IMPROVING AND EVALUATING POW-BASED CONSENSUS ATTACKS WITH BRIBERY

COMPUTER NETWORKING COURSE PROJECT REPORT

SHANGHAI JIAO TONG UNIVERSITY, IEEE PIONEER CLASS, GROUP 8

Hongbin Chen
Student ID: 516030910544
k160438@sjtu.edu.cn

Xinyu Wang
Student ID: 516030910557
wang_x_y@sjtu.edu.cn

June 17, 2019

ABSTRACT

PoW-based consensus protocols are vulnerable to consensus attacks such as bribery attack, selfish mining. We propose an improvement strategy which can be applied to many traditional attacks including selfish mining, stubborn mining and their variants. Evaluation shows performance improvement in realistic scenarios. Our strategy not only poses a greater risk for the blockchain community, but also provides a possible solution to the difficulty in applying realistic bribery attacks.

Keywords Blockchain · Consensus Attack · Proof-of-Work

1 Introduction

Blockchain has become a hot topic in recent years. People are amazed about this decentralized system which makes it possible to transact with others without central authorities. Blockchain is a combination of crytography, computer network, network security, etc. [1] Blockchain has been developed into monetary-related applications such as Bitcoin, and becomes an attack surface threatened by cyber-criminals.

While the proof-of-work based consensus protocols serve as a key component which ensures the stability of cryptocurrencies, Many recent studies attempt to improve this protocol by proposing PoW-based consensus attacks such as bribery attack [2], selfish mining [3], and stubborn mining [1].

We propose a general improvement strategy which can be applied to many PoW-based consensus attacks, including selfish mining, stubborn mining, and many of their variants. The improvement strategy is inspired by the insight of the bribery attack, which we conclude as aa temporal increase in mining power by bribery. To evaluate the improvement strategy, we design two experiments to improve selfish mining, and stubborn mining, respectively. Results show that our improvement achieve the highest expected revenue ratio in a large range of parameter space comparing to traditional strategies. We also perform a comprehensive evaluation of the improvement strategy under different settings of parameters.

Our improvement not only poses a greater risk for the blockchain community, but also provides a solution to the difficulty in applying the bribery attack in realistic scenarios. We show that with appropriate bribery cost, our improvement still outperforms the traditional strategies in a large parameter space.

This paper is organized in the following order: section 2 introduces our improvement strategy. Section 3 designs two experiments for evaluation of the improvement strategy. Section 4 displays experimental results and provides simple observations. Section 5 further discusses experiments and results in the previous sections. Section 6 summarizes the paper.

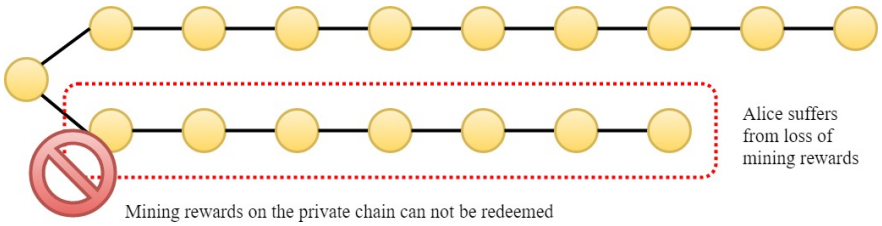

Figure 1: A stubborn miner Alice suffers from large loss of mining rewards caused by losing to the main branch

2 Improvement Strategy

In this section we will introduce the improvement strategy. We start by discussing the problems in traditional PoW-based consensus attacks that we want to solve. We focus on the large computational power loss when long private chain is abandoned. We will analyze its cause in the lead stubborn mining strategy [1], and propose our improvement strategy which can be applied to different genres of PoW-based consensus attacks, including selfish mining [3], stubborn mining [1], and many of their variations.

2.1 Cause of Revenue Loss

Traditional PoW-based consensus attacks potentially risk large revenue loss when long private branch is abandoned. As displayed in figure 1, in a long-lasting mining competition, Alice may suffer from large loss of computational power from losing to the main branch.

We try to provide an insight into this risk by analyzing its cause in the lead stubborn mining strategy [1]. In a lead stubborn mining, Alice can have an extreme long private branch parallel to the main branch, because a stubborn miner, unlike a selfish miner, prefers witholding unpublished blocks in a competition to gaining revenue by publishing new blocks. This preference, as shown by our analysis in figure 2 outperforms selfish mining in a large range of parameter settings. Because of the essense of probabilistics and the minority of Alice's mining power, it is very likely that a long private branch is finally overtaken by the main branch. Whenever Alice finds she is behind the main branch by more than one block, she abandones the private chain and thus gives up all the revenue that has been accumulated on the private chain.

2.2 Bribery as a Complement

Bribery attack, first proposed by Bonneau [2], launches a double-spending by paying rational miners and gaining extra computational power. With the extra mining power, adversary is probable to invert a confirmed transaction by generating a fork and using all the mining power to overtake the main branch. Some studies intended to improve the original bribery attack, such as the whale attack proposed by Liao et al. [4], which increases the anonymity of the adversary.

However, bribery attack and its variants can not threaten the blockchain community in reality because even if the adversary obtains mining power from rational miners by bribery, the total power controlled by the adversary is still a minority. Therefore, the adversary and rational miners who are willing to take the bribery suffer from the risk of wasting computational power if the main branch wins in the long run. Considering the potential risk, rational miners incline not to take the bribery at the first place, which makes the bribery attack impossible to launch in realistic scenarios.

Although bribery attack, as an approach to a double-spending, does not impose realistic risks, its insight, which we conclude as a temporal increase in mining power by bribery, can be a solution to the revenue loss in stubborn mining. When the main branch takes the lead, Alice can save a long private branch by temporarily renting large mining power to win the following rounds of the mining competition over honest miners. The intuition behind our improvement is that when the private branch is long, it is profittable for Alice to save rewards on a long branch with the cost of bribery for only a few blocks.

2.3 Threat Model

The design of our strategy follows the rationality assumption rather than honest majority assumption. Although the latter one was adopted by Garay et al. to prove of the backbone protocol [5], recent studies [6, 7] pay more attention to the rationality assumption because risks under this assumption is considered more realistic.

Based disussion in section 2.1 and 2.2, we formulate the threat model as follows.

Rationality Assumption Honest miners intend to take actions other than the honest mining strategy that will maximize their expected revenue and minimize the variance (risk), regardless of any potential harm to the health of blockchain community. Therefore, honest miners may take a bribery from an adversary which provides greater expected revenue or less variance than the honest strategy.

Adversary The adversary, Alice, controls a minority of mining power, which takes proportion α among all mining power. Honest miners control the rest $(1-\alpha)$ mining power. Alice is the only miner who launches a PoW-based consensus attack. Other rational miners can take a bribery from Alice and rent their mining power to Alice. After paying bribery b to rational miners, Alice gains axtra β mining power, and controls $(\alpha+\beta)$ mining power in total. We define parameter $bribery\_ratio$ as β/α, so Alice's total mining power with bribery can be expressed as $\alpha\cdot(1+bribery\_ratio)$.

Parameters in PoW-based Consensus Attacks In a consensus attack such as selfish mining or stubborn mining, a parameter γ represents the fraction of honest miners who mine on Alice's branch when the private and public branches have equal length. Thus, in a equal-length competition, the mining power on the private chain is $\alpha+\gamma\cdot(1-\alpha)$, and on public chain is $(1-\gamma)\cdot(1-\alpha)$.

Table 1 summarizes the notations.

Notation	Explanation
Alice	the adversary
α	fraction of mining power controlled by Alice
β	fraction of mining power rented by Alice from honest miners
$bribery\_ratio$	$bribery\_ratio = \beta/\alpha$
γ	fraction of honest miners who mine on Alice's branch in an equal fork
b	the amount of bribery paid to rational miners

Table 1: Notations

3 Experimental Setup

In this section we introduce our experimental setup in the evaluation of improved attacks. We apply our improvement to two classical PoW-based consensus attacks, namely, selfish mining [3] and lead stubborn mining [1].

Our improvement increases Alice's tolerence of disadvantage in a mining competition. We notice that a bribery can be expensive, so Alice should only pay bribery for a relatively long private branch. We call the length of private chain to start the bribery as the "start length", denoted by a parameter $start\_length$. Also, Alice should not keep chasing the main branch if the disadvantage is very large. We call the number of blocks by which the private branch falls behind the main branch as the "end length", denoted by a parameter end_length.

We assume the bribery is paid in each mining round. If Alice rents the mining power and a new block is mined, no matter who mines it, the rented mining power costs Alice $penalty\times\beta$. Therefore, in a simulation, the total cost of bribery is

$$b = penalty\times\beta\times\text{mining rounds with bribery} \tag{1}$$

We evaluate the improved attacks by simulations over all possible parameter space, and displays the best strategy comparing to traditional strategies such as the honest mining strategy, and selfish mining strategy. Figure 2 is an example of a parameter space evaluation for honest mining, selfish mining, and stubborn mining.

We design two experiments to provide a comprehensive evaluation over parameter settings including α, β, γ, $start\_length$, end_length, and $penalty$.

3.1 Improving Selfish Mining with Bribery

We first apply the improvement strategy on selfish mining. In selfish mining, selfish miner abandon current private chain if the main branch exceeds the private chain considering the difficulty in chasing the main branch. However, we argue that private chain can exceed main branch again by renting hash power from other miners.

We extend the state space of selfish mining, as shown in Figure 3. The state $0''$ and $-x$ are new states compared to selfish mining. α in Figure 3 means hash power that selfish miner has and β means the hash power that selfish miner

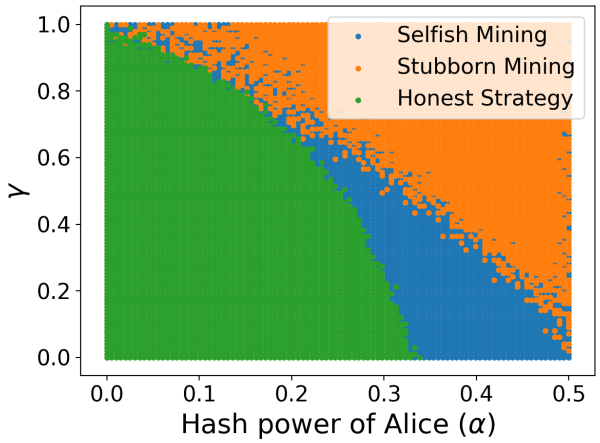

Figure 2: An example of the parameter space evaluation for traditional mining strategies

rents from others. State $0''$ means private chain is as long as main branch and selfish miner still rents hash power expecting to exceed main branch. State $-x$ means private chain lag behind main branch x blocks. We don't want to keep chasing the main branch if the disadvantage is very large, so we set the maximum tolerance is N, which is equal to parameter end_length. If the number of blocks lagging behind is larger than N, selfish miner has to abandon the whole private chian.

In our simulation experiment, the iteration of each simulation is set as 200,000 and we repeat every simulation for 5 times. We equally sample four different settings of $start\_length$, end_length and $bribery\_ratio$, as shown in table 2. Since there are three parameters to evaluate, we perform two experiments. The first experiment evaluates the parameter space under different settings of end_length and $bribery\_length$, and the second one evaluates the parameter space under different settings of $start\_length$ and $bribery\_length$.

Parameter	Values
$start\_len$	{1, 5, 10, 15}
end_len	{10, 15, 20, 25}
$bribery\_ratio$	{0.00, 0.17, 0.33, 0.50}

Table 2: Experimental settings of parameters

3.2 Improving Stubborn Mining with Bribery

We also apply our method on stubborn mining. Classic stubborn mining extends three strategies based on selfish mining, in our experiment, we only consider about lead-stubborn mining because lead-stubborn mining can cause a long private chain abandoned if private chain lags behind main branch.

As same as improved selfish mining, we extend the state space of lead-stubborn mining, as shown in Figure 4. The state $0''$ and $-x$ are new states compared to lead-stubborn mining. α in Figure 3 means hash power that stubborn miner has and β means the hash power that stubborn miner rents from others. State $0''$ means private chain is as long as main branch and stubborn miner still rents hash power expecting to exceed main branch. State $-x$ means private chain lag behind main branch x blocks. We don't want to keep chasing the main branch if the disadvantage is very large, so we set the maximum tolerance is N, which is equal to parameter end_length. If the number of blocks lagging behind is larger than N, stubborn miner has to abandon the whole private chian.

In our simulation experiment, the iteration of each simulation is set as 200,000 and we repeat every simulation for 5 times. We equally sample four different settings of $start\_length$ and $bribery\_ratio$, the same as settings in table 2.

We perform two experiments. The first one chooses $penalty$ as 0.00, and the second one as 1.25. We intend to evaluate the realistic risk of our improvement by comparing results of an appropriate $penalty$ (1.25) to that of an ideal setting (0.00).

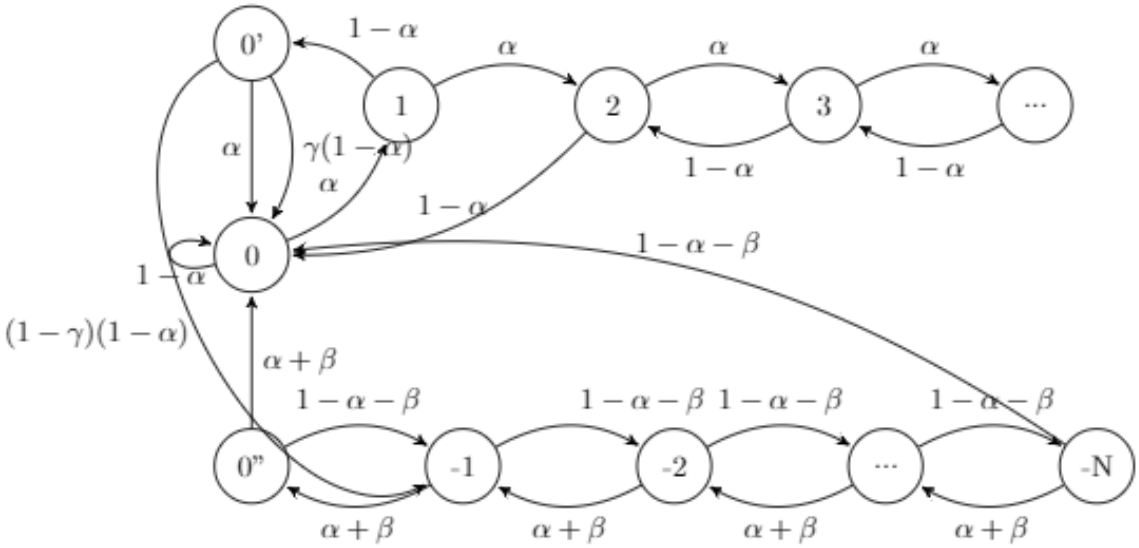

Figure 3: Improved selfish mining with bribery

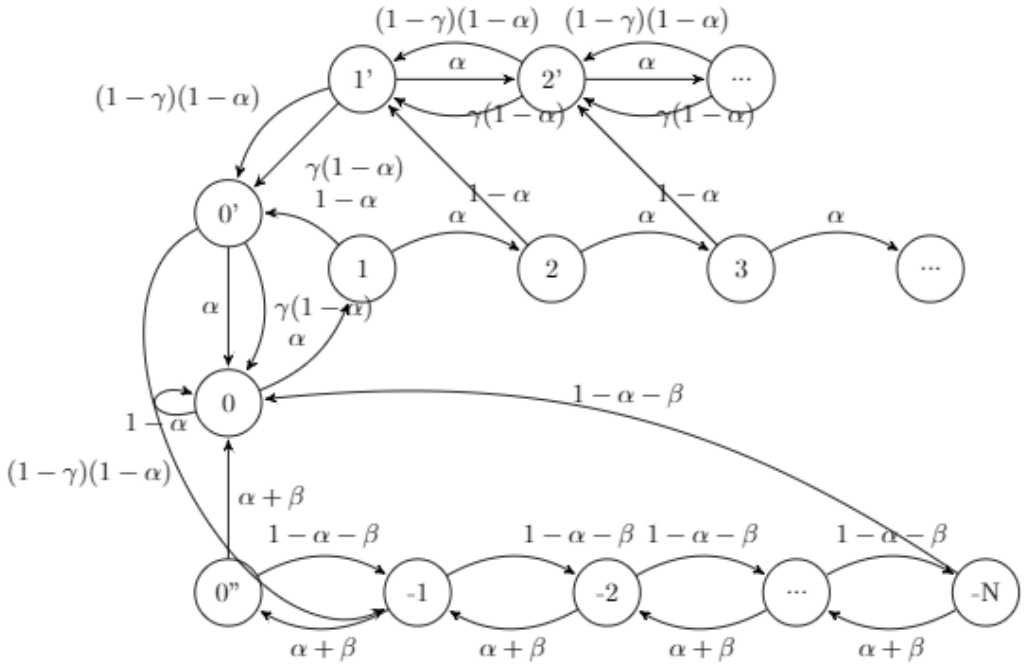

Figure 4: Improved stubborn mining with bribery

4 Evaluation

In this section we visualize the results of our evaluation and draw observations from them. Observations will be further discussed in section 5

Figure 5 shows the optimal strategies on the parameter space under different settings of $bribery\_ratio$ and $start\_length$. We can observe that as parameter end_length increases, selfish-mining-with-bribery strategy shows a slight decrease in performance.

Figure 6 shows the optimal strategies on the parameter space under different settings of $bribery\_ratio$ and $start\_length$. We can observe that when $start\_length$ is larger than 1, selfish-mining-with-bribery strategy can not be differentiated from the selfish mining. In another word, the improved strategy deteriorates to the selfish mining. A simple explanation is that an equal fork with length more than 1 is impossible in selfish mining and our improved strategy. If $start\_length$ is set greater than 1, the state where bribery takes effects will never occur.

Figure 7 shows the optimal strategies on the parameter space under different settings of $bribery\_ratio$ and $start\_length$, regardless of the bribery payment. We draw two observations as follows.

- As $bribery\_ratio$ increases, the parameter space where stubborn-mining-with-bribery strategy is the optimal strategy becomes larger.
- As $start\_length$ increases, the improved strategy experiences an increase in performance followed by a decrease. Therefore, parameter $start\_length$ has an optimal selection which can achieve the best performance.

Figure 8 shows the optimal strategies on the parameter space under different settings of $bribery\_ratio$ and $start\_length$, considering the bribery cost as $1.25 \times \beta \times$ number of bribery rounds. By comparing this result with figure 7, we observe that a decent bribery cost causes a slight decrease in performance of the improved strategy, but still remains optimality in a large range of parameter space. Therefore, we argue that the stubborn-mining-with-bribery strategy poses a realistic risk to the health of the blockchain community.

5 Discussion

5.1 Rationality of Our strategy

The discussion of our strategy's rationality can be divided into two parts.

The first is that whether backward private chain can chase and exceed main branch again with the help of extra hash power from bribery. In this part, we don't consider about the bribery expense and just focus on the original revenue. We simulate on imporved selfish mining and improved stubborn mining, results are shown in Figure 5 and Figure 7. We find that bribery indeed helps backward private chain chase and exceed main branch under some parameters settings. As $bribery\_ratio$ increases, the parameter space where improved attack strategy is the optimal strategy becomes larger.

The other part is that whether one can gain more with bribery considering the bribery expense. In this case, we have to take length of private chain into consideration, that is whether the current private chain is worth saving. Saving a short private chain may cost more bribery expense. In this way, our strategy has no practical use on selfish mining because the length of equal fork in selfish mining won't exceed 1 block, as shown in Figure 6, the $start\_length$ has no effect when it is larger than 1. Further simulations show that our strategy worses the revenue of selfish mining considering bribery expense. Because of long equal forks exist in lead-stubborn minig, our strategy has practical use on stubborn mining. As shown in Figure 8, we see that with high α, our strategy tends to be optimal, however, larger $start\_length$ will destroy this advantage. The reason is that as $start\_length$ increases, the number of cases where the length of equal fork is longer than $start\_length$ decreases sharply and the effect of bribery will be weaker. Finally, stubborn mining with bribery tends to be same as stubborn mining.

5.2 Cost of Bribery Expense

In simulation experiments, we set parameter $penalty\_ratio(> 1)$ to measures the penalty of bribery behavior, so the bribery cost is $penalty\_ratio \times \beta \times$ number of bribery rounds. When we calculate the revenue, we subtract the attacker's blocks by bribery cost. However, from our experiments, we find $penalty\_ratio$ hardly effects the parameter space where stubborn mining with bribery is optimal, but it may effects the final revenue.

In experiments, we consider the value of each block is constant independent from time. However, as some blocks are invalid finally, the value of each block is relatively higher than without equal fork and our calculation of bribery cost is actually larger than real cost.

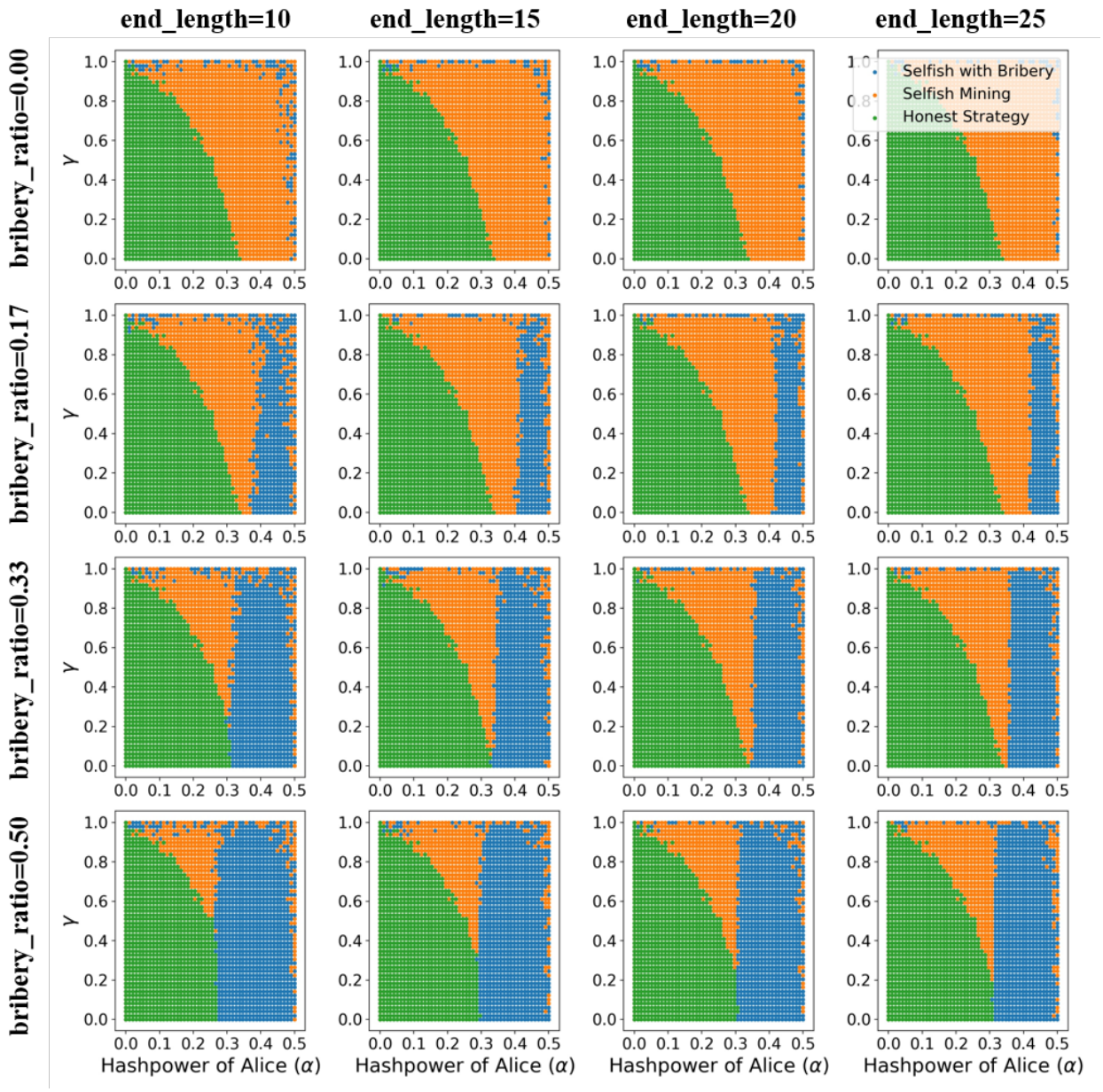

Figure 5: Optimal strategies in the experiment of selfish mining improvement with different settings of $bribery\_ratio$ and end_length

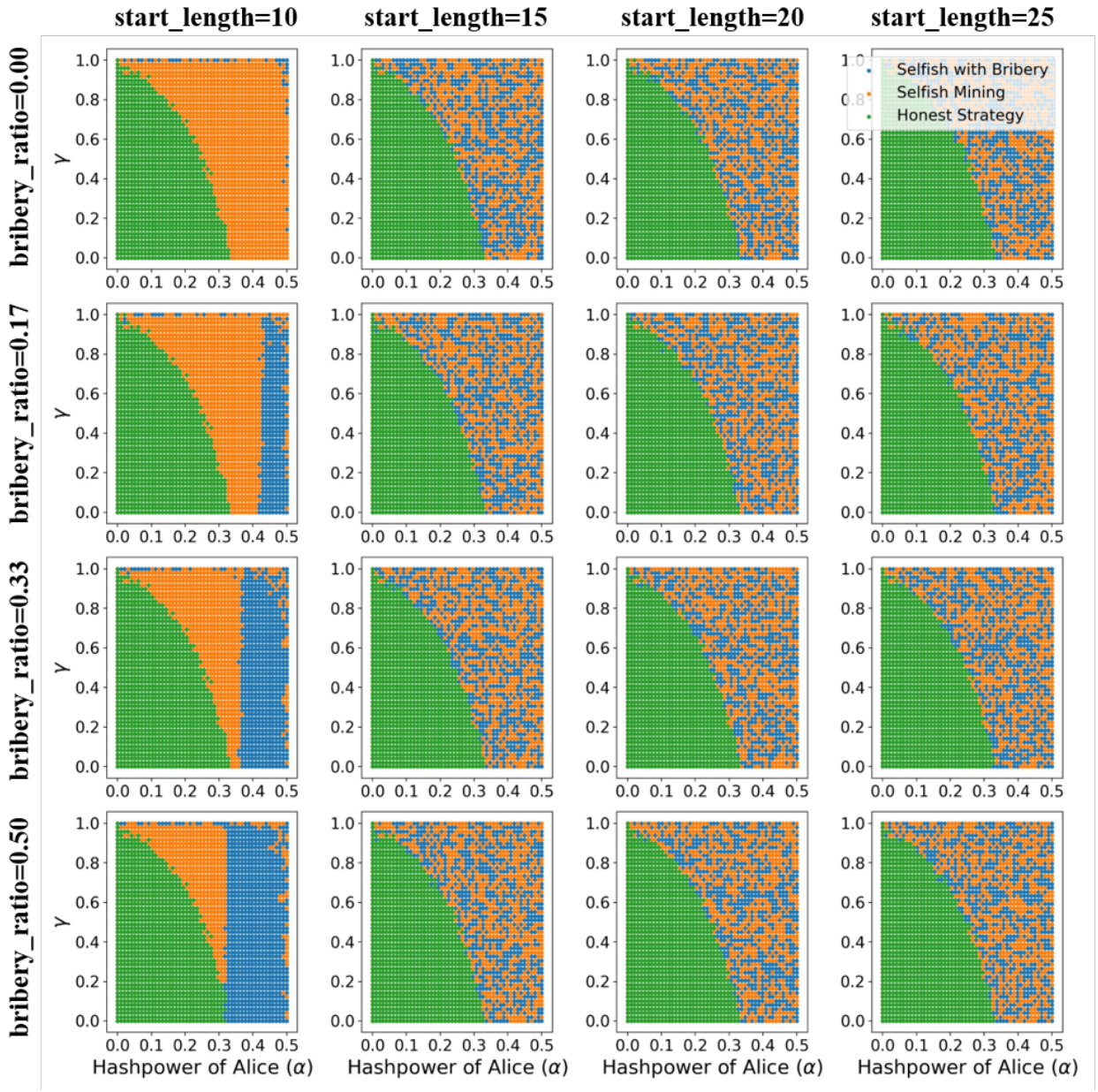

Figure 6: Optimal strategies in the experiment of selfish mining improvement with different settings of $bribery\_ratio$ and $start\_length$

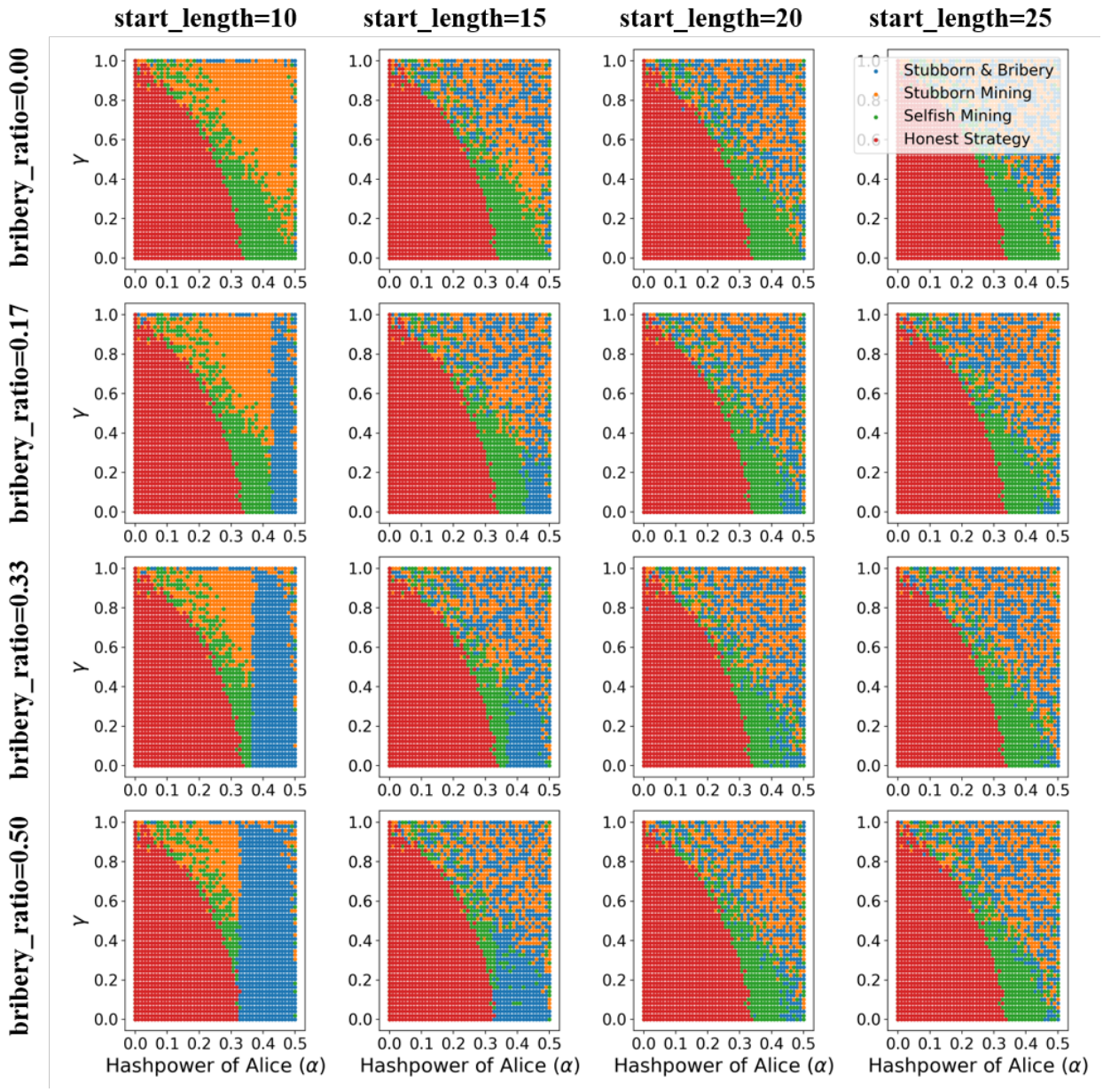

Figure 7: Optimal strategies in the experiment of lead stubborn mining improvement with $penalty = 0.00$ and different settings of $bribery\_ratio$ and $start\_length$

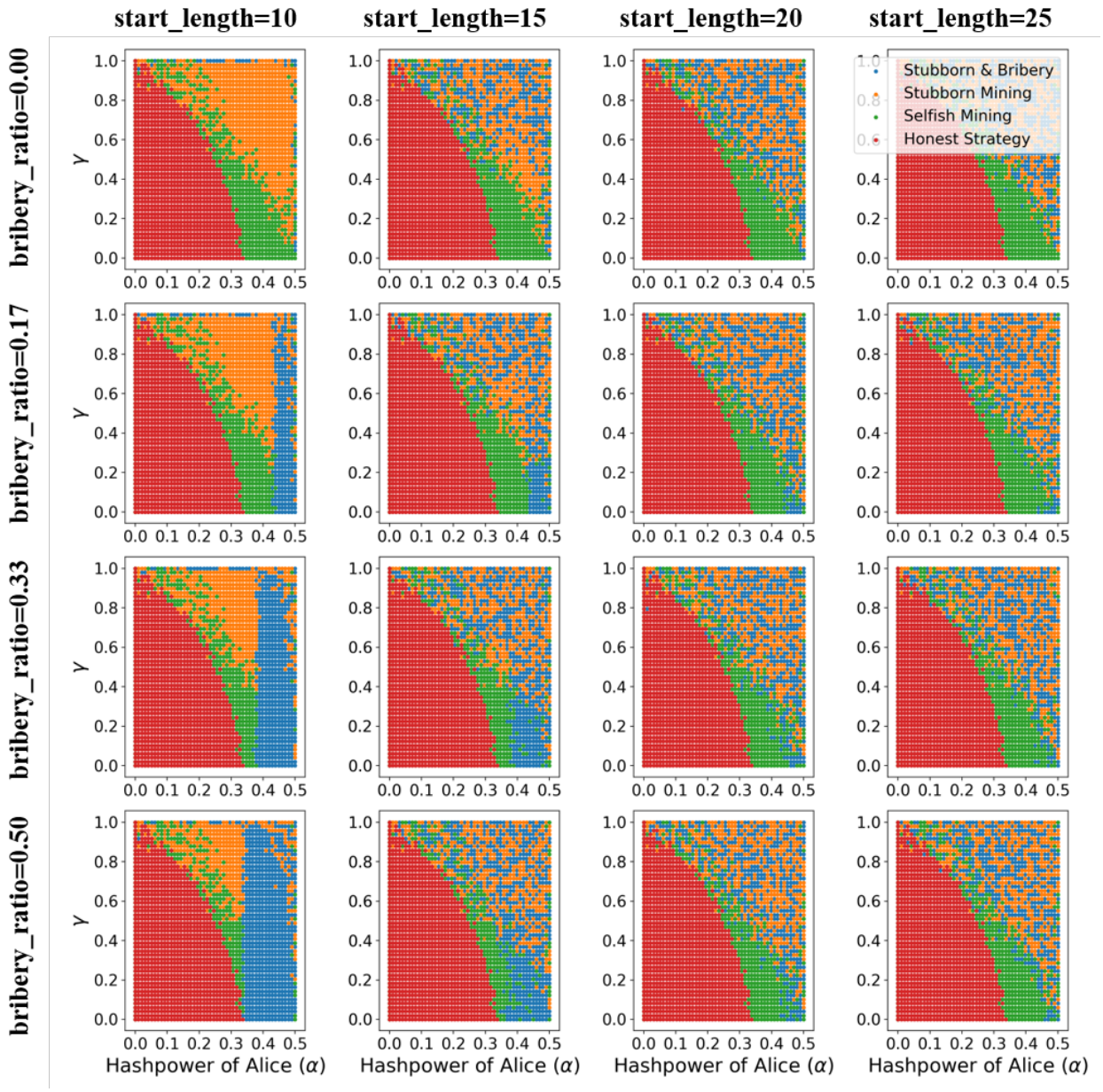

Figure 8: Optimal strategies in the experiment of lead stubborn mining improvement with $penalty = 1.25$ and different settings of $bribery\_ratio$ and $start\_length$

5.3 Selection of Bribery Ratio

In our simulation experiments, we always choose $bribery\_ratio$ as a constant, however, it is obvious that the selection of $bribery\_ratio$ shouldn't be such simple, we believe further study on selection of $bribery\_ratio$ can strengthen our strategy.

6 Conclusions

In this project, we propose an improvement strategy for PoW-based consensus attacks with bribery. To evaluate the improvement strategy, we design two experiments which apply our strategy on selfish mining and stubborn mining by extending the state space. Simulation results show performance improvement in a large range of parameter space. Based on experimental results, we discuss the rationalty of the improvement strategy and potential directions of our future work.

References

[1] Kartik Nayak, Srijan Kumar, Andrew K. Miller, and Elaine Shi. Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 305–320, 2015.

[2] Joseph Bonneau. Why buy when you can rent? - bribery attacks on bitcoin-style consensus. In *Financial Cryptography Workshops*, 2016.

[3] Ittay Eyal and Emin Gün Sirer. Majority is not enough: bitcoin mining is vulnerable. In *Commun. ACM*, 2014.

[4] Kevin Liao and Jonathan Katz. Incentivizing blockchain forks via whale transactions. In *Financial Cryptography Workshops*, 2017.

[5] Juan A. Garay, Aggelos Kiayias, and Nikos Leonardos. The bitcoin backbone protocol: Analysis and applications. *IACR Cryptology ePrint Archive*, 2014:765, 2014.

[6] Ittay Eyal. The miner's dilemma. *2015 IEEE Symposium on Security and Privacy*, pages 89–103, 2015.

[7] Patrick McCorry, Alexander Hicks, and Sarah Meiklejohn. Smart contracts for bribing miners. In *Financial Cryptography Workshops*, 2018.