Simulating a double selfish mining attack using the P2P Cryptocurrency Network

CS 765 - Introduction of Blockchains, Cryptocurrencies, and Smart Contracts

Spring 2024

Indian Institute of Technology Bombay

Authors:

Debdoot (23M0765)

Nilava Sarkar (22M0753)

Niraj Jaiswal (22M0779)

Lecturer:

Prof. Vinay Joseph Ribeiro

Dept. of CSE, IIT Bombay

March 26, 2024

1. Introduction

A selfish mining attack in a blockchain network, such as Bitcoin's Proof of Work (PoW) consensus mechanism, involves malicious miners withholding newly discovered blocks from the network, secretly mining additional blocks on top of the withheld blocks, and strategically releasing their longer private chain to force a chain reorganization event. This unethical strategy enables selfish miners to unfairly increase their mining rewards and potentially gain control over the blockchain. By monopolising the block rewards and transaction fees, selfish miners undermine the network's security and decentralization, threatening its integrity.

2. State Diagram

The state diagram of the experiment is shown in the figure 1.

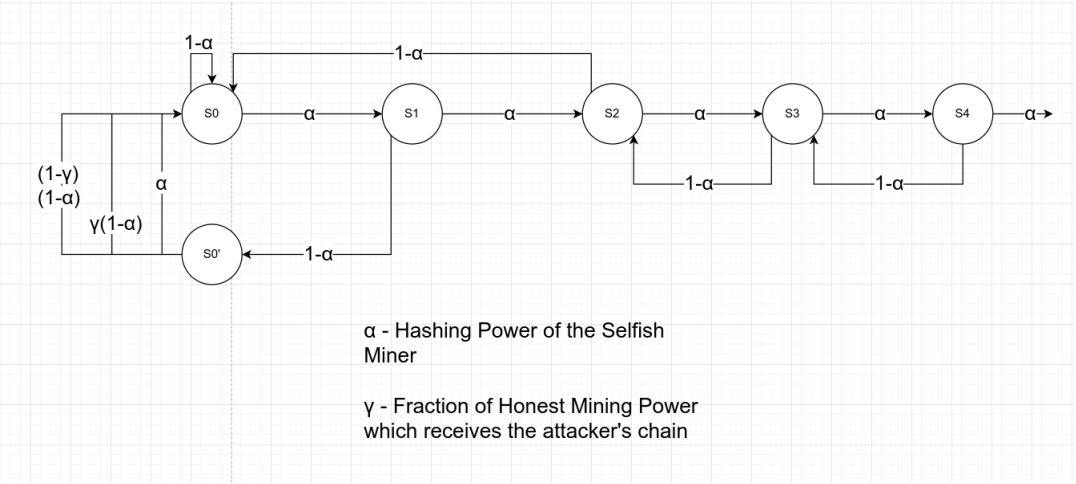

Figure 1: State Diagram

3. Simulation Parameters

- N = Number of Honest Miners
- ζ_1 = Hashing Power of Selfish Miner 1
- ζ_2 = Hashing Power of Selfish Miner 2
- Ttx = Mean Interarrival Time of Transactions
- I = Mean Mining Time of Blocks
- T = Simulation Time

4. Experiment and Analysis

The Evaluation ratio is shown in the figure 2.

$$MPU_{node_{adv}} = \frac{\text{Number of block mined by an adversary in final public main chain}}{\text{Total number of blocks mined by this adversary overall}}$$

$$MPU_{node_{overall}} = \frac{\text{Number of block in the final public main chain}}{\text{Total number of blocks generated across all the nodes}}$$

Figure 2: Evaluation Ratio

MPUnode(adv) - Measures the success of the selfish miner. The success is determined by the number of selfish blocks it is able to insert in the public blockchain. An increase in hashing power of the Selfish Node will increase this ratio more.

MPUnode(overall) - Measures the number of blocks not wasted due to forking. It is determined by the total number of blocks mined by all the nodes in the longest chain with respect to all the chains/forks. The ratio is

low when there are a lot of forks, and many blocks are orphaned. Similarly, the ratio is high when there are fewer or no forks.

4.1 Experiment

Experiment done with the following parameters unless otherwise specified.

N = 10, ttx = 10 sec, I = 300 sec, T = 6000 sec, $\zeta_1 = 0.3$, $\zeta_2 = 0.3$.

- $\zeta_1 = \mathbf{0.3}$, $\zeta_2 \approx \mathbf{0}$ Total Nodes mined by Adv1 = 8

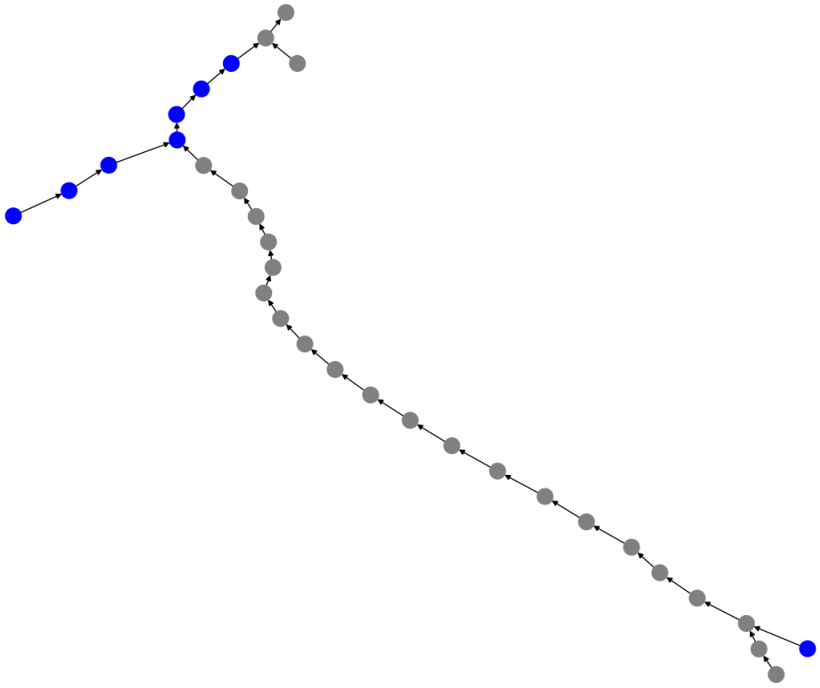

Figure 3: Blockchain1

Total Nodes mined by Adv1 in the longest chain = 4

Total Nodes mined by Adv2 = 0

Total Nodes mined by Adv2 in the longest chain = 0

Total Nodes mined by all nodes = 32

Total Nodes mined in the longest chain = 27

Only one of the selfish miners has a hash power of 30% to attack the main chain but not enough to continue the selfish attack for a longer time. Thus creating very few forks with a success ratio MPU(adv) of about 50%, while MPU(overall) is about 90%.

An increase in ζ_1 up to a certain extent will increase MPU(adv), thus creating more forks and creating more longer attacks in the chain.

- ζ_1 **= 0.3,** ζ_2 **= 0.3** Total Nodes mined by Adv1 = 8

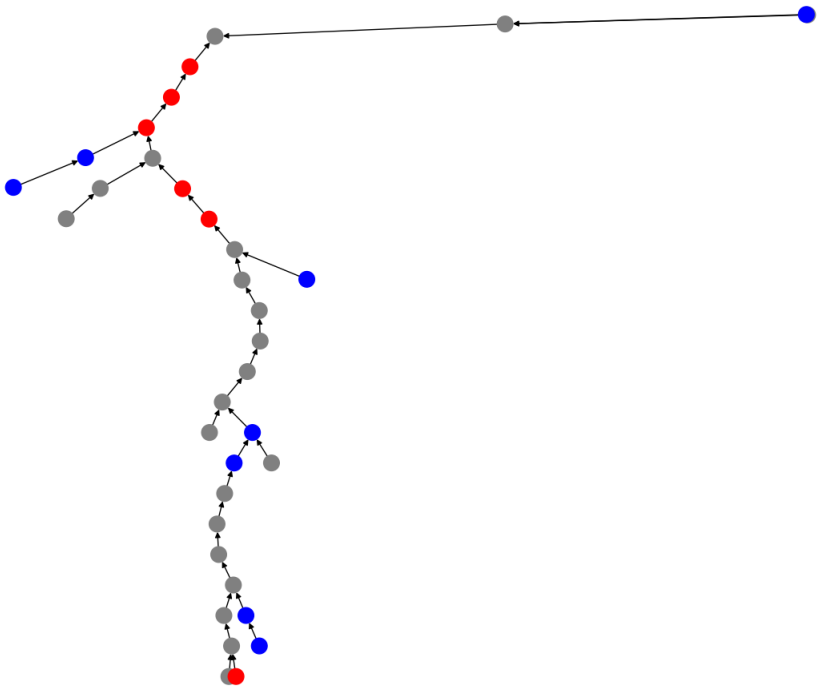

Figure 4: Blockchain2

Total Nodes mined by Adv1 in the longest chain = 2

Total Nodes mined by Adv2 = 6

Total Nodes mined by Adv2 in the longest chain = 4

Total Nodes mined by all nodes = 35

Total Nodes mined in the longest chain = 22

Both the selfish miners have a hash power of 30% each, thus allowing both of them to create selfish attacks on the chain. These will create more selfish blocks than the previous chain due to the lowering of total hash power for the honest miners.

But both of the selfish miners are unable to continue the attack for longer. Selfish miners need more hash powers to hold attacks for longer.

- $\zeta_1 = \mathbf{0.6}$, $\zeta_2 \approx \mathbf{0}$ Total Nodes mined by Adv1 = 20

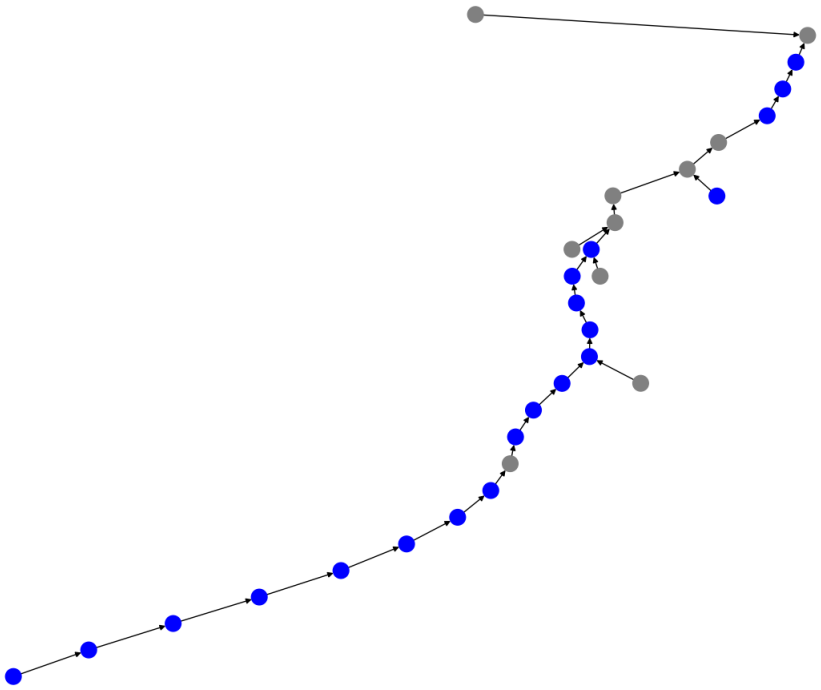

Figure 5: Blockchain3

Total Nodes mined by Adv1 in the longest chain = 19

Total Nodes mined by Adv2 = 0

Total Nodes mined by Adv2 in the longest chain = 0

Total Nodes mined by all nodes = 30

Total Nodes mined in the longest chain = 25

Selfish Miner 1 has been given a hash power of almost 60%. As expected it mined significant number of blocks in the main chain, while the honest nodes and Selfish Miner 2 struggled to make a block.

Forking are less than in the previous case due to the huge hash power assigned to the Selfish Miner. The honest miners are not only struggling to make a longer chain but also to keep up with mine time of the blocks.

- $\zeta_1 = \mathbf{0.9}$, $\zeta_2 \approx \mathbf{0}$ Total Nodes mined by Adv1 = 20

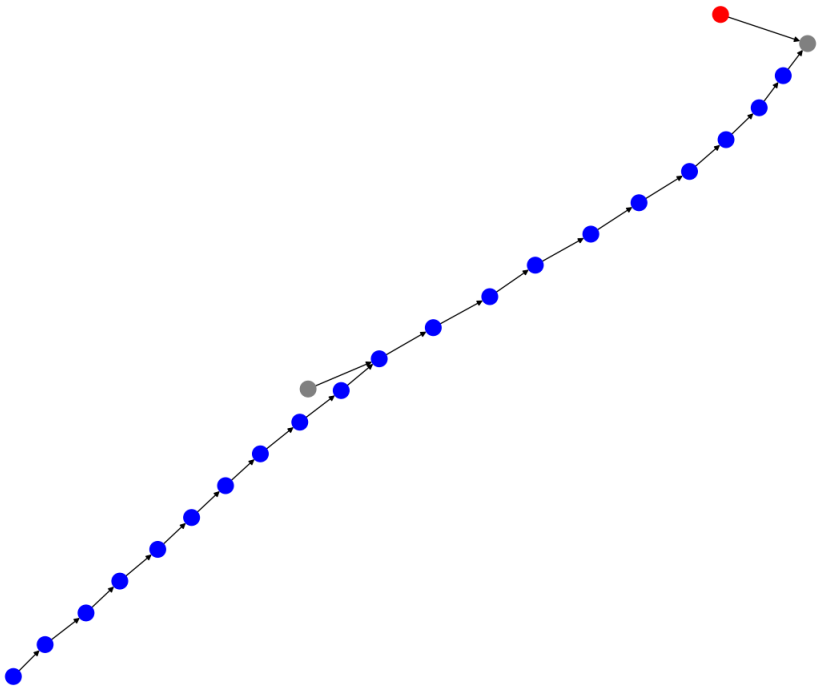

Figure 6: Blockchain4

Total Nodes mined by Adv1 in the longest chain = 20

Total Nodes mined by Adv2 = 1

Total Nodes mined by Adv2 in the longest chain = 0

Total Nodes mined by all nodes = 23

Total Nodes mined in the longest chain = 21

The worst case of all is where one of the selfish miners is assigned almost 90% of the hash power of the entire network. Thus creating little to no fork in the main chain, as honest miners will not be able to keep up with the selfish miner hash power and selfish attack.

- $\zeta_1 \approx \mathbf{0.5}$, $\zeta_1 \approx \mathbf{0.5}$ Total Nodes mined by Adv1 = 13

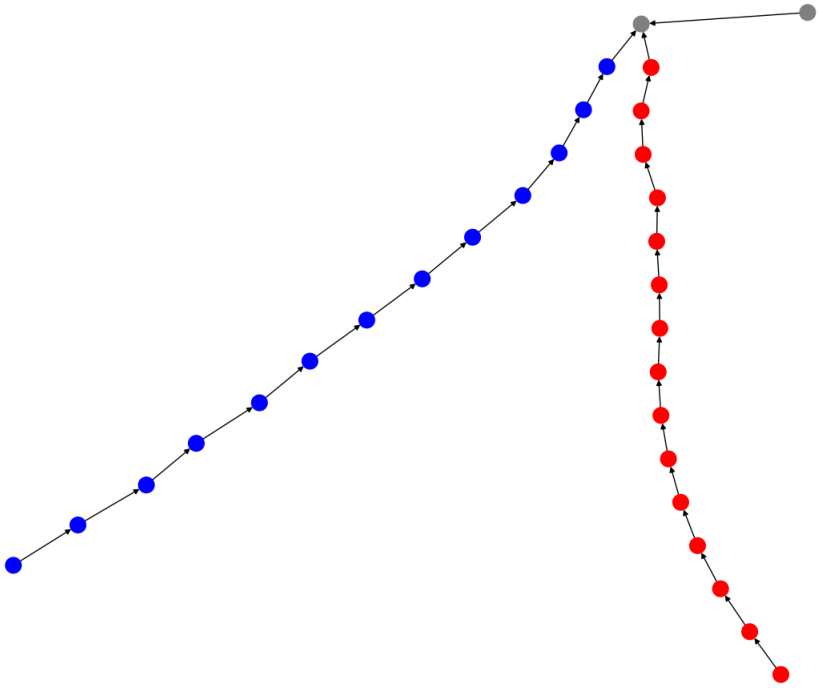

Figure 7: Blockchain5

Total Nodes mined by Adv1 in the longest chain = 13

Total Nodes mined by Adv2 = 15

Total Nodes mined by Adv2 in the longest chain = 15

Total Nodes mined by all nodes = 29

Total Nodes mined in the longest chain = 16

Both the selfish miner has a cumulative hash power of almost 100%, each having a hash power of close to 50%. Due to this, both miners selfishly mine on their private chain and avoid releasing the chain, thus creating two different independent chains for each of the selfish miners and not knowing of the other private chain.

Both release the chain after the simulation time is over, creating two

long chains of blocks mined only by the miners releasing the chain.

4.2 Analysis

4.2.1 MPU(adv) Vs. MPU(overall)

MPU(adv) - As ζ (Hashing Power of a Selfish Node) increases, the ratio MPU(adv) also increases due to a simultaneous decrease in the hashing power of the honest miners. More ζ means the hashing power of honest miners will struggle to compete with the selfish node. Around ζ=80% we can notice, that the selfish miner is achieving almost 100% ratio, i.e. all blocks mined by it are in the longest chain.

MPU(overall) - As ζ increases up to a certain extent, MPU(overall) decreases due to more forking in the chains. It reaches the minimum when the hashing power is equally divided among both honest and selfish miners (Around ζ=50%). As ζ increases above 50%, the hashing power tends to one side (i.e. Selfish Miner gets more hashing power than honest miners), creating less forks in the main chain and thus again increasing the MPU(overall) ratio. It reaches 100% again when the selfish miner has around 100% of the hashing power.

The graph of the two ratios is shown in figure 8.

4.2.2 MPU(overall) Vs. Ratio of Number of Selfish Blocks in the Longest Chain

If we fix the hashing power of one of the selfish miner (ζ=10%) and increase the number of honest miners gradually we can notice the selfish miner gradually taking over the longest chain. Since the hashing power of the honest miners are fixed, so gradual increase of number of honest miners will slowly dilute the hash power of each honest miner. As a result, as we increase N,

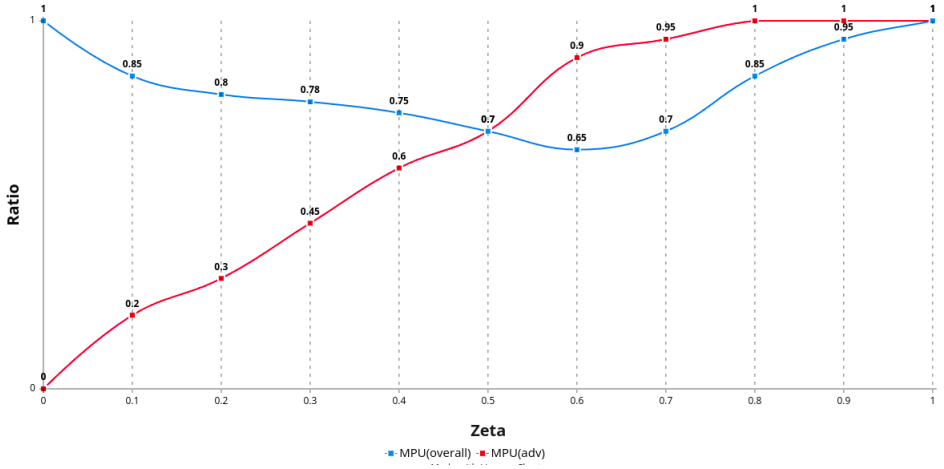

Figure 8: Ratio Graph

number of selfish blocks increases gradually in the longer and MPU first decreases overall up to a certain point then again increases as forking gets less due to dilution of honest miners hash power. The graph of the two ratios is shown in figure 9.

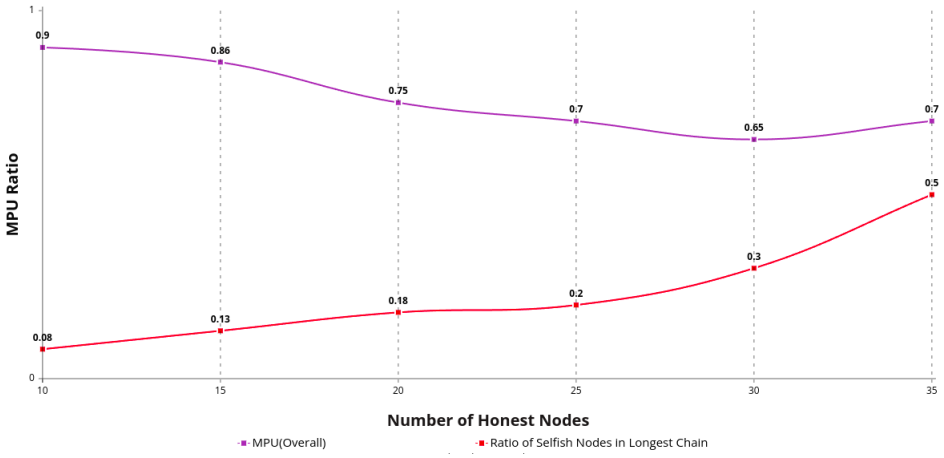

Figure 9: Ratio Graph

Further increase of N could not be done due to hardware constraints.