Tree Representation, Growth Rate of Blockchain and Reward Allocation in Ethereum With Multiple Mining Pools

Quan-Lin Li, Yan-Xia Chang, and Chi Zhang

***Abstract*—It is interesting but difficult and challenging to study Ethereum with multiple mining pools. One of the main difficulties comes from not only how to represent such a general tree with multiple block branches (or sub-chains) related to the multiple mining pools, but also how to analyze a multi-dimensional stochastic system due to the mining competition among the multiple mining pools. In this paper, we first set up a mathematical representation for the tree with multiple block branches. Then we provide a block classification of Ethereum: Regular blocks (in the main chain), orphan blocks, uncle blocks, stale blocks, and nephew blocks, and give some key ratios and probabilities of generating the different types of blocks by applying the law of large numbers. Based on this, we further discuss the growth rate of blockchain and the reward allocation among the multiple mining pools through applying the renewal reward theorem. Finally, we use some simulation experiments to verify our theoretical results, and show that the approximate computation approaches developed, such as the key ratios and probabilities, the long-term growth rate of blockchain, and the long-term reward allocation (rate) among the multiple mining pools, can have a faster convergence. Therefore, we provide a powerful tool for observing and understanding the influence of the selfish mining attacks on the performance of Ethereum with multiple mining pools. We believe that the methodology and results developed in this paper will shed light on the study of Ethereum with multiple mining pools, such that a series of promising research can be inspired potentially.**

***Index Terms*—Ethereum, selfish mining, multiple mining pools, tree representation, growth rate of blockchain, reward allocation, the law of large numbers, renewal reward theory.**

I. Introduction

BITCOIN and blockchain have opened a new era of automatically processing and storing transactions since the pioneering work by Nakamoto [26]. The transactions can be automatically processed in the form of Bitcoin scripts through a data structure of Merkle tree in a P2P network. However, the Bitcoin script language is not Turing-complete owing to robust concerns. On the other hand, Ethereum can break such a limitation by introducing Ethereum Virtual Machine featuring smart contract functionality, e.g., see [6], [7]. Note that Bitcoin and Ethereum are the two largest and most popular blockchain-based cryptocurrencies in the world. Ethereum has gained great attention in the development of blockchain technology. Interested readers are referred to Ethereum survey papers, for example, Wood [40], Vujičić *et al.* [35], Angelo and Salzer [10], Mohammed *et al.* [25]; Ethereum system security by Chen *et al.* [9] and Praitheeshan *et al.* [28]; and smart contracts by Wang *et al.* [36], Dika and Nowostawski [11], Wang *et al.* [37] and Atzei *et al.* [1]. At the same time, Ethereum has been developed into many practical applications from industry and academia, e.g., see sharing economy by Bogner *et al.* [5], healthcare by Sookhak *et al.* [34], IoT and logistics by Augusto *et al.* [2], emergency service by Aung and Tantidham [3], decentralized marketplace by Ranganthan *et al.* [29], and so forth.

Bitcoin and Ethereum applied the most widely used consensus mechanism: Proof-of-Work (PoW). In a PoW blockchain system, many miners (or mining pools) competitively mine each block which is generated by means of finding a nonce through solving a cryptographic puzzle of using all the foregoing information of that blockchain in front of this block and then peg the block with the nonce to the blockchain. See Li *et al.* [20], [21] for more details. For the PoW blockchain system with two mining pools (honest and dishonest), Eyal and Sirer [12] introduced the selfish mining attacks and applied a simple Markov chain to study a few advantages of selfish mining. Li *et al.* [21] proposed the two-block leading competitive criterion and set up a pyramid Markov (reward) process to analyze the operations efficiency and economic benefit of the blockchain selfish mining system. It is seen from [12] and [21] that the mining competition between the two mining pools can be described as a comparison of lengths of the two block branches, and the dishonest mining pool develops a selfish mining attack policy, see [21, Figs. 2 and 4]. Clearly, the longest one of the two block branches is called the main chain, which is pegged on the blockchain; while another block branch is regarded as the chain of orphan blocks, which cannot be connected to the blockchain and is returned to the transaction pool for reprocessing. Thus the orphan blocks generate a lot of waste of computing resources. Li *et al.* [21] applied the pyramid Markov (reward) process to give a detailed analysis of the orphan blocks and their waste of computing resources. Following the two block branches corresponding to the two

Manuscript received 24 March 2022; accepted 27 July 2022. Date of publication 1 August 2022; date of current version 7 March 2023. Quan-Lin Li was supported by the National Natural Science Foundation of China under grants No. 71671158 and 71932002. The associate editor coordinating the review of this article and approving it for publication was A. Santos. *(Corresponding author: Yan-Xia Chang.)*

The authors are with the School of Economics and Management, Beijing University of Technology, Beijing 100124, China (e-mail: changyanxia@emails.bjut.edu.cn).

Digital Object Identifier 10.1109/TNSM.2022.3195292

mining pools, this paper analyzes the PoW blockchain system with multiple mining pools, and finds a general tree with multiple block branches where the dishonest block branches can fork at different positions of the honest block branch (see Section III). Furthermore, under the two-block leading competitive criterion, this paper provides a mathematical representation of the general tree with multiple block branches, and shows that the main chain in the general tree can be determined easily by means of the longest chain principle. Note that here our general tree is different from that of the GHOST protocol given in Sompolinsky and Zohar [32], [33], since our general tree is directly built in the competing process of multiple mining pools, and the longest chain (main chain) can be easily determined by using the mathematical representation of the general tree with multiple block branches.

So far, only a few studies have been done on the PoW blockchain system with multiple mining pools. Important examples include the two different classes: (a) Simulation, and (b) extending the Markov chain method of Eyal and Sirer [12]. For (a) simulation, since the blockchain system with multiple mining pools is very complicated, the simulation method becomes effective and feasible. Leelavimolsilp *et al.* [17] used simulation to provide a preliminary investigation on the selfish mining strategy adopted by multiple miners, and analyzed the relative reward, the power threshold of selfish miners, and the safety level of the Bitcoin system. Under the assumptions of [17], Leelavimolsilp *et al.* [18] further studied the effectiveness of the selfish mining strategy. For (b) extending the Markov chain method of Eyal and Sirer [12], readers are referred to, for example, Liu *et al.* [22], Marmolejo-Cossío *et al.* [24], Bai *et al.* [4], Jain [15] and Xia *et al.* [41]. Note that Li *et al.* [21] indicated that the Markov chain given in Eyal and Sirer [12] has some deficiencies and defects compared to the theory of Markov processes.

The GHOST protocol was first introduced by Sompolinsky and Zohar [32], [33] in order to improve the security and throughput of the Bitcoin system by using the heaviest chain principle in a tree. On the other hand, Ethereum implements a simplified GHOST protocol which refers to orphan blocks when observing which chain is the longest. In this case, the referenced blocks are called the uncle blocks while the referencing blocks are called the nephew blocks. For the uncle and nephew blocks, the Ethereum system with multiple mining pools will be faced with two basic challenges. The first challenge is how to set up a tree structure with multiple block branches and forked positions, which expresses the competition process of multiple mining pools. The second challenge is how to design the rewards of uncle blocks and nephew blocks, which are used to increase the mining enthusiasm of the multiple mining pools, especially when those mining pools cannot access the main chain. For the two challenging problems, so far there has not been a clear answer or better research yet. This motivates us in this paper to explore both setting up the tree structure and designing the rewards of the uncle and nephew blocks. By finding the mathematical representation of a general Ethereum tree, this paper applies the law of large numbers and the reward renewal theorem to make some key and important progress in the study of Ethereum systems with multiple mining pools.

There have been a few works on the uncle and nephew blocks and their reward design in Ethereum up to now. On the basis of the previous short work [43], Zhang *et al.* [44] analyzed the benefits of selfish mining in Ethereum, and chose the maximum, 7/8 units as the uncle block reward, and 1/32 units as the nephew block reward. Lerner [19] found that the uncle block strategy may cause the deliberate increase in the supply of Ethereum, thus it indirectly reduces the value of Ethereum. Ritz and Zugenmaier [30] set up a Monte Carlo simulation platform to quantify how the uncle blocks affect the probability of selfish mining. Chang *et al.* [8] introduced the uncle block attacks to discuss the incentive compatibility among the different attacks. Werner *et al.* [39] formally reconstructed a Sybil attack to exploit the uncle block distribution policy in a queue-based mining pool. Chang *et al.* [8] and Werner *et al.* [39] provided the simulation analysis for the uncle blocks in Ethereum. Zhang [42] developed a Markov decision process model to analyze the profitability and threshold of the three-player attacks. In addition, the uncle block mechanism can improve the security of Ethereum systems with multiple mining pools. For the details of the two (honest and dishonest) mining pools, interested readers are referred to Feng and Niu [13], Grunspan and Pérez-Marco [14], Kang *et al.* [16], Liu *et al.* [23] and Wang *et al.* [38]. Compared with the studies above, this paper proposes a new method with two consecutive rounds of mining competition for analyzing the uncle and nephew blocks, in which we set up a basic relation among the uncle and nephew blocks (see Fig. 11) so that the rewards of the uncle and nephew blocks can be estimated easily (see Sections IV and VIII). Therefore, one key finding of this paper is to reveal that the uncle blocks and the nephew blocks must appear in two different rounds of mining competition. This is crucial and interesting in the research of Ethereum.

Based on the above analysis, we summarize the main contributions of this paper as follows:

1. Under the two-block leading competitive criterion, we describe the Ethereum system with one honest mining pool and multiple dishonest mining pools, set up a general tree with multiple block branches where the multiple dishonest block branches can fork at different positions of the one honest block branch, and provide a mathematical representation of the general tree. By using the mathematical representation of tree, we provide an effective method to easily determine the main chain by means of the longest chain principle. (See Sections II and III)
2. By using the mathematical representation of tree, we propose a two-stage mechanism to find the uncle and nephew blocks and then design the uncle block and nephew block rewards in two consecutive rounds of mining competition. (See Sections IV and VIII)
3. We apply the law of large numbers to study some key probabilities in the Ethereum system with multiple mining pools, and define and compute some key ratios: The main chain ratio, the orphan block ratio, the uncle block ratio, the stale block ratio, and the chain quality. Note that the key ratios are necessary and useful in the security

analysis of Ethereum systems with multiple mining pools. (See Sections V and VI)

4. On the one hand, we provide expression for the long-term growth rate of blockchain by using the renewal reward theory, which is one of the most important indicators for the Ethereum system, where the growth rate of blockchain is the block number in all the main chains increasing per unit time. (See Section VII). On the other hand, once the uncle and nephew block rewards are determined, we provide expressions both for the long-term reward allocation and for the long-term reward allocation rate to each mining pool by using the renewal reward theory. (See Sections VIII and IX)
5. We use some simulation experiments to discuss the Ethereum system with one honest mining pool and two dishonest mining pools, verify how the key probabilities of Ethereum are obtained approximately by using the law of large numbers, and analyze the performance measures of the Ethereum system, for example, the long-term growth rate of blockchain, the long-term reward allocation, and the long-term reward allocation rate to each mining pool. We show that the approximative computation of the performance measures of the Ethereum system can have a faster convergence. (See Section X)

The remainder of this paper is organized as follows. Section II describes the Ethereum system with multiple mining pools and provides the mathematical representation of a general tree with multiple block branches. Section III gives some examples with one honest mining pool and two dishonest mining pools for analyzing the mathematical representation of the general tree. Section IV introduces a classification of blocks, gives some conditions under which the orphan block can become an uncle block, and provides a two-stage mechanism to determine the uncle and nephew blocks. Section V studies some key probabilities of Ethereum by using the law of large numbers. Section VI defines some key ratios for the general tree with multiple block branches. Section VII applies the renewal reward theory to discuss the long-term growth rate of blockchain. Section VIII provides a long-term reward allocation to each mining pool by means of the renewal reward theory. Section IX applies the renewal reward theory to study the reward allocation rates among the multiple mining pools. Section X conducts simulation experiments to analyze the performance measures of the Ethereum system. Section XI provides some concluding remarks.

II. Model Description

In this section, we describe the Ethereum system with multiple mining pools. The mining competition among the multiple mining pools directly leads to a general tree forked at the different positions of the honest block branch by the multiple dishonest block branches. Based on this, one of our key findings is to provide a mathematical representation for such a tree with multiple block branches. In addition, we introduce some mathematical notations used in our later study.

In a PoW Ethereum with multiple mining pools, the mining competition among the multiple mining pools is a main way to build the blockchain by means of solving the PoW mathematical puzzles. In the process of mining competition, some transactions are first packaged as a block with finite sizes (see Li *et al.* [20]), and then the block needs a nonce which is found through solving the PoW mathematical puzzle among the multiple mining pools, in which the success of one mining pool is based on its mining power in proportion to the total mining power. Once the nonce is solved by one mining pool and is written into the block, then the block is successfully mined so that it can be pegged to the block branch of this mining pool.

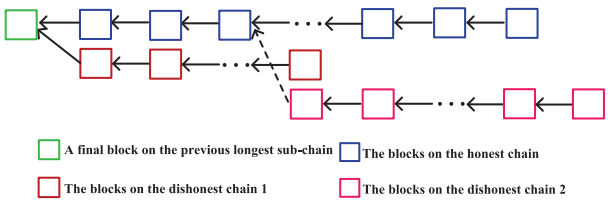


Fig. 1. A tree with multiple sub-chains.

In the multiple mining pools, when the last round of mining competition ends, it is easy to see the honest mining pool is major, so that it can begin to mine from the final block on the previous sub-chain. However, the first block mined by a dishonest mining pool can be connected to any position of the honest sub-chain, including the final block in the last round of mining competition, while it cannot be connected to one dishonest sub-chain mined by the other dishonest mining pools. Based on this, we describe the tree structure with multiple sub-chains. See Fig. 1 for more details. Clearly, it is the key to provide a mathematical representation for the tree given in Fig. 1.

Note that the two-block leading competitive criterion was first proposed in Li *et al.* [21] to discuss the blockchain selfish mining with two mining pools. In this paper, such a two-block leading competitive criterion is further extended to selfish mining with multiple mining pools. Based on this, we can set up the termination rule of mining competition among the multiple mining pools.

Now, we provide model description for the PoW Ethereum system with multiple mining pools as follows:

(1) *Structure of PoW Ethereum system:* There are $m+1$ mining pools in the PoW Ethereum system, where there are one honest mining pool and m dishonest mining pools for $m \geq 1$. For simplicity, we regard the blocks or sub-chain mined by the honest mining pool as the honest blocks or sub-chain, and the blocks or sub-chains mined by dishonest mining pools as the dishonest blocks or sub-chains.

(2) *Honest mining pool:* The honest mining pool in Ethereum follows the PoW protocol with the two-block leading competitive criterion (also see **4-a** below). Once a block is mined by the honest mining pool, the complete information of this block is immediately broadcasted to the entire P2P network so that each dishonest mining pool can monitor its block information. Thus, all the dishonest mining pools can learn about the length of the honest sub-chain in a timely manner.

(3) *Dishonest mining pools:* The dishonest mining pools in Ethereum follow the PoW protocol with the two-block leading competitive criterion (also see **4-b** below), and they can carry out various selfish mining attacks. That is, when the dishonest mining pools launch selfish attacks, the blocks mined by each dishonest mining pool may not be immediately broadcasted to the entire P2P network. In this case, only a part of the block branch mined by one dishonest mining pool may be pegged on the blockchain when this dishonest mining pool can set up the main chain; while another part of the block branch is left to keep the mining competitive advantage of this dishonest mining pool in the next round of mining competition. Thus, the honest mining pool and the other dishonest mining pools cannot know the accurate information of blocks mined by this dishonest mining pool.

(4) *The two-block leading competitive criterion and its modification:*

(4-a) If the honest mining pool takes the lead in mining and as long as the number of blocks mined by the honest mining pool is 2 blocks ahead of the second-longest sub-chain mined by one of the dishonest mining pools, then the round of mining competition terminates immediately, and the longest sub-chain mined by the honest mining pool becomes the main chain which is pegged onto the blockchain, while all the other sub-chains mined by the dishonest mining pools become the chains of orphan blocks, all of which are returned to the transaction pool without adding any new transaction fee.

(4-b) If one dishonest mining pool takes the lead in mining and as long as the number of blocks mined by this dishonest pool is at least 2 blocks ahead of the second-longest sub-chain among the other mining pools, then the sub-chain mined by this dishonest mining pool becomes the main chain, while this dishonest pool may release only a part of the main chain into the blockchain under a basic condition that the part of the main chain is still at least 2 blocks ahead of the second-longest sub-chain, and another part of the main chain is reserved for the next round of mining competition in order to keep the mining competitive advantage of this dishonest mining pool. Once the part of the main chain begins to peg onto the blockchain, the round of mining competition terminates immediately, and all the other sub-chains become the chains of orphan blocks, all of which are returned to the transaction pool without adding any new transaction fee.

(5) *A key mathematical representation for the $m+1$ sub-chains of the tree:*

Let $L = \{L_0, L_1, L_2, \ldots, L_m\}$ denote the tree with $m+1$ sub-chains mined by one honest mining pool and m dishonest mining pools, where L_0 is the sub-chain mined by the honest mining pool and L_i is the sub-chain mined by the ith dishonest mining pool for $i = 1, 2, \ldots, m$. We write

$$L_0 = \{H_1, H_2, \ldots, H_v\},$$

where H_l denotes the lth block mined by the honest mining pool for $l = 1, 2, \ldots, v$. That is, there are v blocks in the sub-chain mined by the honest mining pool. Similarly, we write

$$L_i = \left\{H_1, H_2, \ldots, H_{k_i}; D^{(i)}_{k_i,1}, D^{(i)}_{k_i,2}, \ldots, D^{(i)}_{k_i,l_i}\right\},$$

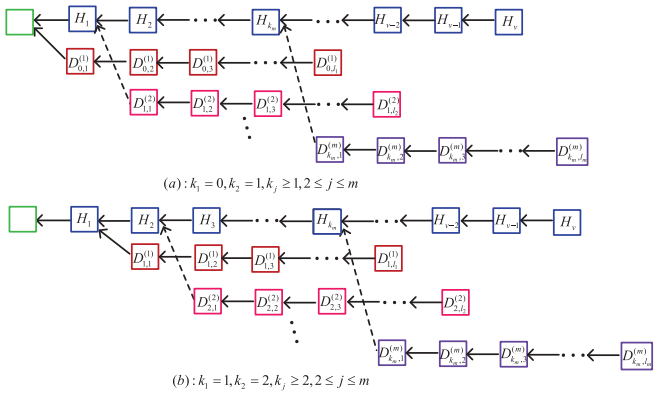


Fig. 2. A mathematical representation for the tree with $m+1$ sub-chains.

where k_i represents the number of honest blocks which have been mined by the honest mining pool before the ith dishonest mining pool begins to fork after the block H_{k_i}, such that a new sub-chain with l_i blocks is mined by the ith dishonest mining pool, where $D^{(i)}_{k_i,l}$ denotes the lth block mined by the ith dishonest mining pool for $l = 1, 2, \ldots, l_i$,. Fig. 2 provides a more intuitive understanding for the $m+1$ sub-chains of the tree.

(6) *A mathematical expression for the mining terminative rules:* From $L_0 = \{H_1, H_2, \ldots, H_v\}$ and $L_i = \{H_1, H_2, \ldots, H_{k_i}; D^{(i)}_{k_i,1}, D^{(i)}_{k_i,2}, \ldots, D^{(i)}_{k_i,l_i}\}$ for $i = 1, 2, \ldots, m$, we write

$$\Omega = \{v\} \cup \{(k_1, l_1), (k_2, l_2), \ldots, (k_m, l_m)\}.$$

Further, we write

$$S = \{v, k_1 + l_1, k_2 + l_2, \ldots, k_m + l_m\},$$

where, v is the length of sub-chain mined by the honest mining pool, and $k_i + l_i$ is the generalized sub-chain mined by the ith dishonest mining pool for $i = 1, 2, \ldots, m$. Based on this, the elements of the set S are sorted from the largest to the smallest as follows:

$$\omega_1 \geq \omega_2 \geq \cdots \geq \omega_m \geq \omega_{m+1}.$$

(6-a) If ω_1 comes from the honest mining pool and $\omega_1 - \omega_2 = 2$, then this round of mining competition terminates immediately; the sub-chain mined by the honest mining pool is the main chain and is pegged onto the blockchain, while all the sub-chains mined by the m dishonest mining pools are returned to the transaction pool.

(6-b) If ω_1 comes from the ith dishonest mining pool and $\omega_1 - \omega_2 \geq 2$, then this round of mining competition may terminate, the sub-chain mined by the ith dishonest mining pool is the main chain, and the part with φ_i blocks for $\omega_2 + 2 \leq \varphi_i + k_i \leq \omega_1$ of the main chain mined by the ith dishonest mining pool begins to peg onto the blockchain, while another part with $\omega_1 - \varphi_i - k_i$ blocks of the main chain is reserved for the next round of mining competition in order to keep the mining competitive advantage of the ith dishonest mining pool. In this case, all the sub-chains mined by the other (honest and dishonest) mining pools are returned to the transaction pool.

(6-c) If $0 \leq \omega_1 - \omega_2 \leq 1$, then this round of mining competition cannot terminate and the mining pools continue to mine until the two-block leading competitive criterion is satisfied.

(7) *The mining rewards:* When a round of mining competition terminates, the main chain is pegged onto the blockchain, and all the other sub-chains become orphan blocks which are returned to the transaction pool, waiting for the next round of mining competition.

(7-a) If the main chain comes from the honest mining pool and $L_0 = \{H_1, H_2, \ldots, H_v\}$, then the honest mining pool obtains the rewards of v blocks.

(7-b) If the main chain comes from the ith dishonest mining pool, and

$$L_i = \left\{H_1, H_2, \ldots, H_{k_i}; D^{(i)}_{k_i,1}, D^{(i)}_{k_i,2}, \ldots, D^{(i)}_{k_i,l_i}\right\},$$

then the honest mining pool obtains the rewards of k_i blocks. Let $\omega_1 = k_i + l_i$, then the ith dishonest mining pool obtains the rewards of φ_i blocks for $\omega_2 + 2 \leq \varphi_i + k_i \leq \omega_1$. Note that the $\omega_1 - \varphi_i - k_i$ blocks of the main chain cannot be pegged onto the blockchain, thus they cannot lead to any reward for the ith dishonest mining pool.

In addition, the uncle block rewards and associated reference rewards will be assumed and discussed in Section IV.

III. Examples for the Tree Representation

This section provides some examples to analyze the mathematical representation of the general tree with one honest mining pool and two dishonest mining pools. Note that the mathematical representation of tree plays a key role in the study of PoW Ethereum systems with multiple mining pools.

In the PoW Ethereum system with multiple mining pools, the tree with multiple sub-chains expresses the mining competition process among the honest mining pool and the m dishonest mining pools. The following theorem provides an essential feature of the tree with multiple sub-chains.

Theorem 1: In a round of mining competition and from the two-block leading competitive criterion, we have

(a) if the sub-chain of the ith dishonest mining pool is the first forked sub-chain among all the m dishonest mining pools, then either $k_i = 0$ or $k_i = 1$; and

(b) if the sub-chain of the ith dishonest mining pool is not the first forked sub-chain among all the m dishonest mining pools, then $k_i \geq 1$.

Proof: (a) We provide the proof by contradiction. We assume that the sub-chain of the ith dishonest mining pool is the first forked sub-chain among all the m dishonest mining pools, and $k_i \geq 2$. In this case, by using the two-block leading competitive criterion, it is easy to see that the honest mining pool is at least 2 blocks ahead of the second-longest sub-chain among the other mining pools, the honest mining pool immediately terminates this round of mining competition. Obviously, it is impossible that the ith dishonest mining pool can fork at $k_i \geq 2$. Therefore, this gives that either $k_i = 0$ or $k_i = 1$.

(b) If the sub-chain of the ith dishonest mining pool is not the first forked sub-chain among all the m dishonest mining pools, then there must be a jth dishonest mining pool who is the first one to fork at the tree for $j \neq i$. By using (a), either $k_j = 0$ or $k_j = 1$. In this case, our discussion has two different cases as follows:

(i) When $k_j = 0$, since the sub-chain of the ith dishonest mining pool is not the first forked sub-chain among all the m dishonest mining pools, it is easy to see that the ith dishonest mining pool can fork at a later position than the dishonest pool j, this gives $k_i > k_j = 0$, i.e., $k_i \geq 1$. In this case, $k_i \geq 1$.

(ii) When $k_j = 1$, since the sub-chain of the i th dishonest mining pool is not the first forked sub-chain among all the m dishonest mining pools, it is easy to see that the ith dishonest mining pool can fork at a later position than the dishonest pool j, this gives $k_i > k_j = 1$, i.e., $k_i \geq 2$. In this case, $k_i \geq 2$.

From the above two cases, we get that $k_i \geq 1$. This completes the proof. ■

In the remainder of this section, we analyze some examples of the honest mining pool and one dishonest mining pool setting up the main chain, respectively.

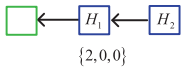


Fig. 3. The first case.

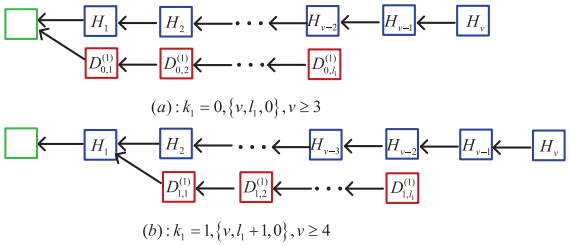


Fig. 4. The second case.

A. The Honest Mining Pool Sets Up the Main Chain

If the honest mining pool sets up the main chain in a round of mining competition, this subsection provides four different examples to express the tree with at most three sub-chains.

Tree one: The 1st and 2nd dishonest mining pools have not mined any block yet, while the honest mining pool has mined two blocks in a round of mining competition. Thus the round of mining competition is terminated immediately. See Fig. 3.

Tree two: The sub-chain lengths of the 1st and 2nd dishonest mining pools and the honest mining pool are l_1, 0 and v, respectively. If $k_1 = 0$, then the round of mining competition ends due to $l_1 = v - 2$. If $k_1 = 1$, then the round of mining competition ends due to $l_1 + 1 = v - 2$. See Fig. 4.

Tree three: The sub-chain lengths of the two dishonest mining pools are not 0, and the two sub-chains fork at the same position. If $k_1 = k_2 = 0$, the round of mining competition ends at the condition under which either $l_1 = v - 2, 1 \leq l_2 \leq v - 2$ or $l_2 = v - 2, 1 \leq l_1 \leq v - 2$. If $k_1 = k_2 = 1$, the round of mining competition ends at the condition under which either $l_1 + 1 = v - 2, 2 \leq l_2 + 1 \leq v - 2$ or $l_2 + 1 = v - 2, 2 \leq l_1 + 1 \leq v - 2$. See Fig. 5.

Tree four: The sub-chain lengths of the two dishonest mining pools are not 0, and the two sub-chains fork at two different

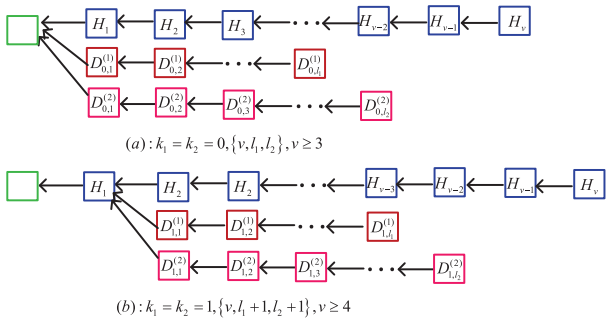


Fig. 5. The third case.

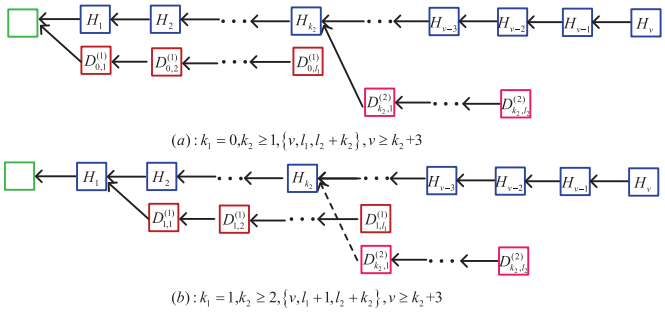


Fig. 6. The fourth case.

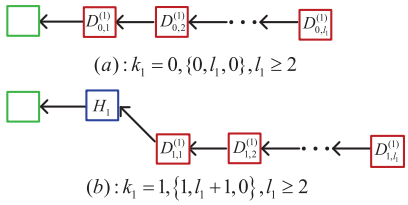


Fig. 7. The first case.

positions. If $k_1 = 0, k_2 \geq 1$, then the round of mining competition ends at the condition under which either $l_1 = v - 2, 2 \leq l_2 + k_2 \leq v - 2$ or $l_2 + k_2 = v - 2, 1 \leq l_1 \leq v - 2$. If $k_1 = 1, k_2 \geq 2$, then the round of mining competition ends at the condition under which either $l_1 + 1 = v - 2, 3 \leq l_2 + k_2 \leq v - 2$ or $l_2 + k_2 = v - 2, 2 \leq l_1 + 1 \leq v - 2$. See Fig. 6.

B. One Dishonest Mining Pool Sets Up the Main Chain

If one dishonest mining pool sets up the main chain in a round of mining competition, this subsection provides four different examples to express the tree with at most three sub-chains.

Tree one: The sub-chain length of the 1st dishonest mining pool is at least 2, but the sub-chain lengths of the 2nd dishonest mining pool and the honest mining pool after the 1st dishonest mining pool forks are 0. See Fig. 7.

Tree two: The sub-chain lengths of the 1st dishonest mining pool and the honest mining pool after the 1st dishonest mining pool forks are positive, but the sub-chain length of the 2nd dishonest mining pool is 0. If $k_1 = 0$, then the round of mining competition ends at the condition under which $l_1 \geq v + 2$, $v \geq 1$. If $k_1 = 1$, then the round of mining competition ends at the condition under which $l_1 + 1 \geq v + 2, v \geq 2$. See Fig. 8.

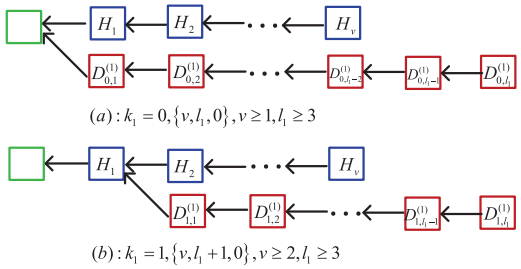


Fig. 8. The second case.

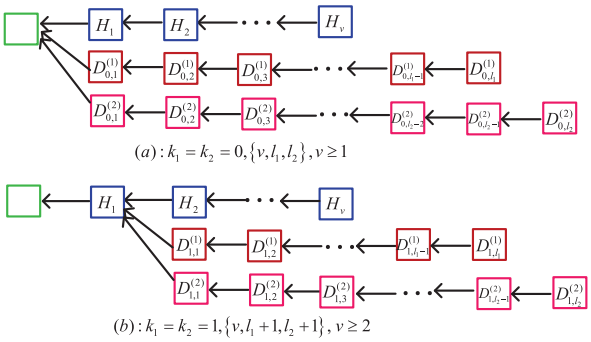


Fig. 9. The third case.

Tree three: The 1st and 2nd dishonest mining pools fork at the same position, and $l_1 \neq 0$, $l_2 \neq 0$; while the honest mining pool has a positive sub-chain length after the 1st and 2nd dishonest mining pools fork, thus, $v - k_1 \geq 1$ and $k_1 = k_2$. If $k_1 = k_2 = 0$, then the round of mining competition ends with the condition under which either $l_1 \geq \max\{v + 2, l_2 + 2\}$ or $l_2 \geq \max\{v + 2, l_1 + 2\}$. If $k_1 = k_2 = 1$, then the round of mining competition ends at the condition under $l_1 + 1 \geq \max\{v + 2, l_2 + 3\}$ or $l_2 + 1 \geq \max\{v + 2, l_1 + 3\}$. See Fig. 9.

Tree four: The two dishonest mining pools do not fork at the same position, and $l_1 \neq 0$, $l_2 \neq 0$; while the honest mining pool has a positive sub-chain length after the 1st dishonest mining pool forks, i.e., $v - k_1 \geq 1$. Now, we consider two cases: $k_1 = 0$, $k_2 \geq 1$ and $k_1 = 1$, $k_2 \geq 2$. If the 1st dishonest mining pool sets up the main chain, then the round of mining competition ends at the condition under which either

$$l_1 \geq \max\{v + 2, l_2 + k_2 + 2\}, k_1 = 0, k_2 \geq 1;$$

or

$$l_1 + 1 \geq \max\{v + 2, l_2 + k_2 + 2\}, k_1 = 1, k_2 \geq 2.$$

If the 2nd dishonest mining pool sets up the main chain, then the round of mining competition ends at the condition under which either

$$l_2 + k_2 \geq \max\{v + 2, l_1 + 2\}, k_1 = 0, k_2 \geq 1;$$

or

$$l_2 + k_2 \geq \max\{v + 2, l_1 + 3\}, k_1 = 1, k_2 \geq 2.$$

See Fig. 10.

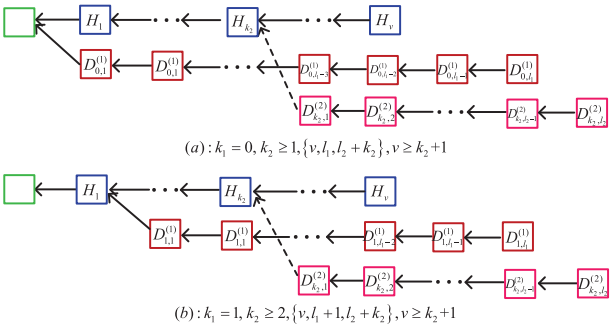


Fig. 10. The fourth case.

IV. The Uncle Blocks and Reward Design

In this section, we introduce a classification of blocks, and give some conditions under which the orphan block can become an uncle block. Furthermore, we provide a two-stage mechanism to determine the uncle blocks and the nephew blocks.

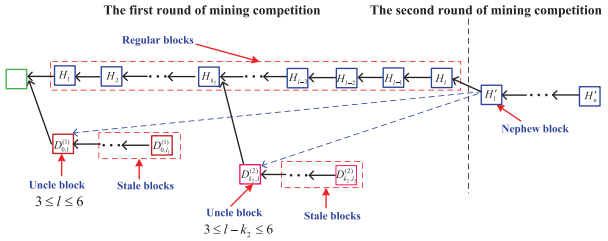
In the multiple sub-chains of the tree corresponding to the PoW Ethereum system with multiple mining pools, we divide the blocks into five different types: Regular blocks, orphan blocks, uncle blocks, stale blocks, and nephew blocks. Also, the orphan blocks are further divided into the uncle blocks and the stale blocks, if any. See Fig. 11 for an intuitive understanding.

The regular block is a block of the main chain. The uncle block must satisfy a key condition under which the distance between the uncle block and the nephew block does not exceed 7 blocks. The nephew block is the first block in the next round of mining competition, and it is used to determine the uncle blocks from the sub-chains of the tree, as seen in Fig. 11. The stale blocks follow from an uncle block in a sub-chain of the tree, or the stale blocks are all the blocks in each sub-chain of orphan blocks.

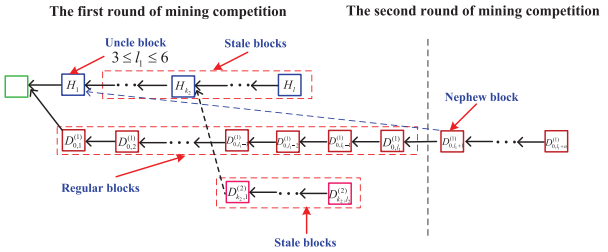
For the mining pools in Ethereum, their reward includes 4 parts: A regular block reward is given by Ethereum, transaction costs (i.e., gas fee), an uncle block reward, and a nephew reference reward. In this paper, we take the unit of economic measure as "block", in other words, the regular reward given by the Ethereum systems is assumed to be 1.

In the PoW Ethereum system with multiple mining pools, the main purpose of introducing the uncle blocks is to inspire the mining enthusiasm of each mining pool, especially for those mining pools that cannot set up the main chain. In this case, they cannot obtain any reward but have to cover some costs associated with the mining processes, such as electricity, equipment investment, maintenance costs, staff salary, management fees and so on. Thus, providing the uncle block reward and the nephew reference reward can be a necessary and valuable support for the multiple mining pools that take an active part in each round of mining competition.

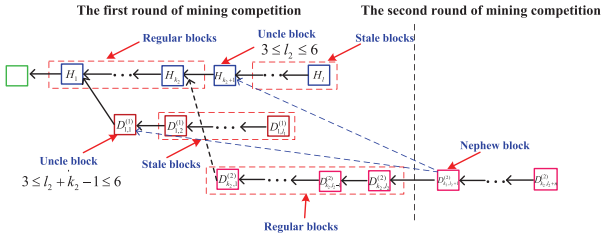
To show how to allocate the uncle block reward and the nephew reference reward, we provide a two-stage reward allocation mechanism through analyzing the position among the uncle blocks and the nephew blocks. For simplicity of analysis, the uncle blocks are determined at the moment that a



(a) The honest mining pool sets up the main chain



(b) The first forked dishonest mining pool sets up the main chain



(c) The second forked dishonest mining pool sets up the main chain

Fig. 11. A classification of blocks.

TABLE I
Table of Uncle Rewards at Different Distances

Distance	1	2	3	4	5	6
Uncle reward	7/8	6/8	5/8	4/8	3/8	2/8

round of mining competition has been over, and a nephew block (or the first block) is just generated in the next round of mining competition. That is, to determine the uncle blocks, we first need to determine the nephew block in order that we can judge that the distance between the uncle block and the nephew block does not exceed 7 blocks. Note that the nephew block comes from two different cases: (i) If all the blocks of the main chain are released at the ending moment of the last round of mining competition, then the nephew block is the first block mined by one of the $m + 1$ mining pools in the next round of mining competition. (ii) If there are some blocks of the main chain reserved at the ending moment of the last round of mining competition, then these reserved blocks must come from one dishonest mining pool, and the nephew block is the first one of the reserved blocks.

It is worthwhile to note that such a reward allocation is well related to the distance between the uncle block and the nephew block. Table I shows the uncle block rewards at some different distances.

In what follows we discuss some conditions under which the orphan block can become an uncle block. Our description contains two different cases as follows:

Case one: If the honest mining pool sets up the main chain, then one of the orphan blocks mined by the *i*th dishonest mining pool becomes an uncle block and must satisfy the following two conditions (a-1) and (a-2), also see (a) of Fig. 11.

(a-1) The distance between the orphan block and its corresponding nephew block does not exceed 7 blocks; and

(a-2) the orphan block is the first block $D_{k_i,1}^{(i)}$ mined by the *i*th dishonest mining pool.

Case two: If the *i*th dishonest mining pool sets up the main chain, then either one of the orphan blocks mined by the *j*th dishonest mining pool for $j \neq i$ or one of the orphan blocks mined by the honest mining pool becomes an uncle block and must satisfy the following two conditions (b-1) and (b-2), also see (b) and (c) of Fig. 11.

(b-1) The distance between the orphan block and its corresponding nephew block does not exceed 7 blocks; and

(b-2) either the orphan block is the first block $D_{k_j,1}^{(j)}$ mined by the *j*th dishonest mining pool, or the orphan block is the block H_{k_i+1} mined by the honest mining pool.

In addition to the uncle reward given in a round of mining competition, we still need to provide a reward to the nephew block, called the nephew reference reward, which is given in the next round of mining competition. A nephew block can obtain the nephew reference reward: $N_{\rm U}/32$, where $N_{\rm U}$ is the number of uncle blocks in this round of mining competition. Note that, our above reward allocation is designed in two consecutive rounds of mining competition, and it is called a two-stage reward allocation mechanism.

V. The Law of Large Numbers

In this section, we apply the law of large numbers to study some key probabilities, which are necessary and useful in our later study, such as the key ratios of Ethereum, the growth rate of blockchain, the reward allocation rates among the mining pools, and so forth.

In a round of mining competition, both the honest mining pool and one of the dishonest mining pools are possible to set up the main chain. Now, we provide an analysis for the number of main chains that can be set up by either the honest mining pool or the ζth dishonest mining pool. During the N rounds of mining competition, we denote by $N_{\rm H}$ and N_ζ the numbers that the honest mining pool and the ζth dishonest mining pool can set up the main chain, respectively. It is clear that $0 \leq N_{\rm H} \leq N$, $0 \leq N_\zeta \leq N$, and $N_{\rm H} + \sum_{\zeta=1}^{m} N_\zeta = N$. Thus we have

$$\frac{N_{\rm H}}{N} + \sum_{\zeta=1}^{m} \frac{N_\zeta}{N} = 1\,.$$

In the PoW Ethereum system with multiple mining pools, the competitively mining processes can be repeated round after round, as we repeat the similar experiments round after round under the same conditions. Thus, we can apply the law of large numbers to study the frequencies: $N_{\rm H}/N$ and N_ζ/N, and to show that they can steadily approach some fixed values for $\zeta = 1, 2, \ldots, m$. The following two theorems are obvious by the law of large numbers, and their proof is easy and omitted here for brevity.

Theorem 2: In the PoW Ethereum system with multiple mining pools, by using the law of large numbers, as $N \to \infty$, we have

$$\frac{N_{\rm H}}{N} \to \mathbf{p}_{\rm H},\ \text{a.s.},$$

and for $\zeta = 1, 2, \ldots, m$,

$$\frac{N_\zeta}{N} \to \mathbf{p}_\zeta,\ \text{a.s.}.$$

Also, it is clear that

$$\mathbf{p}_{\rm H} + \sum_{\zeta=1}^{m} \mathbf{p}_\zeta = 1.$$

When the honest mining pool sets up the main chain, during the N rounds of mining competition, we denote by $N_{\rm H,H}^{(\rm N)}$, $N_{{\rm H},i}^{(\rm N)}$ and $N_{{\rm H},i}^{(\rm U)}$ the numbers that the honest mining pool and the *i*th dishonest mining pool can contain one nephew block, and the *i*th dishonest mining pool can contain one uncle block, respectively.

When the ζth dishonest mining pool sets up the main chain for $\zeta = 1, 2, \ldots, m$, during the N rounds of mining competition, we denote by $N_{\zeta,\zeta}^{(\rm N)}$, $N_{\zeta,{\rm H}}^{(\rm N)}$, $N_{\zeta,i}^{(\rm N)}$, $N_{\zeta,{\rm H}}^{(\rm U)}$ and $N_{\zeta,i}^{(\rm U)}$ the numbers that the ζth dishonest mining pool, the honest mining pool, and the *i*th dishonest mining pool can contain one nephew block, and the honest mining pool and the *i*th dishonest mining pool can contain one uncle block, respectively.

Theorem 3: In the PoW Ethereum with multiple mining pools, by using the law of large numbers, as $N \to \infty$, for $i = 1, 2, \ldots, m$,

$$\frac{N_{\rm H,H}^{(\rm N)}}{N} \to \mathbf{q}_{\rm H,H}^{(\rm N)},\ \text{a.s.},$$
$$\frac{N_{{\rm H},i}^{(\rm N)}}{N} \to \mathbf{q}_{{\rm H},i}^{(\rm N)},\ \frac{N_{{\rm H},i}^{(\rm U)}}{N} \to \mathbf{q}_{{\rm H},i}^{(\rm U)},\ \text{a.s.},$$

and for $i \neq \zeta$ and $i = 1, 2, \ldots, m$,

$$\frac{N_{\zeta,\zeta}^{(\rm N)}}{N} \to \mathbf{q}_{\zeta,\zeta}^{(\rm N)},\ \text{a.s.},$$
$$\frac{N_{\zeta,{\rm H}}^{(\rm N)}}{N} \to \mathbf{q}_{\zeta,{\rm H}}^{(\rm N)},\ \frac{N_{\zeta,{\rm H}}^{(\rm U)}}{N} \to \mathbf{q}_{\zeta,{\rm H}}^{(\rm U)},$$
$$\frac{N_{\zeta,i}^{(\rm N)}}{N} \to \mathbf{q}_{\zeta,i}^{(\rm N)},\ \frac{N_{\zeta,i}^{(\rm U)}}{N} \to \mathbf{q}_{\zeta,i}^{(\rm U)},\ \text{a.s.}.$$

It is clear that $0 \leq \mathbf{q}_{\rm H,H}^{(\rm N)}, \mathbf{q}_{{\rm H},i}^{(\rm N)} \leq 1$ and $0 \leq \mathbf{q}_{{\rm H},i}^{(\rm U)} \leq 1$ for $i = 1, 2, \ldots, m$; and $0 \leq \mathbf{q}_{\zeta,\zeta}^{(\rm N)}, \mathbf{q}_{\zeta,{\rm H}}^{(\rm N)}, \mathbf{q}_{\zeta,i}^{(\rm N)} \leq 1$ and $0 \leq \mathbf{q}_{\zeta,{\rm H}}^{(\rm U)}, \mathbf{q}_{\zeta,i}^{(\rm U)} \leq 1$ for $i \neq \zeta$ and $i = 1, 2, \ldots, m$.

VI. Some Key Ratios of Ethereum

This section defines some key ratios of the PoW Ethereum system with multiple mining pools, and provides a detailed analysis for the key ratios by means of the mathematical representation of tree with multiple sub-chains.

In a round of mining competition, the multiple mining pools use their mined blocks to set up a tree with multiple sub-chains. From the tree, we can classify five different types of blocks: Regular blocks, orphan blocks, uncle blocks, nephew blocks, and stale blocks. Based on this, we can set up some key ratios of the PoW Ethereum system with multiple mining pools. To this end, we define some key ratios of Ethereum from two perspectives of efficiency and benefit, such as chain quality, main chain length ratio, orphan block ratio, uncle block ratio, and stale block ratio.

Definition 1: In a round of mining competition, we define

(a) *The chain quality c_Q:* It is defined as the ratio that some blocks on the main chain mined by the honest mining pool occupy all the blocks of the main chain.

(b) *The main chain length ratio r_M:* It is defined as the ratio that the number of blocks on the main chain over the number of blocks on the tree.

If the honest mining pool sets up the main chain, from $L_0 = \{H_1, H_2, \ldots, H_v\}$ and $L_i = \{H_1, H_2, \ldots, H_{k_i}; D^{(i)}_{k_i,1}, D^{(i)}_{k_i,2}, \ldots, D^{(i)}_{k_i,l_i}\}$ for $i = 1, 2, \ldots, m$, we have

$$c_{\mathrm{Q,H}} = \frac{v}{v} = 1$$

and

$$r_{\mathrm{M,H}} = \frac{v}{v + \sum_{i=1}^{m} l_i}.$$

If the ith dishonest mining pool sets up the main chain for $i = 1, 2, \ldots, m$, due to the part with $\varphi_i + k_i$ blocks for $\omega_2 + 2 \le \varphi_i + k_i \le \omega_1$ of the main chain are pegged on the blockchain, while the $\omega_1 - \varphi_i - k_i$ blocks of the main chain cannot be observed by all the other mining pools in the P2P network, then

$$c_{\mathrm{Q},\,i} = \frac{k_i}{k_i + \varphi_i}$$

and

$$r_{\mathrm{M},\,i} = \frac{\varphi_i + k_i}{v + \varphi_i + \sum_{j=1,\,j\neq i}^{m} l_j}.$$

Definition 2: In a round of mining competition, we define

(a) *The orphan block ratio r_O:* It is defined as the ratio of the number of orphan blocks to the number of blocks on the tree.

(b) *The uncle block ratio r_U:* It is defined as the ratio of the number of uncle blocks to the number of blocks on the tree.

(c) *The stale block ratio r_S:* It is defined as the ratio of the number of stale blocks to the number of blocks on the tree.

It is easy to see from Definitions 1 and 2 that

$$r_{\mathrm{M}} + r_{\mathrm{O}} = 1$$

and

$$r_{\mathrm{O}} = r_{\mathrm{U}} + r_{\mathrm{S}}.$$

In what follows, we first express the orphan block ratio r_{O}. To this end, our computation needs to consider two different cases: The honest mining pool sets up the main chain, and one dishonest mining pool sets up the main chain.

If the honest mining pool sets up the main chain, then

$$r_{\mathrm{O,H}} = \frac{\sum_{i=1}^{m} l_i}{v + \sum_{i=1}^{m} l_i}. \tag{1}$$

If the ith dishonest mining pool sets up the main chain for $i = 1, 2, \ldots, m$, due to the part with $\varphi_i + k_i$ blocks for $\omega_2 + 2 \le \varphi_i + k_i \le \omega_1$ of the main chain are pegged onto the blockchain, while the $\omega_1 - \varphi_i - k_i$ blocks of the main chain cannot be observed by all the other mining pools in the P2P network, then

$$r_{\mathrm{O},\,i} = \frac{(v - k_i) + \sum_{j=1,\,j\neq i}^{m} l_j}{v + \varphi_i + \sum_{j=1,\,j\neq i}^{m} l_j}. \tag{2}$$

Now, we compute the uncle block ratio r_{U}. Note that such a computation is a little bit complicated.

To compute the uncle block ratio r_{U}, it is necessary to first determine how many orphan blocks can become uncle blocks. To this end, our computation also needs to consider two different cases: The honest mining pool sets up the main chain, and one dishonest mining pool sets up the main chain.

Case one: The honest mining pool sets up the main chain

In this case, from the tree with multiple sub-chains (e.g., see (a) of Fig. 11), it is easy to determine the nephew block, and the uncle blocks and their number. Note that $N_{\mathrm{U,H}}$ is the number of uncle blocks in the tree in this round of mining competition, thus the uncle block ratio is given by

$$r_{\mathrm{U,H}} = \frac{N_{\mathrm{U,H}}}{v + \sum_{i=1}^{m} l_i}. \tag{3}$$

Case two: One dishonest mining pool sets up the main chain

In this case, we assume that the ith dishonest mining pool sets up the main chain, it is clear that $\omega_1 = k_i + l_i$. We further assume that the part with $\varphi_i + k_i$ blocks for $\omega_2 + 2 \le \varphi_i + k_i \le \omega_1$ of the main chain are pegged onto the blockchain, while another part with $\omega_1 - \varphi_i - k_i$ blocks of the main chain is left in the next round of mining competition. It is easy to see that $D^{(i)}_{k_i,\varphi_i+1}$ is the first block in the next round of mining competition. Now, our first task is to use the block $D^{(i)}_{k_i,\varphi_i+1}$ to determine which blocks of H_{k_i+1} and $D^{(j)}_{k_j,1}$ for $j \neq i$ can become uncle blocks. If H_{k_i+1} is an uncle block, then for $j \neq i$, $D^{(j)}_{k_j,1}$ is not an uncle block for $k_j \ge k_i + 1$.

From the tree with multiple sub-chains (e.g., see (b) and (c) of Fig. 11), it is easy to determine the nephew block, and the uncle blocks and their number. Note that $N_{\mathrm{U},\,i}$ is the number of uncle blocks in the tree, thus the uncle block ratio is given by

$$r_{\mathrm{U},\,i} = \frac{N_{\mathrm{U},\,i}}{v + \varphi_i + \sum_{j=1,\,j\neq i}^{m} l_j}. \tag{4}$$

In the remainder of this section, we apply the law of large numbers to discuss the key ratios of the Ethereum system.

In the rth round of mining competition, we denote by $c^{(r)}_{\mathrm{Q}}$, $r^{(r)}_{\mathrm{M}}$, $r^{(r)}_{\mathrm{O}}$, and $r^{(r)}_{\mathrm{U}}$ the chain quality, the main chain length

ratio, the orphan block ratio, and the uncle block ratio, respectively. We assume that the part with $\varphi_i^{(r)} + k_i^{(r)}$ blocks for $\omega_{i,2}^{(r)} + 2 \le \varphi_i^{(r)} + k_i^{(r)} \le \omega_{i,1}^{(r)}$ of the main chain mined by the *i*th dishonest mining pool are pegged onto the blockchain, where $\omega_{i,1}^{(r)}$ and $\omega_{i,2}^{(r)}$ are the first and second elements of the sorted set *S* related to the *i*th dishonest mining pool, respectively.

Theorem 4: In the PoW Ethereum system with multiple mining pools, by using the law of large numbers, as $N \to \infty$, we have

$$\frac{\sum_{r=1}^{N} c_{\mathrm{Q}}^{(r)}}{N} \to \overline{c}_{\mathrm{Q}},\ \text{a.s.}, \qquad \frac{\sum_{r=1}^{N} r_{\mathrm{M}}^{(r)}}{N} \to \overline{r}_{\mathrm{M}},\ \text{a.s.},$$
$$\frac{\sum_{r=1}^{N} r_{\mathrm{O}}^{(r)}}{N} \to \overline{r}_{\mathrm{O}},\ \text{a.s.}, \qquad \frac{\sum_{r=1}^{N} r_{\mathrm{U}}^{(r)}}{N} \to \overline{r}_{\mathrm{U}},\ \text{a.s.},$$

where

$$\overline{c}_{\mathrm{Q}} = \mathbf{p}_{\mathrm{H}} + \sum_{\zeta=1}^{m} \mathbf{p}_{\zeta} \times \overline{c}_{\mathrm{Q},\zeta},$$
$$\overline{r}_{\mathrm{M}} = \mathbf{p}_{\mathrm{H}} \times \overline{r}_{\mathrm{M,H}} + \sum_{\zeta=1}^{m} \mathbf{p}_{\zeta} \times \overline{r}_{\mathrm{M},\zeta},$$
$$\overline{r}_{\mathrm{O}} = \mathbf{p}_{\mathrm{H}} \times \overline{r}_{\mathrm{O,H}} + \sum_{\zeta=1}^{m} \mathbf{p}_{\zeta} \times \overline{r}_{\mathrm{O},\zeta},$$
$$\overline{r}_{\mathrm{U}} = \mathbf{p}_{\mathrm{H}} \times \overline{r}_{\mathrm{U,H}} + \sum_{\zeta=1}^{m} \mathbf{p}_{\zeta} \times \overline{r}_{\mathrm{U},\zeta}.$$

Proof: The proof is easy. We only take the first one as an example. To do this, we need to consider two different cases:

Case one: If the honest mining pool sets up the main chain in the *r*th round of mining competition for $r = 1, 2, 3, \ldots N$. In this case, we have $c_{\mathrm{Q,H}}^{(r)} = 1$, and

$$\frac{\sum_{r=1}^{N_{\mathrm{H}}} c_{\mathrm{Q,H}}^{(r)}}{N} = \frac{N_{\mathrm{H}}}{N}.$$

This gives that as $N \to \infty$,

$$\frac{\sum_{r=1}^{N_{\mathrm{H}}} c_{\mathrm{Q,H}}^{(r)}}{N} \to \mathbf{p}_{\mathrm{H}},\ \text{a.s.}.$$

Case two: If the ζth dishonest mining pool sets up the main chain in the *r*th round of mining competition for $r = 1, 2, 3, \ldots, N$. In this case, it is worthwhile to note that $k_{\zeta}^{(r)}$ is the number of honest blocks which have been mined by the honest mining pool before the ζth dishonest mining pool begins to fork to a new sub-chain with $l_{\zeta}^{(r)}$ blocks, and note that the part with $\varphi_{\zeta}^{(r)} + k_{\zeta}^{(r)}$ blocks of the main chain are pegged onto the blockchain, thus we have

$$c_{\mathrm{Q},\zeta}^{(r)} = \frac{k_{\zeta}^{(r)}}{k_{\zeta}^{(r)} + \varphi_{\zeta}^{(r)}}.$$

Since the competitively mining processes of the multiple mining pools are repeated round after round, the random variable $c_{\mathrm{Q},\zeta}^{(r)}$ can be repeated by the similar experiments round after round under the same conditions. Based on this, the random variable sequence $\{c_{\mathrm{Q},\zeta}^{(r)}, r = 1, 2, 3, \ldots\}$ can be regarded as independent and identically distributed. When the number that the ζth dishonest mining pool can set up the main chain is N_{ζ}, we obtain that as $N_{\zeta} \to \infty$,

$$\frac{\sum_{r=1}^{N_{\zeta}} c_{\mathrm{Q},\zeta}^{(r)}}{N_{\zeta}} \to \overline{c}_{\mathrm{Q},\zeta},\ \text{a.s.},$$

which holds for each $\zeta = 1, 2, \ldots, m$. This gives that as $N \to \infty$,

$$\frac{\sum_{r=1}^{N_{\zeta}} c_{\mathrm{Q},\zeta}^{(r)}}{N} = \frac{\sum_{r=1}^{N_{\zeta}} c_{\mathrm{Q},\zeta}^{(r)}}{N_{\zeta}} \cdot \frac{N_{\zeta}}{N} = \mathbf{p}_{\zeta} \cdot \overline{c}_{\mathrm{Q},\zeta}, \text{a.s.}.$$

From the above two cases, we get

$$\overline{c}_{\mathrm{Q}} = \mathbf{p}_{\mathrm{H}} + \sum_{\zeta=1}^{m} \mathbf{p}_{\zeta} \cdot \overline{c}_{\mathrm{Q},\zeta}. \text{a.s.}.$$

This completes the proof. ■

VII. The Growth Rate of Blockchain

In this section, we apply the renewal reward theory to study the long-term growth rate of blockchain in the PoW Ethereum system with multiple mining pools.

In the *r*th round of mining competition, we denote by $v^{(r)}$ the number of blocks on the main chain when the honest mining pool sets up the main chain, denote by $k_i^{(r)}$ and $l_i^{(r)}$ the number of honest blocks first mined by the honest mining pool and the number of dishonest blocks mined by the *i*th dishonest mining pool after forked when the *i*th dishonest mining pool sets up the main chain, respectively. Note that the part with $\varphi_i^{(r)} + k_i^{(r)}$ blocks of the main chain is pegged onto the blockchain.

Lemma 1: In the PoW Ethereum system with multiple mining pools, by using the law of large numbers, as $N \to \infty$, we have

$$\frac{\sum_{r=1}^{N} v^{(r)}}{N} \to \overline{v},\ \text{a.s.},$$

for $i = 1, 2, \ldots, m$,

$$\frac{\sum_{r=1}^{N} k_i^{(r)}}{N} \to \overline{k}_i,\ \text{a.s.},$$
$$\frac{\sum_{r=1}^{N} l_i^{(r)}}{N} \to \overline{l}_i,\ \text{a.s.},$$

and

$$\frac{\sum_{r=1}^{N} \varphi_i^{(r)}}{N} \to \overline{\varphi}_i,\ \text{a.s.},$$

where $\overline{v}$, $\overline{k}_i$, $\overline{l}_i$ and $\overline{\varphi}_i$ are the means of four random variables $v^{(r)}$, $k_i^{(r)}$, $l_i^{(r)}$ and $\varphi_i^{(r)}$, respectively.

It is worthwhile to note that in the PoW Ethereum system, the competitively mining processes of the multiple mining pools are repeated round after round, as we repeat the experiments under the same conditions. Thus, the moments that one round of mining competition is over and the next round of mining competition begins immediately are all renewal points.

That is, let T_k be the kth moment that the kth round of mining competition is over and the $(k+1)$ th round of mining competition begins immediately. In fact, $[T_{k-1}, T_k)$ represents a time interval that the kth round of mining competition is underway. For simplicity of analysis, we assume that $T_0 = 0$, i.e., the first round of mining competition begins at time $T_0 = 0$. Let $N(t) = \max\{k, T_k \le t\}$, then $\{N(t), t \ge 0\}$ is a renewal process.

We assume that there are M_k blocks of the main chain in the time interval $[T_{k-1}, T_k)$, and M_k is independent of the time interval $[T_{k-1}, T_k)$. Let $M(t)$ be the number of blocks of all the main chains generated in the time interval [0, t). Then, the following theorem provides the growth rate $E[M(t)]/t$ of blockchain in the PoW Ethereum system with multiple mining pools.

Theorem 5: In the PoW Ethereum system with multiple mining pools, if $E[M_1] < +\infty$ and $E[T_1] < +\infty$, then as $t \to +\infty$, we have

$$\frac{M(t)}{t} \to \frac{E[M_1]}{E[T_1]}, \text{a. s.}, \tag{5}$$

$$\frac{E[M(t)]}{t} \to \frac{E[M_1]}{E[T_1]}, \tag{6}$$

where

$$E[M_1] = \mathbf{p}_{\mathrm{H}} \cdot \overline{v} + \sum_{i=1}^{m} \mathbf{p}_i \cdot \left\{\overline{k}_i + \overline{\varphi}_i\right\}.$$

Proof: We give the proof for the equation (5) only. We write

$$\frac{M(t)}{t} = \frac{\sum_{k=1}^{N(t)} M_k}{t} = \frac{\sum_{k=1}^{N(t)} M_k}{N(t)} \cdot \frac{N(t)}{t}.$$

Note that

$$T_k = (T_1 - T_0) + (T_2 - T_1) + \cdots + (T_k - T_{k-1})$$

is the time length of the k renewal periods, and the random variables $T_1 - T_0,\ T_2 - T_1, \ldots, T_k - T_{k-1}$ are independent and identically distributed, $E[T_k - T_{k-1}] = E[T_1 - T_0] = E[T_1]$ for $k \ge 1$. At the same time, the random variables $M_k, k \ge 1$ are also independent and identically distributed, $E[M_k] = E[M_1]$.

Note that as $t \to +\infty$, $N(t) \to +\infty$. By the strong law of large numbers, we obtain that as $t \to \infty$

$$\frac{\sum_{k=1}^{N(t)} M_k}{N(t)} \to E[M_1].$$

By using the elementary renewal theorem ([31, Ch. 7]), we obtain that

$$\frac{N(t)}{t} \to \frac{1}{E[T_1]}, \text{as } t \to +\infty.$$

Therefore, we obtain that as $t \to +\infty$

$$\frac{M(t)}{t} \to \frac{E[M_1]}{E[T_1]}.$$

This completes the proof. ■

VIII. Average Reward Allocation Among the Mining Pools

In this section, we propose a new method to compute the uncle block and nephew block rewards in two consecutive rounds of mining competition, and provide expressions for the long-term reward allocation and for the long-term reward allocation rate to each mining pool by using the renewal reward theory.

To set up the reward allocation, it is easy to see that a regular block is paid the reward of 1 block; an uncle block is paid the reward of $(8 - l)/8$ blocks for $1 \le l \le 6$, where l is the distance between the uncle block and the nephew block; a nephew block is paid the reward of $N_{\mathrm{U}}/32$ blocks, where N_{U} is the number of uncle blocks; and a stale block is paid no reward.

In the PoW Ethereum system with multiple mining pools, it is worthwhile to note that in the tree with multiple sub-chains, the main chain can be obtained by either the honest mining pool or one of the m dishonest mining pools, thus our reward allocation is considered as the following two different cases.

Case one: The honest mining pool sets up the main chain.

In this case, by observing (a) of Fig. 11, we consider two different cases as follows:

(i) The reward of the honest mining pool is given by

$$R_{\mathrm{H,H}} = v + R_{\mathrm{H,H}}^{(\mathrm{N})},$$

where v is the number of blocks in the main chain, $R_{\mathrm{H,H}}^{(\mathrm{N})}$ is the reward of a nephew block which is the first block of the main chain. Note that the nephew block refers to the uncle blocks in the previous round of mining competition. Thus we obtain

$$R_{\mathrm{H,H}}^{(\mathrm{N})} = \begin{cases} \frac{N_{\mathrm{U,H}}}{32}, & \text{if the first block of the main chain} \\ & \text{is a nephew block,} \\ 0, & \text{otherwise.} \end{cases}$$

(ii) Note that each of the m dishonest mining pools may mine only the orphan blocks, that is, either the uncle blocks or the stale blocks. In addition, it has a nephew block **either** if the main chain of the previous round of mining competition is competitively pegged onto the blockchain, and this dishonest mining pool is the first one to mine a block, that is, the firstly mined block is the nephew block; **or** if this dishonest mining pool sets up the main chain in the previous round of mining competition, and a non-empty part of this main chain is left to this round of mining competition, that is, the nephew block is the first block of the non-empty part. Based on this, the reward of the ith dishonest mining pool is given by

$$R_{\mathrm{H},i} = R_{\mathrm{H},i}^{(\mathrm{U})} + R_{\mathrm{H},i}^{(\mathrm{N})}, \quad i = 1, 2, \ldots, m,$$

where $R_{\mathrm{H},i}^{(\mathrm{U})}$ is the reward that if the ith dishonest mining pool has an uncle block, and

$$R_{\mathrm{H},i}^{(\mathrm{U})} = \frac{8 - l}{8}$$

for $1 \le l \le 6$, where l is the distance between the uncle block and the nephew block of the next round of mining competition;

and

$$R_{\mathrm{H},i}^{(\mathrm{N})} = \frac{N_{\mathrm{U},\,i}}{32},$$

where $N_{\mathrm{U},\,i}$ is the number of uncle blocks in the previous round of mining competition if the nephew block belongs to the ith dishonest mining pool. Based on this, we have

$$R_{\mathrm{H},i}^{(\mathrm{U})} = \begin{cases} \frac{8-l}{8}, & \text{if the } i\text{th dishonest mining pool} \\ & \text{has an uncle block,} \\ 0, & \text{otherwise;} \end{cases}$$

and

$$R_{\mathrm{H},i}^{(\mathrm{N})} = \begin{cases} \frac{N_{\mathrm{U},i}}{32}, & \text{if the nephew block belongs to} \\ & \text{the } i\text{th dishonest mining pool,} \\ 0, & \text{otherwise.} \end{cases}$$

Case two: The ζth dishonest mining pool sets up the main chain.

In this case, by observing (b) and (c) of Fig. 11, we consider three different cases as follows:

(i) The reward of the ζth dishonest mining pool

Note that the ζth dishonest mining pool sets up the main chain, it is clear that $\omega_1 = k_\zeta + l_\zeta$. We assume that the part with $\varphi_\zeta + k_\zeta$ blocks for $\omega_2 + 2 \le \varphi_\zeta + k_\zeta \le \omega_1$ of the main chain are pegged onto the blockchain, while another part with $\omega_1 - \varphi_\zeta - k_\zeta$ blocks of the main chain is left to the next round of mining competition.

The reward of the ζth dishonest mining pool is given by

$$R_{\zeta,\zeta} = \varphi_\zeta + R_{\zeta,\zeta}^{(\mathrm{N})},$$

where φ_ζ is the number of blocks in the part of the main chain, which is pegged onto the blockchain; and

$$R_{\zeta,\zeta}^{(\mathrm{N})} = \frac{N_{\mathrm{U},\zeta}}{32},$$

where $N_{\mathrm{U},\zeta}$ is the number of uncle blocks in the previous round of mining competition if the nephew block belongs to the ζth dishonest mining pool. Based on this, we have

$$R_{\zeta,\zeta}^{(\mathrm{N})} = \begin{cases} \frac{N_{\mathrm{U},\zeta}}{32}, & \text{if the nephew block belongs to} \\ & \text{the } \zeta\text{ th dishonest mining pool,} \\ 0, & \text{otherwise.} \end{cases}$$

(ii) The reward of the honest mining pool

If the ζth dishonest mining pool sets up the main chain, then the honest mining pool can mine only the orphan blocks after the ζth dishonest mining pool forks, that is, either the uncle blocks or the stale blocks. In addition, it may have a nephew block if the honest mining pool is the first one to mine a block in this round of mining competition. In this case, the reward of the honest mining pool contains the reward of k_ζ regular blocks, the reward of one uncle block and the reward of a nephew block. Based on this, the reward of the honest mining pool is given by

$$R_{\zeta,\mathrm{H}} = k_\zeta + R_{\zeta,\mathrm{H}}^{(\mathrm{U})} + R_{\zeta,\mathrm{H}}^{(\mathrm{N})},$$

where $R_{\zeta,\mathrm{H}}^{(\mathrm{U})}$ is the reward that if the honest mining pool has an uncle block, and

$$R_{\zeta,\mathrm{H}}^{(\mathrm{U})} = \frac{8-l}{8}$$

for $1 \le l \le 6$, where l is the distance between the uncle block and the nephew block of the next round of mining competition; and

$$R_{\zeta,\mathrm{H}}^{(\mathrm{N})} = \frac{N_{\mathrm{U,H}}}{32},$$

where $N_{\mathrm{U,H}}$ is the number of uncle blocks in the previous round of mining competition if the nephew block belongs to the honest mining pool. Based on this, we have

$$R_{\zeta,\mathrm{H}}^{(\mathrm{U})} = \begin{cases} \frac{8-l}{8}, & \text{if the honest mining pool has an uncle block,} \\ 0, & \text{otherwise;} \end{cases}$$

and

$$R_{\zeta,\mathrm{H}}^{(\mathrm{N})} = \begin{cases} \frac{N_{\mathrm{U,H}}}{32}, & \text{if the nephew block belongs to} \\ & \text{the honest mining pool,} \\ 0, & \text{otherwise.} \end{cases}$$

(iii) The reward of the ith dishonest mining pool for $i \neq \zeta$

If the ζth dishonest mining pool sets up the main chain, then the ith dishonest mining pool can mine only the orphan blocks, that is, either the uncle blocks or the stale blocks. In addition, it has a nephew block **either** if the main chain of the previous round of mining competition is completely pegged onto the blockchain, and the ith dishonest mining pool is the first one to mine a block, that is, the firstly mined block is the nephew block that belongs to the ith dishonest mining pool; **or** if the ith dishonest mining pool sets up the main chain in the previous round of mining competition, and a non-empty part of this main chain is left to this round of mining competition, that is, the nephew block is the first block of the non-empty part. Based on this, the reward of the ith dishonest mining pool is given by

$$R_{\zeta,i} = R_{\zeta,i}^{(\mathrm{U})} + R_{\zeta,i}^{(\mathrm{N})}, \quad i = 1, 2, \ldots, m,$$

where $R_{\zeta,i}^{(\mathrm{U})}$ is the reward that if the ith dishonest mining pool has an uncle block, and

$$R_{\zeta,i}^{(\mathrm{U})} = \frac{8-l}{8}$$

for $1 \le l \le 6$, where l is the distance between the uncle block and the nephew block of the next round of mining competition; and

$$R_{\zeta,i}^{(\mathrm{N})} = \frac{N_{\mathrm{U},i}}{32},$$

where $N_{\mathrm{U},i}$ is the number of uncle blocks in the previous round of mining competition if the nephew block belongs to the ith dishonest mining pool. Based on this, we have

$$R_{\zeta,i}^{(\mathrm{U})} = \begin{cases} \frac{8-l}{8}, & \text{if the } i\text{th dishonest mining pool} \\ & \text{has an uncle block,} \\ 0, & \text{otherwise;} \end{cases}$$

and

$$R_{\zeta,i}^{(\mathrm{N})} = \begin{cases} \frac{N_{\mathrm{U},i}}{32}, & \text{if the nephew block belongs to} \\ & \text{the } i\text{th dishonest mining pool, .} \\ 0, & \text{otherwise.} \end{cases}$$

When the reward of each mining pool is regarded as random variables, we can apply the law of large numbers to further

study the reward of each mining pool. In this situation, we obtain some interesting results, which can be applied to solving many practical problems owing to the fact that by using the law of large numbers, our experimental reward of each mining pool can steadily approach their corresponding fixed values. Such a statistical method is effective and useful in the study of the PoW Ethereum system with multiple mining pools, because analysis of the tree with multiple sub-chains always has a higher computational complexity.

In what follows, we apply the law of large numbers to study the reward obtained by each multiple mining pool in the PoW Ethereum system. Let $R_{\rm H}^{(r)}$ and $R_i^{(r)}$ be the reward obtained by the honest mining pool or the *i*th dishonest mining pool in the *r*th round of competition for $i = 1, 2, \ldots, m$.

Theorem 6: In the PoW Ethereum system with multiple mining pools, by using the law of large numbers, as $N \to \infty$, we have

$$\frac{\sum_{r=1}^{N} R_{\rm H}^{(r)}}{N} \to \overline{R}_{\rm H},\ \text{a.s.},$$

and for $i = 1, 2, \ldots, m$,

$$\frac{\sum_{r=1}^{N} R_i^{(r)}}{N} \to \overline{R}_i,\ \text{a.s.}.$$

Proof: The proof is easy. We only take the first one as a proof example. Here, we also need to consider two different cases:

Case one: The honest mining pool sets up the main chain in the *r*th round of mining competition for $r = 1, 2, 3, \ldots, N$. In this case,

$$R_{\rm H,H}^{(r)} = v^{(r)} + R_{\rm H,H}^{({\rm N},r)},$$

where in the *r*th round of mining competition, $v^{(r)}$ and $R_{\rm H}^{({\rm N},r)}$ are the number of blocks in the main chain, and the reward of a nephew block which is the first block of the main chain, respectively. Note that the main chain is mined by the honest mining pool.

Note that the competitively mining processes of the multiple mining pools are repeated round after round, the random variables $v^{(r)}$ and $R_{\rm H}^{({\rm N},r)}$ (thus $R_{\rm H}^{(r)}$) can be repeated by the experiments round after round under the same conditions. Based on this, the random variable sequences $\{v^{(r)}, r = 1, 2, 3, \ldots\}$ and $\{R_{\rm H,H}^{({\rm N},r)}, r = 1, 2, 3, \ldots\}$ (thus $\{R_{\rm H}^{(r)}, r = 1, 2, 3, \ldots\}$) can be regarded as independent and identically distributed. Therefore, we obtain

$$\frac{\sum_{r=1}^{N_{\rm H}} v^{(r)}}{N_{\rm H}} \to \overline{v},\ \text{a.s.},$$

and

$$\frac{\sum_{r=1}^{N_{\rm H}} R_{\rm H,H}^{({\rm N},r)}}{N_{\rm H}} \to \overline{R}_{\rm H,H}^{({\rm N})},\ \text{a.s.}.$$

This gives

$$\begin{aligned}\frac{\sum_{r=1}^{N_{\rm H}} R_{\rm H,H}^{(r)}}{N_{\rm H}} &= \frac{\sum_{r=1}^{N_{\rm H}} v^{(r)}}{N_{\rm H}} + \frac{\sum_{r=1}^{N_{\rm H}} R_{\rm H,H}^{({\rm N},r)}}{N_{\rm H}} \\ &\to \overline{v} + \overline{R}_{\rm H,H}^{({\rm N})}, \qquad \text{a.s.}.\end{aligned}$$

Case two: The ζth dishonest mining pool sets up the main chain in the *r*th round of mining competition for $r = 1, 2, 3, \ldots$.

In this case, the honest mining pool can mine only the orphan blocks, that is, either the uncle blocks or the stale blocks. In addition, it may have a nephew block if the honest mining pool is the first one to mine a block in a round of mining competition, that is, the firstly mined block is the nephew block. Thus, we have

$$R_{\zeta,{\rm H}}^{(r)} = k_\zeta^{(r)} + R_{\zeta,{\rm H}}^{({\rm U},r)} + R_{\zeta,{\rm H}}^{({\rm N},r)},$$

where in the *r*th round of mining competition, $R_{\zeta,{\rm H}}^{({\rm U},r)}$ is the reward of 1 uncle block if the honest mining pool has an uncle block.

Note that the competitively mining processes of the multiple mining pools are repeated round after round, the random variables $k_\zeta^{(r)}$, $R_{\zeta,{\rm H}}^{({\rm U},r)}$ and $R_{\zeta,{\rm H}}^{({\rm N},r)}$ (thus $R_{\zeta,{\rm H}}^{(r)}$) can be repeated by the experiments round after round under the same conditions. Based on this, the random variable sequences $\{k_\zeta^{(r)}, r = 1, 2, 3, \ldots\}$, $\{R_{\zeta,{\rm H}}^{({\rm U},r)}, r = 1, 2, 3, \ldots\}$ and $\{R_{\zeta,{\rm H}}^{({\rm N},r)}, r = 1, 2, 3, \ldots\}$ (thus $\{R_{\zeta,{\rm H}}^{(r)}, r = 1, 2, 3, \ldots\}$) can be regarded as independent and identically distributed. Therefore, we obtain

$$\frac{\sum_{r=1}^{N_\zeta} k_\zeta^{(r)}}{N_\zeta} \to \overline{k}_\zeta,\ \text{a.s.},$$

$$\frac{\sum_{r=1}^{N_\zeta} R_{\zeta,{\rm H}}^{({\rm U},r)}}{N_\zeta} \to \overline{R}_{\zeta,{\rm H}}^{({\rm U})},\ \text{a.s.},$$

and

$$\frac{\sum_{r=1}^{N_\zeta} R_{\zeta,{\rm H}}^{({\rm N},r)}}{N_\zeta} \to \overline{R}_{\zeta,{\rm H}}^{({\rm N})},\ \text{a.s.}.$$

This gives

$$\begin{aligned}\frac{\sum_{r=1}^{N_\zeta} R_{\zeta,{\rm H}}^{(r)}}{N_\zeta} &= \frac{\sum_{r=1}^{N_\zeta} k_\zeta^{(r)}}{N_\zeta} + \frac{\sum_{r=1}^{N_\zeta} R_{\zeta,{\rm H}}^{({\rm U},r)}}{N_\zeta} + \frac{\sum_{r=1}^{N_\zeta} R_{\zeta,{\rm H}}^{({\rm N},r)}}{N_\zeta} \\ &\to \overline{k}_\zeta + \overline{R}_{\zeta,{\rm H}}^{({\rm U})} + \overline{R}_{\zeta,{\rm H}}^{({\rm N})}, \qquad \text{a.s.}.\end{aligned}$$

Therefore, we can get that as $N \to \infty$

$$\begin{aligned}\frac{\sum_{r=1}^{N} R_{\rm H}^{(r)}}{N} &= \frac{\sum_{r=1}^{N_{\rm H}} R_{\rm H}^{(r)}}{N} + \frac{\sum_{\zeta=1}^{m}\sum_{r=1}^{N_\zeta} R_{\zeta,{\rm H}}^{(r)}}{N} \\ &= \frac{\sum_{r=1}^{N_{\rm H}} R_{\rm H}^{(r)}}{N_{\rm H}} \cdot \frac{N_{\rm H}}{N} + \frac{\sum_{\zeta=1}^{m}\sum_{r=1}^{N_\zeta} R_{\zeta,{\rm H}}^{(r)}}{N_\zeta} \cdot \frac{N_\zeta}{N} \\ &= \mathbf{p}_{\rm H}\left(\overline{v} + \overline{R}_{\rm H,H}^{({\rm N})}\right) + \sum_{\zeta=1}^{m} \mathbf{p}_\zeta \left(\overline{k}_\zeta + \overline{R}_{\zeta,{\rm H}}^{({\rm U})} + \overline{R}_{\zeta,{\rm H}}^{({\rm N})}\right).\end{aligned}$$

This completes the proof. ■

IX. Reward Rates Allocated Among the Mining Pools

In this section, we apply the renewal reward processes to study the long-term reward allocation rates among the multiple mining pools.

Note that the competitively mining processes of the multiple mining pools are repeated round after round, as we repeat the experiments under the same conditions. Thus, the moments that one round of mining competition is over and the next round of mining competition begins immediately are all renewal points. Let T_k be the kth moment that the kth round of mining competition is over and the $(k+1)$ th round of mining competition begins immediately. In fact, $[T_{k-1}, T_k)$ represents such a time interval that the kth round of mining competition is underway. For simplicity, we assume that $T_0 = 0$. Let $N(t) = \max\{k, T_k \le t\}$, then $\{N(t), t \ge 0\}$ is a renewal process.

Let $R_{\mathrm{H}}^{(k)}$ and $R_{\zeta}^{(k)}$ be the rewards allocated to the honest mining pool and the ζth dishonest mining pool in the time interval $[T_{k-1}, T_k)$, respectively. Meanwhile, $R_{\mathrm{H}}^{(k)}$ and $R_{\zeta}^{(k)}$ are independent of the time interval $[T_{k-1}, T_k)$. Let $R_{\mathrm{H}}(t)$ and $R_{\zeta}(t)$ be the rewards allocated to the honest mining pool and the ζth dishonest mining pool in the time interval [0, t). The following theorem respectively provides the reward allocation rates to the honest mining pool and the ζth dishonest mining pool in the PoW Ethereum system with multiple mining pools.

Theorem 7: In the PoW Ethereum system with multiple mining pools, if $E[R_{\mathrm{H}}^{(1)}] < +\infty$ and $E[T_1] < +\infty$, then as $t \to \infty$,

$$\frac{R_{\mathrm{H}}(t)}{t} \to \frac{E\left[R_{\mathrm{H}}^{(1)}\right]}{E[T_1]}, \text{ a. s. }, \tag{7}$$

$$\frac{E[R_{\mathrm{H}}(t)]}{t} \to \frac{E\left[R_{\mathrm{H}}^{(1)}\right]}{E[T_1]}; \tag{8}$$

$$\frac{R_{\zeta}(t)}{t} \to \frac{E\left[R_{\zeta}^{(1)}\right]}{E[T_1]}, \text{ a. s. }, \tag{9}$$

$$\frac{E\left[R_{\zeta}(t)\right]}{t} \to \frac{E\left[R_{\zeta}^{(1)}\right]}{E[T_1]}, \tag{10}$$

where

$$\begin{aligned} E\left[R_{\mathrm{H}}^{(1)}\right] &= \mathbf{p}_{\mathrm{H}}\left(\overline{v} + \mathbf{q}_{\mathrm{H,H}}^{(\mathrm{N})}\overline{R}_{\mathrm{H,H}}^{(\mathrm{N})}\right) \\ &\quad + \sum_{\zeta=1}^{m} \mathbf{p}_{\zeta}\left(\overline{k}_{\zeta} + \mathbf{q}_{\zeta,\mathrm{H}}^{(\mathrm{U})}\overline{R}_{\zeta,\mathrm{H}}^{(\mathrm{U})} + \mathbf{q}_{\zeta,\mathrm{H}}^{(\mathrm{N})}\overline{R}_{\zeta,\mathrm{H}}^{(\mathrm{N})}\right), \\ E\left[R_{\zeta}^{(1)}\right] &= \mathbf{p}_{\zeta}\left(\overline{\varphi}_{\zeta} + \mathbf{q}_{\zeta,\zeta}^{(\mathrm{N})}\overline{R}_{\zeta,\zeta}^{(\mathrm{N})}\right) \\ &\quad + \mathbf{p}_{\mathrm{H}}\left(\mathbf{q}_{\mathrm{H},\zeta}^{(\mathrm{U})}\overline{R}_{\mathrm{H},\zeta}^{(\mathrm{U})} + \mathbf{q}_{\mathrm{H},\zeta}^{(\mathrm{N})}\overline{R}_{\mathrm{H},\zeta}^{(\mathrm{N})}\right) \\ &\quad + \sum_{k\neq\zeta}^{m} \mathbf{p}_{k}\left(\mathbf{q}_{k,\zeta}^{(\mathrm{U})}\overline{R}_{\mathrm{k},\zeta}^{(\mathrm{U})} + \mathbf{q}_{k,\zeta}^{(\mathrm{N})}\overline{R}_{\mathrm{k},\zeta}^{(\mathrm{N})}\right). \end{aligned}$$

Proof: We give the proof for equation (7) only. To do this, we write

$$\frac{R_{\mathrm{H}}(t)}{t} = \frac{\sum_{k=1}^{N(t)} R_{\mathrm{H}}^{(k)}}{t} = \frac{\sum_{k=1}^{N(t)} R_{\mathrm{H}}^{(k)}}{N(t)} \cdot \frac{N(t)}{t}.$$

Note that

$$T_k = (T_1 - T_0) + (T_2 - T_1) + \cdots + (T_k - T_{k-1})$$

is the time length of the k renewal periods, and the random variables $T_1 - T_0,\ T_2 - T_1, \ldots, T_k - T_{k-1}$ are independent and identically distributed, $E[T_k - T_{k-1}] = E[T_1 - T_0] = E[T_1]$ for $k \ge 1$. At the same time, the random variables $R_{\mathrm{H}}^{(k)}, k \ge 1$ are also independent and identically distributed, $E[R_{\mathrm{H}}^{(k)}] = E[R_{\mathrm{H}}^{(1)}]$.

Note that $t \to +\infty$, $N(t) \to +\infty$. By the strong law of large numbers, we obtain that as $t \to \infty$

$$\frac{\sum_{k=1}^{N(t)} R_{\mathrm{H}}^{(k)}}{N(t)} \to E\left[R_{\mathrm{H}}^{(1)}\right].$$

According to the elementary renewal theorem ([31, Ch. 7]), we obtain that

$$\frac{N(t)}{t} \to \frac{1}{E[T_1]}, \text{as } t \to +\infty.$$

Therefore, we obtain that as $t \to +\infty$,

$$\frac{R_{\mathrm{H}}(t)}{t} \to \frac{E\left[R_{\mathrm{H}}^{(1)}\right]}{E[T_1]}.$$

This completes the proof. ■

X. Simulation Experiments

In this section, we use some simulation experiments to discuss the Ethereum system with one honest mining pool and two dishonest mining pools, verify how the key probabilities of Ethereum are obtained approximately by using the law of large numbers, and analyze the performance measures of the Ethereum system by means of the renewal reward theorem.

A. An Experiment Design

In order to analyze the competitive mining processes of multiple mining pools, our simulation experiment is designed as follows:

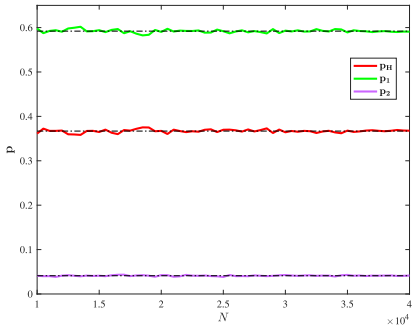
(1) In our simulation, we take a round of mining competition as a sampling, and then we repeat such a sampling round after round.

(2) Under the two-block leading competitive criterion, one dishonest mining pool may release a part of its sub-chain into the Ethereum system. Here, our simulation experiments consider the mining rule: $\omega_1 - \omega_2 = 2$. That is, the longest sub-chain length is 2 blocks longer than the second-longest sub-chain in the system.

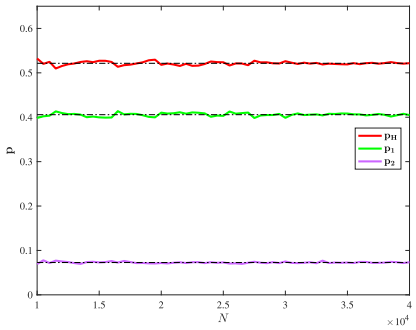
Although $\omega_1 - \omega_2 = 2$ is taken as a terminate rule to end a round of mining competition among the multiple mining pools, the dishonest mining pools can fork at any position of the sub-chain mined by the honest mining pool. Thus, the tree with multiple block sub-chains is still more complicated, and also represents the general practical mining structure of the PoW Ethereum system with multiple mining pools.

(3) For simplicity of simulation, our design is to consider the mining processes of the multiple mining pools. To do this, we take $m+1$ random variables as

$$X_i = X \cdot \left(\frac{1}{\alpha_i} + \frac{1}{\gamma}\right), i = 0, 1, 2, \ldots, m,$$



(a) $\alpha_{\mathrm{H}} = 0.6, \alpha_1 = 0.3, \alpha_2 = 0.1$



(b) $\alpha_{\mathrm{H}} = 0.7, \alpha_1 = 0.2, \alpha_2 = 0.1$

Fig. 12. The probabilities are approximately computed by the law of large numbers.

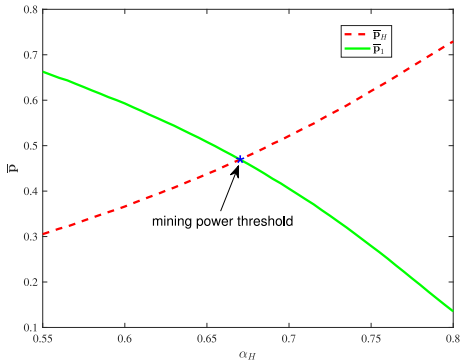


Fig. 13. The average probabilities and the mining power threshold.

where, X_i is the block-generating and block-pegging time of the *i*th mining pool, α_i is the mining power of the *i*th mining pool, and γ is the communication ability of the P2P network. Note that $1/\alpha_i + 1/\gamma$ is used to show the independence between the mining power and the communication ability. At the same time, it is easy to see that X_i decreases as α_i or γ increases, this is consistent with our intuitive understanding of the mining times.

We assume that the random variable *X* obeys an exponential distribution of the mean 15 seconds, where the 15 seconds are always chosen as the expected mining time of one block in the PoW Ethereum system. Let *d* be a random number generated by the exponential distribution of *X*, then we have

$$d_i = d \cdot (\frac{1}{\alpha_i} + \frac{1}{\gamma}), i = 0, 1, 2, \ldots, m.$$

B. Simulation and Results

In this subsection, we describe and analyze some interesting simulation results.

(1) *The law of large numbers:* Note that the probability that each mining pool wins the mining competition plays a key role in our research on the key ratios, the growth rate of blockchain, the reward allocation rates and so on. Here, it is necessary to verify how this probability is obtained approximately by using the law of large numbers. To this end, we take $N \in [10000, 40000]$ and $\gamma = 10$.

From Fig. 12, it is easy to see that the three probabilities: $\mathbf{p}_{\mathrm{H}}$, $\mathbf{p}_1$,and $\mathbf{p}_2$ fluctuate around a certain value. This shows that the law of large numbers is well applied to determine these probabilities.

From Fig. 12(a), it is seen that $\mathbf{p}_1 > \mathbf{p}_{\mathrm{H}}$, while $\mathbf{p}_1 < \mathbf{p}_{\mathrm{H}}$ in Fig. 12(b). It shows that as α_{H} increases, there exists an α^*_{H}, such that $\mathbf{p}_1 = \mathbf{p}_{\mathrm{H}}$. Furthermore, it is observed that as α_{H} increases, the probability $\mathbf{p}_1 + \mathbf{p}_2$ of the dishonest mining pools decreases. Thus, the influence of the dishonest pools decreases as α_{H} increases.

In order to observe the interesting value α^*_{H}, we use a special experiment. Let $N = 20000, \gamma = 10, \alpha_2 = 0.1$, $\alpha_{\mathrm{H}} \in [0.55, 0.8]$, $\alpha_1 = 1 - \alpha_{\mathrm{H}} - \alpha_2$. Each of our simulations with 20000 rounds of mining competition is repeated 100 times to calculate the average of the approximate probabilities $\mathbf{p}_{\mathrm{H}}$ and $\mathbf{p}_1$. We denote the two average values by $\overline{\mathbf{p}}_{\mathrm{H}}$ and $\overline{\mathbf{p}}_1$. The results are shown in Fig. 13.

From Fig. 13, it is seen that as α_{H} increases, the average probability $\overline{\mathbf{p}}_{\mathrm{H}}$ increases, and the average probability $\overline{\mathbf{p}}_1$ decreases. Also, it is observed that there exists an α^*_{H}, when $\alpha_{\mathrm{H}} < \alpha^*_{\mathrm{H}}$, $\overline{\mathbf{p}}_1 > \overline{\mathbf{p}}_{\mathrm{H}}$; and when $\alpha_{\mathrm{H}} > \alpha^*_{\mathrm{H}}$, $\overline{\mathbf{p}}_1 < \overline{\mathbf{p}}_{\mathrm{H}}$. In our simulation experiments, we obtain that the 95% confidence interval of the mining power α^*_{H} is [0.6662, 0.6731]. This shows that when the mining power of the 1st dishonest mining pool exceeds the mining power threshold of 23.38%, the 1st dishonest pool has the biggest probability of setting up the main chain. Based on this, the dishonest mining pool with a smaller mining power can have the same probability of setting up the main chain as the honest mining pool. This is why the dishonest mining pool may fork at any position of the honest sub-chain.

In the remainder of this subsection, we explore how the performance measures of the Ethereum system depend on the mining powers of the three mining pools. To this end, we take the parameters as follows: The mining power of the honest mining pool $\alpha_{\mathrm{H}} \in [0.51, 0.82]$, the mining power of the 2nd dishonest mining pool $\alpha_2 = 0.13$, and the mining power of the 1st dishonest mining pool $\alpha_1 = 1 - \alpha_{\mathrm{H}} - \alpha_2$, the rate at which the block is pegged to the corresponding sub-chain $\gamma = 10$, and the numbers of mining competition rounds are $N_1 = 15000$, $N_2 = 18000, N_3 = 20000, N_4 = 25000, N_5 = 30000$.

(2) *Some key probabilities vs. α_H:* From Fig. 14, it is seen that each of the probabilities is stably close to a certain

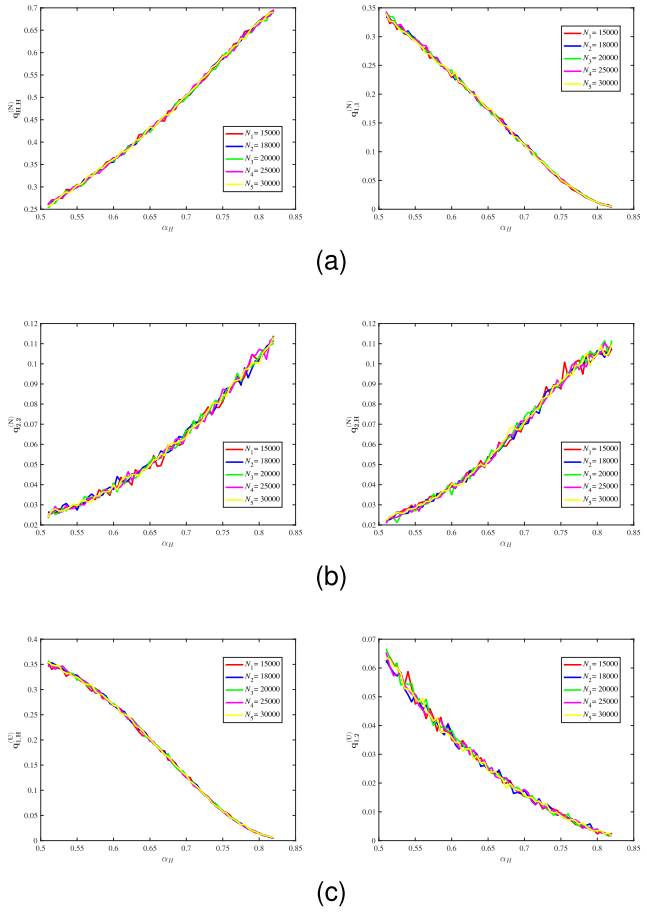


Fig. 14. Some probabilities that each main chain contains uncle or nephew blocks.

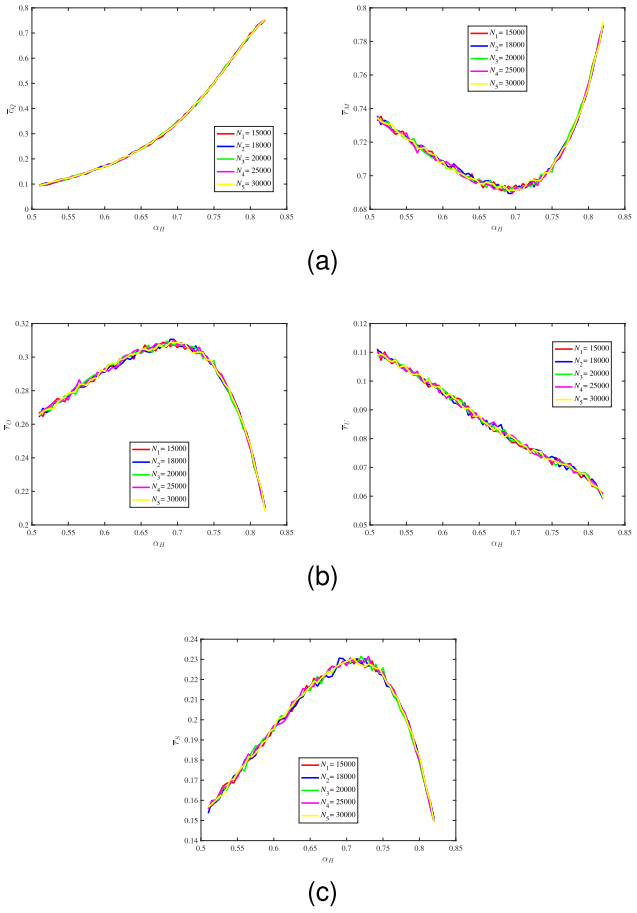


Fig. 15. The key ratios of blockchain.

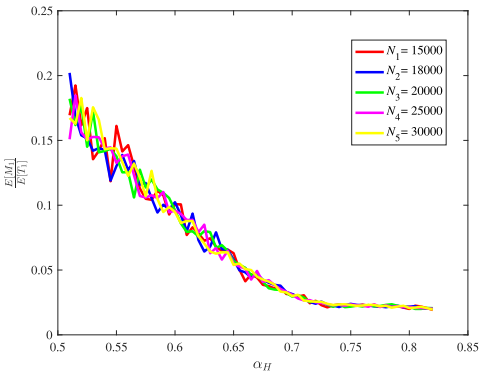


Fig. 16. The growth rate of blockchain.

value. This shows that the law of large numbers is successfully applied in our computation. See Theorem 3 for a comparison.

(3) *Some key ratios vs. α_H:* From Fig. 15, it is observed that for some different values of *N*, the key ratios $\overline{c}_{\rm Q}$, $\overline{r}_{\rm M}$, $\overline{r}_{\rm O}$, $\overline{r}_{\rm U}$, and $\overline{r}_{\rm S}$ are all approximately stable in our computation by means of the law of large numbers. Also, the chain quality $\overline{c}_{\rm Q}$ increases as $\alpha_{\rm H}$ increases, while the uncle block ratio $\overline{r}_{\rm U}$ decreases as the $\alpha_{\rm H}$ increases. For the main chain length ratio $\overline{r}_{\rm M}$, it first decreases and then increases as the $\alpha_{\rm H}$ increases. For the stale block ratio $\overline{r}_{\rm S}$ (resp. $\overline{r}_{\rm O}$), it first increases and then decreases as $\alpha_{\rm H}$ increases. It shows from $\overline{r}_{\rm M}$, $\overline{r}_{\rm O}$ and $\overline{r}_{\rm S}$ that when the mining power of the honest mining pool is close to 0.7, the mining competition among the three mining pools is the most intense, so a lot of mining resources are wasted.

(4) *The growth rate of blockchain vs. α_H:* From Fig. 16, we can see that for some different values of *N*, the growth rate of the blockchain $lim_{t\to+\infty}M(t)/t = E[M_1]/E[T_1]$ can be effectively computed by means of the law of large numbers and the renewal theorem. In addition, there exists a value $\alpha_0 \in (0,1)$ such that when $\alpha_{\rm H} < \alpha_0$, the growth rate of blockchain decreases as $\alpha_{\rm H}$ increases; while when $\alpha_{\rm H} > \alpha_0$, the growth rate of blockchain is almost unchanged as $\alpha_{\rm H}$ increases. It indicates that the growth rate of blockchain reaches the lowest level once the honest mining pool masters the major mining power of the entire network. Also, it shows from Fig. 16 that the total mining power of the Ethereum system is dispersed into multiple mining pools benefits the growth rate of blockchain.

(5) *The reward of the honest mining pool vs. α_H:* Fig. 17 shows the reward of the honest mining pool. For some different values of *N*, the reward $\overline{R}_{\rm H}$ can be approximately computed by means of the law of large numbers. Also, the reward $\overline{R}_{\rm H}$ increases as $\alpha_{\rm H}$ increases, which is consistent with the fact that the reward of the honest mining pool is positively correlated with its mining power.

(6) *The reward allocation rate of the honest mining pool vs. α_H:* Fig. 18 shows the reward allocation rate of the honest

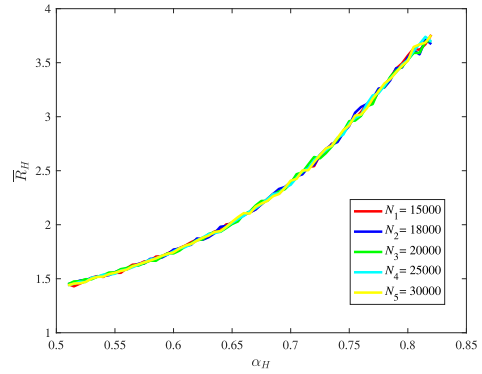


Fig. 17. The reward obtained by the honest mining pool.

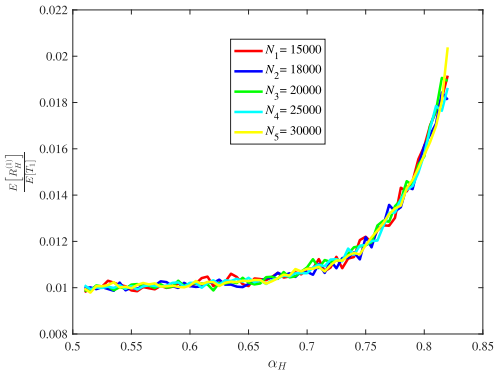


Fig. 18. The reward allocation rate of the honest mining pool.

mining pool. For different values of N, the reward allocation rate of the honest mining pool $lim_{t\to+\infty} R_{\mathrm{H}}(t)/t = E[R_{\mathrm{H}}^{(1)}]/E[T_1]$ can be effectively computed by using the law of large numbers and the renewal theorem. Also, the reward allocation rate increases as α_{H} increases, which is also consistent with the fact that the reward allocation rate of the honest mining pool is positively correlated with its mining power.

XI. Conclusion

The growth of the PoW Ethereum system with multiple mining pools has created the need for not only the development of blockchain technology but also for setting up a general mathematical representation of tree and dealing with the multi-dimensional stochastic systems related to the multiple block branches of tree. In general, the research on such a tree and associated mathematical analysis is very difficult and challenging. It is worthwhile to note that our mathematical representation of tree is the first one in the study of blockchain with multiple mining pools, and it is different from that tree of the GHOST protocol given in Sompolinsky and Zohar [32], [33].

For a blockchain system with two mining pools, Eyal and Sirer [12] found the selfish mining and constructed a simple tree with two block branches. Based on their analysis and assumptions, this simple tree structure has also been analyzed by other researchers, including Li *et al.* [21] who established the two-dimensional Markov (reward) processes to analyze the efficiency and benefit of blockchain. However, so far little research has worked on the blockchain systems with multiple mining pools to answer many questions such as how to mathematically represent a general tree with multiple block branches and how to analyze a complicated multi-dimensional stochastic system running on the general tree. It is obvious that the study of PoW Ethereum system with multiple mining pools will need to apply the multi-dimensional stochastic processes on a general tree, even simply, we will need to apply the fluid and diffusion approximations on a general tree.

In this paper, we describe a PoW Ethereum system with multiple mining pools, which is controlled by the two-block leading competitive criterion proposed in Li *et al.* [21]. Here, a block branch will be generated by only one mining pool, Thus the mining competition among the multiple mining pools can generate a general tree with multiple block branches. When observing the general tree, one of our key findings is to learn that the block branches of the multiple dishonest mining pools can be forked at any (different) positions of the block branch of one honest mining pool. Based on this, we can provide a mathematical representation for the general tree with multiple block branches. Also, we can easily determine the main chain by means of the principle of longest chain, e.g., see Li *et al.* [21] for the blockchain with multiple mining pools.

By using the tree representation and observing multiple rounds of mining competitions, we can provide a block classification of Ethereum: Regular blocks (i.e., the main chain), orphan blocks, uncle blocks, stale blocks, and nephew blocks, and set up an approximate computation for the key ratios and probabilities of generating the different types of blocks by applying the law of large numbers. Based on the key probabilities, together with the tree representation, we develop an economic framework for computing the rewards allocated to the multiple mining pools. This is one of our key theoretical findings in the study of the PoW Ethereum system with multiple mining pools.

By applying the renewal reward theorem, we further discuss the growth rate of blockchain, the reward allocation among the multiple mining pools, and the reward rates allocated among multiple mining pools, three of which become the key performance measures of PoW Ethereum system with multiple mining pools. Furthermore, we use simulation experiments to verify our theoretical results, and show that our approximate computation is fast and effective for dealing with the three performance measures. Therefore, this paper provides a powerful tool for the performance evaluation of the PoW Ethereum system with multiple mining pools.

To the best of our knowledge, this paper is the first one to provide the mathematical representation of the general tree, and to analyze the PoW Ethereum system with multiple mining pools through applying the law of large numbers and the renewal reward theorem. Therefore, we hope that our methodology and results given in this paper are applicable to the study of more general PoW Ethereum systems with multiple mining pools. Along the research line, there are still a number of interesting directions for future research:

- Setting up a new tree representation for the PoW Ethereum system with multiple honest mining pools and multiple dishonest mining pools. In the more complicated

case, how to determine the main chain from such a tree? How to give the performance evaluation of the PoW Ethereum systems?

- Developing some more effective simulation techniques in the study of the PoW Ethereum system with multiple (honest and dishonest) mining pools through applying the law of large numbers and the renewal reward theorem.
- Developing fluid approximation and/or diffusion approximation to analyze the PoW Ethereum system with multiple (honest and dishonest) mining pools.
- Providing optimal methods and dynamic control (e.g., Markov decision processes and stochastic game) in the study of PoW Ethereum system with multiple (honest and dishonest) mining pools.

REFERENCES

[1] N. Atzei, M. Bartoletti, and T. Cimoli, "A survey of attacks on Ethereum smart contracts (SOK)," in *Proc. Int. Conf. Principles Security Trust*, 2017, pp. 164–186.

[2] L. Augusto, R. Costa, J. Ferreira, and R. Jardim-Gonçlves, "An application of Ethereum smart contracts and IoT to logistics," in *Proc. Int. Young Eng. Forum*, 2019, pp. 1–7.

[3] Y. N. Aung and T. Tantidham, "Ethereum-based emergency service for smart home system: Smart contract implementation," in *Proc. 21st Int. Conf. Adv. Commun. Technol.*, 2019, pp. 147–152.

[4] Q. Bai, X. Zhou, X. Wang, Y. Xu, X. Wang, and Q. Kong, "A deep dive into blockchain selfish mining," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2019, pp. 1–6.

[5] A. Bogner, M. Chanson, and A. Meeuw, "A decentralised sharing app running a smart contract on the Ethereum blockchain," in *Proc. 6th Int. Conf. the Internet Things*, 2016, pp. 177–178.

[6] V. Buterin. "Ethereum Whitepaper." 2013. [Online]. Available: https://github.com/Ethereum/wiki/wiki/White-Paper

[7] V. Buterin. "A Next-Generation Smart Contract and Decentralized Application Platform." 2014. [Online]. Available: https://github.com/Ethereum/wiki/wiki/White-Paper.

[8] S. Y. Chang, Y. Park, S. Wuthier, and C. W. Chen, "Uncle-block attack: Blockchain mining threat beyond block withholding for rational and uncooperative miners," in *Proc. Int. Conf. Appl. Cryptogr. Netw. Security*, 2019, pp. 241–258.

[9] H. Chen, M. Pendleton, L. Njilla, and S. Xu, "A survey on Ethereum systems security: Vulnerabilities, attacks, and defenses," *ACM Comput. Surveys*, vol. 53, no. 3, pp. 1–43, 2020.

[10] M. D. Angelo and G. Salzer, "A survey of tools for analyzing Ethereum smart contracts," in *Proc. IEEE Int. Conf. Decentralized Appl. Infrastruct.*, 2019, pp. 69–78.

[11] A. Dika and M. Nowostawski, "Security vulnerabilities in Ethereum smart contracts," in *Proc. IEEE Int. Conf. Internet Things IEEE Green Comput. Commun. IEEE Cyber Phys. Social Comput. IEEE Smart Data*, 2018, pp. 955–962.

[12] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," *Commun. ACM*, vol. 61, no. 7, pp. 95–102, 2018.

[13] C. Feng and J. Niu, "Selfish mining in Ethereum," in *Proc. 39th IEEE Int. Conf. Distrib. Comput. Syst.*, 2019, pp. 1306–1316.

[14] C. Grunspan and R. Pérez-Marco, "Selfish mining in Ethereum," in *Mathematical Research for Blockchain Economy*. Cham, Switzerland: Springer, 2020, pp. 65–90.

[15] P. Jain, "Revenue generation strategy through selfish mining focusing multiple pools of honest miners," M.S. thesis, Dept. Comput. Sci. Appl. Math., Indraprastha Inst. Inf. Technol., New Delhi, India, 2019.

[16] H. Kang, X. Chang, R. Yang, J. Mišić, and V. B. Mišić, "Understanding selfish mining in imperfect bitcoin and Ethereum networks with extended forks," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 3, pp. 3079–3091, Sep. 2021. doi: 10.1109/TNSM.2021.3073414.

[17] T. Leelavimolsilp, L. Tran-Thanh, and S. Stein, "On the preliminary investigation of selfish mining strategy with multiple selfish miners," 2018, *arXiv:1802.02218*.

[18] T. Leelavimolsilp, V. H. Nguyen, S. Stein, and L. Tranthanh, "Selfish mining in Proof-of-Work blockchain with multiple miners: An empirical evaluation," in *Proc. Int. Conf. Principles Pract. Multi-Agent Syst.*, 2019, pp. 219–234.

[19] S. D. Lerner. "Uncle Mining, An Ethereum Consensus Protocol Flaw." 2016. [Online]. Available: https://bitslog.wordpress.com/2016/04/28/uncle-mining-an-Ethereum-consensus-protocol-aw

[20] Q. L. Li, J. Y. Ma, and Y. X. Chang, "Blockchain queue theory," in *Proc. Int. Conf. Comput. Social Netw.*, 2018, pp. 25–40.

[21] Q. L. Li, Y. X. Chang, X. Wu, and G. Zhang, "A new theoretical framework of pyramid markov processes for blockchain selfish mining," *J. Syst. Sci. Syst. Eng.*, vol. 30, no. 6, pp. 667–711, 2021.

[22] H. Liu, N. Ruan, R. Du, and W. Jia, "On the strategy and behavior of Bitcoin mining with N-attackers," in *Proc. Asia Conf. Comput. Commun. Security*, 2018, pp. 357–368.

[23] Y. Liu, Y. Hei, T. Xu, and J. Liu, "An evaluation of uncle block mechanism effect on Ethereum selfish and stubborn mining combined with an eclipse attack," *IEEE Access*, vol. 8, pp. 17489–17499, 2020.

[24] F. J. Marmolejo-Cossío, E. Brigham, B. Sela, and J. Katz, "Competing (semi-) selfish miners in bitcoin," in *Proc. 1st ACM Conf. Adv. Financ. Technol.*, 2019, pp. 89–109.

[25] A. H. Mohammed, A. A. Abdulateef, and I. A. Abdulateef, "Hyperledger, Ethereum and blockchain technology: A short overview," in *Proc. 3rd Int. Congr. Human-Comput. Interact. Optim. Robot. Appl.*, 2021, pp. 1–6.

[26] S. Nakamoto. "Bitcoin: A Peer-to-Peer Electronic Cash System." 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[27] C. Pinzón, C. Rocha, and J. Finke, "Algorithmic analysis of blockchain efficiency with communication delay," in *Fundamental Approaches to Software Engineering*. Cham, Switzerland: Springer, 2020, pp. 400–419.

[28] P. Praitheeshan, L. Pan, J. Yu, and R. Doss, "Security analysis methods on Ethereum smart contract vulnerabilities: A survey," 2019, *arXiv:1908.08605*.

[29] V. P. Ranganthan, R. Dantu, A. Paul, P. Mears, and K. Morozov, "A decentralized marketplace application on the Ethereum blockchain," in *Proc. IEEE 4th Int. Conf. Collaboration Internet Comput.*, 2018, pp. 90–97.

[30] F. Ritz and A. Zugenmaier, "The impact of uncle rewards on selfish mining in Ethereum," in *Proc. IEEE Eur. Symp. Security Privacy Work-Shops*, London, U.K., 2018, pp. 50–57.

[31] S. Ross, *Introduction to Probability Models*. Amsterdam, The Netherlands: Academic, 2014.

[32] Y. Sompolinsky and A. Zohar, "Accelerating bitcoin's transaction processing. Fast money grows on trees, not chains," Int. Assoc. Cryptol. Res., Lyon, France, Rep. 2013/881, 2013.

[33] Y. Sompolinsky and A. Zohar, "Secure high-rate transaction processing in bitcoin," in *Proc. Int. Conf. Financ. Cryptogr. Data Security*, 2015, pp. 507–527.

[34] M. Sookhak, M. R. Jabbarpour, N. S. Safa, and F. R. Yu, "Blockchain and smart contract for access control in healthcare: A survey, issues and challenges, and open issues," *J. Netw. Comput. Appl.*, vol. 178, Mar. 2021, Art. no. 102950.

[35] D. Vujičić, D. Jagodić, and S. Randić, "Blockchain technology, bitcoin, and Ethereum: A brief overview," in *Proc. 17th Int. Symp. Infoteh-Jahorina*, 2018, pp. 1–6.

[36] S. Wang, Y. Yuan, X. Wang, J. Li, R. Qin, and F. Y. Wang, "An overview of smart contract: Architecture, applications, and future trends," in *Proc. IEEE Intell. Veh. Symp.*, 2018, pp. 108–113.

[37] Z. Wang, H. Jin, W. Dai, K. K. R. Choo, and D. Zou, "Ethereum smart contract security research: Survey and future research opportunities," *Front. Comput. Sci.*, vol. 15, no. 2, pp. 1–18, 2021.

[38] Z. Wang, J. Liu, Q. Wu, Y. Zhang, H. Yu, and Z. Zhou, "An analytic evaluation for the impact of uncle blocks by selfish and stubborn mining in an imperfect Ethereum network," *Comput. Security*, vol. 87, Nov. 2019, Art. no. 101581.

[39] S. M. Werner, P. J. Pritz, A. Zamyatin, and W. J. Knottenbelt, "Uncle traps: Harvesting rewards in a queue-based Ethereum mining pool," in *Proc. 12th EAI Int. Conf. Perform. Eval. Methodol. Tools*, 2019, pp. 127–134.

[40] G. Wood, "Ethereum: A secure decentralized generalised transaction ledger EIP-150 revision," Ethereum, Bern, Switzerland, Yellow Paper, 2017.

[41] Q. Xia *et al.*, "The impact analysis of multiple miners and propagation delay on selfish mining," in *Proc. IEEE 45th Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, 2021, pp. 694–703.

[42] S. Zhang, "Analyzing the success of selfish mining with multiple players," M.S. thesis, School Comput. Sci., McGill Univ., Montreal, QC, Canada, 2020.

[43] S. Zhang, K. Zhang, and B. Kemme, "A simulation-based analysis of multiplayer selfish mining," in *Proc. IEEE Int. Conf. Blockchain Cryptocurrency*, 2020, pp. 1–5.
[44] S. Zhang, K. Zhang, and B. Kemme, "Analysing the benefit of selfish mining with multiple players," in *Proc. IEEE Int. Conf. Blockchain*, 2020, pp. 36–44.

Quan-Lin Li received the Ph.D. degree from the Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing, China. He is a Full Professor with the School of Economics and Management Sciences, Beijing University of Technology, Beijing, China. He has published an English monograph *Constructive Computation in Stochastic Models with Applications: The RG-Factorizations* (Springer, 2010), has edited three proceedings of international conferences (2012, 2017, and 2019) by Springer, and has published over 60 research papers in a variety of international journals, such as, *Advances in Applied Probability*, *Queueing Systems*, *Stochastic Models*, *European Journal of Operational Research*, *Computer Networks*, *Performance Evaluation*, *Discrete Event Dynamic Systems*, *Computers & Operations Research*, *Computers & Mathematics with Applications*, *Annals of Operations Research*, and *International Journal of Production Economics*. His current research interests include stochastic models, stochastic processes, the mean-field theory, stochastic process algebra, game theory, queueing networks, computer networks, resource management in big networks, health care systems, mathematical models of blockchain, and sharing economy.

Yan-Xia Chang is pursuing the Ph.D. degree with the School of Economics and Management, Beijing University of Technology, Beijing, China. Her current research interests include blockchain systems, resource management of big networks, queueing networks, the mean-field theory, and service systems.

Chi Zhang received the B.S. degree in industrial engineering and the M.S. degree in management science and engineering from Xian Jiaotong University and the Ph.D. degree in systems engineering from the Stevens Institute of Technology. He is currently an Associate Professor with the School of Economics and Management, Beijing University of Technology. His research efforts focus on complex networked systems reliability analysis and optimization, maintenance optimization, critical infrastructure resilience, and warranty policy optimization.