Optimal Selfish Mining Strategies in Bitcoin: A Comprehensive Summary

Panos Lin
Computer Science Department
Virginia Commonwealth University
Richmond, VA
ling8@vcu.edu

***Abstract*—This paper presents a comprehensive summary of "Optimal Selfish Mining Strategies in Bitcoin" by Sapirshtein, Sompolinsky, and Zohar. The original work extends the model of selfish mining attacks in Bitcoin and provides an algorithm to compute optimal strategies for attackers. Our summary covers five key contributions: (1) the generalization and optimization of selfish mining strategies beyond previous approaches, (2) the discovery of a lower profit threshold required for profitable attacks, (3) the evaluation of proposed protocol modifications to mitigate such attacks, (4) the impact of network delays on the viability of selfish mining, and (5) the interaction between selfish mining and double spending attacks.**

Together, these findings demonstrate that Bitcoin's security against selfish mining is more tenuous than previously believed, with significant implications for the cryptocurrency's long-term incentive compatibility.

***Index Terms*—Bitcoin, Selfish Mining, Blockchain Security, Optimization, Double Spending**

I. Introduction

A. Background and Motivation

The security of Bitcoin and other proof-of-work cryptocurrencies relies on miners following the protocol honestly. However, in 2014, Eyal and Sirer identified a vulnerability called "selfish mining," where miners can increase their relative rewards by deviating from the protocol [1]. Their strategy, known as SM1, showed that miners controlling more than a certain threshold of computational power could profitably withhold blocks instead of publishing them immediately.

The paper we summarize, "Optimal Selfish Mining Strategies in Bitcoin" by Sapirshtein et al., extends this work by finding optimal selfish mining strategies and examining their implications for Bitcoin's security [2]. This summary is motivated by the need to understand the full extent of selfish mining vulnerabilities, which could potentially undermine Bitcoin's incentive structure and lead to centralization.

B. Objectives

This summary paper aims to provide a comprehensive overview of the key findings presented in the original paper, focusing on:

- The algorithm for finding ε-optimal selfish mining strategies and its theoretical foundations
- The lower profit threshold for selfish mining compared to previous work
- The effectiveness of proposed protocol modifications against optimal selfish mining
- The impact of network delays on the viability of selfish mining attacks
- The relationship between selfish mining and double spending attacks

II. Generalization and Optimization of Selfish Mining

This section examines how Sapirshtein et al. extended the selfish mining attack model and developed an algorithm to find optimal attack strategies. While the original selfish mining strategy (SM1) demonstrated that Bitcoin is not incentive-compatible, the authors show that even more effective strategies exist and provide a framework for finding them.

A. The Basic Selfish Mining Model

The selfish mining strategy, first described by Eyal and Sirer [1], exploits Bitcoin's fork resolution mechanism to increase an attacker's relative revenue. In the standard Bitcoin protocol, miners immediately publish any blocks they discover [3]. In selfish mining, attackers strategically withhold blocks to force honest miners to waste computational power on blocks that will eventually be discarded.

Two key parameters characterize the attack model:

- α: The fraction of the total network hashrate controlled by the attacker (between 0 and 0.5)
- γ: The connectivity parameter representing the fraction of honest miners who would mine on the attacker's blocks in case of a block race (between 0 and 1)

In the SM1 strategy, the attacker follows specific rules based on their private chain's lead over the public chain:

- When the attacker finds a block, they keep it private instead of publishing it
- If the honest miners find a block while the attacker has a one-block lead, the attacker immediately publishes their secret chain, creating a fork
- If the honest miners build on the attacker's block during a fork, the attacker gains revenue from the previously withheld block
- If the attacker extends their private chain to have a two-block lead, they publish their entire chain, causing honest miners to switch to it and abandoning their work

The original analysis showed that SM1 becomes profitable when $\alpha > \frac{1-\gamma}{3-2\gamma}$, which for $\gamma = 0$ means attackers need at least 33% of the network hashrate, and for $\gamma = 1$ need just 25%.

B. Markov Decision Process Formulation

Sapirshtein et al. generalize the selfish mining problem by formulating it as a Markov Decision Process (MDP). This approach allows them to find optimal strategies rather than analyzing a single predefined strategy.

An MDP consists of states, actions, transition probabilities, and rewards:

- **States**: The state space (a, h) captures the length of the attacker's private chain (a) and the length of the honest network's chain (h) relative to the last common block.
- **Actions**: At each state, the attacker can perform three possible actions:
  - **Adopt**: Abandon their private chain and mine on the public chain
  - **Override**: Publish enough blocks to overtake the public chain
  - **Match**: Publish exactly enough blocks to match the public chain length
  - **Wait**: Continue mining on their private chain without publishing blocks
- **Transitions**: Transitions between states occur when either the attacker or honest miners find new blocks. The probability of the attacker finding the next block is α, while honest miners find it with probability $1 - \alpha$.
- **Rewards**: The attacker's reward is measured as the expected fraction of blocks in the main chain that belong to them. The long-term average reward ρ represents the attacker's revenue relative to their resources.

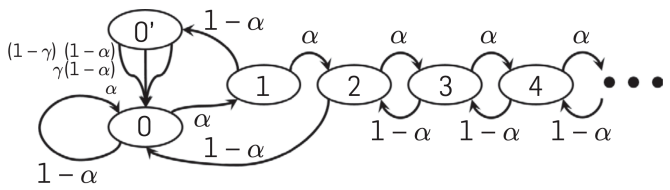

Fig. 1. State machine representation of the SM1

The authors define a policy π as a mapping from states to actions. An optimal policy maximizes the attacker's relative revenue. Unlike SM1, which prescribes fixed actions for each state, as shown in Figure 1, optimal policies can adapt based on precise state configurations.

C. The ε-Optimal Algorithm

Finding an optimal selfish mining strategy equates to finding an optimal policy for the MDP. The authors develop an algorithm that can compute an ε-optimal policy for any desired precision $\varepsilon > 0$.

The authors show that their algorithm can find policies with revenues arbitrarily close to the theoretical upper bound of $\frac{\alpha}{1-\alpha}$ when $\gamma = 1$, which represents the maximum theoretical revenue an attacker can achieve.

D. Verification and Implementation

To validate their approach, the authors implemented both the algorithm and a selfish mining simulator:

- **Simulation Framework**: They created a discrete-event simulation of the Bitcoin network that models block discovery, propagation, and fork resolution.
- **Comparison with SM1**: The simulator allowed direct comparison between the optimal policies and SM1, confirming that optimal policies indeed produce higher revenue in all parameter settings.
- **Error Analysis**: The authors analyzed the error bounds of their algorithm and showed that the practical error is typically much smaller than the theoretical bound, especially for reasonable values of α and γ.

The simulation results confirmed that the algorithm successfully finds policies that outperform SM1, particularly for attackers with lower hashrate or better network connectivity. This verification process established the practical viability of the optimization approach and demonstrated that the gap between optimal strategies and SM1 is significant enough to lower the profit threshold for selfish mining attacks.

III. Lower Profit Threshold for Selfish Mining

One of the most significant findings of Sapirshtein et al. is that optimal selfish mining strategies can be profitable at lower thresholds of computational power than previously established. This section examines the comparison between optimal strategies and SM1

A. Comparison with SM1

The authors' algorithm generates selfish mining strategies that consistently outperform SM1 across all parameter configurations. As shown in Table I, the key findings from their comparison include:

TABLE I
THE REVENUE OF THE ATTACKER UNDER SM1 AND UNDER THE ε-OPT POLICIES, COMPARED TO THE COMPUTED UPPER BOUND, FOR VARIOUS α AND WITH $\gamma = 0$.

α	$SM1$	ε-OPT	Upper-Bound
$1/3$	$1/3$	0.33705	0.33707
0.35	0.36650	0.37077	0.37079
0.375	0.42118	0.42600	0.42604
0.4	0.48372	0.48866	0.48904
0.425	0.55801	0.56808	0.57226
0.45	0.65177	0.66891	0.70109
0.475	0.78254	0.80172	0.90476

- For all values of α and γ, the optimal strategy either matches or exceeds the revenue of SM1.
- The improvement over SM1 is most pronounced for miners with computational power near the original profitability threshold.
- The gap between SM1 and optimal strategies narrows as α increases, indicating that SM1 approaches optimality for attackers with significant computational resources.
- For smaller attackers (those with α close to the profit threshold), optimal strategies involve more nuanced decisions on when to adopt the honest chain versus when to continue selfish mining.

The performance difference arises primarily from the optimal strategy's more sophisticated approach to managing the private chain under specific state configurations. Where SM1 follows fixed rules, optimal strategies dynamically adjust based on the precise state of both chains.

IV. EVALUATION OF PROTOCOL MODIFICATIONS

Sapirshtein et al. evaluated countermeasures against selfish mining, focusing on Eyal and Sirer's uniform tie-breaking rule.

A. The Uniform Tie-Breaking Rule

When miners encounter competing chains of equal length, Eyal and Sirer proposed they should select one uniformly at random rather than choosing the first-seen chain. This aims to neutralize the attacker's network advantage by setting effective $\gamma = 0.5$ during block races.

B. Effectiveness Against Optimal Strategies

The authors found this modification:

- The modification is somewhat effective against strong attackers (high α), reducing their maximum potential revenue
- For $\gamma = 1$, the profit threshold increases from approximately 23.2% to 25%
- The uniform selection rule effectively forces $\gamma = 0.5$ for block races, reducing the impact of network advantages

However, contrary to Eyal and Sirer's conjecture that this modification would increase the profit threshold to 25% for all γ values, the authors found that attackers with less than 25% of computational power can still profit from selfish mining under certain conditions.

C. Unexpected Benefits for Medium-Sized Attackers

Counter-intuitively, the modification benefits miners with poor network connectivity:

- Attackers with 25-32% hashrate and low γ gain higher revenue
- The random choice rule effectively improves their network position
- These miners achieve better results under the modified protocol than the original

This counterintuitive result demonstrates that protocol modifications may have unexpected consequences when attackers employ optimal strategies rather than fixed approaches like SM1.

V. IMPACT OF NETWORK DELAYS

Sapirshtein et al. extended their analysis to include network propagation delays, revealing far more concerning implications for Bitcoin's security.

A. Network Delay Model

The authors developed a more realistic model of the Bitcoin network by incorporating actual block propagation dynamics. Unlike the original model which assumed instantaneous block propagation, this approach accounts for variable transmission times between nodes throughout the network. The model captures how an attacker's blocks might reach some miners before honest blocks do, creating natural forks even without deliberate withholding. By parameterizing different connectivity scenarios, the authors could analyze how selfish mining performs under various network conditions that more closely reflect real-world deployments.

B. Vanishing Profit Threshold

The most striking result from this analysis is that when realistic network delays are considered, the profit threshold for selfish mining completely vanishes. This means that miners with arbitrarily small computational power can potentially profit from deviating from the protocol – a dramatic departure from previous findings that established minimum threshold requirements. The advantage becomes more pronounced for miners with better network positions (higher effective γ values), but exists even for poorly connected attackers. This fundamentally challenges the notion that Bitcoin remains

secure as long as no miner controls a significant fraction of the network's computational power.

VI. Interaction with Double Spending

The final significant contribution of Sapirshtein et al. is their analysis of how selfish mining interacts with double spending attacks, revealing a concerning synergy between these two threat vectors.

Bitcoin's security model assumes double spending attacks (reversing confirmed transactions) are costly. Nakamoto's analysis suggested attackers must invest significant computational resources with diminishing success probability as confirmations increase, which is why users wait for multiple confirmations for high-value transactions.

The authors demonstrate, however, that miners who can profitably engage in selfish mining can perform double spending at no additional cost. A selfish miner already maintaining a private chain can include double-spend transactions in this chain, and if their chain wins, they both earn the higher selfish mining revenue and successfully reverse transactions on the public chain.

This finding undermines Nakamoto's security analysis by invalidating the assumption that attackers bear opportunity costs when attempting transaction reversals. The economic disincentives against double spending are significantly weakened when combined with selfish mining strategies. Consequently, any miner who can profitably selfish mine (including small miners under network delay conditions) can also execute double spending attacks without additional cost, effectively lowering the security threshold for Bitcoin transactions.

VII. Conclusion

The original paper makes significant contributions to understanding Bitcoin's security against selfish mining attacks. By finding optimal selfish mining strategies, the authors demonstrated that the profit threshold is lower than previously thought, protocol modifications may have unexpected effects, network delays eliminate the threshold entirely, and selfish mining enables cost-free double spending.

These findings highlight the need for continued research into Bitcoin's incentive compatibility and security model. Future work could focus on developing more robust protocol modifications, exploring the practical feasibility of these attacks in the wild, and addressing the fundamental tension between network efficiency and security in decentralized consensus systems.

References

[1] Ittay Eyal and Emin Gün Sirer. Majority is not enough: Bitcoin mining is vulnerable. *Communications of the ACM*, 61(7):95–102, 2018.

[2] Ayelet Sapirshtein, Yonatan Sompolinsky, and Aviv Zohar. Optimal selfish mining strategies in bitcoin. In *International conference on financial cryptography and data security*, pages 515–532. Springer, 2016.

[3] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008.