An Analytical Study of Selfish Mining Attacks on Chainweb Blockchain

Suyang Wang, Bo Yin, Shuai Zhang, Yu Cheng
Department of Electrical and Computer Engineering, Illinois Institute of Technology, USA 60616
Email: $\{swang133, byin, szhang104\}$*@hawk.iit.edu;* $\{cheng\}$*@iit.edu*

2022 19th Annual International Conference on Privacy, Security & Trust (PST) | 978-1-6654-7398-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/PST55820.2022.9851985

***Abstract*—Chainweb and some other parallel blockchain systems have recently been proposed, with the objectives of improving the throughput and enhancing the tamper-proof capability. While many security related studies have been conducted for traditional single-chain based blockchain systems, the security aspect of parallel chain systems is yet to be well studied and understood. Our paper presents a systematic study on selfish mining attacks in Chainweb based on mathematical modeling. Specifically, selfish mining is conducted by concentrating the computation power on a subset of parallel chains and operating a proper withholding strategy. We demonstrate how to establish a Markov chain based analytical model with innovative techniques to handle the very large state space. Our Markov chain model is also capable of handling different number of parallel chains. The mathematical analysis brings an insightful, in fact counterintuitive, finding that the attackers need less computation power to harvest additional rewards through withholding when Chainweb contains a larger number of chains; while the common understanding is that the more chains are used, the more tamper-proof the system is. The accuracy of the Markov chain analysis is demonstrated via comparison to the simulation results.**

***Index Terms*—blockchain, Proof-of-Work, Chainweb, scalability, mining attacks.**

I. Introduction

Since the seminal work of Satoshi Nakamoto in 2008 [1], blockchain has been incorporated into a wide variety of applications such as electronic payment, asset management, wireless network, and Internet of things [2]. The key enabler of blockchain is a consensus protocol based on the cryptographic hash named proof-of-work (PoW). In order to achieve stable consensus under the impact of the propagation delay over the global Internet, the PoW design in classic Bitcoin on average allows a block generation rate of one block every 10 minutes, with the block size up to 1 megabyte. Such a low transaction processing rate is known as the scalability or the throughput issue and significantly hinders blockchain's practical use.

People have made efforts from different angles to address the throughput issue. The directed acyclic graph (DAG) method [3], [4] allows blocks not necessarily on the main chain to contribute to transaction confirmation. The Prism mechanism proposed in [5], [6] achieves the best throughput and a confirmation latency bounded only by the physical limits. OHIE [7] claims to have higher throughput than that in Bitcoin, however, without much improvement in latency. The sharding approaches in [8]–[10] increases the throughput of blockchain by dividing participants into groups that achieve parallel processing of transactions. Some studies propose to leverage the help of off-chain payment [11], [12]. The basic idea is to establish payment channels between nodes without immediate committing transactions. The parallel chain techniques [5]–[7], [13], [14] offer a complementary dimension over the single-chain techniques to address the throughput issue. The throughput is expected to linearly increase with the number of chains exploited in the blockchain design. Among the parallel chain protocols, Chainweb [14] is of our particular interest. It has the advantage that the throughput increases linearly as the number of chains grows with the total mining power requirement unchanged. In addition, it designs a cross-referencing feature that enhances the capability of resisting hostile forks.

In blockchain, security related issues are of equal importance as the throughput. In blockchain, mining blocks to get block rewards is in essence a computation power competition among the miners. Therefore, mining attacks related studies are of great importance, revealing that the strategy of utilizing the computation power can significantly impact the reward and offer insights on enhancing the mining protocol design. For example, the selfish mining proposed in [15] allows attackers to get higher benefits than their fair share through rationally keeping and releasing the secretly mined blocks according to the number of leading blocks. The block withholding attack (BWH) and its advanced version fork after withholding attack (FAW) have been proposed and studied in [16]–[19]. In stubborn mining, an attacker gains higher block rewards by not easily giving up the secret chain [20]. The denial of service (DoS) related issues such as routing attacks and eclipse attacks have also been studied in [21] and [22]. We would like to emphasize that the existing security studies on blockchain were mainly conducted for single chain based systems. There are very few references about security analysis for parallel chain systems. For example, the study in [23] analyzes the consistency of Cliquechain, a variation of Chainweb with only 2-chain and 3-chain structures. It is worth noting that most of the security analysis from single chain scenarios cannot be directly extended to parallel blockchains, where different chains are not just operating independently but under certain interplaying rules.

This paper presents a systematic study on selfish mining attacks in the Chainweb system. We fully describe the procedure to launch a selfish mining attack in Chainweb and demonstrate how to develop a Markov chain model for analysis. Establish-

ing such a model is very challenging due to the interactions between the closely coupled multiple chains. Following the protocol, Chainweb miners would allocate their mining power evenly on all minable blocks. However, during the mining procedure, the miners' power allocation changes rapidly with the dynamics of the number and location of existing blocks. Introducing selfish miners will further aggravate the complex situation. To deal with the complexity, we aim to focus on the long-term block distribution on the attackers' target chains. We delicately design a three-dimensional Markov chain to describe the system's behavior. Specifically, each 3-d state vector consists of the aggregate number of blocks secretly mined by attackers over the subset of chains being attacked, the aggregate number of blocks released by normal miners over the attacked chains, and the aggregate number of blocks released over the chains other than the attacked ones. Even though, the state space in the Markov chain model is still large, requiring much effort to solve the problem to be developed in this paper.

Our model is capable of analyzing Chainweb with an arbitrary number of parallel chains as long as the size of the target chain set is 4. The Markov chain based analysis brings an insightful, in fact counterintuitive finding that the attackers need less computation power to harvest additional rewards through withholding when Chainweb contains a larger number of chains; while the common understanding is that the more chains are used, the more tamper-proof the system is. Our analysis will give technical explanations for this interesting finding. The main contributions of this paper are summarized as follows.

1) We demonstrate in Chainweb how the selfish mining attackers can concentrate their mining power on part of chains, and strategically conduct either selfish mining or normal mining according to the system state for long-term reward gains.
2) We develop a Markov chain model for studying the selfish mining attacks on Chainweb. This model captures the complex interactions among the attackers and normal miners in the system.
3) We utilize the Markov chain to quantitatively and rigorously analyze the reward gain by selfish mining. The analytical results give us a counterintuitive finding that the attackers would succeed easier when Chainweb has a larger number of chains.
4) We conduct simulations with different chain settings. The simulation results match our analytical results well, justifying the effectiveness and accuracy of the mathematical analysis.

The remainder of this paper is organized as follows. Section II gives the background of Chainweb. Section III describes the selfish mining attack model. Section IV develops the Markov chain modeling. Section V shows the numerical results and the performance analysis. Section VI reviews more related work. Section VII concludes this paper.

II. Overview of Chainweb

This section briefly introduces the PoW, blockchain mining, and Chainweb mining.

A. PoW and blockchain mining

PoW is a consensus protocol that enables blockchain participants to validate new blocks and maintain the common block history. The action of trying to generate new blocks is called block mining. The blockchain participants who involve working on generating new blocks are called block miners. PoW requires miners to compute the hash of a random number (nonce) combined with the transactions and other metadata to check if the hashed value meets the criteria. If so, the miner successfully mines one block and immediately broadcasts it. Otherwise, the miner needs to try a different nonce until she/he finds a valid block or receives a valid block from other miners. Once a new block arrives at all miners and is verified as valid, the length of the blockchain is extended by 1. If miners receive more than one valid block with the same *block height* (the number of blocks preceding a particular block) in a small time interval, they may choose to mine on one of the new blocks. In this way, PoW assures that no miner can generate a valid block without devoting enough effort. As a result, it is nearly impossible for an adversary to develop a whole counterfeit chain to replace the genuine chain. The blockchain's tamper-proof property comes from here.

B. Chainweb Mining

Unlike traditional blockchain mining, Chainweb miners can mine blocks on different chains simultaneously. The transaction processing rate increases linearly with the number of chains. Chainweb was originally launched with a 10-chain configuration and then extended to 20 chains with all the history blocks reserved. The developers can further increase the number of chains to meet the needs of the transaction processing rate. Chainweb binds all the chains together using a cross-referencing feature. The cross-referencing relationships are demonstrated using undirected graphs. Since settings with different number of chains have different graphs, for the illustration purpose, we redraw the originally 10-chain graph in the Chainweb white paper [14] to show the relationships. In Fig. 1, each vertex represents a single chain, and the edges between the vertices indicate the mutual cross-referencing relationships. For example, The four red vertices show that *Chain 3* cross-references *Chain 1, 5, and 8*. Suppose a miner wants to mine a block on a certain chain with block height $N+1$, the miner needs to refer to not only the block on this chain with block height N, but also the blocks on its cross-referencing chains with block height N.

Fig. 2a shows the mining process in Chainweb blockchain. The blocks indicated by vertices on each vertical line are with the same block height. The same-chain referencing and cross-chain referencing are shown by the solid arrows and dotted arrows, respectively. The block on $Chain\ m$ with block

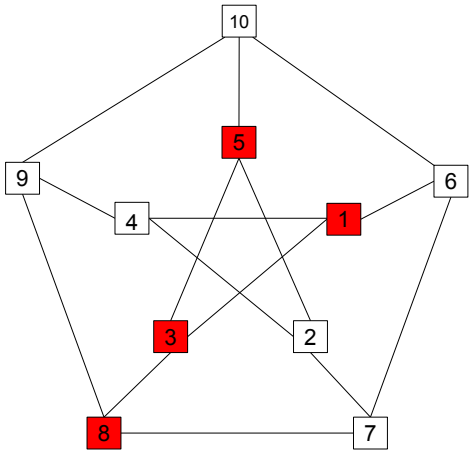
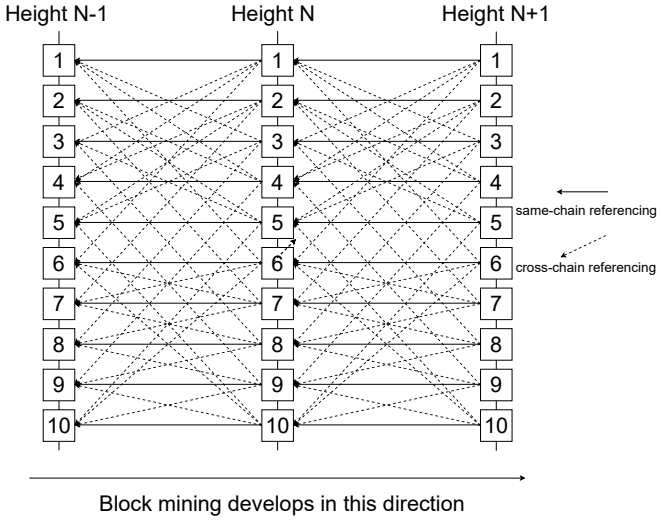
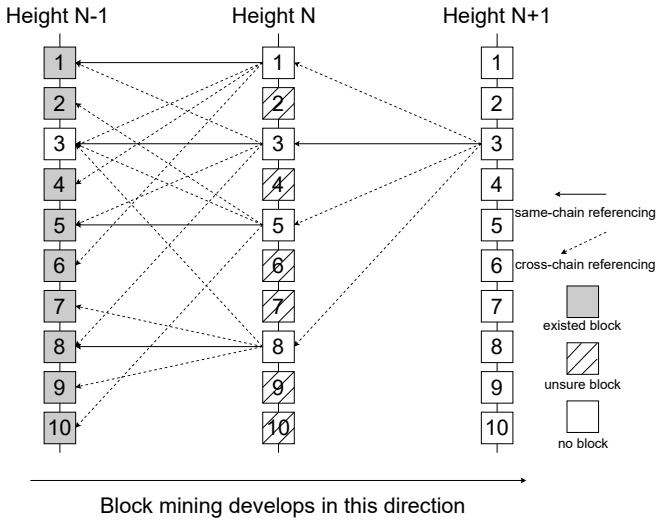


Fig. 1: A graph of 10-chain chainweb showing cross-referencing relationship.

height n can be uniquely identified as $\langle m, n\rangle$. According to Fig. 1, $Chain$ 3 cross-references with $Chain$ 1, 5, and 8. Also, as shown in Fig. 2a, the block $\langle 3, N+1\rangle$ can be mined only when the blocks $\langle 1, N\rangle, \langle 3, N\rangle, \langle 5, N\rangle$ and $\langle 8, N\rangle$ are available, which further requires the availability of all the blocks with block height $N-1$. In this way, no chain can be extended more than two blocks away from the shortest chain. Chainweb miners follow similar PoW rules to generate new blocks. Normally, miners can choose to do block mining on arbitrary chains. However, suppose a miner constantly mines on a specific chain. In that case, she/he may encounter a situation where no block on that chain is minable due to the aforementioned cross-referencing relationship. Considering that each newly mined block would be released immediately and finally propagated through the entire network, the optimal mining strategy is to allocate identical mining power to all minable blocks among all chains. As the number of minable blocks fluctuates, miners will redistribute their mining power accordingly.



(a) An illustration of Chainweb mining.



(b) An illustration of selfish attack on Chainweb.

Fig. 2: An illustration of Chainweb mining and selfish attack.

III. SELFISH MINING IN CHAINWEB

In this section, we illustrate selfish mining in single-chain blockchain and Chainweb blockchain, respectively.

A. Selfish Mining in Single-chain Blockchain

Bitcoin miners follow the longest chain rule. As the name indicates, miners would accept the longest chain when there is more than one conflicting chain (fork). When all miners obey this rule, one exclusive chain is deemed as the valid chain. However, instead of obeying the rules, a selfish miner may mine blocks without immediately broadcasting newly generated blocks. The selfish miner would observe the growth of the public blockchain and mine privately on its own block without releasing them. Once the selfish miner is lucky to get a private chain longer than the public one, depending on the number of blocks the selfish miner is ahead of the public chain, the selfish miner would decide to release part or all of its private blocks. Thus, the released selfish miner's private chain would override the public chain as it is longer. By doing so, if the selfish miner's blocks are finally included in the blockchain, the selfish miner can get all the block rewards of her/his private chain. By selfish mining, the selfish miner would gain extra rewards compared with honest mining because the selfish behaviors also waste honest miners' mining power on the blocks which do not eventually bring any block rewards.

B. Selfish Mining in Chainweb

In order to resist malicious forks, chains in Chainweb are tightly tied by a cross-referencing feature. As the Chainweb white paper [14] indicates, the security level increases as the number of chains grows. This feature binds the chains

together like a rope and prevents certain chains from growing much faster than other chains, which leaves a difficult task to attackers who try to secretly mine a longer single chain or a longer whole braid of chains to override the public chains, similar to what the selfish miner does in the single-chain blockchain system. The Chainweb white paper [14] gives the probability of an attacker mining privately being able to override the full public chain braid, which shows that the attack is nearly impossible.

Though it is barely feasible to attack the entire chain braid, concentrating mining power on a part of chains, withholding and releasing mined blocked later may give an attacker some extra mining reward. This paper aims to study this kind of mining attack on Chainweb blockchain. Although the real Chainweb has upgraded from a 10-chain configuration to 20 chains, for ease of demonstration, we focus on the 10-chain Chainweb as the logic of both settings is similar. As the Fig. 1 shows, any chain can reach all the other chains in no more than 2 "hops". This indicates that as long as one or more blocks with height $N-1$ are unavailable, miners can not start to mine any blocks with height $N+1$. Fig. 2b gives an example that the absence of block $\langle 3, N-1\rangle$ restricts the subsequent development of block $\langle 1, N\rangle, \langle 3, N\rangle, \langle 5, N\rangle$, $\langle 8, N\rangle$, and all the blocks with height $N+1$. In our attack model, there are miners with two roles: attackers and honest miners. No matter how many miners participate in the Chainweb network, only the proportion of the computational power counts. All the miners mine honestly and individually by default, whereas attackers need to work as a party with central coordination. The attackers and honest miners can be treated as two mining pools [16], taking a percentage of α and $1-\alpha$ of total mining power, respectively.

The attackers act like opportunists. Taking the Fig. 2b as an example, during the block mining process, all the blocks except the one on $Chain$ 3 with height $N-1$ are available. At this specific time, all miners, including attackers and honest miners, are still mining honestly. Based on the cross-referencing relationship, the block $\langle 1, N\rangle, \langle 3, N\rangle, \langle 5, N\rangle$, and $\langle 8, N\rangle$ do not exist since block $\langle 3, N-1\rangle$, the block they reference to, is not available. Moreover, all the blocks with height $N+1$ do not exist for the same reason. We do not know whether the other blocks with height N exist since they are minable.

Once the block $\langle 3, N-1\rangle$ is released by a certain miner and noticed by the attackers, the block $\langle 1, N\rangle, \langle 3, N\rangle, \langle 5, N\rangle$, and $\langle 8, N\rangle$ becomes minable. Instead of honestly mining on all minable blocks among all chains, the attackers will start the attack by first putting all the mining power evenly on the block $\langle 1, N\rangle, \langle 3, N\rangle, \langle 5, N\rangle$, and $\langle 8, N\rangle$. The corresponding chains of these four blocks form a *target chain set*. In this case, $Chain$ 3 is the center chain of the target chain set since it has referencing relationships with all the other three chains. Once the attackers have generated the four blocks on the target chains with height N, the attackers will concentrate all mining power on the block $\langle 3, N+1\rangle$. These five blocks form an *umbrella*. The essential principle is that the attackers would never release any secretly mined block until the collection of the whole umbrella blocks is complete. Similar to the single-chain blockchain's block confirmation scheme, a certain block is confirmed to be ultimately on the chain with a high probability only when several blocks are buried on it. Blocks in the Chainweb blockchain also follow this rule. However, attackers mining secretly on the umbrella blocks and broadcasting them all together would significantly improve the odds that other miners switch from the blocks with smaller block height to the umbrella, accept and finally include the umbrella in the Chainweb ledger. Plus, the corresponding blocks mined by the honest miners will be totally orphaned, and the devoted mining power is wasted.

Consequently, when the umbrella revealed by the attackers reaches the whole Chainweb network ahead of other honest miners', we assume that the attack succeeds, and the attackers can gain all the five-block rewards. Only when all the blocks belonging to the umbrella are released by the honest miners earlier than the attackers can the attack be judged as a failure. The attackers would switch to honest mining after either success or failure of the attack and wait for the next chance to launch the attack. Please be noted that the example includes but is not limited to the scenario with the umbrella centering on $Chain$ 3. The center of the target umbrella constantly changes according to the last block completed at a particular block height.

IV. Markov Chain Modeling

This section first develops a Markov chain based modeling of the selfish mining attack on Chainweb blockchain. To simplify the analysis, we constrain the honest miners' behaviors to the fact that the honest miner will not allocate their mining resource on any minable blocks with block height $N+1$ except for block $\langle 3, N+1\rangle$. This constraint forces the honest miners to put more mining power to the target umbrella, making it harder for the attackers to compete in the umbrella race. As a result, the selfish mining attack's numerical result can be considered a lower bound of the attackers' revenue. After that, by calculating the stationary probability distribution of each state, we obtain the overall probability of the attack being successful and the absolute rewards of the selfish mining attack. Furthermore, we obtain the weights of each starting state by normalizing the probability distribution of the number of blocks on the chains other than the target chain set. Then convert the absolute block rewards to long-term block rewards of the attack by analyzing the proportion of blocks mined by the attackers to the total number of blocks mined during the attack. Finally, we define *extra rewards* and *attack efficiency* as additional performance metrics to evaluate for a better attacking timing according to the mining power.

A. Markov Chain Model

We leverage a discrete Markov chain to model the selfish mining attack process. The key factors of the process are the

dynamics of the number of blocks generated on the target block set and the blocks on the other chains. Also, during the Chainweb mining, the mining power distributed among the chains would change rapidly according to the dynamics of the block distribution. Furthermore, during the attack, the block distribution on the target block set and the other chains, and whether the blocks are mined by honest miners, who publish blocks immediately, or attackers, who keep blocks secretly, have different impacts on the mining power distribution, which further influences the mining process. Therefore, developing a Markov chain model is very challenging to completely represent the attack procedure without too complex model.

Our model can analyze Chainweb with an arbitrary number of parallel chains as long as the size of the target chain set is 4. As for the states of the model, as shown in Fig. 3, besides of two states "Honest miners win" and "Attackers win" which are denoted as "HW" and "AW", all the other states are denoted by a 3-tuple (i, j, k) to represent the distribution of the existed blocks with height N. The first two elements i and j denote the attackers' secretly mined blocks and the blocks mined and released by the honest miners among the four target chains, respectively. The third element k represents the number of generated blocks on the chains other than the target chains, all by the honest miners. i and j are integers having the same range, from 0 to 4, while k ranges from 0 to $T-4$, where T denotes the total number of parallel chains designed in the Chainweb blockchain system, taking the value either 10 or 20 in our analytical model. Recall that the attackers initialize the attack on a target chain set right after the last block with a certain block height, say $N-1$, being mined out. At this initial stage, none of the target blocks exists. However, since all blocks with height $N-1$ were already there except the last one, at this moment, the $T-4$ blocks with height N are minable. Consequently, at the beginning of the attack, the number of blocks with height N excluding the target blocks could vary from 0 to $T-4$. Thus, the possible initial state of the attack in the Markov chain model is denoted as $(0, 0, K)$, where $K \in \{0, \ldots, T-4\}$ is the value of k. For ease of demonstration, we call that states with the same value of k are in the same layer.

The discrete random process can be considered as the sequence $\{X_n\}$ with values from a finite set $A = \{HW, AW, (i, j, k) | i, j \in \{0, 1, \ldots, 4\}, k \in \{0, \ldots, T-4\}\}$. The process is said to be in state u at time n if $X_n = u$ with $u \in A$. The state transition happens when any new block is mined out during the attack. From the Markov chain property, the state $X_n + 1$ depends only upon the previous state X_n, where the transition probability is

$$P_{uv} = P\{X_{n+1} = v | X_n = u\}, \quad u, v \in A. \tag{1}$$

The Markov chain model shown in Fig. 3 depicts the attacking process. Fig. 4 shows the probability transitions between states in layer k. Let us take an example attack with

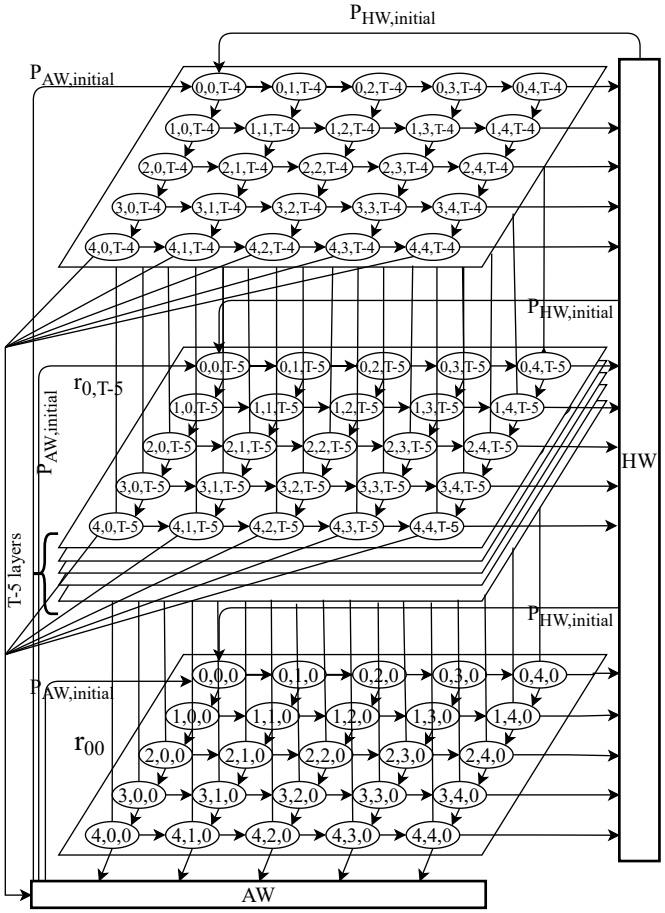


Fig. 3: Markov chain model of the selfish mining attack.

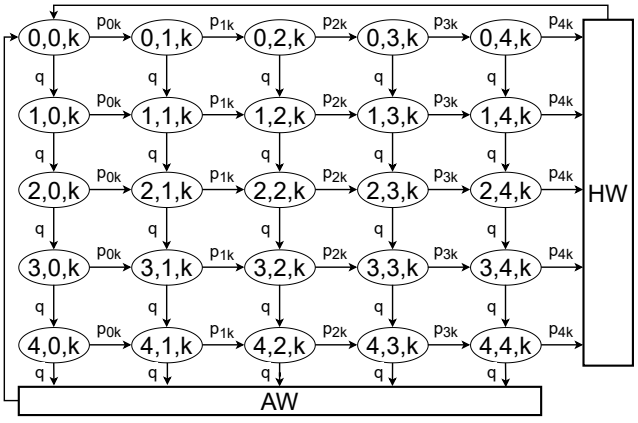


Fig. 4: The probability transitions between states in layer k.

an attacker Alice equipped with mining power α and honest miners with power $1-\alpha$ in total. Before the attack is launched, Alice mines honestly with power α. When Alice notices that the last block is mined out with height $N-1$, she will switch her mining power to the corresponding target chains to mine the four target blocks. At this time, suppose that there already exist l blocks with height N, the first state should be $(0, 0, l)$. The state will transit no matter which new block being mined

out. The state transits from (i,j,k) to $(i+1,j,k)$ if Alice solves a new block out of the target blocks, with transition probability q equalling to Alice's mining power α, i.e.,

$$\begin{aligned} &q = P_{(i,j,k),(i+1,j,k)} = \alpha, \\ &i \in \{0,1,\ldots,3\}, j \in \{0,1,\ldots,4\}, k \in \{0,\ldots,T-4\}. \end{aligned} \tag{2}$$

However, honest miners mining out a new block out of Alice's target blocks will lead to a transition from (i,j,k) to $(i,j+1,k)$, with probability p_{jk}, i.e.,

$$\begin{aligned} &p_{jk} = P_{(i,j,k),(i,j+1,k)} = \frac{(1-\alpha)(4-j)}{T-k-j}, \\ &i \in \{0,1,\ldots,4\}, j \in \{0,1,\ldots,3\}, k \in \{0,\ldots,T-4\}, \end{aligned} \tag{3}$$

and $p_{4k} = p_{3k}$, because the completion of blocks with height N in the umbrella will invoke the top block of the umbrella with height $N+1$ minable. Similarly, a new block mined out of the blocks other than Alice's target blocks by any honest miner can trigger the state to transit from (i,j,k) to $(i,j,k+1)$. This cross-layer transition probability can be written as

$$\begin{aligned} &r_{jk} = P_{(i,j,k),(i,j,k+1)} = \frac{(1-\alpha)(T-4-k)}{T-k-j}, \\ &i \in \{0,1,\ldots,4\}, j \in \{0,1\ldots,3\}, k \in \{0,\ldots,T-5\}, \end{aligned} \tag{4}$$

and $r_{4k} = r_{3k}$. There are also the transitions from either the "Honest miners win" or "Attackers win" state back to one of the initial states $(0,0,K)$ with probability

$$P_{HW,initial} = P_{AW,initial} = \frac{1}{T-4+1}, \tag{5}$$

which indicate that the current umbrella is completed, and Alice starts the attack on a new umbrella. The probabilities of the transitions other than those mentioned above are set to 0. Thus, including the "HW" and "AW" states, the total number of states would be $25\times(T-3)+2$. The Markov chain can be described by a $[25\times(T-3)+2]\times[25\times(T-3)+2]$ transition probability matrix $\mathbf{P}$ as

$$\begin{pmatrix} P_{000,000} & P_{000,010} & \cdots & P_{000,(4,4,T-4)} & P_{000,HW} & P_{000,AW} \\ P_{010,000} & P_{010,010} & \cdots & P_{010,(4,4,T-4)} & P_{010,HW} & P_{010,AW} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ P_{HW,000} & P_{HW,010} & \cdots & P_{HW,(4,4,T-4)} & P_{HW,HW} & P_{HW,AW} \\ P_{AW,000} & P_{AW,010} & \cdots & P_{AW,(4,4,T-4)} & P_{AW,HW} & P_{AW,AW} \end{pmatrix}. \tag{6}$$

B. Stationary Distribution and Probability of the Attack Being Successful

Let π_u denote the stationary probabilities of the Markov chain, where $u \in A$, which can be solved by the following equations:

$$\begin{aligned} \pi_{(i,j,k)|i\neq 0,j\neq 0} = {} & \pi_{(i-1,j,k)}q + \pi_{(i,j-1,k)}p_{j-1,k} \\ & + \pi_{(i,j,k-1)}r_{j,k-1}, \\ & i,j \in \{0,1,\ldots,4\}, k \in \{0,\ldots,T-4\}, \end{aligned} \tag{7}$$

$$\begin{aligned} \pi_{(0,0,k)} = \frac{1}{T-4+1}(\pi_{HW}+\pi_{AW}) + \pi_{(0,0,k-1)}r_{0,k-1}, \\ k \in \{0,\ldots,T-4\} \end{aligned} \tag{8}$$

$$\pi_{HW} = \sum_{i=0}^{4}\sum_{k=0}^{T-4}\pi_{(i,4,k)}p_{4k}, \tag{9}$$

$$\pi_{AW} = \sum_{j=0}^{4}\sum_{k=0}^{T-4}\pi_{(4,j,k)}q, \tag{10}$$

$$\sum_{u\in A}\pi_u = 1. \tag{11}$$

For those invalid states, i.e., $(-1,*,*),(*,-1,*)$ and $(*,*,-1)$, the values of π are set to 0.

The large state space makes it hard to get the stationary probabilities with a variable α in the symbolic form. Therefore, we just represent the stationary probability as π_u and use the specific values of α to get the results.

The stationary probability is the proportion of the total time the process will be in a certain state. "HW" and "AW" are the only two states representing the termination of each attack round. Thus, we can obtain the success probability of the attackers as

$$P_{suc} = \frac{\pi_{AW}}{\pi_{HW}+\pi_{AW}}. \tag{12}$$

C. Weighted Initial States and Attackers' Rewards

Since each success of the attack would bring the attackers rewards of all the five blocks in the umbrella, a naive and straightforward way to evaluate the attackers' block rewards would be

$$B_1 = 5 \times P_{suc}. \tag{13}$$

However, in the long-term run, to alleviate the effect of difficulty adjustment, the ratio of the blocks obtained by the attackers through the attack to the total blocks mined out during each attack round is more suitable to justify the effectiveness of the selfish attack, where an attack round is defined as a process from the beginning of an attack to the completion of an umbrella. Therefore, in this subsection, we define this ratio as the *relative rewards* and quantitatively analyze the attackers' relative rewards.

The total blocks mined out throughout each attack round consist of four parts, which are shown as follows.

The expectation of blocks obtained by

1) the attackers on the umbrella when "Attackers win", denoted as B_1;
2) the honest miners on the umbrella when "Honest miners win", denoted as B_2;
3) the honest miners on the other $T-4$ chains when "Attackers win", denoted as B_3;

4) the honest miner on the other $T-4$ chains when "Honest miner wins", denoted as B_4.

The attackers' *relative rewards* can be defined as

$$R_{attack} = \frac{B_1}{B_1 + B_2 + B_3 + B_4}. \tag{14}$$

B_1 has been discussed in (13). In the denominator, B_1 and B_2 can be treated as a whole with $B_1 + B_2 = 5$ since all the blocks in an umbrella would be obtained at the end of each round either by attackers or honest miners. As for the analysis of B_3 and B_4, we should focus on the distribution of the number of already existing blocks on the $T-4$ chains at the beginning of each attack round. Suppose that there are $l \in \{0, \ldots, T-4\}$ blocks on the $T-4$ non-target chains with height N at the start of an attack round. Only $T-4-l$ blocks remain to be mined by the honest miners with respect to B_3 and B_4. The corresponding initial state of the attack in this round would be $(0, 0, l)$. Specifically for this round, states in Markov chain model with $(*, *, k')$ where $k' < l$ are all invalid. All the valid states $(4, *, *)$ and $(*, 4, *)$ are of our interest as they are one transition away to the "AW" or "HW" state.

As we need to count the different situations based on the $T-4$ potential initial states of an attack round, through the stationary probability of the Markov chain model discussed above according to (7)-(11), we can get the conditional initial state probabilities as follows.

$$\pi_{init}^{l} = \frac{\pi_{(0,0,l)}}{\sum_{k=0}^{T-4} \pi_{(0,0,k)}}, \quad l \in \{0, 1, \ldots, T-4\} \tag{15}$$

Let π'_u, $u \in A$ denote the probability that the state transiting from an initial state $(0, 0, l)$ to u , represented by Fig. 3 and derived by the transition probability matrix from each initial state as follows.

$$\begin{aligned} \pi'_{(i,j,k)|i\neq 0, j\neq 0} &= \pi_{(i-1,j,k)} q + \pi_{(i,j-1,k)} p_{j-1,k} \\ &\quad + \pi_{(i,j,k-1)} r_{j,k-1}, \\ &\quad i, j \in \{0, 1, \ldots, 4\}, k \in \{0, \ldots, T-4\}, \end{aligned} \tag{16}$$

$$\pi'_{(0,0,k=l)} = 1, \quad l \in \{0, \ldots, T-4\} \tag{17}$$

The transition probabilities of the Markov chain model still hold. Given an attack round with initial state $(0, 0, l)$, the expected number of blocks B_3^l mined out by the honest miners if the attack round termination triggered by the attackers completing the umbrella ahead of the honest miners would be

$$B_3^l = \sum_{k=l}^{T-4} \sum_{j=0}^{4} q(k-l) \pi'_{(4,j,k)}. \tag{18}$$

Summing up the expected blocks weighted by π_{init}^{l} would give the result of B_3

$$B_3 = \sum_{l=0}^{T-4} \pi_{init}^{l} B_3^l. \tag{19}$$

Similarly, given a attack round with initial state $(0, 0, l)$, the expected number of blocks B_4^l mined out by the honest miners if the attack round termination triggered by the honest miners completing the umbrella ahead of the attackers would be

$$B_4^l = \sum_{k=l}^{T-4} \sum_{i=0}^{4} p_{4k}(k-l) \pi'_{(i,4,k)}. \tag{20}$$

B_4 can be expressed as

$$B_4 = \sum_{l=0}^{T-4} \pi_{init}^{l} B_4^l. \tag{21}$$

Thus we can combine (13)-(21) to get the attackers' relative rewards.

D. Performance Metrics

In a healthy block mining environment, all miners mine honestly. Suppose Bob's mining power takes up β percentage of total mining power. Bob mines honestly in a healthy environment with all his mining power in a relatively long period τ. The number of blocks generated during τ is B_τ. On average, Bob would earn $\beta \times B_\tau$ block rewards. In other words, an honest miner's block rewards are always proportional to his percentage of mining power.

In this way, Alice, owing α percentage of mining power, would gain $\alpha\%$ of the total number of blocks generated in the whole blockchain network. If Alice conducts our proposed selfish mining attack, she will receive $R_{attack}\%$ of the total number of blocks generated during the attack. Noted that the R_{attack} contains a variable α. Thus, assuming attackers with power of α, we define *extra rewards* as the difference between the attackers' relative rewards earned by attacking and expected rewards earned by normal mining, i.e.,

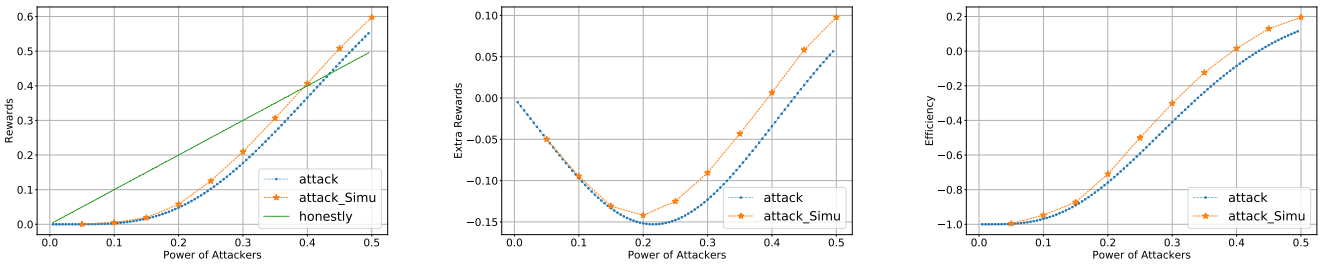
$$E_{extra} = R_{attack} - \alpha. \tag{22}$$

Obviously, potential miners would only launch the selfish mining attack when they possess mining power that makes $E_{extra} > 0$.

It is also worth discovering the relationship between the mining power devoted and the extra rewards gained. We define *attack efficiency* η to describe it.

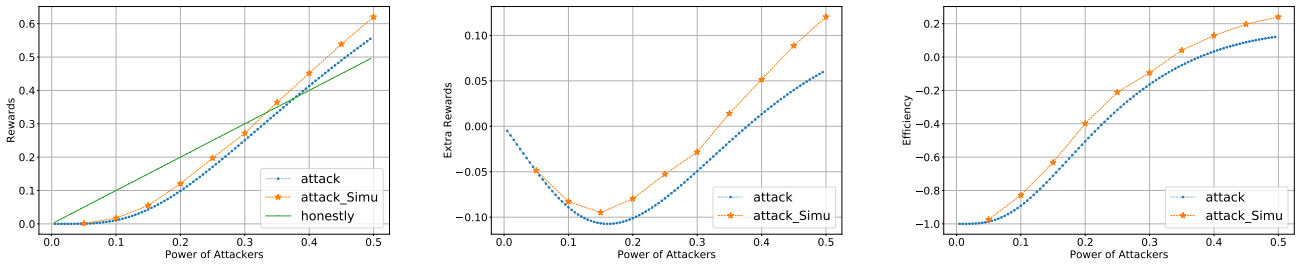
$$\eta = \frac{E_{extra}}{\alpha} \tag{23}$$

The attack efficiency indicates the capability of the selfish mining attack. Therefore, rational attackers should plan their mining power to balance achieving high extra rewards and high attack efficiency.



(a) Relative Rewards (b) Extra Rewards (c) Attack Efficiency

Fig. 5: Attacking performance in 10-chain setting



(a) Relative Rewards (b) Extra Rewards (c) Attack Efficiency

Fig. 6: Attacking performance in 20-chain setting

V. Performance Evaluation

In this section, we conduct simulations to evaluate the performance of the proposed attack on the original 10-chain setting and today's 20-chain setting, respectively. We employ the ns-3 simulation framework to build a discrete event model that emulates the dynamics of Chainweb systems over peer-to-peer networks. The results are averaged over 50 simulations.

A. Relative Rewards, Extra Rewards, Attack Efficiency, and Profit Threshold

In this section, we obtain the performance of the attacks by showing the results of key performance metrics. Like the traditional PoW blockchain, Chainweb also obeys that the mining power of any mining entity should not exceed 50%, to prevent the classic 51% attack. Considering attackers equipped with mining power $\alpha \in [0.005, 0.495]$ on a 10-chain Chainweb, the relative rewards, extra rewards, and attack efficiency are illustrated in Fig. 5.

Fig. 5a shows the attackers' rewards and the rewards they would gain without launching the selfish attack. The attackers' attacking rewards grow monotonically as the mining power increases. The maximum rewards reach 55.15%. However, the attackers actually suffer loss when her/his mining power is not high enough. Fig. 5b shows that, as the mining power grows, the extra rewards of the attackers are negative and decrease until reach the minimum. After that, the extra rewards increase and turn positive, which means the miners can benefit from the attack. The maximum extra rewards reach 5.65%. The mining power which leads to the critical point between negative and positive extra rewards is defined as the *profit threshold*. It shows that the attackers can make extra rewards when they own at least 43.5% of total mining power in the network. The profit threshold is also a measure to evaluate the security level of the blockchain system. Fig. 5c gives a relationship between power and attack efficiency. The attack efficiency also increases monotonically as the power increases. The highest efficiency would be 11.42%. We can also extract the attackers' performance in a 20-chain scenario from Fig. 6. The maximum relative rewards, extra rewards, and attack efficiency are 55.45%, 5.95%, and 12.03%, respectively. The profit threshold for 20-chain setting is 38%.

Simulation results reflect that the attackers' profit threshold is lower than the analytical results obtained through Markov chain modeling. As our model elaborates, we restrict the honest miners from allocating their mining power on any minable blocks with block height $N + 1$. However, miners would mine on all the minable blocks in the real system, making the umbrella less competitive, leading to attackers getting higher rewards than the analytical relative rewards. The simulation results verify that our model indeed provides a lower bound of the selfish mining attack's rewards.

B. Performance Versus Number of Chains

As discussed above, attackers' maximum relative rewards, extra rewards, and attack efficiency in a 20-chain Chainweb

are better than those in a 10-chain scenario. Besides that, the difference between the profit threshold delivers an exciting finding: attacking in the 20-chain setting is easier than in the 10-chain setting. This contradicts the claim in Chainweb white paper, saying the security level increases as the number of chains in the system expands. To analyze the reason, we need to recall that the core of the attack is to compete with the so-called umbrella. Suppose that attackers devote all their power to the target umbrella. However, the attackers' competitor is actually part of the honest miner's power distributed on the attackers' target umbrella. Since the honest miners always allocate their mining power evenly among all minable blocks, the more chains in the Chainweb, the less portion of power on the umbrella. In other words, the power of attackers' competitors is diluted. Therefore, the claim in the white paper may be valid regarding the attack of replacing the whole braid of chains, but not suitable for the selfish attack in Chainweb.

C. Countermeasures Against the Attack

The mining attacks can cause harm and devastate a blockchain system. Therefore, it is critical and urgent to study the countermeasures against the potential attack.

1) Pool size limitation: Although the blockchain is a distributed system, the mining pool concept somewhat brings centralization. Our results indicate the attackers can earn extra benefits with at least 38% power. However, owning this amount of power is not feasible for any individual. The attack can be prevented by setting a maximum pool size threshold to regulate large mining pools.

2) Miners' reactions to forks: The attack leverages the longest chain rule to achieve the goal. Therefore, if miners are more conservative and cautious about switching to the longer chains, the effect of the attack can be alleviated. For example, the system could set a block height interval and suggest the miners switch to a longer chain only when the block difference exceeds the block height interval.

VI. RELATED WORK

A. Throughput Improvement

There are many designs targeting throughput improvement. Directed acyclic graph (DAG) proposed in [4] adds more flexibility to generate new blocks. [24] systematically conclude the DAG based blockchains, where Prism [5], [6] defines three types of blocks that collaborate to achieve the best throughput and a confirmation latency, and OHIE [7] achieves high throughput with long confirmation delay. [25] proposes a weak consensus algorithm that maintains only relative positions of messages to construct a blockchain system Sphinx with high transaction throughput. Chainweb [14] adopts a parallel chain structure, each of which has cross-referencing relationships with its peer chains. The parallelism design increases the system throughput linearly as the number of chains grows. In addition, the cross-referencing feature offers an excellent fork resistance level. These extraordinary designs provide a massive opportunity for Chainweb to become a popular blockchain. The public version of Chainweb has been released by Kadena and is now fully accessible to the public for block mining, token trading, etc [26].

In addition to the consensus mechanisms, reducing the number of transactions processed by miners can also improve the throughput. Either sharding or off-chain payment channels can achieve this. Sharding technology partitions the nodes into small portions call shards. All shards can work on fewer transactions in parallel. OmniLedger [27] is a typical scalable and secure sharding blockchain system. Bitcoin-NG proposed in [28] defines key blocks and micro blocks to collect transactions. Once the key block is generated, its corresponding miner becomes the leader in generating micro blocks. Off-chain payment channel method [11] such as Lightning Network [12] is designed for Bitcoin blockchain to enable fast transactions.

B. Security in Blockchain

Mining attack is a security issue that severely destroys the block mining incentive mechanism and vastly impacts the health of the cryptocurrency ecosystem. Usually, attackers conduct mining attacks by deviating their mining behaviors from the original protocols to earn more block rewards than mining normally. Selfish mining proposed by Eyal et al. [15] allows attackers to harvest more block rewards through mining secretly and disclosing blocks later. The block withholding attack (BWH) has been proposed and studied in [16]. The philosophy of BWH is to split and reallocate the attacker's mining power into different mining pools to share the rewards from the infiltrated pool without contributing any valid blocks. Fork after withholding attack (FAW) has been proposed by Y. Kwon et al. in [19] as an advanced version of BWH attack to hold the rewards gained via BWH as a lower bound. The FAW attack enables the infiltrating part of the attackers to deliberately generate a fork in the filtrated pool as far as possible to improve the probability of winning more block rewards. Other mining attacks, such as stubborn mining, also undermine the security of the traditional blockchain [20]. Eclipse attack [22] is a DoS based attack that isolates the victim nodes by jamming all connections to earn extra mining rewards. Some systematical studies in [29]–[31] evaluate the impacts on various blockchains of various factors such as network delay, mining power distribution, transaction throughput, and block size.

VII. CONCLUSION

Our work presents the first systematical model to analyze the selfish mining on the Chainweb blockchain. We describe the attack procedure in detail and develop a complete and accurate Markov chain based model to analyze the selfish mining attack. The result reflects that the attack is effective.

We also conduct simulations that verify our results. The profit threshold shows an insightful and counterintuitive finding that the attackers are easier to succeed in 20-chain than in 10-chain setting.

VIII. ACKNOWLEDGMENT

This work was supported in part by the NSF under grants CNS-1816908 and CNS-2008092.

REFERENCES

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

[2] J. Xie, H. Tang, T. Huang, F. R. Yu, R. Xie, J. Liu, and Y. Liu, "A survey of blockchain technology applied to smart cities: Research issues and challenges," *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2794–2830, 2019.

[3] Q. Wang, J. Yu, S. Chen, and Y. Xiang, "Sok: Diving into dag-based blockchain systems," *arXiv preprint arXiv:2012.06128*, 2020.

[4] Y. Li, B. Cao, M. Peng, L. Zhang, L. Zhang, D. Feng, and J. Yu, "Direct acyclic graph-based ledger for internet of things: Performance and security analysis," *IEEE/ACM Transactions on Networking*, vol. 28, no. 4, pp. 1643–1656, 2020.

[5] V. Bagaria, S. Kannan, D. Tse, G. Fanti, and P. Viswanath, "Prism: Deconstructing the blockchain to approach physical limits," in *Proc. of ACM SIGSAC Conference on Computer and Communications Security*, 2019, p. 585–602.

[6] L. Yang, V. Bagaria, G. Wang, M. Alizadeh, D. Tse, G. Fanti, and P. Viswanath, "Prism: Scaling bitcoin by 10,000x," *arXiv preprint arXiv:1909.11261*, 2019.

[7] H. Yu, I. Nikolic, R. Hou, and P. Saxena, "Ohie: Blockchain scaling made simple," in *Proc. of IEEE Symposium on Security and Privacy*, 2020, pp. 112–127.

[8] I. Eyal, A. E. Gencer, E. G. Sirer, and R. V. Renesse, "Bitcoin-ng: A scalable blockchain protocol," in *Proc. of 13th USENIX Symposium on Networked Systems Design and Implementation*, Mar. 2016, pp. 45–59.

[9] H. Dang, T. T. A. Dinh, D. Loghin, E.-C. Chang, Q. Lin, and B. C. Ooi, "Towards scaling blockchain systems via sharding," in *Proc. of International Conference on Management of Data*, 2019, p. 123–140.

[10] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich, "Algorand: Scaling byzantine agreements for cryptocurrencies," in *Proc. of the 26th symposium on operating systems principles*, 2017, pp. 51–68.

[11] P. Li, T. Miyazaki, and W. Zhou, "Secure balance planning of off-blockchain payment channel networks," in *Proc. of IEEE INFOCOM*, 2020, pp. 1728–1737.

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

[13] S. Popov, "The tangle," *White paper*, vol. 1, no. 3, 2018.

[14] W. Martino, M. Quaintance, and S. Popejoy, "Chainweb: A proof-of-work parallel-chain architecture for massive throughput," 2018. [Online]. Available: http://kadena.io/docs/chainweb-v15.pdf

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

[16] M. Rosenfeld, "Analysis of bitcoin pooled mining reward systems," *arXiv preprint arXiv:1112.4980*, 2011.

[17] I. Eyal, "The miner's dilemma," in *Proc. of IEEE Symposium on Security and Privacy*, 2015, pp. 89–103.

[18] L. Luu, R. Saha, I. Parameshwaran, P. Saxena, and A. Hobor, "On power splitting games in distributed computation: The case of bitcoin pooled mining," in *Proc. of IEEE Computer Security Foundations Symposium*, 2015, pp. 397–411.

[19] Y. Kwon, D. Kim, Y. Son, E. Vasserman, and Y. Kim, "Be selfish and avoid dilemmas: Fork after withholding (faw) attacks on bitcoin," in *Proc. of ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 195–209.

[20] K. Nayak, S. Kumar, A. Miller, and E. Shi, "Stubborn mining: Generalizing selfish mining and combining with an eclipse attack," in *Proc. of IEEE European Symposium on Security and Privacy*, 2016, pp. 305–320.

[21] M. Apostolaki, A. Zohar, and L. Vanbever, "Hijacking bitcoin: Routing attacks on cryptocurrencies," in *Proc. of 2017 IEEE Symposium on Security and Privacy*, pp. 375–392.

[22] E. Heilman, A. Kendler, A. Zohar, and S. Goldberg, "Eclipse attacks on bitcoin's peer-to-peer network," in *Proc. of 24th USENIX Security Symposium*, 2015, pp. 129–144.

[23] L. Kiffer, R. Rajaraman, and A. Shelat, "A better method to analyze blockchain consistency," in *Proc. of ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 729–744.

[24] Q. Wang, J. Yu, S. Chen, and Y. Xiang, "Sok: Diving into dag-based blockchain systems," *arXiv preprint arXiv:2012.06128*, 2020.

[25] Q. Wang and R. Li, "A weak consensus algorithm and its application to high-performance blockchain," *arXiv preprint arXiv:2102.00872*, 2021.

[26] "Kadena." [Online]. Available: https://www.kadena.io/kadena

[27] E. Kokoris-Kogias, P. Jovanovic, L. Gasser, N. Gailly, E. Syta, and B. Ford, "Omniledger: A secure, scale-out, decentralized ledger via sharding," in *Proc. of Symposium on Security and Privacy*. IEEE, 2018, pp. 583–598.

[28] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse, "Bitcoin-ng: A scalable blockchain protocol," in *Proc. of 13th USENIX Symposium on Networked Systems Design and Implementation*, 2016, pp. 45–59.

[29] N. Papadis, S. Borst, A. Walid, M. Grissa, and L. Tassiulas, "Stochastic models and wide-area network measurements for blockchain design and analysis," in *Proc. of IEEE INFOCOM*, 2018, pp. 2546–2554.

[30] B. Cao, M. Li, L. Zhang, Y. Li, and M. Peng, "How does csma/ca affect the performance and security in wireless blockchain networks," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4270–4280, 2019.

[31] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou, "Modeling the impact of network connectivity on consensus security of proof-of-work blockchain," in *Proc. of IEEE INFOCOM*, 2020, pp. 1648–1657.