Competitive Selfish Bitcoin Mining

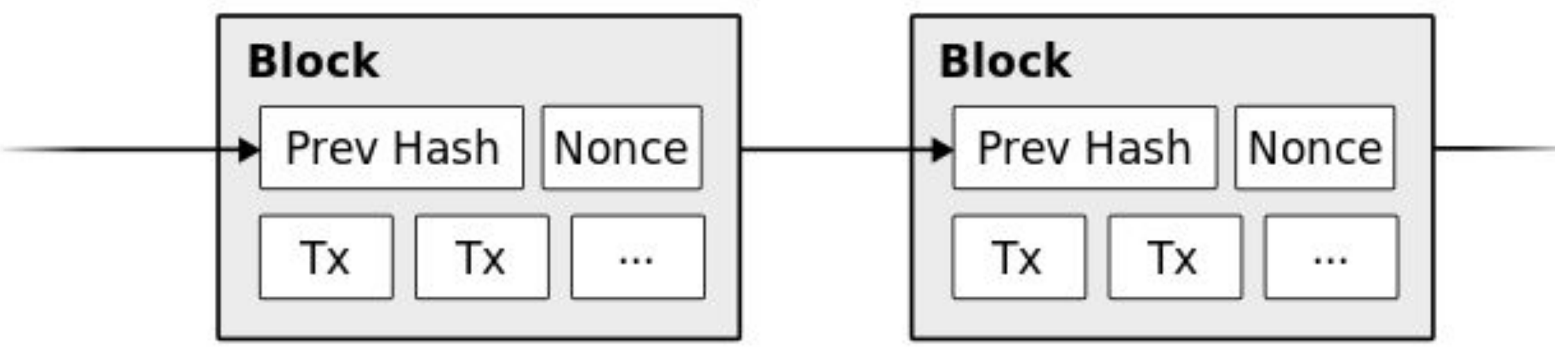
Hamid Azimy, Ali A. Ghorbani

Canadian Institute for Cybersecurity (CIC), University of New Brunswick (UNB)

ABSTRACT

Bitcoin mining is the process of generating new blocks in Bitcoin blockchain. This process is itself vulnerable to different types of attacks. One of these attacks is Selfish Mining, introduced by Eyal and Sirer in 2014. This attack is essentially a strategy that a sufficiently powerful mining pool can follow to obtain more revenue than its fair share. This is a tempting attack to deploy because every pool is trying to increase its revenue and this is a good opportunity. However, in most of the works to date, the effect of only one selfish pool has been studied. This is an attempt to analyze selfish mining in a competitive environment, where there are two selfish miners in play. To do so, we created a Bitcoin network simulator, and used it to simulate different configurations of miners to be able to address this problem.

Proof-of-Work and Mining



Proof-of-Work in Bitcoin (Source: Satoshi Nakamoto, 2008)

- Mining:
  - The process of creating a new block out of unconfirmed transactions (TXs) and appending it to the blockchain
  - Not all blocks will be accepted. Only blocks with a hash lower than a target difficulty will be accepted
  - Therefor it needs a considerable amount of computational (hash) power for the trial and error process

Selfish Mining

A strategy that miner could follow to obtain revenue more than its fair share.

- If you found a block:
  a. If there was a tie beforehand (meaning that you are now ahead), publish all your branch and you will win the race.
  b. Else (meaning that you are way ahead), continue mining on your private branch.
- If they (honest miners) found a block:
  a. If they are ahead, it means the won the race. Switch to their branch immediately.
  b. Else if there is a tie, reveal your private branch immediately and hope you will win the race.
  c. Else if you are one ahead, reveal all your chain. You would win because you are ahead of them.
  d. Else, (meaning you are way ahead of them), just reveal your first unpublished block, and continue mining on your private branch

Problem Formulation

General Case

M : Number of honest pools (typically equals to 1)

N : Number of selfish pools

$H_j \ , \ j \in \{1, 2, ..., M\}$ Honest pool number i

$\alpha_j \ , \ j \in \{1, 2, ..., M\}$ Mining power of H_i, in which

$S_i \ , \ i \in \{1, 2, ..., N\}$ Selfish pool number i

$\beta_i \ , \ i \in \{1, 2, ..., N\}$ Mining power of S_i

$$\sum_{j\in\{1,\ldots,M\}} \alpha_j = 1 - \sum_{i\in\{1,\ldots,N\}} \beta_i$$

Two selfish miners case

$N = 2$ (Two selfish miners, S_1 & S_2)

$\beta_t = \beta_1 + \beta_2$ in which

$$\beta_1 = \frac{\beta_t}{2} \times (1 + \delta)$$

$$\beta_2 = \frac{\beta_t}{2} \times (1 - \delta)$$

$M = 1$ (One honest miner, H_1)

$$\alpha_1 = 1 - \beta_t$$
$$= 1 - (\beta_1 + \beta_2)$$

Evaluation Metrics

R_P : Revenue of Miner P_k

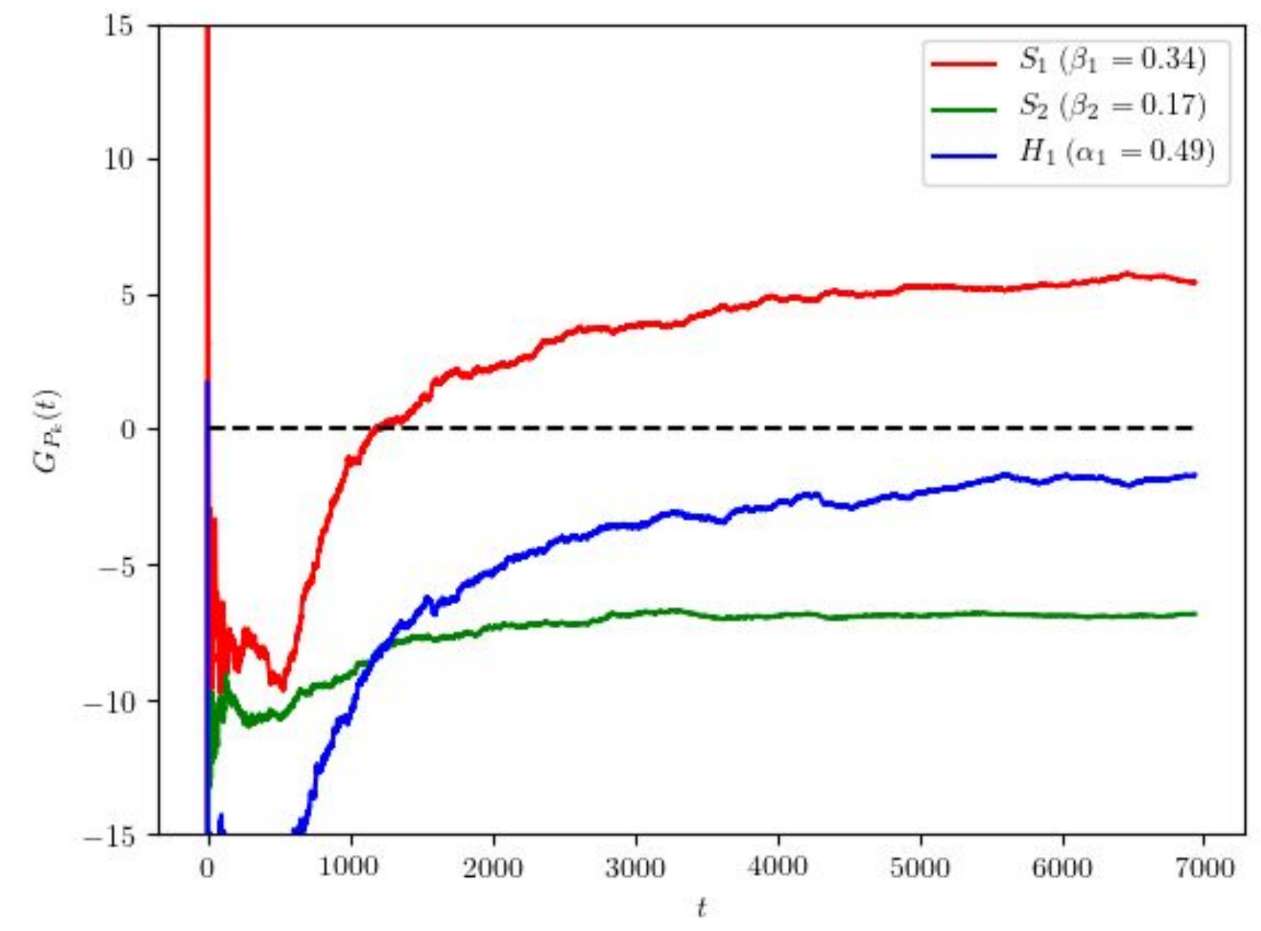
$\overline{R}_{P_k}$: Expected revenue of Miner P_k, when all miners are honest

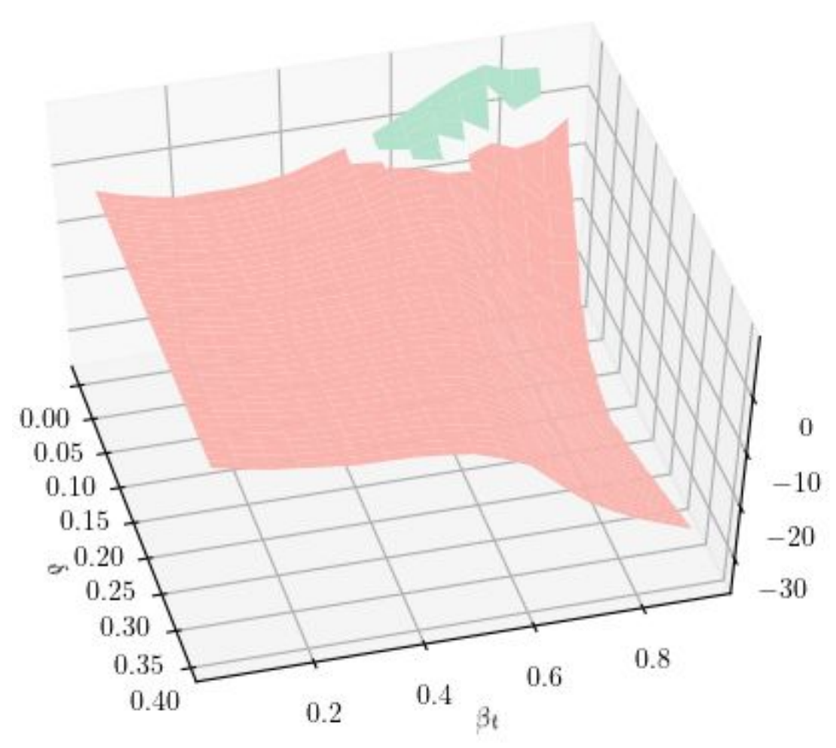
$\overline{R}_{P_k}(t) \approx \alpha_k \times t \times 6$

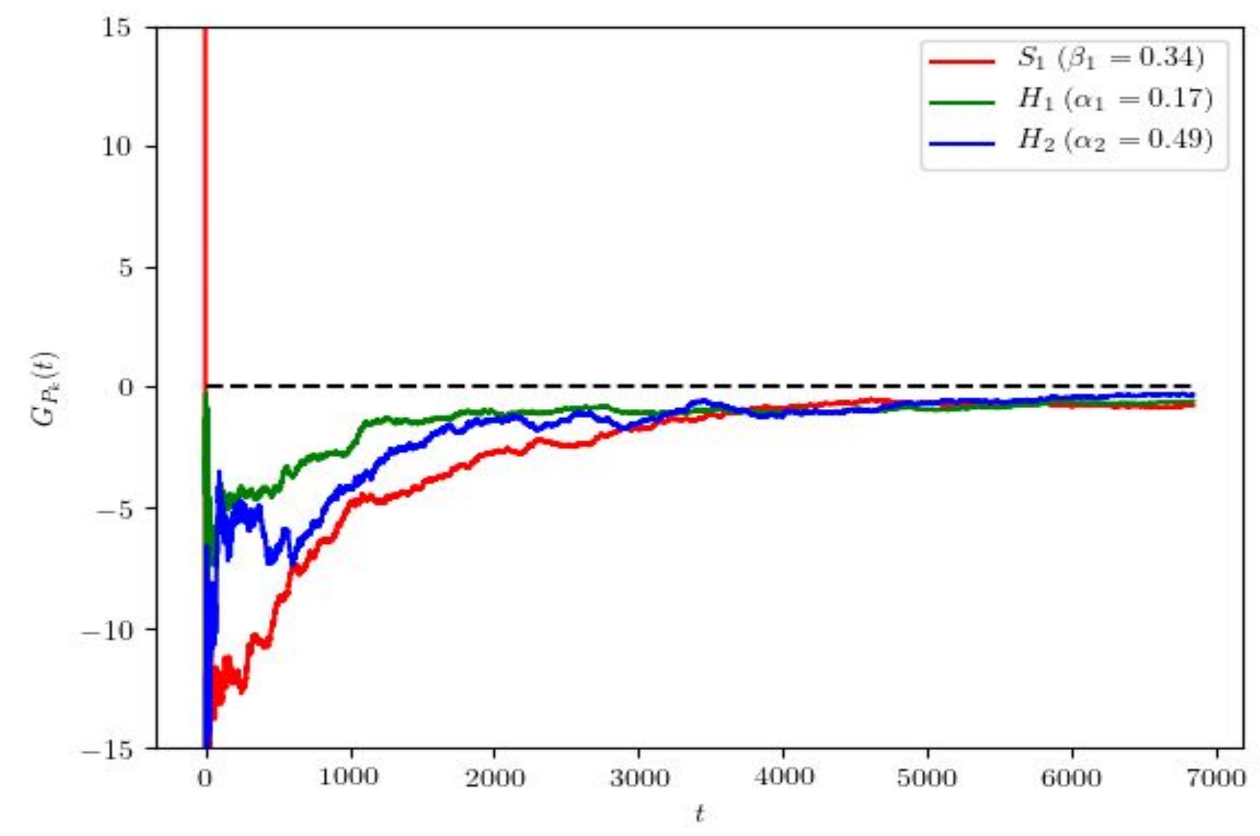
$R_{P_k}(t) \approx \overline{R}_{P_k}(t)$, if all miners are honest $(N = 0)$

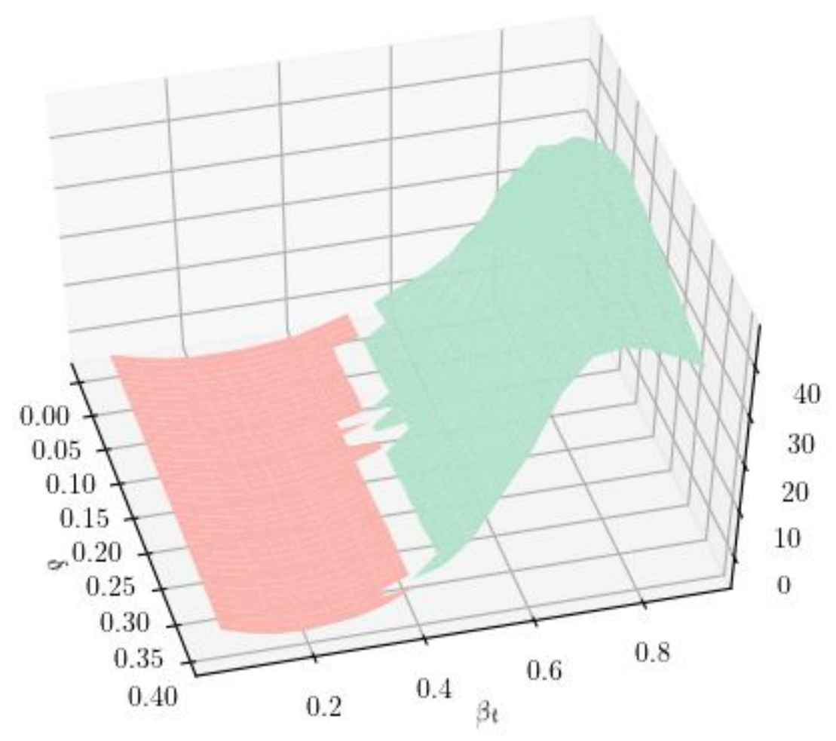
$$G_{P_k}(t) = \frac{R_{P_k}(t) - \overline{R}_{P_k}(t)}{t \times 6} \times 100$$

Simulation Results









Conclusion

- In case that both selfish miners have at least half of the network's hash power together:
  - It is profitable for the stronger selfish miner to continue mining selfishly.
  - The weaker selfish miner with suffer a loss of revenue.
- If the weaker selfish miner switches to honest mining:
  - Selfish mining will not be profitable for the (stronger) selfish miner anymore.