{5}$$

which means that for a block with depth d, in all its transaction outputs, each output has been used with probability $C(d)$.

The probability of UTXOs in each block is

$$U(d) = 1 - C(d) \tag{6}$$

which is the survival function of UTXOs' survival block.

When the number of transaction outputs is uniform, we can get $u(d)$, the probability density function of UTXOs,

$$u(d) = \frac{U(d)}{\int_0^{+\infty} U(d)\mathrm{d}d}. \tag{7}$$

According to the $f(x)$ of obtained from our statistics,

$$u(d) = (0.0247e^{-2.005d}) + 0.1286e^{-0.1302d}. \tag{8}$$

When $d > 0$, $u(d)$ decreases.

C. Downsampling Blockchain Algorithm

Broadcast only concerns about whether the transaction is valid. If each block is viewed independently, the information entropy of different blocks at some point is

$$H(d) = E[-\log u(d)] = -u(d)\log u(d). \tag{9}$$

We can downsample the blockchain with Algorithm 2.

Algorithm 2 Downsampling Blockchain Algorithm

1: Determine the downsampling factor M.
2: Calculate the base δ of the reserved set $\mathcal{D}$ according to the downsampling factor M and the total block number d_{max}, where $\delta = d_{max}/M$.
3: Obtain the information entropy of block $H(d)$ according to the probability distribution of each block used by the new transaction $u(d)$.
4: Calculate the range of the reserved block depth according to the base δ of the reserved set $\mathcal{D}$ and the information entropy of block $H(d)$ to determine the reserved set $\mathcal{D}$, where $\mathcal{D}$ is the set of δ blocks with the largest $H(d)$.
5: A DS node stores all block headers of the entire blockchain and block bodies of reserved set $\mathcal{D}$.

When selecting M, the situation of nodes and network should be considered comprehensively. A detailed analysis of the influencing factors will be given in Section IV.

When a new T is received by a DS node, it will use block bodies of reserved set $\mathcal{D}$ to generate UTXOs pool $\mathcal{P}_{UTXO}$ and will perform the checks with Algorithm 3.

Algorithm 3 Transaction Verification of Downsampling Nodes

1: $T \notin \mathcal{S}_{processed}$.
2: T is valid, including the address is valid, the originator is the legal owner of the input address and $T_{in} \in \mathcal{P}_{UTXO}$.
3: $\sum T_{in} \geq \sum T_{out}$.

If all the checks are passed, T will be regarded as a valid unconfirmed transaction and broadcast.

It can be seen that the downsampling blockchain algorithm is concise. In the absence of a priori knowledge of actual block bodies, a method of selecting block bodies with high information entropy, that is, a greater likelihood of being used by a new transaction is given.

TABLE I
THE PROBABILITY OF BROADCAST OF DOWNSAMPLING NODES

Probability	Broadcast	Discard
Valid	$\frac{N_{su}}{N_u}$	$\frac{N_u - N_{su}}{N_u}$
Invalid	$\frac{N_{st} - N_{su}}{N_t - N_u}$	$\frac{N_t - N_u - (N_{st} - N_{su})}{N_t - N_u}$

IV. ANALYSIS AND SIMULATION RESULTS

A. Performance analysis

Let N_t and N_u be the number of the entire blockchain's transaction outputs and UTXOs, respectively. Let N_{st} and N_{su} be the number of a DS node's transaction outputs and UTXOs, respectively. In Table I, we analyze the probability of broadcast of DS nodes. When the transaction is valid or invalid, there are two options: broadcast transaction or discard transaction.

To illustrate the quality of a node, we define broadcast accuracy and storage efficiency.

Definition 2. *The broadcast accuracy, denoted by φ, is the probability that a node broadcasts valid transactions.*

Definition 3. *The storage efficiency, denoted by R, is broadcast accuracy storage data size ratio.*

For a DS node,

$$\varphi_{\mathcal{D}} = \frac{N_{su}}{N_u} \tag{10}$$

$$R_{\mathcal{D}} = \frac{\varphi_{\mathcal{D}}}{S_{\mathcal{D}}}, \tag{11}$$

where $\mathcal{D} \subseteq \{d_1, d_2, \ldots, d_\delta\}$ is reserved set, δ is the number of reserved block bodies, $\delta = d_{max}/M$ and $S_{\mathcal{D}}$ is the total block size of reserved set.

As the probability distribution of the block bodies used by the new transaction is non-uniform, we can get a better storage efficiency than full nodes, SPV nodes and RS nodes by downsampling.

We analyze the broadcast accuracy and mutual information in the case of uniform transaction outputs number and block size, multiplied by the correction function when non-uniform.

If transaction outputs number and block size are uniform,

$$\frac{N_{su}}{N_u} = \int_{\mathcal{D}} u(d)\mathrm{d}d \tag{12}$$

$$\frac{N_u - N_{su}}{N_u} = 1 - \int_{\mathcal{D}} u(d)\mathrm{d}d \tag{13}$$

$$\frac{N_{st} - N_{su}}{N_t - N_u} = \frac{\int_{\mathcal{D}}(1 - u(d))\mathrm{d}d}{\int_0^{d_{max}}(1 - u(d))\mathrm{d}d} \tag{14}$$

$$\frac{N_t - N_u - (N_{st} - N_{su})}{N_t - N_u} = 1 - \frac{\int_{\mathcal{D}}(1 - u(d))\mathrm{d}d}{\int_0^{d_{max}}(1 - u(d))\mathrm{d}d} \tag{15}$$

$$\varphi_{\mathcal{D}} = \int_{\mathcal{D}} u(d)\mathrm{d}d \tag{16}$$

$$R_{\mathcal{D}} = \frac{\int_{\mathcal{D}} u(d)\mathrm{d}d}{S_{avg}\delta}, \tag{17}$$

where S_{avg} is the average block size.

TABLE II
THE PROBABILITY OF BROADCAST OF FULL NODES, RANDOM SAMPLING NODES AND DOWNSAMPLING NODES

Probability	Broadcast & Valid	Discard & Valid	Broadcast & Invalid	Discard & Invalid	R
Full Nodes	1	-	-	-	6.049e-6
RS Nodes	0.001	0.999	0.001	0.999	6.049e-6
DS Nodes	1.000	0.000	0.001	0.999	0.006

In Table II, we give the probability of broadcast of full nodes, RS nodes and DS nodes at $M = 1024$. Here we take $S_{avg} = 0.32$MB and $d_{max} = 519000$, which are consistent with the Bitcoin blockchain parameters on April 19, 2018. The difference is that the transaction outputs number and block size of the Bitcoin blockchain are non-uniform. When analyzing Bitcoin and other blockchains, we only need to multiply results by the correction function, so the analysis of uniform transaction outputs number and block size is representative. The examples in the later analysis use the same parameters.

It can be seen that when the M is less than 1024, DS nodes have a similar broadcast accuracy to full nodes, and have better broadcast accuracy than RS nodes. The DS nodes also have higher storage efficiency than full nodes and RS nodes.

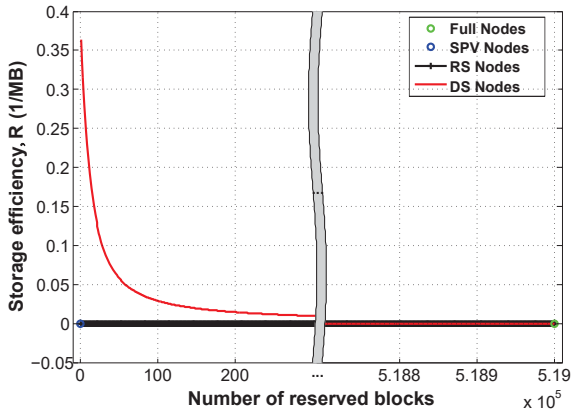

Fig. 4. Storage efficiency of full nodes, Simplified Payment Verification nodes, random sampling nodes, and downsampling nodes.

In Fig. 4, we compare R of full nodes, SPV nodes, RS nodes and DS nodes. The horizontal coordinate is number of reserved blocks, and the vertical coordinate is storage efficiency R. It shows that DS nodes have a better storage efficiency than full nodes, SPV nodes and RS nodes. For other blockchains, the analysis results are similar. Because the broadcast accuracy under different R is different, selecting a proper R can improve the broadcast accuracy and suppress the error rate.

When choosing the downsampling factor, besides the storage efficiency and broadcast accuracy, it is also necessary to consider the ability of DS nodes to request full nodes to correct broadcast errors. Under different network conditions, this ability is different. In the case where full nodes are more reliable, the ability of full nodes to help DS nodes to correct errors is stronger; in another case where full nodes are less reliable, this ability is weaker.

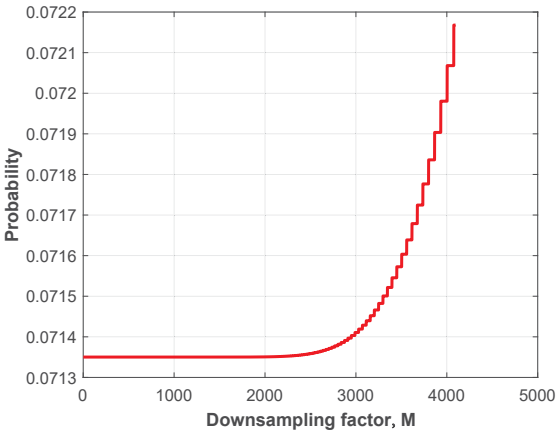

Fig. 5. Probability of requiring full nodes support when verifying transactions at different downsampling factor.

In Fig. 5, we analysis probability that DS nodes require full nodes support when verifying transactions at different downsampling factor. The horizontal coordinate is downsampling factor M, and the vertical coordinate is probability. It shows that with M increases, the probability increases. Therefore, when selecting M, storage efficiency, broadcast accuracy and the ability of DS nodes to request full nodes to correct broadcast errors should be considered comprehensively.

Then we compare the difference in mutual information between RS nodes and DS nodes. If UTXOs are randomly sampled, $N_{su}/N_{st} = N_u/N_t$ when we do not have a priori information other than distribution. The mutual information is

$$\begin{aligned} I(EC;RS) &= H(EC) - H(EC|RS) \\ &= -\tfrac{N_u}{N_t}\log\tfrac{N_u}{N_t} - (-\tfrac{N_{su}}{N_{st}}\log\tfrac{N_{su}}{N_{st}}) = 0, \end{aligned} \tag{18}$$

where EC represents the entire blockchain.

If UTXOs are non-uniform distribution, $N_{su}/N_{st} \geq N_u/N_t$ when using the downsampling blockchain algorithm. The mutual information is

$$\begin{aligned} I(EC;DS) &= H(EC) - H(EC|DS) \\ &= -\tfrac{N_u}{N_t}\log\tfrac{N_u}{N_t} - (-\tfrac{N_{su}}{N_{st}}\log\tfrac{N_{su}}{N_{st}}) \geq 0. \end{aligned} \tag{19}$$

It illustrates that the downsampling blockchain algorithm could have higher mutual information than random sampling, which is also consistent with the previous accuracy analysis results.

B. Complexity analysis

The selection of blocks in the downsampling blockchain algorithm is a one-time job. Although the block depth is constantly changing, the distribution of the reserved block depth is stable. Thus, the complexity of DS nodes is determined by the number of downloaded block bodies. In Table III, we give the average downloaded blocks bodies number of full nodes, RS nodes, and DS nodes.

It is shown that the downsampling blockchain algorithm with appropriate downsampling factor only has about $1/M$ average number of downloaded blocks of full nodes.

TABLE III
AVERAGE DOWNLOADED BLOCKS NUMBER OF FULL NODES, RANDOM SAMPLING NODES, AND DOWNSAMPLING NODES

M	1	16	256	1024	4096
Full Nodes	519000	-	-	-	-
RS Nodes	519000	32438	2028	507	127
DS Nodes	519000	32438	2028	507	127

TABLE IV
THE EXPECTATION OF DECEIVED NODES OF FULL NODES, RANDOM SAMPLING NODES AND DOWNSAMPLING NODES

M	1	16	256	1024	4096
Full Nodes	0	-	-	-	-
RS Nodes	0	0.001	0.004	0.016	0.067
DS Nodes	0	1.000	0.032	0.008	0.002

C. Security analysis

For payment, a new transaction usually needs 6 confirmations. Since a DS node stores all block headers, the lower bound of security is same as the SPV node. Because a DS node loses part of the information, the upper bound of security is same as the full node. The security of DS nodes can approach full nodes with appropriate M.

For broadcast, we can obtain the security by introducing randomness, like randomly reserving part of the block bodies of the non-reserved set. Assuming that the probability of successful cheating is p, the probability of successful cheating n times is p^n if the block bodies stored in the two nodes are independent. It can be seen that the possibility of diffusion is rapidly decreasing.

If the maximum number of connections for every node is m, the largest broadcast of a fake transaction is a tree that each parent node has m child nodes. The biggest expectation of deceived nodes (DC nodes) is

$$E[DC] = \sum_{l=1}^{+\infty} p^l m^l = \sum_{l=1}^{+\infty} (pm)^l. \tag{20}$$

When p is smaller than $1/m$, the expectation is convergence,

$$E[DC] = \frac{pm}{1-pm}. \tag{21}$$

Because of the sequential feature of blockchain, it can be determined whether a transaction output is UTXO if all blocks after that transaction output are known. The expectation of DC nodes can be further reduced with this feature. For example, if a node reserves the continuous latest blocks in blockchain, the expectation of DC nodes could approach zero. In Table IV, we analyze the expectation of the DC nodes of full nodes, RS nodes and DS nodes at different M in the broadcast.

It can be seen that when the downsampling factor is greater than 1024, DS Nodes and full nodes have a similar expectation of DC nodes in the broadcast.

V. CONCLUSIONS

This paper downsampled blockchain M times to reduce the storage of nodes. To improve the storage efficiency, the proposed downsampling blockchain algorithm reserved block bodies with high information entropy. Performance analysis showed that the downsampling blockchain algorithm can perform better than full nodes, SPV nodes and RS nodes on storage efficiency. Simulations showed that the downsampling blockchain algorithm with appropriate downsampling factor M can achieve almost the same broadcast accuracy as full nodes with only at most $1/M$ storage of full nodes.

REFERENCES

[1] A. M. Antonopoulos, *Mastering Bitcoin: unlocking digital cryptocurrencies*. O'Reilly Media, Inc., 2014.

[2] G. D'Angelo, S. Ferretti, and M. Marzolla, "A blockchain-based flight data recorder for cloud accountability," in *1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems (CryBlock'18)*. Munich, Germany: ACM, June 2018, pp. 93–98.

[3] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," [Online]. Available: https://bitcoin.org/bitcoin.pdf, 2008.

[4] Z. Gao, L. Xu, G. Turner, B. Patel, N. Diallo, L. Chen, and W. Shi, "Blockchain-based identity management with mobile device," in *1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems (CryBlock'18)*. Munich, Germany: ACM, June 2018, pp. 66–70.

[5] A. Ranchal Pedrosa and G. Pau, "Chargeitup: On blockchain-based technologies for autonomous vehicles," in *1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems (CryBlock'18)*. Munich, Germany: ACM, June 2018, pp. 87–92.

[6] K. Wang, J. Mi, C. Xu, Q. Zhu, L. Shu, and D. J. Deng, "Real-time load reduction in multimedia big data for mobile internet," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 12, no. 5s, pp. 76:1–76:20, 2016.

[7] M. Dai, S. Zhang, H. Wang, and S. Jin, "A low storage room requirement framework for distributed ledger in blockchain," *IEEE Access*, vol. 6, pp. 22 970–22 975, 2018.

[8] A. Back, M. Corallo, L. Dashjr, M. Friedenbach, G. Maxwell, A. Miller, A. Poelstra, J. Timn, and P. Wuille, "Enabling blockchain innovations with pegged sidechains," [Online]. Available: https://blockstream.com/sidechains.pdf, 2014.

[9] B. Core, "Segregated witness benefits," [Online]. Available: https://bitcoincore.org/en/2016/01/26/segwit-benefits/, 2016.

[10] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," [Online]. Available: https://lightning.network/lightning-network-paper.pdf, 2016.

[11] D. Frey, M. X. Makkes, P.-L. Roman, F. Taïani, and S. Voulgaris, "Bringing secure bitcoin transactions to your smartphone," in *Proceedings of the 15th International Workshop on Adaptive and Reflective Middleware*. Trento, Italy: ACM, December 2016, pp. 3:1–3:6.

[12] A. D. Poularikas, *Handbook of Formulas and Tables for Signal Processing (1 ed.)*. CRC Press, 1998.

[13] A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. Princeton University Press, 2016.

[14] O. J. A. Pinno, A. Grgio, and L. C. E. D. Bona, "Controlchain: Blockchain as a central enabler for access control authorizations in the iot," in *IEEE GLOBECOM*, 2017.

[15] L. Lamport, R. E. Shostak, and M. C. Pease, "The byzantine generals problem," *ACM Transactions on Programming Languages and Systems*, vol. 4, no. 3, pp. 382–401, 1982.

[16] C. Decker and R. Wattenhofer, "Information propagation in the bitcoin network," in *IEEE Thirteenth International Conference on Peer-To-Peer Computing (P2P'13)*. Trento, Italy: IEEE, September 2013, pp. 1–10.

[17] M. Möser, K. Soska, E. Heilman, K. Lee, H. Heffan, S. Srivastava, K. Hogan, J. Hennessey, A. Miller, A. Narayanan *et al.*, "An empirical analysis of traceability in the monero blockchain," *Proceedings on Privacy Enhancing Technologies*, vol. 2018, no. 3, pp. 143–163, 2018.

[18] S. Delgado-Segura, C. Pérez-Solà, G. Navarro-Arribas, and J. Herrera-Joancomartı, "Analysis of the bitcoin utxo set," in *Proceedings of the 5th Workshop on Bitcoin and Blockchain Research*. Porta Blancu, Curaçao: Springer Lecture Notes in Computer Science, March 2018.