Published: 25 August 2025

Citation in BibTeX format

ASIA CCS '25: 20th ACM Asia Conference on Computer and Communications Security
August 25 - 29, 2025
Hanoi, Vietnam

Conference Sponsors:
SIGSAC

RESEARCH-ARTICLE

Mining Attack with Zero Knowledge in the Blockchain

YU JIAPING, The Hong Kong Polytechnic University, Hong Kong, Hong Kong, Hong Kong

GAO SHANG, The Hong Kong Polytechnic University, Hong Kong, Hong Kong, Hong Kong

SONG RUI, The Hong Kong Polytechnic University, Hong Kong, Hong Kong, Hong Kong

ZHIPING CAI, National University of Defense Technology China, Changsha, Hunan, China

XIAO BIN, The Hong Kong Polytechnic University, Hong Kong, Hong Kong, Hong Kong

Open Access Support provided by:

The Hong Kong Polytechnic University

National University of Defense Technology China

ASIA CCS '25: Proceedings of the 20th ACM Asia Conference on Computer and Communications Security (August 2025)
https://doi.org/10.1145/3708821.3733906
ISBN: 9798400714108

Mining Attack with Zero Knowledge in the Blockchain

Yu Jiaping
Ocean University of China
School of Computer Science and Technology
Qingdao, Shandong, China
The Hong Kong Polytechnic University
Department of Computing
Hong Kong, Hong Kong
yujiaping@ouc.edu.cn

Gao Shang
The Hong Kong Polytechnic University
Department of Computing
Hong Kong, Hong Kong
shanggao@comp.polyu.edu.hk

Song Rui
The Hong Kong Polytechnic University
Department of Computing
Hong Kong, Hong Kong
csrsong@comp.polyu.edu.hk

Zhiping Cai
National University of Defense Technology
College of Computer
Changsha, China
zpcai@nudt.edu.cn

Xiao Bin
The Hong Kong Polytechnic University
Department of Computing
Hong Kong, Hong Kong
b.xiao@polyu.edu.hk

Abstract

Mining attacks remain a serious threat to Proof-of-Work (PoW) blockchain systems, as malicious miners can deviate from standard mining rules to gain extra rewards. While classic selfish mining tactics conceal entire blocks or release them at once to cause forks, we extend the strategy space by introducing *partial block sharing* and leveraging *zero-knowledge proofs*. Specifically, we propose a novel **Mining Attack with Zero Knowledge**, encompassing two main strategies: **Partial Selfish Mining (PSM)** and its advanced variant **Advanced PSM (A-PSM)**.

By selectively releasing only partial block data, an attacker can attract *rational miners* to join its private branch without fully revealing the mined block. A zero-knowledge-proof-based mechanism ensures that these rational miners can be convinced of the block's validity without learning its complete content, thereby incentivizing them to collude for higher individual gains. Our theoretical and experimental results show that, under certain conditions on mining power distribution and network latency, PSM can yield higher rewards than both honest and selfish mining, and A-PSM attackers can achieve profits that match or exceed selfish mining and even honest mining. This work highlights a novel *zero-knowledge-enabled* collusion threat in blockchain mining and calls for broader security measures to protect against such sophisticated strategies.

CCS Concepts

• **Security and privacy → Logic and verification**; **Economics of security and privacy**.

ACM ISBN 979-8-4007-1410-8/25/08
https://doi.org/10.1145/3708821.3733906

Keywords

Blockchain, Selfish Mining, Mining Attack

ACM Reference Format:
Yu Jiaping, Gao Shang, Song Rui, Zhiping Cai, and Xiao Bin. 2025. Mining Attack with Zero Knowledge in the Blockchain. In *ACM Asia Conference on Computer and Communications Security (ASIA CCS '25), August 25–29, 2025, Hanoi, Vietnam.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3708821.3733906

1 Introduction

Decentralized digital currencies like Bitcoin have captured public interest for years [11, 36]. By the end of 2024, Bitcoin has a market value of more than 1,840 billion USD. In Bitcoin, the first miner who solves the puzzle and broadcasts the result will be rewarded with 6.25 bitcoins (BTC), which are worth 591,038 USD by January 2025 [7]. The more computing resources a miner applies, the more likely it can solve the puzzle and claim the reward first [21, 34]. The result of this puzzle is known as "Proof-of-Work" (PoW) in Bitcoin. Such openness and decentralization promote wide adoption but also make Bitcoin a target of malicious behavior [1, 13, 14, 16, 38].

Recent studies show that *mining attacks* can undermine PoW-based systems by deviating from standard mining protocols. Attackers can hide or discard mined blocks, or strategically release them to cause forks [31–33, 35]. As one of the most fundamental and well-known mining attacks, *selfish mining* [8] withholds newly discovered blocks in a private branch. When honest miners publish their blocks, the attacker releases its hidden blocks to cause a fork. If the attacker's branch is selected as the main chain, honest miners effectively waste their computational power.

Although selfish mining can yield higher profits for the attacker if its computational power is sufficiently large (e.g., exceeding 33%), such conditions are hard to meet in large-scale public blockchains [5, 30]. Extensive work [14, 15] has thus explored alternative tactics (like block withholding, forking, or combined strategies) to gain

(a) Selfish Mining (b) Partial Selfish Mining

Figure 1: (a) Selfish mining withholds discovered blocks in its private chain; (b) Partial selfish mining can first withhold a discovered block, then share partial block data, and finally broadcast the whole block.

an edge even with less mining power. Meanwhile, other incentive-based attacks (e.g., DoS attacks [20]) leverage the notion that most miners are *rational*, seeking to maximize their rewards by choosing the most profitable chain or strategy to mine on.

Selfish mining is not frequently observed in major cryptocurrencies, partly because it is difficult to detect and partly because attackers cannot easily surpass the power threshold on their own. Colluding with other *rational* miners is a potential approach: if enough miners join the attacker's private chain, they can collectively tilt the odds in favor of the attack. However, rational miners require assurances of profitability and verifiability—particularly the guarantee that the attacker truly holds valid blocks.

To address these challenges, we propose a *partial block sharing* strategy, whereby an attacker reveals only part of its newly found block (e.g., transaction headers) while withholding critical pieces (e.g., the nonce). This approach can attract profit-driven miners to join the attacker's private chain without fully disclosing the block. Based on partial block sharing, we develop a new **Partial Selfish Mining** (PSM) attack: an attacker withholds a discovered block, selectively shares partial data, and only reveals the "secret" part (such as the correct nonce) before or right after another miner finds a new block on the public chain. Figure 1(b) illustrates how this partial release can entice "attracted" miners to abandon the public branch, boosting the attacker's chance of winning the block race.

While partial block sharing broadens the attacker's strategy space, it raises a critical question: how can rational miners be certain the attacker actually possesses a valid block without revealing its entire content? To solve this, we incorporate *zero-knowledge proofs* that allow the attacker to prove the validity of the withheld block data *without fully exposing it*. This mechanism fosters trust among rational miners that the shared partial block is genuine, thereby ensuring that working on the attacker's private chain is indeed worthwhile. Additionally, we design an economic-based protection scheme—e.g., requiring the attacker to place deposits in a trusted third party or smart contract—to deter the attacker from performing a denial-of-service (DoS) by never revealing the full block.

Moreover, we propose an **Advanced PSM (A-PSM)** that can further increase the attacker's profit by actively monitoring the public chain's height (e.g., via an oracle [37]) and releasing hidden blocks at more opportune moments. Under A-PSM, even if the attacker's self-owned mining power is limited, the additional fraction of rational miners attracted by partial block sharing and verifiable zero-knowledge proofs can still make the attack profitable or at least no worse than classic selfish mining. We also analyze multi-attacker scenarios in which more than one malicious party adopts either PSM or A-PSM. Under such conditions, rational miners must decide which branch to join based on the timing of partial-block releases and the overall power behind each attacker's private chain.

Our theoretical and simulation results confirm that PSM can outperform selfish mining in certain ranges of network and hashing parameters, and A-PSM can achieve even higher revenues. For instance, when 50% of miners are rational, an attacker holding only 20% of total mining power can still earn approximately 1.25% more than honest mining and 9.79% more than classic selfish mining. In a similar vein, when 30% of miners are rational, an attacker with merely 10% mining power can gain up to 23.6% and 13.1% more profit than honest and selfish mining, respectively. These results highlight how partial block sharing and zero-knowledge proofs jointly reshape the power dynamics in PoW systems.

Paper Organization. In Section 2, we present an overview of the blockchain background and related work. Section 3 describes our system model and assumptions regarding rational and honest miners. In Section 4, we detail the PSM attack and show why partial block sharing can attract enough miners to the private branch. Next, Section 5 addresses how zero-knowledge proofs and a deposit scheme ensure trust and feasibility. Section 6 provides numerical analysis comparing PSM with honest and selfish mining. In Section 7, we propose the A-PSM strategy and investigate its profitability. Section 8 further explores optimal strategies for attackers and rational miners under different network scenarios, and Section 9 discusses multi-attacker situations and potential countermeasures. Finally, Section 10 concludes the paper.

2 Related Work

Selfish Mining: Selfish mining was first proposed by Eyal et al. [8]. A selfish mining attacker can earn extra rewards by intentionally generating a fork. When an attacker discovers a new block in selfish mining, it will keep the block as its private branch and keeps mining after it. When other miners find a block, the attacker will release the withheld blocks to cause a fork. The attacker can earn extra rewards once its private branch is selected as the main chain. According to [8], miners with more than 33% computational power can surely get an extra reward compared with honest mining. In [22], researchers find that for a larger parameter space, by following a more 'stubborn' strategy, miners can gain more rewards than selfish mining when their mining power is large enough. Liao et al. [18] present the whale transactions. By deploying large fees, attackers can incentivize miners to fork the blockchain. In recent years, Negy et al. [23] further analyzed the profits of selfish mining and proposed intermittent selfish mining. It assures the attacker can get more profits than honest mining even after the difficult adjustment. Li et al. [17] analyze the mining attack strategy from the honest miner's perspective and optimize the selfish mining with a hidden Markov decision process. Sapirshtein et al.[28] propose an extension to the model of selfish mining attacks in the Bitcoin protocol, providing an algorithm to find optimal policies for attackers and tight upper bounds on revenue. Our methods can further improve the attacker's rewards by attracting other miners to work in the attacker's branch. For instance, consider a scenario where half of the miners are rational with parameters $\alpha_A = 1/3$ and $\gamma = 0$. In this case, the revenue of the original selfish mining strategy is 1/3. The strategy proposed in [28] achieves a revenue of 0.33707, which is 1.12% higher than the original selfish mining strategy. However, with Partially Selfish Mining (PSM), the revenue attained is 0.3403, which is 0.96% higher than the revenue achieved by the strategy in [28] and 2.1% higher than the revenue of the original selfish mining strategy.

Other Cryptocurrency-Related Attacks: After the proposal of selfish mining, researchers have proposed various mining attack tactics against the PoW Blockchain. For those who do not have enough mining power, Loi et al. [19] proposed the block withholding (BWH) attack. By strategically splitting the mining power, attackers can get more rewards than honest mining in the long run. Kwon et al. [15] combine the BWH with selfish mining and propose the Fork After Withholding (FAW) attack. Instead of simply discarding the full block finding in the victim pools, in FAW, attackers mine on the newly mined block to generate a private branch. In recent years, Gao et al. [9] proposed a Power Adjusting Withholding (PAW) attack. PAW attackers adjust the mining power between the victim pools and solo mining, unlike FAW attackers. By doing so, PAW attackers can gain twice as much revenue as FAW attacks.

Besides mining attacks, researchers also proposed other incentive-based attacks. For example, based on the assumption that miners are inclined to choose the most profitable mining strategy to work, Michael et al. [20] propose the Blockchain Denial of Service (BDoS) attack. Instead of getting a higher reward, BDoS attacker invests resources to reduce other miners' profits and lure them away from mining. By publishing the block header of the newly mined block, the attacker signal to the miners that the system is in a state that reduces its profits. In our study, PSM attackers share the full block except for the secret data instead of sharing the block with the rational miner. Miners receiving partial block data are not likely to suffer a loss. Instead, they can reevaluate their profits and choose the most profitable way to work.

When there is a fork and a race occurs, miners follow the block they received first as the legal block. In previous work [24], researchers tend to believe that each branch has about 50% of miners to work on. However, Saad et al. investigated the Bitcoin safety properties and concluded from different angles. In [26], they pointed out that the assumption about the strong network synchrony may not hold in real-world deployment. They also implemented a HashSplit attack that allowed the attacker to orchestrate the mining power distribution when a race occurred. Their work [27] further shows that the unstable hash rate distribution can make it possible for attackers to launch a double-spending attack without mining power.

3 Model and Assumptions

3.1 Mining Model

We consider a blockchain system with n miners whose normalized mining powers are denoted by $\alpha_1, \alpha_2 ... \alpha_n$, and an attacker with mining power α_A, such that $\alpha_A + \sum_{i=1}^{n} \alpha_i = 1$.

As shown in Figure 2, in PSM attacks, without loss of generality, we assume that miners are divided into the following groups:

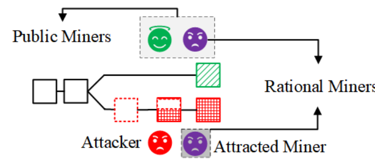


Figure 2: Miners' roles in PSM scenario.

- **Attacker:** A miner or a colluding minority pool that has a newly mined block(s) and follows the PSM strategy. It can preserve a mined block(s), form a private branch, and share partial block data with rational miners like [20]. Besides, it also has access to the smart contract to share the proof of block possession and other needed data with rational miners.
- **Rational miners:** A group of miners that are profit-driven but still follow PoW to mine. In most cases, this group is the minority. Receiving a partial block released by an attacker as in PSM, rational miners can choose their optimal strategy (i.e., mining on which branch) to get a higher reward.
- **Attracted miners:** Part of rational miners that choose to work on the attacker's private branch, mining after the partial block. We name the strategy taken by attracted miners as the *greedy mining strategy*.
- **Public miners:** Miners that take honest mining to truthfully generate and release new blocks in the longest chain. We name the strategy taken by public miners as the *public mining strategy*. Non-attracted rational miners are considered public miners and follow the public mining strategy.

Note that all attracted miners are rational miners. The reverse does not always hold since some rational miners may choose to mine on the original public branch according to their optimal strategy even under PSM attacks. A miner, rather than the attacker, is either an attracted miner or a public miner under attacks. If another miner (e.g., a non-public miner) finds a new block and launches the PSM attack simultaneously, we have multiple attackers and such cases are discussed in Section 9.

We denote the rushing ability of the attacker by γ. If the attacker publishes a new whole block from its private branch to race with other miners (e.g., a new block broadcast in the public branch), γ is the expected ratio of public miners that receive the attacker's block first. γ mainly depends on the miner's network condition. In previous work, γ is commonly considered as $\frac{1}{2}$ [8, 28]. However, some researchers also propose methods that allow the attacker to get a larger γ [26]. In this paper, we consider $0 < \gamma < 1$.

3.2 Assumptions

We have the following assumptions that are consistent with other PoW-based mining attacks, such as [8], [22], [10], [9].

(1) Instead of 6.25 BTC, the reward of a new block is normalized to 1. Our analysis gives the expected reward for every participant [2].
(2) Each miner/pool's computational power should not be greater than 0.5 to avoid "51% attacks" [3].
(3) A miner's expected normalized reward equals the probability of finding a valid block in each round. This assumption

is reasonable because according to [10], the probability of unintentional forks is around 0.41%, which is negligible.

(4) Rational miners do not trust the attacker and join its private branch unless the attacker can provide an extra method to guarantee the profits of rational miners. Consequently, the traditional selfish miner cannot attract rational miners.
(5) Not all the miners in the blockchain system are rational. Though we proposed some measures to address the concern of miners working on the attacker's private branch, we take the possibility into consideration that they cannot allay the concerns of all miners. Thus, we discuss the fraction of rational miners to be no more than 50% in this paper.
(6) Rational miners tend to pull external information and adjust their mining strategy accordingly. This assumption is commonly made in previous work [6, 20].

4 Partial Selfish Mining Attack

In PSM, an attacker follows the partial block sharing strategy and shares the partial block information with rational miners. The partial block has some data covered, e.g., *nonce* and part of arbitrary bytes in the coinbase transaction. Miners can mine after it to get a new block. The hidden data can be recovered by others, spending considerable mining power. We denote the hidden data as *secret*.

The generic PSM can first withhold the new block, then release the partial block, and finally broadcast the complete block. If an attacker does not release the partial block or complete block in PSM, it becomes selfish mining. The attacker can also bypass the first two steps and directly publish the complete block, and it is honest mining. Or the attacker can bypass the first step (when selfish mining is not the dominant strategy as discussed in Section 8) but continue the second and third steps, which has not been discussed previously and is the focus of this section.

4.1 Attack Overview

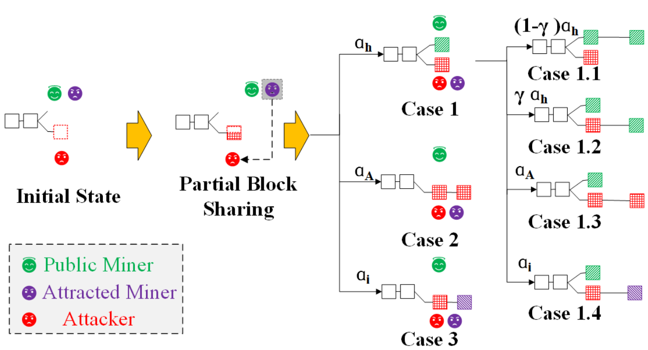


Figure 3: Workflow of PSM strategy. Instead of publishing a new block, the attacker shares partial block data with other miners to attract them to join its private branch.Three possible cases of finding another new block after the announcement of the partial block. Case 1: By public miners; Case 2: By the attacker; Case 3: By attracted miners.

The workflow of PSM is shown in Figure 3. In PSM, when the attacker finds a block, it keeps the block private instead of immediately releasing it. In the meantime, the attacker releases the partial block with proof of block possession. With these released data, rational miners could mine on the private branch. To assure the profits of attracted miners, the attacker will also announce a smart contract that allows the rational miner to get the attacker's collateral if it does not release the secret promised. For simplicity and fairness, we assume that all honest and rational miners will work on the branch first received if a race happens. An attacker can even lure miners to work on his branch in the race case by adding transactions with lower fees to its block and leaving high-fee transactions to be collected by others.

After the release of the partial block, three possible cases may happen: the attacker finds the new block on its private branch, public miners find a new block on the public branch or attracted miners find a new block on the private branch.

In the first case, when miners find a new block on the public branch, the attacker releases its private branch and starts a 0-lead racing. In this scenario, the attacker and the attracted miners will mine on the previously private branch, and public miners will choose to mine on either branch. As defined earlier, γ public miners will work on the private branch, and $1 - \gamma$ public miners will mine on the public branch.

In the second case, when the attacker finds a new block on the private branch, the attacker will release the whole private branch and get the two blocks' revenue. Then the system goes back to the single-branch state.

In the third case, when the attracted miners find a new block in the private branch, the attacker will immediately release the whole private branch to get the revenue of all the blocks on the private branch. Similar to the first case, the blockchain returns to the single-branch state.

Attackers can decide among PSM, honest or selfish mining based on profitability. In the following, we illustrate the reward of PSM. The reward of honest and selfish mining is given in Section 6 for comparisons. We show the optimal mining strategy for the attacker in Section 8.

4.2 PSM Reward

We use the following parameters in our analysis.

- γ: Ratio of public miners choosing attacker's branch when a race occurs.
- α_i: Mining power of attracted miners.
- α_k: Mining power of one rational miner k.
- α_A: Mining power of the attacker.
- α_h: Mining power of public miners.
- R_m^n: Revenue of miner n in case m.
- H: Public mining strategy.
- S: Selfish mining strategy.
- P: Partial selfish mining strategy.

Figure 4 shows the progress of the PSM attack as a state machine. The state of the system shows the lead status of the attacker's private branch. Zero lead is separated into states 0 and 0'. State 0 means there is no branch in the blockchain, and the public chain is the longest chain. State 0' is the state where there are two branches

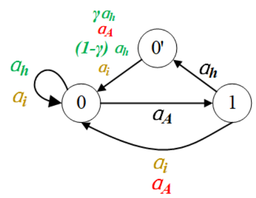


Figure 4: State machine of PSM when miners with overall α_i mining power working greedy.

on the blockchain, one is the original public chain, and the other is the attacker's previous private chain. The value on each arrow indicates the possibility of state transition.

Based on the above state machine, we can derive the PSM attacker's profit as follows:

Theorem 4.1. *The profit of a PSM attacker is*

$$R_P^A = \frac{\alpha_A\alpha_h^2\gamma + (\alpha_A\alpha_h + \alpha_A)\alpha_i + 2\alpha_A^2\alpha_h + 2\alpha_A^2}{(2\alpha_A\alpha_h + 2\alpha_A + 1)\alpha_i + 2\alpha_A\alpha_h^2 + (2\alpha_A^2 + 1)\alpha_h + 2\alpha_A^2} \quad (1)$$

The proof of Theorem 4.1 is given in Appendix A.

For simplicity, we exclude the block reward variable and network-related factors other than rushing ability in our analysis. This is due to the relatively minor impact of these factors when compared with the mining power and rushing ability.

4.3 Rational Miners' Profits in PSM

To attract rational miners to work on its private branch, the attacker needs to assure that the rational miners can get more rewards than public mining, and make a solid promise to guarantee that the attacker's cost of breaking the promise is unbearably high. In this section, based on our assumption in Section 3, we analyze the rational miner's strategy space and its profits.

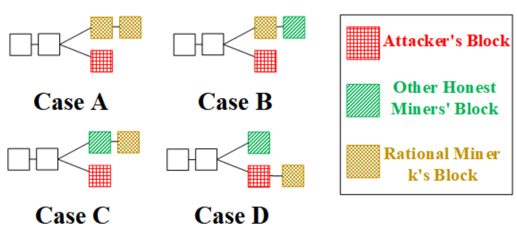


Figure 5: Four reward cases exist when a rational miner k follows the public mining strategy.

Assuming in a bitcoin-like blockchain network, rational miner k will choose the most profitable branch to work based on the information it received. Miner k can know the α_A from the attacker's message. In a balanced blockchain network with low congestion, γ is usually $\frac{1}{2}$ [8, 28]. In practice, the network status changes dynamically, and miners can hardly get the exact γ value. Rational miners could estimate the γ value based on the current network status or simply consider the worst case ($\gamma = 0$). We are interested in which strategies can maximize the miner's profit with different mining power, γ, and α_A.

From the state machine in Figure 4, The miner k's revenue R_G^k can be expressed as follows:

Theorem 4.2. *When miner k follows the greedy mining strategy, miner k's revenue is*

$$R_G^k = \frac{(\alpha_A\alpha_h + \alpha_A + 1)\alpha_k}{(2\alpha_A\alpha_h + 2\alpha_A + 1)\alpha_k + 2\alpha_A\alpha_h^2 + (2\alpha_A^2 + 1)\alpha_h + 2\alpha_A^2}. \quad (2)$$

Proof. According to Appendix A, the overall revenue of all attracted miners can be expressed as:

$$R_P^G = \frac{(\alpha_A\alpha_h + \alpha_A + 1)\alpha_i}{(2\alpha_A\alpha_h + 2\alpha_A + 1)\alpha_i + 2\alpha_A\alpha_h^2 + (2\alpha_A^2 + 1)\alpha_h + 2\alpha_A^2} \quad (3)$$

Rational miners make their decisions independently. Therefore, miner k faces difficulty in accurately determining the number of rational miners in the network. As a cautious rational miner, miner k should consider the worst-case scenario when the attacker has small mining power, assuming it is the only attracted miner in the network, which means $\alpha_i = \alpha_k$. Then, we have Equation (2). □

When miner k follows the public mining strategy, there are four possible reward cases, as shown in Figure 5. Its revenue can be derived as follows:

Theorem 4.3. *When miner k follows the public mining strategy, its revenue is:*

$$R_P^k = \frac{(\alpha_A\alpha_k^2 + (\alpha_A^2 - \alpha_A)\alpha_k)\gamma + ((-2\alpha_A^2) + 2\alpha_A + 1)\alpha_k}{\alpha_A + 1}. \quad (4)$$

The proof of Theorem 4.3 is in Appendix B.

Here, we first consider a specific case: an attacker with the computational power of 0.1 who executes the PSM attack against the blockchain. For different α_k, the expected profits of following greedy and public mining strategies are shown in Figure 6.

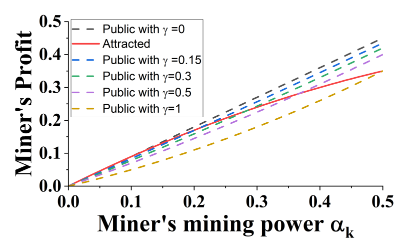


Figure 6: Revenue comparisons for a miner to choose greedy or public mining strategies when attacker's mining power $\alpha_A = 0.1$. Note that the rushing ability γ has no impact on the revenue of the greedy strategy.

To evaluate the profit of the greedy strategy over the public mining strategy, we calculate the relative extra rewards (RER). The expected RER of miner k when adopting strategy s_1 rather than s_2 can be expressed as:

$$RER_k^{s_1,s_2} = \frac{R_{s_1}^k - R_{s_2}^k}{R_{s_2}^k}, \quad (5)$$

where $R^k_{s_1}$ and $R^k_{s_2}$ represent the reward of k when adopting s_1 and s_2 strategies respectively.

Theorem 4.4. *For rational miner k, working on the attacker's private branch is always more profitable when $\alpha_k < \alpha_A$.*

Proof. Rational miners, including mining pools, have independent mining decisions and processes. It is hard to predict the ratio of attracted miners in the network. When judging his profit, miner k should consider the worst case that there are no other attracted miners in the whole network, i.e., $\alpha_i = 0$.

The RER of miner k when following greedy mining instead of public mining strategy is:

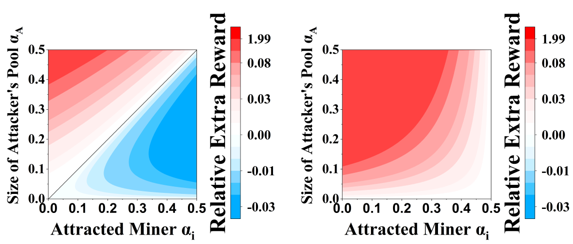
$$RER^{G,H}_k = \frac{R^G_k - R^H_k}{R^H_k} = \frac{((-\alpha_A\alpha_i) - \alpha_A^2 + \alpha_A)\gamma - \alpha_A\alpha_i + \alpha_A^2}{(\alpha_A\alpha_i + \alpha_A^2 - \alpha_A)\gamma - 2\alpha_A^2 + 2\alpha_A + 1}. \tag{6}$$

$RER^{G,H}_k > 0$ means that being greedy is more profitable for the miners. A negative value means that the miner suffers from a loss when choosing the greedy strategy.

With $\alpha_h = 1 - \alpha_A - \alpha_k$ ($\alpha_i = 0$), when $RER^{G,H}_k > 0$, we have:

$$\begin{gathered} ((-\alpha_A\alpha_k) - \alpha_A^2 + \alpha_A)\gamma - \alpha_A\alpha_k + \alpha_A^2 > 0 \\ \alpha_k < \frac{\alpha_A - (\alpha_A - 1)\gamma}{1+\gamma} \end{gathered} \tag{7}$$

From Equation (7), we can see that the miner's RER mainly depends on γ and α_A. In the worst case when $\gamma = 0$, miner k will get more rewards on the attacker's private branch if $\alpha_k < \alpha_A$. Figure 7 shows the quantitative simulation results. □



(a) Rushing ability γ=0 (b) Rushing ability γ=1

Figure 7: With a PSM attacker, the rational miner's relative extra reward when choosing greedy mining strategy instead of public mining strategy ($RER^{G,H}_k$). The solid line represents no extra reward. For miner k, when $\gamma = 0$, working on the private branch is more profitable only when $\alpha_k < \alpha_A$.

To verify the theoretical results, we simulate the RER of an attracted miner with a mining power of 0.2, using a Monte Carlo method over 10^9 rounds, with an upper bound of 10^{-4} error. Table 1 shows the Monte Carlo simulation results. The results are consistent with our expectation from Equation (7).

Section 5 provides a detailed mechanism design to convince rational miners that working on the attacker's private branch is a profitable endeavor. It includes a thorough examination of various aspects, such as the proof of block possession and a profit protection mechanism that ensures that rational miners receive the revenue promised.

Table 1: Monte Carlo simulation result of rational miner k's relative extra reward ($RER^{G,H}_k$).

γ	α_A			
	0.1	0.2	0.3	0.4
0	−5.00 (−5.00)	0.02 (0)	6.25 (6.25)	14.30 (14.29)
0.5	22.56 (22.56)	28.47 (28.47)	35.96 (36.00)	45.34 (45.45)
1	72.80 (72.73)	79.98 (80)	88.79 (88.89)	100.02 (100.0)

5 Feasibility of PSM

In this section, we describe the mechanism to prove that rational miners working on the attacker's private branch are profitable in detail. To assure that the attacker will follow the rules announced, the attacker needs to provide proof of block possession and a profit protection mechanism that assures the rational miners can get the promised revenue.

5.1 Proof of Block Possession

Specifically, to provide proof of block possession, the attacker writes some random information r to the coinbase transaction of the new block to provide sufficient randomness and uses r and the *nonce* of the block as the witness. Other parts of the block header besides *nonce* and r, denoted as b, can be used as the public statement, including information such as the Merkle root of transactions in the block, and the hash h of the block which satisfies the difficulty requirement. After this, the attacker needs to prove that:

(1) It knows *nonce* and r such that $H(b, nonce, r) = h$; and
(2) The *nonce* and b are well-formed, i.e., *nonce* is a 32-bit integer.

To this end, the attacker needs to generate a proof of block possession $\pi_b \xleftarrow{R} \mathsf{Prove}((b,h),(nonce,r))$ to prove the following relation

$$H(b, nonce, r) = h \ \wedge\ 0 \le nonce \le 2^{32} - 1 \tag{8}$$

is satisfied. The attacker then makes the tuple (π_b, b, h) publicly available on a dedicated website. In this way, other miners can calculate

$$\mathsf{Verify}((b,h),\pi_b) \stackrel{?}{=} 1 \tag{9}$$

to verify whether the proof π_b holds. If the above verification passes, the other miners can be sure that the attacker holds a specific *nonce* and r, which enables the hash h of the new block to satisfy the difficulty requirement.

If the attacker does not want to share the block information with every miner, it can also share the partial block data b with a specific miner. For this purpose, it can take advantage of the zero-knowledge contingent payment (ZKCP) protocol [4]. We propose an example block-sharing strategy as follows:

Before the exchange starts, the attacker can deploy a smart contract as an arbiter. Then, the attacker and the miner achieve a fair exchange of the partial block data through a simple interaction:

(1) The attacker generates a random key k and encrypts the b value with k, i.e., $\hat{b} = \mathsf{Enc}(b, k)$. Next, it computes the hash

of k $h_k \leftarrow H(k)$ and generates a proof:

$$\pi_e \xleftarrow{R} \mathsf{Prove}((\hat{b}, h, h_k), (b, k, nonce, r)) \tag{10}$$

to prove the following relation:

$$\begin{aligned} H(b, nonce, r) = h \ \wedge \ 0 \leq \text{nonce} \leq 2^{32} - 1 \\ \wedge \ \hat{b} = \mathsf{Enc}(b, k) \wedge h_k = H(k) \end{aligned} \tag{11}$$

is satisfied. Finally, it sends the tuple $(\pi_e, \hat{b}, h, h_k)$ to the miner.

(2) After receiving the tuple $(\pi_e, \hat{b}, h, h_k)$ from the attacker, the miner can verify whether π_e is valid by computing $b \leftarrow \mathsf{Verify}((\hat{b}, h, h_k), \pi_e)$. If $b = 1$, the miner deposits to the arbiter contract the payment agreed upon by both parties beforehand, as well as the hash h_k.
(3) The attacker checks whether the h_k provided by the miner is valid and whether the payment deposited is as previously agreed. If so, the attacker sends the key k to the arbiter contract.
(4) The arbiter contract verifies that $h_k = H(k)$ holds. If it is the case, the contract transfers the payment to the attacker, otherwise, it returns the payment to the miner. Since the k disclosed to the contract will also be available to the miner at the same time, the miner can decrypt $\hat{b}$ by $b \leftarrow \mathsf{Dec}(\hat{b}, k)$.

Any miner as a buyer cannot obtain any information about b without completing the payment. Meanwhile, any attacker acting as a seller cannot cheat the payment by submitting the wrong b. In this way, the attacker is able to sell the partial block data b to a specific miner and gain revenue.

To evaluate the performance of proof generation and verification, a computer running Ubuntu 22.04 with a 3.50 GHz Intel i9-11900k CPU and 32 GB of RAM was used. We constructed the above zero-knowledge proofs based on libsnark and implemented the proof of block possession and ZKCP-based block information exchange using about 3100 lines of C++ code.

For the performance of Block System Possession, the proving time is 2.77 s, the verification time is 19 ms, and the proof size is 517 Byte. As can be seen from the results, it takes only 2-3 seconds to generate proof of block possession. The process of generating such proofs is all at once. Once generated, these proofs can be made public along with the statement, and anyone can verify the correctness of the declared relations by evaluating the proofs. The time required to verify such succinct proofs is in the millisecond range and is, therefore, very efficient.

For the performance of Block System Possession, the proving time is 2.77 s, the verification time is 19 ms, and the proof size is 517 Byte. As can be seen from the results, it takes only 2-3 seconds to generate a proof of block possession. and about 10 seconds to generate a proof for block information exchange.

5.2 PSM-DoS Attack and The Promise of Secret Publication

It should be noted that the above mechanism can only prove to other miners that the attacker has indeed mined a block that satisfies the requirement. But there is no guarantee that the attacker will share the secret in the future. By potentially withholding the reserved block, the attacker could waste the attracted miner's mining power and may cause a DoS attack on the public branch. Since this new attack has to launch with PSM, we call it PSM-DoS attack.

The workflow of the PSM-DoS attack is shown in Figure 8. First, the attacker distributes the partial block data to all the miners and attracts attracted miners to join the attacker's private branch. In the meantime, the attacker leaves the private branch and puts all the mining power back into the public branch. Then, once attracted miners on the private branch find the new block, the attacker refuses to release the first block. In this case, the public chain will not accept the new block because its previous block is not published. Thus, miners who try to follow the attacker will suffer losses if the attacker consistently fails to disclose the full block to other miners.

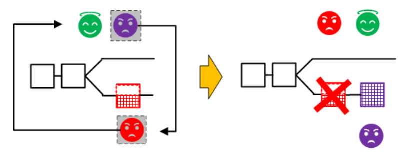


Figure 8: Workflow of PSM-DoS Attack. After attracting the attracted miners to work on its private branch, the attacker discards the secret and goes back to work on the public branch.

The PSM attacker must implement countermeasures to mitigate the potential concerns of attracted miners regarding PSM-DoS attacks. Here, we present two examples:

First, the attacker can address the concerns of attracted miners who discover the new block by carefully selecting the number of hidden bytes, ensuring that the calculation of these hidden bytes can be performed within an acceptable timeframe. As we have demonstrated in Section 5.1, when the attacker shares the partial block, miners receive the full block without the *nonce* and several bytes of coinbase transaction information. Assuming that the attacker hides n bytes of data, attracted miners need to calculate 2^{4n} times of hash to get the hidden data.

Take Bitcoin as an example. According to [21], with difficulty D, we can approximate the hash power of all the miners in the network as $D \times \frac{2^{32}}{600}$ H/s. Assuming that attacker hides b bytes of data so that fr percent of miners need to calculate for a duration of T_c, then we have

$$T_c = \frac{2^{8b}}{fr \times D \times \frac{2^{32}}{600}}, \quad b = \log_{2^8}(T_c \times fr \times D \times \frac{2^{32}}{600}). \tag{12}$$

According to [5], the Hashrate of the Bitcoin network is $1.55371 \times 10^{16} H/s$. The difficulty of bitcoin $D_{btc} \approx 31.25 \times 10^{12}$. By employing Equation (12), hiding 7 bytes of data enables a 1% mining power miner to compute the hidden bytes in about 1 second. Similarly, hiding 8 bytes of data allows a 1% mining power miner to determine the hidden bytes in about 1 minute. Though we take Bitcoin as the example, the proposed method works not only in Bitcoin but also in other PoW-based public blockchains (or Bitcoin-like) systems.

Another method to incentivize miners and address concerns is to create a smart contract with collateral to cover rewards and costs incurred within smart contracts. The contract includes cryptographic commitment of block data. By doing so, the attacker effectively guarantees that it will redeem the collateral at some agreed point in the future by submitting a secret to the contract that satisfies the requirement and opens the commitment. Otherwise, this collateral will be transferred to the attracted miner who finds the new block. On the other hand, when the collateral value is less than the attacker's cost to mislead other miners, the miner has every incentive to provide an invalid block to deceive other miners. To solve the problem, the collateral value is required to be much larger than the cost of mining for the specified period. Specifically, the flow of this smart contract is as follows.

(1) The attacker pre-collateralizes the contract with some coins in the contract with a value equivalent to the proceeds of mining n blocks.
(2) An attracted miner discovers a subsequent new block of the attacker's partial block and submits the information about the new block to the smart contract.
(3) The smart contract verifies the new block. If the verification passes, the contract opens a challenge period within which the attacker should disclose the partial block. If the attacker discloses the full information of the partial block within the challenge period, it can redeem all the collateral it has previously deposited. Suppose the attacker fails to disclose the block information within the challenge period or discloses an incorrect block. In that case, the contract transfers the attacker's collateral to the attracted miner who finds the new block.

Assuming the challenge period lasts for a duration T_C, the miner will get the revenue, and the system will surely go back to a single branch state after T_C. If the attacker refuses to publish the full block, the attacker will lose both the partial block revenue and the collateral. The attacker can get more profits by launching PSM-DoS only if it could get more than $n+1$ blocks within T_C.

According to [12], the possibility of finding a new block within duration T can be expressed as:

$$R(T) = \alpha_e \times (1 - (1 - p_e)^{\frac{T}{T_{avg}}}), \tag{13}$$

where the T_{avg} is the average block generation time in the blockchain. In bitcoin $T_{avg} = 10$ minutes. $p_e = 64\%$, which means all miners in the network have a possibility of 64% generating a new block within 10 minutes.

If the attacker can provide large enough collateral with a low enough T_C, then launching a PSM-DoS attack is economically not worthwhile.

The attacker has the option to employ either both methods or one of these approaches to address the concerns raised by the attracted miners. Though our proposed method could address most of the concerns of rational miners, we agree that it is not realistic to assume all the miners are rational in practice. In the following, we discuss the attacker's gain with different ratios of the rational miners.

6 PSM Analysis and Comparisons

When calculating the revenue, an attacker can estimate α_A, γ, and mining power distribution in a blockchain. It can also roughly estimate the number of rational miners by the number of applications (e.g., via smart contract) for partial block data. However, it cannot know the fraction of rational miners in the entire network. Thus, the attacker needs to estimate its reward with a different fraction of rational miners. Our assumptions here are consistent with those in recent studies [25, 27].

In this section, we use numeric analysis to evaluate the reward of the PSM strategy.

Table 2: Monte Carlo simulation results of attacker's RER when choosing PSM instead of honest mining.

γ	α_i			
	0.1	0.2	0.3	0.4
0	−29.167 (−29.17)	−20.0 (−20.0)	−12.5 (−12.5)	−6.67 (−6.67)
0.5	−8.75 (−8.75)	−5.0 (−5.0)	−2.08 (−2.08)	0.0 (0.0)
1	11.67 (11.67)	10.0 (10.0)	8.33 (8.33)	6.67 (6.67)

6.1 Comparison with Honest Mining

6.1.1 Quantitative Analysis. We mathematically analyze the revenue of PSM against honest mining. As stated before, the attacker's computational power is α_A. If the attacker chooses to follow the honest mining strategy, the possibility of getting the second block revenue is α_A. Overall, the attacker's expected profit when following honest mining is:

$$R_H^A = \alpha_A. \tag{14}$$

If the attacker chooses the PSM attack, it could attract rational miners with α_i mining power, and the total mining power of public miners is $\alpha_h = 1 - \alpha_A - \alpha_i$. For the attacker, the expected RER of following PSM instead of honest mining can be expressed as follows:

$$\begin{aligned} RER_A^{P,H} &= \frac{R_P^A - R_H^A}{R_H^A} \\ &= \frac{(\alpha_A\alpha_h + \alpha_A + 1)\alpha_i}{(2\alpha_A^2\alpha_h + 2\alpha_A^2 + \alpha_A)\alpha_i + 2\alpha_A^2\alpha_h^2 + (2\alpha_A^3 + \alpha_A)\alpha_h + 2\alpha_A^3} - 1. \end{aligned} \tag{15}$$

In Figure 9, we show the numerous simulation results of RER of the PSM strategy over the honest mining strategy. The higher the γ is, the more likely the attackers can get more rewards than honest mining. When $\gamma = 1$, PSM can surely get more rewards than honest mining. If it can attract enough rational miners, the PSM attacker with enough mining power can still get more rewards than honest mining when $\gamma = 0$.

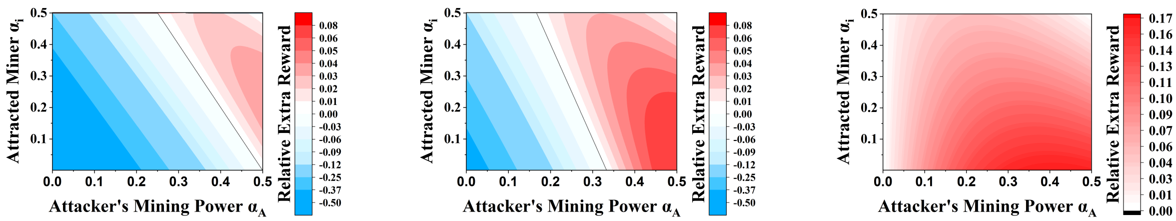
6.1.2 Simulation Results. To further verify the accuracy of our quantitative results, we implement a Monte Carlo simulator in Java to verify our theoretical analysis. We simulate an attacker with $\alpha_A = 0.2$ and run the simulator over 10^9 rounds. The upper bound for error is 10^{-4}. The results are shown in Table 2. The attacker's RER is the same as expected.

6.2 Comparison with Selfish Mining

6.2.1 Quantitative Analysis. According to [8], if choosing selfish mining, the attacker's reward is:

$$R_S^A = \frac{\alpha_A(1-\alpha_A)^2(4\alpha_A + \gamma(1-2\alpha_A)) - \alpha_A^3}{1 - \alpha_A(1 + (2-\alpha_A)\alpha_A)}, \tag{16}$$

and the expected RER of PSM over selfish mining is



(a) Rushing ability γ=0 (b) Rushing ability γ=0.5 (c) Rushing ability γ=1

Figure 9: Attacker's RER when choosing PSM instead of honest mining ($RER_A^{P,H}$). The solid line represents no extra reward.

Table 3: Monte Carlo simulation results of attacker's RER when choosing PSM instead of selfish mining.

γ	α_i			
	0.1	0.2	0.3	0.4
0	8.07 (8.05)	22.04 (22.03)	33.51 (33.47)	42.35 (42.37)
0.5	−1.06 (−1.05)	3.01 (3.01)	6.20 (6.17)	8.42 (8.43)
1	−6.08 (−6.07)	−7.48 (−7.48)	−8.88 (−8.88)	−10.29 (−10.28)

$$RER_A^{P,S} = \frac{\alpha_A^3 - 2\alpha_A^2 - \alpha_A + 1}{\alpha_A + 1} \times \left(\frac{(\gamma - 1)\alpha_i^2 + (2(\alpha_A - 1) - (3\alpha_A - 2)\gamma)\alpha_i}{(2\alpha_A^3 - 5\alpha_A^2 + 4\alpha_A - 1)\gamma - 4\alpha_A^3 + 9\alpha_A^2 - 4\alpha_A} + \frac{(\alpha_A - 1)^2\gamma + 4\alpha_A - 2\alpha_A^2}{(2\alpha_A^3 - 5\alpha_A^2 + 4\alpha_A - 1)\gamma - 4\alpha_A^3 + 9\alpha_A^2 - 4\alpha_A} \right) - 1 \tag{17}$$

In Figure 10, we show the numerous simulation results of the attacker's RER following PSM instead of the selfish mining strategy. The PSM attacker can get a higher reward than the selfish miner when its mining power is relatively small. When the attacker's mining power is large enough, the possibility of finding more than one block on its private branch becomes non-negligible. Thus, the revenue of selfish mining is higher than PSM. We propose an advanced PSM strategy to address this issue in Section 7.

6.2.2 Simulation Results. To further verify the accuracy of our quantitative results, assuming the attacker with a computation power of 0.2, we compare the Monte Carlo simulation results of the RER of PSM over selfish mining with our evaluation results. We run the Java-based simulator over 10^9 rounds. The upper bound for error is 10^{-4}. The Monte Carlo simulation results are shown in Table 3. The attacker's RER is the same as expected.

7 Advanced PSM Strategy

Under the condition that the attacker can have the up-to-date block height from the public chain, we propose an optimized PSM strategy named Advanced PSM (A-PSM), which can further increase the profits of PSM for the attacker.

7.1 Attack Overview

When miners find new blocks, the A-PSM attacker follows the selfish-mining-like strategy to publish the partial-released block instead of simply releasing the partial blocks' secrets. The attacker will keep the secret private until the lead of the private branch is no more than 2 blocks. Attracted miners can immediately release the block it finds, but it will not be recognized as a valid block until the secret of the prior partial block is released.

Specifically, if a public miner finds a new block, two possible cases may happen: if the lead of the private branch is 2, the attacker will release all the partial blocks, and the system goes back to the single branch state. If the private branch's lead is more than 2, then the attacker and rational miners will continue working on its private branch until the lead is 2.

To avoid cases where the private branch's length grows faster than the public branch, we extend our assumption that in the A-PSM scenario, the mining power of the attacker, together with rational miners, is no more than 50%.

We also extend the assumption that the attacker can promise that it will release the secret based on the length of both the private and the public branches. This assumption is reasonable because when launching mining attacks, attackers are motivated to maximize their revenue, and the secret computation mechanism in Section 5 further assures the malicious behavior is economically not worthwhile for the attacker. To further address the rational miner's concern, the attacker can also have the latest block height of the public branch via decentralized oracle [37] or reported by an attracted miner.

7.2 A-PSM Reward

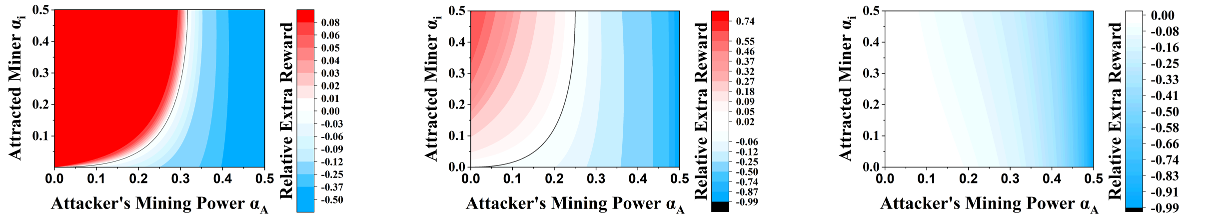
Overall, if choosing the A-PSM strategy, the attacker and attracted miners' expected profits can be derived as follows:

Theorem 7.1. *The profit of an A-PSM attacker is:*

$$R_{AP}^A = \frac{2\alpha_A\alpha_i^3 + (8\alpha_A^2 - 5\alpha_A)\alpha_i^2 + (10\alpha_A^3 - 14\alpha_A^2 + 2\alpha_A)\alpha_i}{\alpha_A\alpha_i^2 + (2\alpha_A^2 - 2\alpha_A - 2)\alpha_i + \alpha_A^3 - 2\alpha_A^2 - \alpha_A + 1} + \frac{(-\alpha_A(\alpha_i + \alpha_A - 1)^2(2\alpha_i + 2\alpha_A - 1))\gamma + 4\alpha_A^4 - 9\alpha_A^3 + 4\alpha_A^2}{\alpha_A\alpha_i^2 + (2\alpha_A^2 - 2\alpha_A - 2)\alpha_i + \alpha_A^3 - 2\alpha_A^2 - \alpha_A + 1}. \tag{18}$$

Theorem 7.2. *The attracted miner's expected profit is*

$$R_{AP}^i = \frac{2\alpha_A\alpha_i^3 + (4\alpha_A^2 - 4\alpha_A - 2)\alpha_i^2 + (2\alpha_A^3 - 4\alpha_A^2 + 1)\alpha_i}{\alpha_A\alpha_i^2 + (2\alpha_A^2 - 2\alpha_A - 2)\alpha_i + \alpha_A^3 - 2\alpha_A^2 - \alpha_A + 1}. \tag{19}$$



(a) Rushing ability γ=0 (b) Rushing ability γ=0.5 (c) Rushing ability γ=1

Figure 10: Attacker's RER when choosing PSM instead of selfish mining. The solid line represents no extra reward.

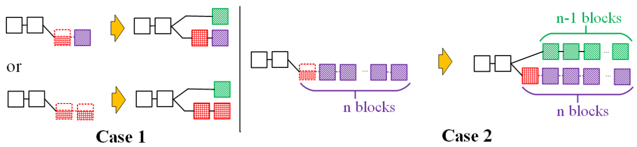


Figure 11: Partial block sharing with A-PSM strategy.

The proof of Theorem 7.1 and Theorem 7.2 is in Appendix C.

7.3 Profit Analysis

Rational Miners' profits Analysis: For rational miner k, when following the public mining strategy, we can derive the following theorems:

Theorem 7.3. *When following the public mining strategy, the rational miner k's expected profit' is:*

$$R_P^{k,APSM} = \frac{(\alpha_A(1-2\alpha_A)\alpha_k(\alpha_k+\alpha_A-1))\gamma}{\alpha_A^3-2\alpha_A^2-\alpha_A+1} + \frac{(4\alpha_A^3-6\alpha_A^2+1)\alpha_k}{\alpha_A^3-2\alpha_A^2-\alpha_A+1} \tag{20}$$

The RER of following the A-PSM instead of public mining is:

$$RER_k^{AP,H} = \frac{((-\alpha_A\alpha_i)-\alpha_A^2+\alpha_A)\gamma-\alpha_A\alpha_i+\alpha_A^2}{(\alpha_A\alpha_i+\alpha_A^2-\alpha_A)\gamma-2\alpha_A^2+2\alpha_A+1}. \tag{21}$$

The proof of Theorem 7.3 is in Appendix D.

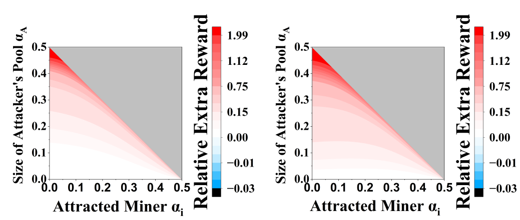
In Figure 12, we show numerous simulation results of miners' RER when following the greedy mining strategy instead of the public mining strategy with an A-PSM attacker.

Attacker's Profit Analysis: When following the honest mining strategy, the attacker's revenue is shown in Equation (14). The RER of the attacker following A-PSM rather than an honest mining strategy is:

$$\begin{aligned} RER_A^{AP,H} = {} & \frac{(-\alpha_A(\alpha_i+\alpha_A-1)^2(2\alpha_i+2\alpha_A-1))\gamma}{\alpha_A^2\alpha_i^2+(2\alpha_A^3-2\alpha_A^2-2\alpha_A)\alpha_i+\alpha_A^4-2\alpha_A^3-\alpha_A^2+\alpha_A} + \\ & \frac{2\alpha_A\alpha_i^3+(8\alpha_A^2-5\alpha_A)\alpha_i^2+(10\alpha_A^3-14\alpha_A^2+2\alpha_A)\alpha_i}{\alpha_A^2\alpha_i^2+(2\alpha_A^3-2\alpha_A^2-2\alpha_A)\alpha_i+\alpha_A^4-2\alpha_A^3-\alpha_A^2+\alpha_A} + \\ & \frac{4\alpha_A^4-9\alpha_A^3+4\alpha_A^2}{\alpha_A^2\alpha_i^2+(2\alpha_A^3-2\alpha_A^2-2\alpha_A)\alpha_i+\alpha_A^4-2\alpha_A^3-\alpha_A^2+\alpha_A} - 1. \end{aligned} \tag{22}$$

The reward of following the selfish mining strategy is shown in Equation (16). The RER of A-PSM over selfish mining strategy is:

$$\begin{aligned} RER_A^{AP,S} = {} & \frac{1-\alpha_A(1+(2-\alpha_A)\alpha_A)}{\alpha_A(1-\alpha_A)^2(4\alpha_A+\gamma(1-2\alpha_A))-\alpha_A^3} \times \\ & (\frac{(-\alpha_A(\alpha_i+\alpha_A-1)^2(2\alpha_i+2\alpha_A-1))\gamma+4\alpha_A^4-9\alpha_A^3+4\alpha_A^2}{\alpha_A\alpha_i^2+(2\alpha_A^2-2\alpha_A-2)\alpha_i+\alpha_A^3-2\alpha_A^2-\alpha_A+1} + \\ & \frac{2\alpha_A\alpha_i^3+(8\alpha_A^2-5\alpha_A)\alpha_i^2+(10\alpha_A^3-14\alpha_A^2+2\alpha_A)\alpha_i}{\alpha_A\alpha_i^2+(2\alpha_A^2-2\alpha_A-2)\alpha_i+\alpha_A^3-2\alpha_A^2-\alpha_A+1}) \end{aligned} \tag{23}$$

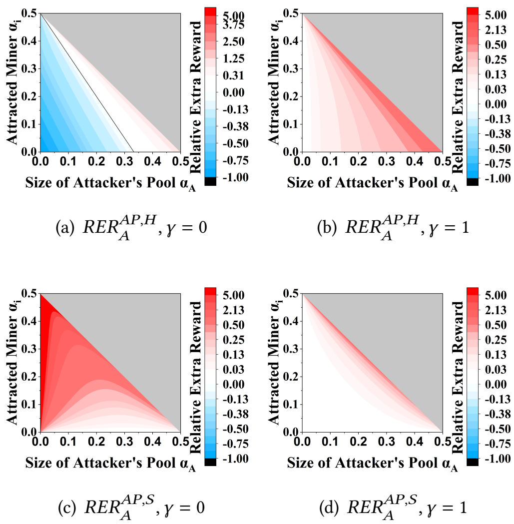


(a) Rushing ability γ=0 (b) Rushing ability γ=1

Figure 12: The relative extra rewards of the rational miner when choosing the greedy mining strategy instead of public mining strategy with an A-PSM attacker ($RER_i^{AP,H}$).

In Figure 13(a) and 13(b), we show the simulation results of the attacker's RER when following A-PSM instead of the honest mining strategy with $\gamma = 0$ and 1, respectively. The results of the attacker's RER of A-PSM over selfish mining are shown in Figure 13(c) and 13(d) with $\gamma = 0$ and 1, respectively.

We compare the Monte Carlo simulation results of the A-PSM attack with honest mining and selfish mining simultaneously. In the simulation, we show the profits for the attacker with a computation



(a) $RER_A^{AP,H}$, $\gamma = 0$ (b) $RER_A^{AP,H}$, $\gamma = 1$

(c) $RER_A^{AP,S}$, $\gamma = 0$ (d) $RER_A^{AP,S}$, $\gamma = 1$

Figure 13: Attacker's RER of A-PSM over honest mining and selfish mining. The solid line represents no extra reward.

Table 4: Attacker's RER when choosing *A-PSM* instead of honest mining. Attacker's mining power $\alpha_A = 0.1$.

γ	α_i			
	0	0.1	0.2	0.3
0	−64.31 (−64.32)	−47.88 (−47.88)	−31.37 (−31.36)	−9.09 (−9.09)
0.5	−27.50 (−27.50)	−18.78 (−18.79)	−9.09 (−9.09)	7.28 (7.27)
1	9.32 (9.32)	10.30 (10.30)	13.18 (13.18)	23.64 (23.64)

Table 5: Attacker's relative extra reward of *A-PSM* over selfish mining. Attacker's mining power $\alpha_A = 0.1$.

γ	α_i			
	0	0.1	0.2	0.3
0	−0.01 (0.00)	46.04 (46.07)	92.39 (92.36)	154.80 (154.78)
0.5	−0.04 (0.00)	12.01 (12.02)	25.39 (25.39)	48.02 (47.96)
1	0.01 (0.00)	0.90 (0.90)	3.54 (3.53)	13.09 (13.10)

power of 0.1 over 10^9 rounds. The upper bound for error is 10^{-4}. The comparison results with honest mining are shown in Table 4, and the comparison results with selfish mining are shown in Table 5.

8 Optimal Mining Strategy

In an ideal case with known α_A and γ, we can get the optimal mining strategy for participants. In this section, we compare PSM, A-PSM with selfish mining and honest mining, and analyze the optimal mining strategy for both rational miners and attackers respectively.

8.1 Rational Miners

Rational miners are profit-driven. They will receive the shared partial block information no matter in PSM or A-PSM.

In PSM, rational miner's optimal strategy can be decided by Equation (6). When $RER_k^{G,H}$ value is positive, rational miner should follow the attacker's branch. Otherwise, it should follow the public mining branch. In reality, it is hard to obtain the optimal strategy for rational miners because Equation (6) contains the parameter γ. γ is unpredictable that represents the ratio of public miners choosing attacker's branch when a race occurs. Thus, we derive the optimal strategy in Theorem 4.4 for the worst-case scenario when $\gamma = 0$. Figure 7 illustrates the optimal strategy for rational miners.

In A-PSM, rational miner's optimal strategy is decided by Equation (21). Since the formula is always > 0 in Equation (21), it is more profitable for rational miners to work on the attacker's branch.

8.2 Attacker

Figure 14 shows the optimal mining strategy simulation results for attackers under different conditions. Specifically, Figure 14(a) compares selfish mining to honest mining in different network settings, revealing that a selfish miner with over 1/3 mining power consistently achieves higher profits.

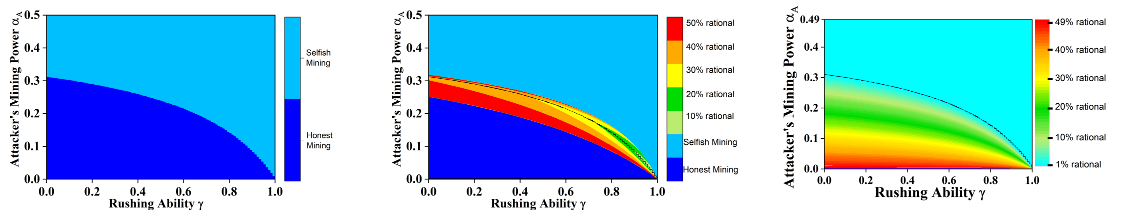
In PSM, an attacker can get its optimal strategy from Equations (15) and (16). When both RER_A values are positive, the attacker's optimal strategy is to launch PSM attacks. Otherwise, it should take either honest mining or selfish mining. The parameters α_A and α_i define the distributions of rational and honest miners in equations. In Figure 14(b), we show in what conditions PSM can outperform both honest and selfish mining. It shows that even if $\alpha_A + \alpha_i < 0.5$, the attacker can still achieve higher profits than both Selfish and Honest mining strategies. For example, when the attacker's mining power $\alpha_A = 0.09$, and $\gamma = 0.9$, the attacker with rational miner controls only 10% of mining power can be the optimal mining strategy. When attracted miners control 50% of mining power, an attacker with 25% of mining power can outperform selfish and honest mining even if $\gamma = 0$.

In A-PSM, an attacker can get its optimal strategy from Equations (22) and (23) similarly. In Figure 14(c), we graphically illustrate the optimal strategy for attackers in A-PSM. As we can see from Figure 14(c), A-PSM can outperform selfish mining even if the attracted miners only control less than 1% of mining power. The fraction of rational miners' mining power needed increases with the decrement of the attacker's mining power. When the attacker's mining power is less than 1%, the amount of rational miner's mining power needed approaches 49%. But the overall mining power of attracted miners and the attacker is always no more than 50%. Whether to take PSM or A-PSM depends on the comparison result between Equation (1)

In reality, α_A, α_i, and γ could change dynamically. The attacker can use network measurement [26, 27] or historical information, e.g., its mining power ratio in the past 1 hour, to infer the current network status and select the attack strategy.

9 Discussion

Multiple Attackers: Except for honest miners, a miner who finds a new block can launch PSM or A-PSM attacks simultaneously. Thus, a PoW-based blockchain system can have multiple attackers. Here, we consider a model with two attackers. Our two-attacker analysis results can be easily extended to the multiple-attacker



(a) Optimal strategy between selfish mining and honest mining.

(b) Optimal strategy among selfish, honest, and PSM with different fractions of rational miners.

(c) Attracted mining power needed for A-PSM attacker to be the optimal strategy.

Figure 14: Optimal strategies for different α_A and γ. We redraw the borderline between selfish and honest mining in (b) and (c).

scenario because the given relative extra reward of attackers still holds. Rational miners can choose to work on one of the attacker's private branches or continue working on the public branch.

First, we consider the case that two attackers release the partial block simultaneously. Assuming there is an attacker A with mining power α_A, and rushing ability γ_A. Attacker B with mining power α_B and rushing ability γ_B. Rational miner k with mining power α_k considers that the rest of the miners are public miners, which means $\alpha_h = 1 - \alpha_A - \alpha_B - \alpha_k$.

For PSM, the relative extra reward of choosing attacker A instead of following the public mining strategy is

$$RER^{A,H}_{k,PSM} = \frac{1 - 2\alpha_k - (1-\gamma)(1 - \alpha_A - \alpha_B - \alpha_k)}{2\alpha_k + (2-\gamma)(1 - \alpha_A - \alpha_B - \alpha_k)} \tag{24}$$

The RER of mining in attacker A over B's branch is always 0. That means the variance of mining power among different attackers has no impact on rational miners. Joining either one of them could ensure its profits.

For A-PSM, the RER of choosing attacker A instead of following the public mining strategy is

$$RER^{A,H}_{k,A-PSM} = \frac{\alpha_h + \frac{1}{\alpha_A+\alpha_i}\left(\frac{(\alpha_A+\alpha_i)^2}{1-2(\alpha_A+\alpha_i)}\right) + 1}{2\alpha_i + (2-\gamma)\alpha_h} - 1. \tag{25}$$

The RER of choosing attacker A instead of attacker B is

$$RER^{A,B}_{k,A-PSM} = \frac{\alpha_A - \alpha_B}{(2\alpha_i + 2\alpha_A - 1)(2\alpha_i + 2\alpha_B - 1)}. \tag{26}$$

It is always more beneficial for rational miners to work with larger mining power attackers.

Then we consider one of the attackers to find the new block first. Assuming attacker A finds the new block at time t_a, the attacker B finds the new block at time t_b, duration $T_d = t_b - t_a > 0$. Since the rational miner trusts both attackers equally, joining the attacker A's private branch means the rational miner can have $\alpha_h R(T_d)\alpha_i + \alpha_i R(T_d)$ more expected revenue during the period T_d. Note that the definition of $R(T)$ is given in Equation (13). Thus, attracted miners will tend to join the attacker A's private branch first to assure higher profits for both PSM and A-PSM strategies. If the rational miners do not find the new block during the period T_d, with the PSM strategy, the attracted miners have no motivation to switch the working branch. For the A-PSM strategy, the attracted miners need to re-evaluate their profits and work with the attacker with higher mining power.

Rationality of Miners: In this paper, we assume that no more than 50% of miners are rational, a strict yet valid assumption.

First, although Nakamoto claimed that block rewards incentivize miners to behave honestly, miners' altruism is not guaranteed [29]. Empirical evidence indicates that miners are profit-driven rather than honest [6]. Indeed, attackers can exploit such profit-driven behavior to undermine blockchains [20]. Hence, altruism alone cannot ensure blockchain security.

Second, security analyses must consider worst-case scenarios rather than optimistic ones. The larger the fraction of rational miners, the higher the potential profit for attackers. Therefore, limiting rational miners to no more than 50% is a reasonable strict assumption.

Finally, as shown in Figure 14(b), Partial Selfish Mining (PSM) attackers can still gain higher rewards than both selfish and honest mining, even when only a small fraction of miners is rational.

Mitigation: A simple way to prevent PSM attacks is to disallow the broadcast of partial block data within the target blockchain network. Since partial blocks can be obtained from public websites, they do not need to be distributed through the network—though blocking all access to them remains challenging.

Public miners can also counter PSM by refusing to mine on the attacker's branch during a race or by carefully analyzing the relevant smart contract. Both strategies reduce the attacker's odds of success, as they cannot distinguish which miners are rational. Some public miners may still retrieve partial block data but choose not to mine on the attacker's branch, lowering γ. As noted in Section 5, the hidden data is termed *secret*. In principle, any miner—especially one with substantial computational power—could compute and publicize the secret once they have a partial block. If the attacker publishes a partial block at time t_0, and a public miner computes the secret att_1, let $T = t_1 - t_0$. The expected profit for attracted miners is $\alpha_h R(T)\alpha_i + \alpha_i R(T)$. As T approaches 0, their extra revenue becomes negligible. Once a public miner discloses the secret, no miners will risk pursuing trivial profits. If $\alpha_i = 0$ in the attacker's A-PSM strategy, the attacker's revenue reverts to selfish mining levels—which are actually higher than PSM profits (see Section 6.2).

Another defense is to hamper partial block dissemination. Public miners can drop any partial block they receive and mark senders as untrusted, discouraging further spread of partial block data.

10 Conclusions

Selfish mining remains a persistent threat to blockchain systems, particularly smaller public chains. While previous attacks typically combine selfish mining with techniques like bribery, BWH, and FAW, they do not fundamentally enhance selfish mining itself. In contrast, this work introduces a new attack paradigm by leveraging partial information sharing, allowing an attacker to collude with rational miners. We first propose Partial Selfish Mining (PSM), which uses partial block sharing to increase rewards for both the attacker and recruited miners. PSM remains practical by offering two mechanisms that guarantee these miners' profits, ensuring they stick to the attacker's private branch. Building on this, Advanced PSM (A-PSM) ensures the attacker's profit is at least on par with selfish mining, making it a lucrative alternative. Although we discuss potential countermeasures to partial selfish mining, an effective real-world solution remains elusive.

Acknowledgments

This work was supported in part by Grants from the HK RGC GRF (PolyU 15209822), NSFC/RGC Joint Research Scheme (N_PolyU529/22 and No. 62261160391), the National Key Research and Development Program of China (2024YFB3311802) and the Shandong Provincial Natural Science Foundation (ZR2024QF055)

References

[1] Lukas Aumayr, Pedro Moreno-Sanchez, Aniket Kate, and Matteo Maffei. 2021. Blitz: Secure Multi-Hop Payments Without Two-Phase Commits. In *30th USENIX Security Symposium (USENIX Security 21)*. USENIX, Online, 4043–4060.

[2] Christopher Bendiksen, Samuel Gibbons, and Eugene Lim. 2018. The Bitcoin mining network-trends, marginal creation cost, electricity consumption & sources. *CoinShares Research* 21 (2018), 3–19.

[3] Danny Bradbury. 2013. The problem with Bitcoin. *Computer Fraud & Security* 2013, 11 (2013), 5–8.

[4] Matteo Campanelli, Rosario Gennaro, Steven Goldfeder, and Luca Nizzardo. 2017. Zero-knowledge contingent payments revisited: Attacks and payments for services. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Dallas, USA, 229–243.

[5] Coinwarz. 2022. Bitcoin Hashrate Chart. https://www.coinwarz.com/mining/Bitcoin/hashrate-chart.

[6] Aryaz Eghbali and Roger Wattenhofer. 2019. 12 Angry Miners. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Springer, Switzerland, 391–398.

[7] Etherscan. 2022. How Many Bitcoins Are There and Will They Ever Run Out? https://capitalcounselor.com/how-many-Bitcoins-are-there/. February 23, 2022.

[8] Ittay Eyal and Emin Gun Sirer. 2014. Majority is not enough: Bitcoin mining is vulnerable. In *International Conference on Financial Cryptography and Data Security*. Springer, Springer, Switzerland, 436–454.

[9] Shang Gao, Zecheng Li, Zhe Peng, and Bin Xiao. 2019. Power adjusting and bribery racing: Novel mining attacks in the Bitcoin system. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. ACM, London, UK, 833–850.

[10] Arthur Gervais, Ghassan O Karame, Karl Wüst, Vasileios Glykantzis, Hubert Ritzdorf, and Srdjan Capkun. 2016. On the security and performance of proof of work blockchains. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Hofburg Palace, Vienna, Austria, 3–16.

[11] Noemi Glaeser, Matteo Maffei, Giulio Malavolta, Pedro Moreno-Sanchez, Erkan Tairi, and Sri Aravinda Krishnan Thyagarajan. 2022. Foundations of coin mixing services. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Los Angeles, USA., 1259–1273.

[12] Ghassan O Karame, Elli Androulaki, and Srdjan Capkun. 2012. Double-spending fast payments in Bitcoin. In *Proceedings of the 2012 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Raleigh, NC, USA, 906–917.

[13] Aggelos Kiayias, Markulf Kohlweiss, and Amirreza Sarencheh. 2022. PEReDi: Privacy-Enhanced, Regulated and Distributed Central Bank Digital Currencies. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Los Angeles, U.S.A., 1739–1752.

[14] Eleftherios Kokoris Kogias, Philipp Jovanovic, Nicolas Gailly, Ismail Khoffi, Linus Gasser, and Bryan Ford. 2016. Enhancing Bitcoin security and performance with strong consistency via collective signing. In *25th USENIX Security Symposium (USENIX Security 16)*. Usenix, Austin, TX, 279–296.

[15] Yujin Kwon, Dohyun Kim, Yunmok Son, Eugene Vasserman, and Yongdae Kim. 2017. Be selfish and avoid dilemmas: Fork after withholding (FAW) attacks on Bitcoin. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, New York,NY,United States, 195–209.

[16] Andrew Lewis-Pye and Tim Roughgarden. 2021. How does blockchain security dictate blockchain implementation?. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Virtual Conference, 1006–1019.

[17] Tao Li, Zhaojie Wang, Guoyu Yang, Yang Cui, Yuling Chen, and Xiaomei Yu. 2021. Semi-selfish mining based on hidden Markov decision process. *International Journal of Intelligent Systems* 36, 7 (2021), 3596–3612.

[18] Kevin Liao and Jonathan Katz. 2017. Incentivizing blockchain forks via whale transactions. In *International Conference on Financial Cryptography and Data Security*. Springer, Springer, Sliema, Malta1, 264–279.

[19] Loi Luu, Ratul Saha, Inian Parameshwaran, Prateek Saxena, and Aquinas Hobor. 2015. On power splitting games in distributed computation: The case of Bitcoin pooled mining. In *2015 IEEE 28th Computer Security Foundations Symposium*. IEEE, IEEE, Verona,Italy, 397–411.

[20] Michael Mirkin, Yan Ji, Jonathan Pang, Ariah Klages-Mundt, Ittay Eyal, and Ari Juels. 2020. BDoS: Blockchain denial-of-service. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Virtual Conference, 601–619.

[21] Satoshi Nakamoto. 2008. Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review* (2008), 21260.

[22] Kartik Nayak, Srijan Kumar, Andrew Miller, and Elaine Shi. 2016. Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. In *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, IEEE, Saarbrücken, Germany, 305–320.

[23] Kevin Alarcón Negy, Peter R Rizun, and Emin Gün Sirer. 2020. Selfish mining re-examined. In *International Conference on Financial Cryptography and Data Security*. Springer, Springer, Kota Kinabalu, Malaysia, 61–78.

[24] Till Neudecker and Hannes Hartenstein. 2019. Short paper: An empirical analysis of blockchain forks in bitcoin. In *Financial Cryptography and Data Security: 23rd International Conference, FC 2019*. Springer, Springer, Frigate Bay, St. Kitts and Nevis, 84–92.

[25] A Pinar Ozisik, George Bissias, and Brian N Levine. 2017. Estimation of miner hash rates and consensus on blockchains. *arXiv preprint arXiv:1707.00082* (2017).

[26] Muhammad Saad, Afsah Anwar, Srivatsan Ravi, and David Mohaisen. 2021. Revisiting Nakamoto Consensus in Asynchronous Networks: A Comprehensive Analysis of Bitcoin Safety and ChainQuality. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Virtual Conference, 988–1005.

[27] Muhammad Saad, Songqing Chen, and David Mohaisen. 2021. SyncAttack: Double-spending in Bitcoin Without Mining Power. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Virtual Conference, 1668–1685.

[28] Ayelet Sapirshtein, Yonatan Sompolinsky, and Aviv Zohar. 2016. Optimal selfish mining strategies in Bitcoin. In *International Conference on Financial Cryptography and Data Security*. Springer, Springer, Christ Church, Barbados, 515–532.

[29] Jakub Sliwinski and Roger Wattenhofer. 2021. Blockchains cannot rely on honesty. In *The 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020)*. ACM, Auckland, New Zealand.

[30] Zhiyuan Sun. 2022. Monero community concerned as leading mining pool nears 51% of ecosystem's total hash rate. https://cointelegraph.com/news/.

[31] Christof Ferreira Torres, Ramiro Camino, et al. 2021. Frontrunner Jones and the Raiders of the Dark Forest: An Empirical Study of Frontrunning on the Ethereum Blockchain. In *30th USENIX Security Symposium (USENIX Security 21)*. Usenix, Virtual Conference, 1343–1359.

[32] Itay Tsabary, Matan Yechieli, Alex Manuskin, and Ittay Eyal. 2021. MAD-HTLC: Because HTLC is crazy-cheap to attack. In *2021 IEEE Symposium on Security and Privacy (S&P)*. IEEE, IEEE, Virtual Conference, 1230–1248.

[33] Xuechao Wang, Viswa Virinchi Muppirala, Lei Yang, Sreeram Kannan, and Pramod Viswanath. 2021. Securing parallel-chain protocols under variable mining power. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Virtual Conference, 1700–1721.

[34] Gavin Wood et al. 2014. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper* 151, 2014 (2014), 1–32.

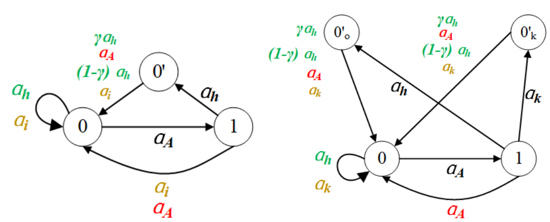
[35] Shuangke Wu, Yanjiao Chen, Minghui Li, Xiangyang Luo, Zhe Liu, and Lan Liu. 2020. Survive and thrive: A stochastic game for DDoS attacks in Bitcoin mining pools. *IEEE/ACM Transactions on Networking* 28, 2 (2020), 874–887.

[36] Yuan Wu, Xu Xu, Liping Qian, Bo Ji, Zhiguo Shi, and Weijia Jia. 2020. Revenue-sharing based computation-resource allocation for mobile blockchain. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops*. IEEE, IEEE, Virtual Conference, 56–61.

[37] Fan Zhang, Deepak Maram, Harjasleen Malvai, Steven Goldfeder, and Ari Juels. 2020. DECO: Liberating web data using decentralized oracles for TLS. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Virtual Conference, 1919–1938.

[38] Mengya Zhang, Xiaokuan Zhang, Yinqian Zhang, and Zhiqiang Lin. 2020. TXSPECTOR: Uncovering attacks in Ethereum from transactions. In *29th USENIX Security Symposium (USENIX Security 20)*. Usenix, Virtual Conference, 2775–2792.

A Proof of Attacker and Greedy Miners' Profits in PSM



(a) State machine of PSM when the miners with overall α_i mining power working as attracted miner.

(b) State machine of PSM when only rational miner k with mining power α_k choose to follow public mining strategy.

Figure 15: State machine of PSM in different conditions.

From the state machine of PSM in Figure 15(a), we can get the following Equations:

$$\alpha_A P_0 = P_1, \tag{27a}$$
$$P_0' = \alpha_h P_1, \tag{27b}$$
$$\alpha_A P_0 = P_0' + (1-\alpha_h)P_1, \tag{27c}$$
$$P_0 + P_0' + P_1 = 1 \tag{27d}$$

From Equation (27a) we can get:

$$P_0 = \frac{1}{\alpha_A} P_1 \tag{28}$$

Plugging the Equation (28) and (27b) into (27d), then we get

$$\begin{aligned} P_1 + \frac{1}{\alpha_A} P_1 + \alpha_h P_1 &= 1 \\ P_1 &= \frac{\alpha_A}{1+\alpha_A+\alpha_A\alpha_h}. \end{aligned} \tag{29}$$

With Equation (29), we can get other probabilities:

$$P_0 = \frac{1}{1+\alpha_A+\alpha_A\alpha_h}, \tag{30a}$$
$$P_0' = \frac{\alpha_A\alpha_h}{1+\alpha_A+\alpha_A\alpha_h}, \tag{30b}$$

Then we can calculate the revenue of the attacker as:

$$\begin{aligned} Rev_{atk}^{PSM} &= 2\alpha_A P_1 + (2\alpha_A + \alpha_i + \gamma\alpha_h)P_0' + \alpha_i P_1 \\ Rev_{atk}^{PSM} &= (2\alpha_A + \alpha_i)\left(\frac{\alpha_A\alpha_h + \alpha_A}{1+\alpha_A+\alpha_A\alpha_h}\right) + \gamma\alpha_h\left(\frac{\alpha_A\alpha_h}{1+\alpha_A+\alpha_A\alpha_h}\right) \\ Rev_{atk}^{PSM} &= \frac{(2\alpha_A+\alpha_i)(\alpha_A\alpha_h+\alpha_A) + \gamma\alpha_A\alpha_h^2}{1+\alpha_A+\alpha_A\alpha_h}. \end{aligned} \tag{31}$$

The overall revenue of all the attracted miners is:

$$Rev_{attracted}^{PSM} = \alpha_i. \tag{32}$$

And the overall revenue of all the honest miners are:

$$\begin{aligned} Rev_{honest}^{PSM} &= (\gamma\alpha_h + 2(1-\gamma)\alpha_h)P_0' + \alpha_h P_0 \\ Rev_{honest}^{PSM} &= \frac{2\alpha_A\alpha_h^2 + \alpha_h - \alpha_A\alpha_h^2\gamma}{1+\alpha_A+\alpha_A\alpha_h}. \end{aligned} \tag{33}$$

Since honest miners or the attacker may work on the blocks that eventually end up outside the blockchain, the overall block generation rate $Rev_{atk}^{PSM} + Rev_{attracted}^{PSM} + Rev_{honest}^{PSM} < 1$. Thus, we can calculate the attacker's expected profit as:

$$\begin{aligned} R_P^A &= \frac{Rev_{atk}^{PSM}}{Rev_{atk}^{PSM} + Rev_{attracted}^{PSM} + Rev_{honest}^{PSM}} \\ &= \frac{\alpha_A\alpha_h^2\gamma + (\alpha_A\alpha_h + \alpha_A)\alpha_i + 2\alpha_A^2\alpha_h + 2\alpha_A^2}{(2\alpha_A\alpha_h + 2\alpha_A + 1)\alpha_i + 2\alpha_A\alpha_h^2 + (2\alpha_A^2+1)\alpha_h + 2\alpha_A^2} \end{aligned} \tag{34}$$

And the attracted miners' overall expected profit is:

$$\begin{aligned} R_P^G &= \frac{Rev_{attracted}^{PSM}}{Rev_{atk}^{PSM} + Rev_{attracted}^{PSM} + Rev_{honest}^{PSM}} \\ &= \frac{(\alpha_A\alpha_h + \alpha_A + 1)\alpha_i}{(2\alpha_A\alpha_h + 2\alpha_A + 1)\alpha_i + 2\alpha_A\alpha_h^2 + (2\alpha_A^2+1)\alpha_h + 2\alpha_A^2} \end{aligned} \tag{35}$$

B Proof of Rational Miner's Profits in PSM

From the state machine in Figure 15(b), we can get the following Equations:

$$\alpha_A P_0 = P_1, \tag{36a}$$
$$P_{0o}' = \alpha_h P_1, \tag{36b}$$
$$P_{0k}' = \alpha_k P_1, \tag{36c}$$
$$P_0 + P_{0o}' + P_{0k}' + P_1 = 1 \tag{36d}$$

From Equation (36a) we can get:

$$P_0 = \frac{1}{\alpha_A} P_1 \tag{37}$$

Plugging the Equation (37) (36b) and (36c) into (36d), then we get

$$\begin{aligned} P_1 + \frac{1}{\alpha_A} P_1 + \alpha_h P_1 + \alpha_k P_1 &= 1 \\ P_1 &= \frac{\alpha_A}{1+\alpha_A(1+\alpha_k+\alpha_h)}. \end{aligned} \tag{38}$$

With Equation 38, we can get other probabilities:

$$P_0 = \frac{1}{1+\alpha_A(1+\alpha_k+\alpha_h)}, \tag{39a}$$
$$P_{0o}' = \frac{\alpha_A\alpha_h}{1+\alpha_A(1+\alpha_k+\alpha_h)}, \tag{39b}$$
$$P_{0k}' = \frac{\alpha_A\alpha_k}{1+\alpha_A(1+\alpha_k+\alpha_h)},. \tag{39c}$$

With $\alpha_h = 1 - \alpha_A - \alpha_k$, we can calculate the revenue of the miner k as:

$$\begin{aligned} Rev_k^{PSM,k} &= (2\alpha_k + (1-\gamma)\alpha_h)P_{0k}' + \alpha_k(P_0 + P_{0o}') \\ Rev_k^{PSM,k} &= \frac{2\alpha_A\alpha_k^2 + (1-\gamma)\alpha_h\alpha_A\alpha_k}{1+\alpha_A(1+\alpha_k+\alpha_h)} + \frac{\alpha_A\alpha_k(\alpha_k+\alpha_h)}{1+\alpha_A(1+\alpha_k+\alpha_h)} \\ Rev_k^{PSM,k} &= \frac{((\alpha_A - \alpha_A^2)\alpha_k - \alpha_A\alpha_k^2)r + (2\alpha_A^2 - 2\alpha_A - 1)\alpha_k}{\alpha_A^2 - 2\alpha_A - 1}. \end{aligned} \tag{40}$$

The overall revenue of all other honest miners is:

$$
\begin{aligned}
Rev_o^{PSM,k} =& \alpha_h(P_0 + P'_{0k}) + (\alpha_h\gamma + 2\alpha_h(1-\gamma))P'_{0o} + \\
& (1-\gamma)\alpha_k P'_{0o} \\
Rev_o^{PSM,k} =& \frac{((\alpha_A^2 - \alpha_A)\alpha_k + \alpha_A^3 - 2\alpha_A^2 + \alpha_A)\gamma}{\alpha_A^2 - 2\alpha_A - 1} + \\
& \frac{((-2\alpha_A^2) + 2\alpha_A + 1)\alpha_k}{\alpha_A^2 - 2\alpha_A - 1} - \frac{2\alpha_A^3 - 4\alpha_A^2 + \alpha_A + 1}{\alpha_A^2 - 2\alpha_A - 1}.
\end{aligned} \tag{41}
$$

And the revenue of the attacker is:

$$
\begin{aligned}
Rev_{atk}^{PSM,k} &= 2\alpha_A(P'_{0o} + P'_{0k} + P_1) + \gamma\alpha_h P'_{0k} + \gamma(\alpha_h + \alpha_k)P'_{0o} \\
Rev_{atk}^{PSM,k} &= \frac{(\alpha_A\alpha_k^2 - \alpha_A^3 + 2\alpha_A^2 - \alpha_A)\gamma + 2\alpha_A^3 - 4\alpha_A^2}{\alpha_A^2 - 2\alpha_A - 1}.
\end{aligned} \tag{42}
$$

And the miner k's overall expected profit is:

$$
\begin{aligned}
R_P^k =& \frac{Rev_k^{PSM,k}}{Rev_{atk}^{PSM,k} + Rev_k + Rev_o} \\
=& \frac{\alpha_A\alpha_h^2\gamma + (\alpha_A\alpha_h + \alpha_A)\alpha_k + 2\alpha_A^2\alpha_h + 2\alpha_A^2}{2\alpha_h^2\gamma + (2\alpha_A\alpha_h + 2\alpha_A + 1)\alpha_k + \alpha_A\alpha_h^2 + 2\alpha_A^2\alpha_h + 2\alpha_A^2} \\
=& \frac{(\alpha_A\alpha_k^2 + (\alpha_A^2 - \alpha_A)\alpha_k)\gamma + ((-2\alpha_A^2) + 2\alpha_A + 1)\alpha_k}{\alpha_A + 1}
\end{aligned} \tag{43}
$$

The relative extra reward of miner k when following attracted mining instead of public mining strategy is:

$$
\begin{aligned}
RER_k^{G,H} &= \frac{R_k^G - R_k^H}{R_k^H} \\
&= \frac{((-\alpha_A\alpha_i) - \alpha_A^2 + \alpha_A)\gamma - \alpha_A\alpha_i + \alpha_A^2}{(\alpha_A\alpha_i + \alpha_A^2 - \alpha_A)\gamma - 2\alpha_A^2 + 2\alpha_A + 1}
\end{aligned} \tag{44}
$$

C Proof of Attacker and attracted miners' Profits in A-PSM

From the state machine in Figure 16(a), we can get the following equations:

$$\alpha_A P_0 = P_1, \tag{45a}$$

$$P'_0 = \alpha_h P_1, \tag{45b}$$

$$(1-\alpha_h)P_n - 1 = \alpha_h P_n, \forall n \geq 2 \tag{45c}$$

$$P'_0 + \sum_{k=0}^{\infty} P_k = 1 \tag{45d}$$

From equation (45b) we can get:

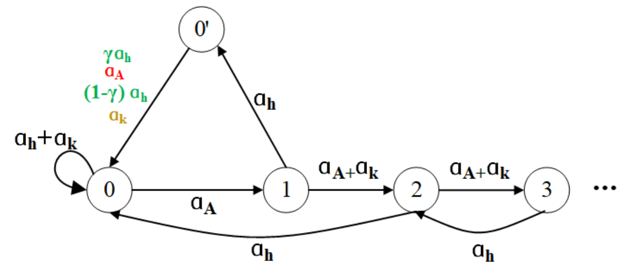
$$P'_0 = \alpha_A\alpha_h P_0 \tag{46}$$

Then, equation (45c) and (45a) give us:

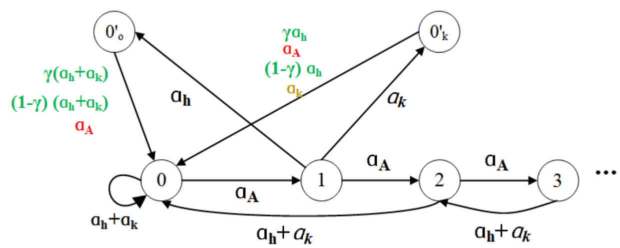
$$P_n = \alpha_A(\frac{1-\alpha_h}{\alpha_h})^{n-1}P_0, \forall n \geq 2 \tag{47}$$

Then we can express equation (45d) as

$$P_0 + \alpha_A(1+\alpha_h)P_0 + \frac{\alpha_A(1-\alpha_h)}{2\alpha_h - 1}P_0 = 1 \tag{48a}$$



(a) State machine of A-PSM when miners with overall α_k mining power working as attracted miner.



(b) State machine of A-PSM when miner k choose to follow public mining strategy.

Figure 16: State machine of PSM in different conditions.

From equation (48a) we can obtain P_0 and other probabilities:

$$P_0 = \frac{2\alpha_h - 1}{2\alpha_A\alpha_h^2 + 2\alpha_h - 1} \tag{49a}$$

$$P'_0 = \frac{\alpha_A\alpha_h(2\alpha_h - 1)}{2\alpha_A\alpha_h^2 + 2\alpha_h - 1} \tag{49b}$$

$$P_1 = \frac{\alpha_A(2\alpha_h - 1)}{2\alpha_A\alpha_h^2 + 2\alpha_h - 1} \tag{49c}$$

$$P_n = (\frac{1-\alpha_h}{\alpha_h})^{n-1}\frac{\alpha_A(2\alpha_h - 1)}{2\alpha_A\alpha_h^2 + 2\alpha_h - 1}, \forall n \geq 2. \tag{49d}$$

Then we can calculate the revenue of the attacker as:

$$
\begin{aligned}
Rev_{atk}^{APSM} =& 2\alpha_A P'_0 + \alpha_i P'_0 + r\alpha_h P'_0 + (1 + \frac{\alpha_A}{\alpha_A + \alpha_i})\alpha_h P_2 \\
& + (\frac{\alpha_A}{\alpha_A + \alpha_i})\alpha_h \sum_{q=3}^{\infty} P_q \\
Rev_{atk}^{APSM} =& \frac{(-\alpha_A(\alpha_i + \alpha_A - 1)^2(2\alpha_i + 2\alpha_A - 1))\gamma}{2\alpha_A\alpha_i^2 + (4\alpha_A^2 - 4\alpha_A - 2)\alpha_i + 2\alpha_A^3 - 4\alpha_A^2 + 1} + \\
& \frac{2\alpha_A\alpha_i^3 + (8\alpha_A^2 - 5\alpha_A)\alpha_i^2 + (10\alpha_A^3 - 14\alpha_A^2 + 2\alpha_A)\alpha_i}{2\alpha_A\alpha_i^2 + (4\alpha_A^2 - 4\alpha_A - 2)\alpha_i + 2\alpha_A^3 - 4\alpha_A^2 + 1} + \\
& \frac{4\alpha_A^4 - 9\alpha_A^3 + 4\alpha_A^2}{2\alpha_A\alpha_i^2 + (4\alpha_A^2 - 4\alpha_A - 2)\alpha_i + 2\alpha_A^3 - 4\alpha_A^2 + 1}.
\end{aligned} \tag{50}
$$

The overall revenue of all the attracted miners is:

$$Rev_{attracted}^{APSM} = \alpha_i. \tag{51}$$

And the overall revenue of all the honest miners are:

$$Rev_{honest}^{APSM} = \alpha_h P_0 + 2\alpha_h(1-\gamma)P_0' + \gamma\alpha_h P_0'$$
$$Rev_{honest}^{APSM} = \frac{\alpha_A(\alpha_i+\alpha_A-1)^2(2\alpha_i+2\alpha_A-1)(\gamma-3)}{2\alpha_A\alpha_i^2+(4\alpha_A^2-4\alpha_A-2)\alpha_i+2\alpha_A^3-4\alpha_A^2+1}. \quad (52)$$

Since honest miners or the attacker may work on the blocks that eventually end up outside the blockchain, the overall block generation rate $Rev_{atk} + Rev_{attracted} + Rev_{honest} < 1$. Thus, we can calculate the attacker's expected profit as:

$$\begin{aligned} R_P^A =& \frac{Rev_{atk}^{APSM}}{Rev_{atk}^{APSM}+Rev_{attracted}^{APSM}+Rev_{honest}^{APSM}} \\ =& \frac{(-\alpha_A(\alpha_i+\alpha_A-1)^2(2\alpha_i+2\alpha_A-1))\gamma}{\alpha_A\alpha_i^2+(2\alpha_A^2-2\alpha_A-2)\alpha_i+\alpha_A^3-2\alpha_A^2-\alpha_A+1}+ \\ & \frac{2\alpha_A\alpha_i^3+(8\alpha_A^2-5\alpha_A)\alpha_i^2+(10\alpha_A^3-14\alpha_A^2+2\alpha_A)\alpha_i}{\alpha_A\alpha_i^2+(2\alpha_A^2-2\alpha_A-2)\alpha_i+\alpha_A^3-2\alpha_A^2-\alpha_A+1}+ \\ & \frac{4\alpha_A^4-9\alpha_A^3+4\alpha_A^2}{\alpha_A\alpha_i^2+(2\alpha_A^2-2\alpha_A-2)\alpha_i+\alpha_A^3-2\alpha_A^2-\alpha_A+1} \end{aligned} \quad (53)$$

And the rational miners' overall expected profit is:

$$\begin{aligned} R_P^G =& \frac{Rev_{attracted}^{APSM}}{Rev_{atk}^{APSM}+Rev_{attracted}^{APSM}+Rev_{honest}^{APSM}} \\ =& \frac{2\alpha_A\alpha_i^3+(4\alpha_A^2-4\alpha_A-2)\alpha_i^2+(2\alpha_A^3-4\alpha_A^2+1)\alpha_i}{\alpha_A\alpha_i^2+(2\alpha_A^2-2\alpha_A-2)\alpha_i+\alpha_A^3-2\alpha_A^2-\alpha_A+1} \end{aligned} \quad (54)$$

D Proof of One Rational Miner's Profits in APSM

From the state machine in Figure 16(b), we can get the following equations:

$$P_1 = \alpha_A P_0, \quad (55a)$$
$$P_{0o}' = \alpha_h P_1, \quad (55b)$$
$$P_{0k}' = \alpha_k P_1, \quad (55c)$$
$$P_q = \frac{\alpha_A}{1-\alpha_A}P_{q-1}, \forall q \geq 2 \quad (55d)$$
$$P_0 + P_{0o}' + P_{0k}' + \sum_{q=1}^{\infty} = 1 \quad (55e)$$

Combining the equation (55a) with (55b) and (55c), then we get

$$\begin{aligned} P_{0k}' &= \alpha_A\alpha_k P_0 \\ P_{0o}' &= \alpha_A\alpha_h P_0. \end{aligned} \quad (56)$$

With equation (55d), we can get:

$$P_n = (\frac{\alpha_A}{1-\alpha_A})^{n-1}\alpha_A P_0 \quad (57)$$

With equation 55e, we can get all states probabilities:

$$P_0 = \frac{1-2\alpha_A}{2\alpha_A^3-4\alpha_A^2+1}, \quad (58a)$$
$$P_{0o}' = \frac{\alpha_A\alpha_h(1-2\alpha_A)}{2\alpha_A^3-4\alpha_A^2+1}, \quad (58b)$$
$$P_{0k}' = \frac{\alpha_A\alpha_k(1-2\alpha_A)}{2\alpha_A^3-4\alpha_A^2+1}, \quad (58c)$$
$$P_n = (\frac{\alpha_A}{1-\alpha_A})^{n-1}\alpha_A\frac{1-2\alpha_A}{2\alpha_A^3-4\alpha_A^2+1} \quad (58d)$$

Then we can calculate the revenue of the miner k as:

$$\begin{aligned} Rev_k^{APSM,k} &= \alpha_k P_0 + 2\alpha_k P_{0k}' + \alpha_k P_{0o}' + (1-\gamma)\alpha_h P_{0k}' \\ Rev_k^{APSM,k} &= \frac{((\alpha_A-2\alpha_A^2)\alpha_k^2+((-2\alpha_A^3)+3\alpha_A^2-\alpha_A)\alpha_k)\gamma}{2\alpha_A^3-4\alpha_A^2+1}+ \\ & \frac{(4\alpha_A^3-6\alpha_A^2+1)\alpha_k}{2\alpha_A^3-4\alpha_A^2+1}. \end{aligned} \quad (59)$$

The revenue of other honest miners are:

$$\begin{aligned} Rev_o^{APSM,k} &= \alpha_h(P_0+P_{0k}') + (\alpha_h\gamma+2\alpha_h(1-\gamma))P_{0o}' + (1-\gamma)\alpha_k P_{0o}' \\ Rev_o^{APSM,k} &= \frac{(2\alpha_A-1)(\alpha_i+\alpha_A-1)(\alpha_A^2\gamma-\alpha_A\gamma-2\alpha_A^2+2\alpha_A+1)}{2\alpha_A^3-4\alpha_A^2+1}. \end{aligned} \quad (60)$$

And the revenue of the attacker is:

$$\begin{aligned} Rev_{atk}^{APSM,k} =& 2\alpha_A(P_{0k}'+P_{0o}') + \gamma(1-\alpha_A)P_{0o}' + \gamma\alpha_h P_{0k}' + \\ & 2(1-\alpha_A)P_2 + (1-\alpha_A)\sum_{q=3}^{\infty}P_q \\ Rev_{atk}^{APSM,k} =& \frac{(\alpha_A(2\alpha_A-1)(\alpha_k-\alpha_A+1)(\alpha_k+\alpha_A-1))\gamma}{2\alpha_A^3-4\alpha_A^2+1}+ \\ & \frac{4\alpha_A^4-9\alpha_A^3+4\alpha_A^2}{2\alpha_A^3-4\alpha_A^2+1}. \end{aligned} \quad (61)$$

And the miner k's overall expected profit is:

$$\begin{aligned} R_P^k =& \frac{Rev_k^{APSM,k}}{Rev_{atk}^{APSM,k}+Rev_k^{APSM,k}+Rev_o^{APSM,k}} \\ =& \frac{(\alpha_A(1-2\alpha_A)\alpha_k(\alpha_k+\alpha_A-1))\gamma+(4\alpha_A^3-6\alpha_A^2+1)\alpha_k}{\alpha_A^3-2\alpha_A^2-\alpha_A+1} \end{aligned} \quad (62)$$

The relative extra reward of miner k when following attracted mining instead of public mining strategy is:

$$\begin{aligned} RER_k^{G,H} &= \frac{R_k^G-R_k^H}{R_k^H} \\ &= \frac{((-\alpha_A\alpha_i)-\alpha_A^2+\alpha_A)\gamma-\alpha_A\alpha_i+\alpha_A^2}{(\alpha_A\alpha_i+\alpha_A^2-\alpha_A)\gamma-2\alpha_A^2+2\alpha_A+1} \end{aligned} \quad (63)$$