P2P Cryptocurreny-Network

Arijit Saha	Aryan Mathe	Khushang Singla
210050017	210050021	210050085

February 2024

1 Introduction

This assignment involves simulating **selfish mining attack** as described in the paper **Majority is not Enough**. The only difference is that, in this assignment we are going to involve **two attackers** and both of them will perform attacks based on the strategy described in the paper. Another important detail is that, both the attackers are unfamiliar of each other and believes that there is only one single attacker present in the network. This assignment is a continuation of the previous assignment in which we have developed a simulator to simulate blockchain events.

2 Running Instruction

```
$ ./blockSim
usage: blockSim [OPTIONS...]

  -t, --interarrival_transaction_time FLOAT (inter arrival time between transactions)
  -k, --interarrival_block_time FLOAT(mean interarrival block time)
  -b, --max_blocks INT (number of blocks to terminate on)
  -a, --initial_amt INT (initial coins for each node)
  -i, --frac_slow FLOAT (fraction of slow (high latency) nodes)
  -s, --seed INT (seed for randomness in simulation)
  -x, --max_transactions INT (Number of Transactions for stopping criteria)
  -g, --graph STRING (path of graph file generated by python)
  -z, --g1 FLOAT (hashing power of selfish miner 1)
  -y, --g2 FLOAT (hashing power of selfish miner 2)
  ?, --help show this help message and exit
```

3 Experiments

Fixed Parameters

- **Mean Interarrival Block Time** = 10
- **Mean Interarrival Transaction Time** = 0.1
- **Number of Miners** = 100
- **Fraction of Slow (on Networking) Nodes** = 0.5

Variable Parameters

- **Attacker-1 hashing power** τ_1
- **Attacker-2 hashing power** τ_2

3.1 Insights and Critique on MPU of Adversary Nodes

This section contains the analysis on MPU of adversary nodes present in the blockchain network.

The equation for calculating the $MPU_{node_{adv}}$ is given below

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

3.1.1 Insights

- When the hashing power of first attacker is zero then it's $MPU_{node_{adv}}$ is also **zero**
- Also, if we fix the hashing power of second attacker then, it can be observed that the $MPU_{node_{adv}}$ for the first attacker increases with the increase in its hashing power
- Another important thing to observe is that the **difference** between the blocks produced by second attacker at different hashing power decreases with increase in the hashing power of the first attacker node in the network
  For example - in case of Adversary-1 MPU bar-graph

$$MPU_{[0.3,0.3]} - MPU_{[0.3,0]} > MPU_{[0.4,0.3]} - MPU_{[0.4,0]} > MPU_{[0.5,0.3]} - MPU_{[0.5,0]}$$

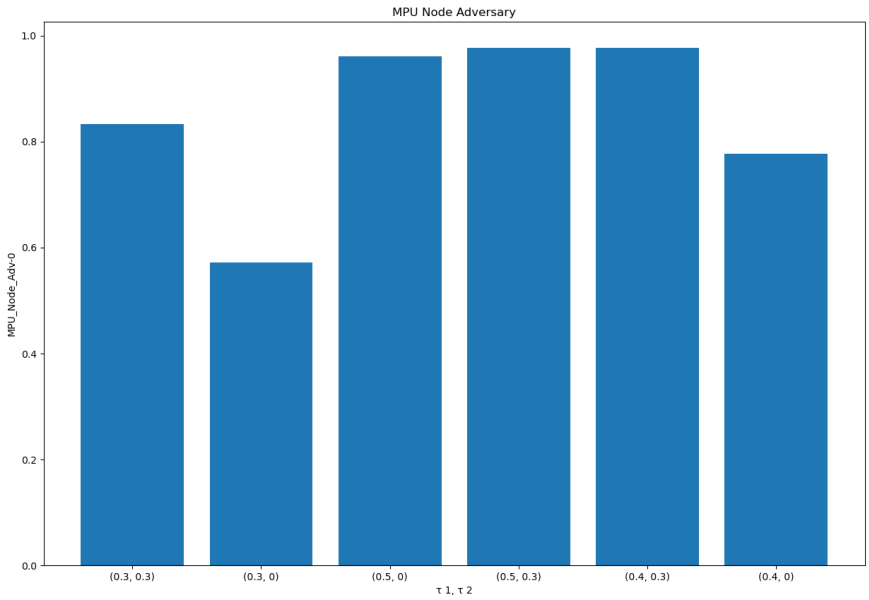

Figure 1: MPU for 1st Adversary-Node

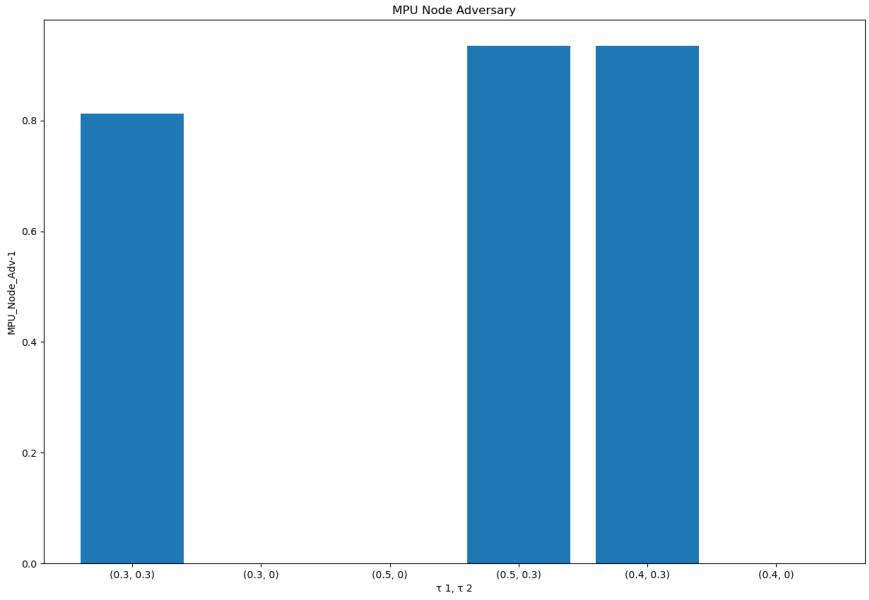

Figure 2: MPU for 2nd Adversary-Node

3.1.2 Critique

- If an attacker has **zero** hashing power then it will not be able to perform any **PoW** and will not contribute in the **LVC**.
- Since the length of private chain is directly proportional to the hashing power of the attacker, and the length of private chain will decide the number of blocks the attacker can insert in **LVC**. Hence keeping hashing power of other attacker fixed we observe that the number of blocks mined by attacker in **lvc** increase with increase in the hashing power which ultimately increases the $MPU_{node_{adv}}$
- The reason for third insight being that, if the hashing power of first attacker increase then it becomes less likely for the other attacker's private chain to successfully join **LVC** because now it will be more likely for the first attacker's block to get accepted and hence it's $MPU_{node_{adv}}$ decreases.

3.2 Insights and Critique on MPU of Overall Nodes

3.2.1 Insights

$MPU_{node_{overall}}$ is found using the formula given below

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

- It can be observed that, as the net hashing power of attackers increases then $MPU_{node_{overall}}$ tends to decrease roughly
- In case of one attacker having hashing power **zero**, the longest main chain has significantly larger fraction of blocks as compared to the case when both attacker nodes has **non-zero** hashing power
- On keeping the hashing power of one attacker constant the $MPU_{node_{overall}}$ increases with decrease in the hashing power of other attacker.
  For example

$$MPU_{[0.3,0.3]} < MPU_{[0.3,0]},\ MPU_{[0.4,0.3]} < MPU_{[0.4,0]},\ MPU_{[0.5,0.3]} < MPU_{[0.5,0]}$$

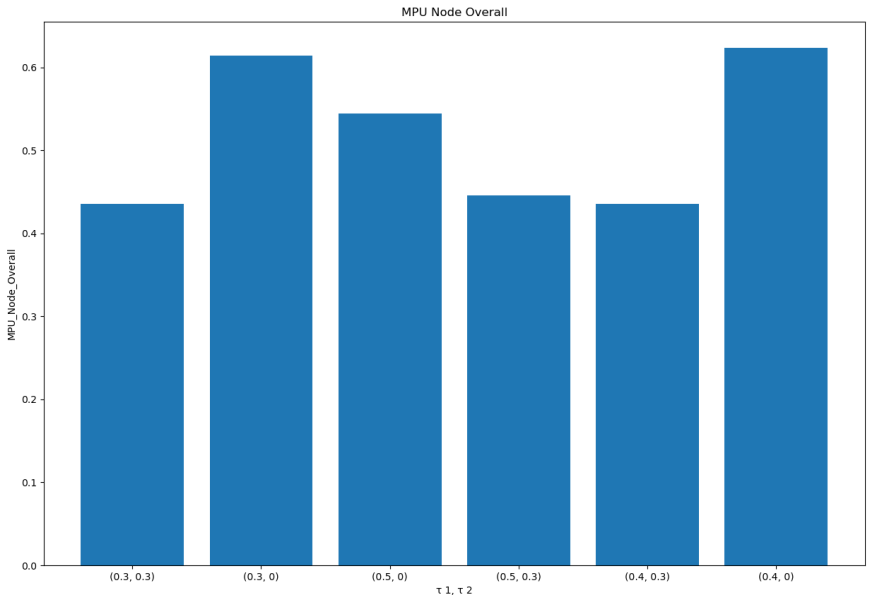

Figure 3: MPU for Overall Nodes

3.2.2 Critique

- The decrease in the $MPU_{node_{overall}}$ with increase in the net hashing power is because of the fact that, as the hashing power of the attackers increase there will be more chances of blocks getting mined on a fork which is not **lvc**, and which makes its more probable for blocks to get wasted increasing the total number of blocks and effectively decreasing the blocks in the **lvc**
- When both attacker has non-zero hashing power it is more likely for many blocks to get wasted because both attacker's are unfamiliar of each other's presence and will compete with each other.
- Keeping one attacker's hashing power constant, increase in the $MPU_{node_{overall}}$ with decrease in other attacker's hashing power is evident from the fact that, when the hashing power of attacker is low it's more likely that the mining will happen on the **lvc** and less blocks will be wasted

3.3 Study on Fraction of Attacker's Block in main chain

NOTE
For this study we have chosen the longest chain which consists of blocks mined by all nodes (adversary + honest) till the simulation end. Because of which the blocks which are not yet released by the adversary will also be considered in the longest chain

3.3.1 Insights

- It can be clearly observed from **Figure:4** and **Figure:5** that the fraction of attacker's block increase with increase in it's hashing power for both cases ($\tau_2 = 0$ and $\tau_2 = 0.3$).

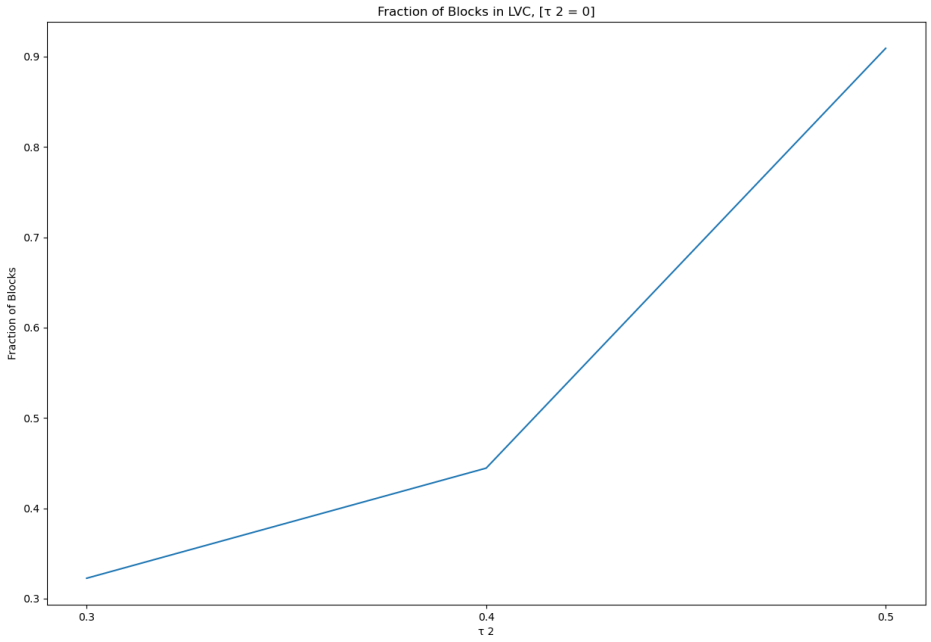

Figure 4: Fraction of Attacker-1 Blocks

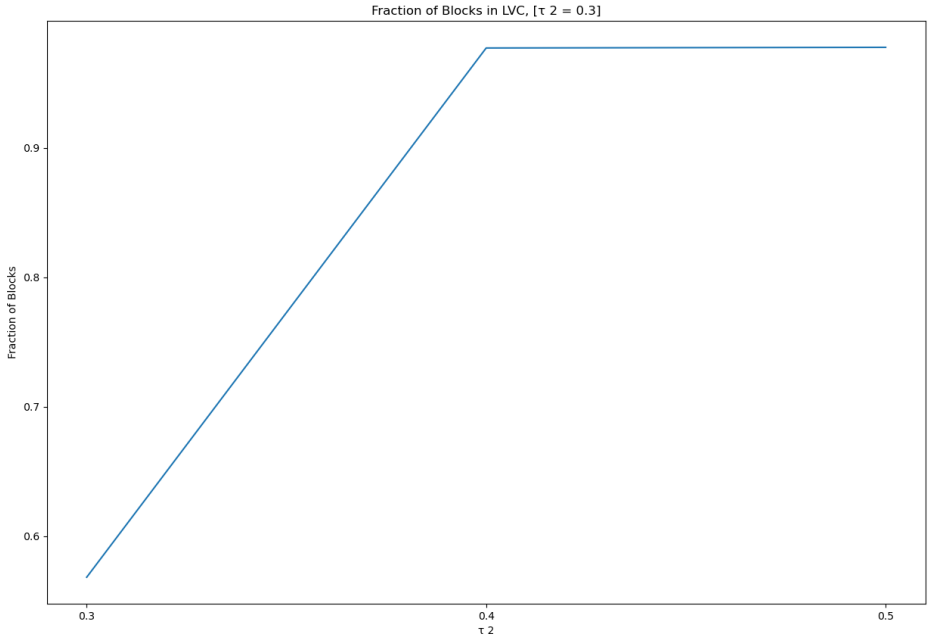

Figure 5: Fraction of Attacker-1 Blocks

3.3.2 Critique

- The number of attacker's blocks with increase in the hashing power increase is clearly evident from the fact that more hashing power will boost the capacity to perform **PoW** and hence the attacker will be able to increase the fraction of blocks mined by her.
- Also, it is interesting to note that the number of blocks mined by the attacker-1 seems to saturates which could be because of the fact that the net hashing power of attackers in the network increases too much (whose rough estimate is ≥ 0.7 in our case) so it's very likely that one of the attacker will share a very large proportion of blocks (in our case it happens to be Attacker-1 both the time)

END OF REPORT