Selfish Mining in Blockchain Systems

Sheng-Wei Wang

Department of Electronic Engineering
National United University
Miaoli, TAIWAN

swwang@nuu.edu.tw

2023-12-04

Outlines

1. **Introductions**
   - Bitcoins & Blockchains
   - Selfish Mining
2. **Our Researches**
   - Selfish Mining in Sharded Blockchains
   - Observations in Blockchains with Multiple Selfish Miners
   - Analytical Model for Blockchains with Multiple Selfish Miners
   - Rational Mining Strategy
3. **Ongoing & Future Works**

Bitcoins

Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

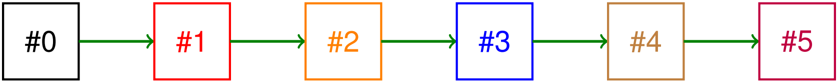
Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

Blockchains

- Blockchain is a decentralized ledger stored in a distributed network
- Transactions are securely stored in distributed networks
- Consecutive blocks form a blockchain using cryptography

Blockchains

- Blockchain is a decentralized ledger stored in a distributed network
- Transactions are securely stored in distributed networks
- Consecutive blocks form a blockchain using cryptography



Mining & Miners

Mining & Proof-of-Works

ID: # 815,184
Hash of Block # 815,183
Transactions
Timestamp
Nonce

- Hash of Block #815,183:

  0000000000000000003d09220e85bbdbb832b86e3dc711c5cda888b1daf5985

Mining & Proof-of-Works

ID: # 815,184
Hash of Block # 815,183
Transactions
Timestamp
893,869,249

ID: # 815,185
Hash of Block # 815,184
Transactions
Timestamp
Nonce

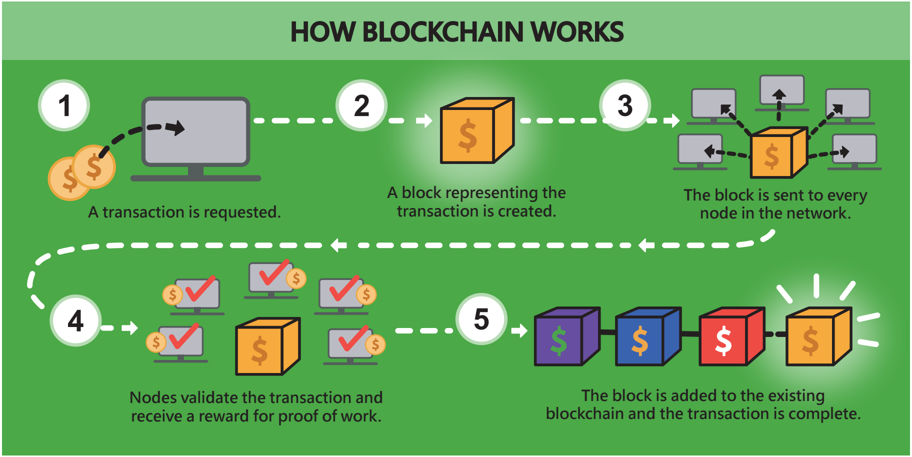
- Hash of Block #815,183:

  00000000000000000003d09220e85bbdbb832b86e3dc711c5cda888b1daf5985

- Hash of Block #815,184:

  00000000000000000000cc167c107a24883b34c16aad188aaa72412cc0ef437a

How Blockchain Works?



Selfish Mining Strategy

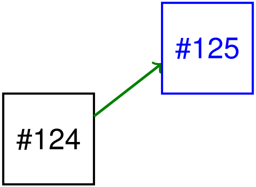
Selfish Mining

- The number of nonces attempted by a miner to solve the puzzle per unit of time is defined as his **mining rate**
- Generally, the probability which the next block is mined by a specific miner shall be proportional to his mining rate
- A mining strategy called **selfish mining** enables a miner to be **profitable**; that is, to earn more rewards than he would be entitled to
- Main idea of selfish mining is not to broadcast the mined blocks when a selfish miner mined a new block
- Define: total honest mining rate h and total selfish mining rate s

Selfish Mining

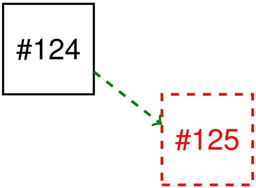
#124

Selfish Mining



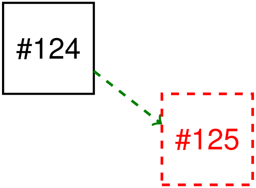
- If honest miner mined the block first, he/she announces the block immediately
- All miners validate the block and start to mine the next one

Selfish Mining



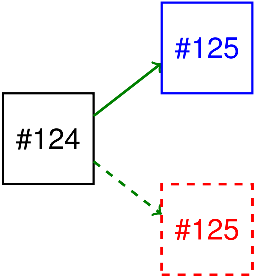
- If honest miner mined the block first, he/she announces the block immediately
- All miners validate the block and start to mine the next one
- If selfish miner mined the block first, he/she hides the block in his/her private branch

Selfish Mining



If honest miner mined the next block first under the above condition:

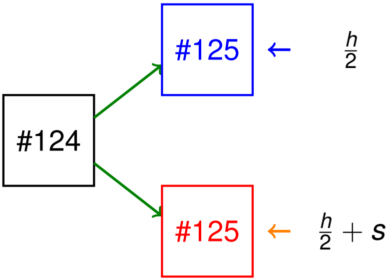
Selfish Mining



If honest miner mined the next block first under the above condition:

- The honest miner announces the block immediately

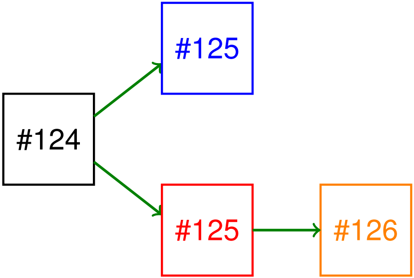
Selfish Mining



If honest miner mined the next block first under the above condition:

- The honest miner announces the block immediately
- The selfish miner releases his/her hidden block immediately

Selfish Mining

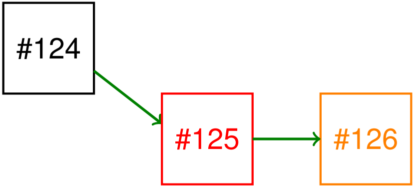


If honest miner mined the next block first under the above condition:

- The honest miner announces the block immediately
- The selfish miner releases his/her hidden block immediately

Longest Chain Rule: The branch on which the next block is mined first becomes the valid chain

Selfish Mining

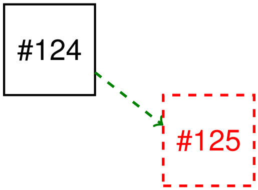


If honest miner mined the next block first under the above condition:

- The honest miner announces the block immediately
- The selfish miner releases his/her hidden block immediately

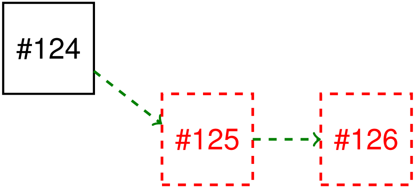
Longest Chain Rule: The branch on which the next block is mined first becomes the valid chain

Selfish Mining



If selfish miner mined the next block first under the above condition:

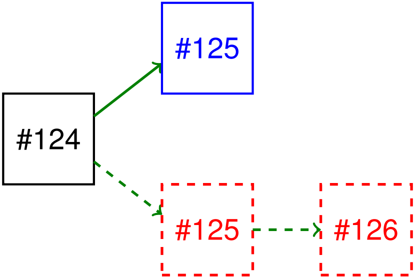
Selfish Mining



If selfish miner mined the next block first under the above condition:

- The selfish miner hides the blocks in his/her branch

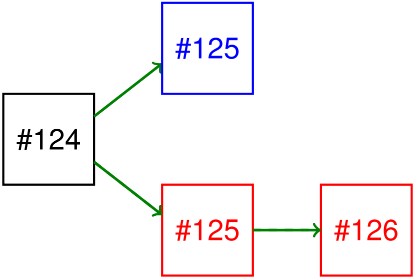
Selfish Mining



If selfish miner mined the next block first under the above condition:

- The selfish miner hides the blocks in his/her branch
- If the honest miner mined the block now, the honest miner release the block immediately

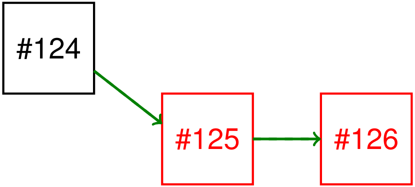
Selfish Mining



If selfish miner mined the next block first under the above condition:

- The selfish miner hides the blocks in his/her branch
- If the honest miner mined the block now, the honest miner release the block immediately
- The selfish miner releases his/her blocks immediately

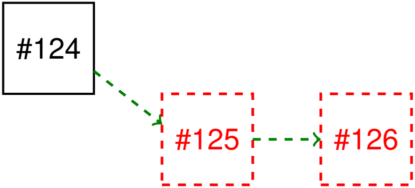
Selfish Mining



If selfish miner mined the next block first under the above condition:

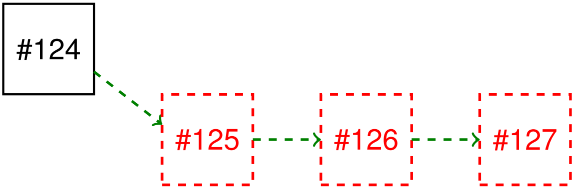
- The selfish miner hides the blocks in his/her branch
- If the honest miner mined the block now, the honest miner release the block immediately
- The selfish miner releases his/her blocks immediately
- Longest chain rule is applied

Selfish Mining



If selfish miner mined the next block first under the above condition:

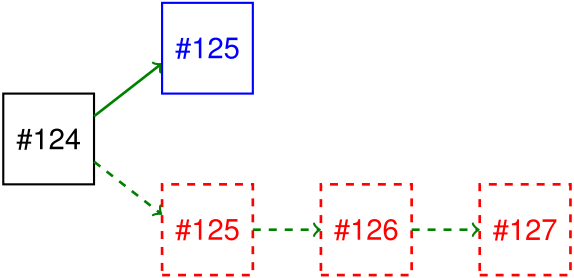
Selfish Mining



If selfish miner mined the next block first under the above condition:

- The selfish miner hides the blocks in his/her branch

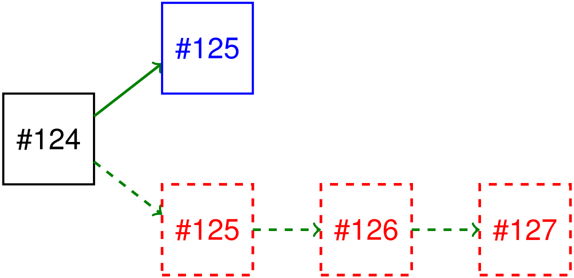
Selfish Mining



If selfish miner mined the next block first under the above condition:

- The selfish miner hides the blocks in his/her branch
- If the honest miner mined the block now, the honest miner releases the block immediately

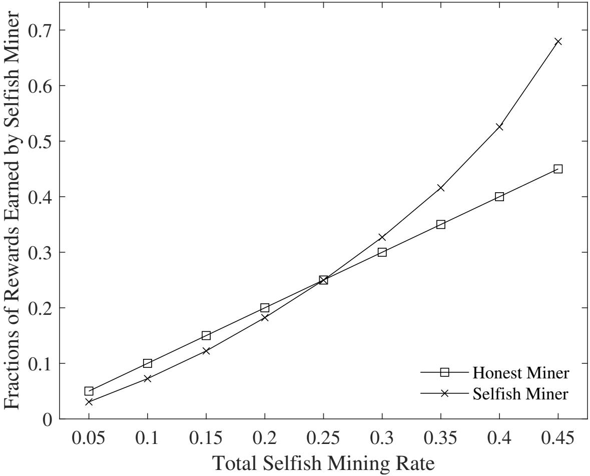
Selfish Mining



If selfish miner mined the next block first under the above condition:

- The selfish miner hides the blocks in his/her branch
- If the honest miner mined the block now, the honest miner releases the block immediately
- The selfish miner do **NOTHING** now

Rewards & Profitable Threshold (25%)



Analytical Model Proposed by *I. Eyal* and *E.G. Sirer*

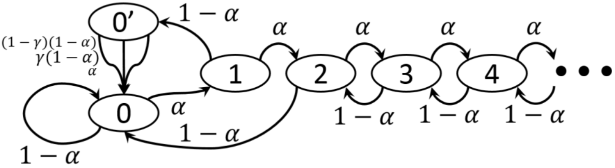


Fig. 1: State machine with transition frequencies.

$$RW(\alpha) = \begin{cases} 1 & \text{if } \alpha \geq 0.5, \\ \frac{\alpha(1-\alpha)^2[4\alpha+\frac{1}{2}(1-2\alpha)]-\alpha^3}{1-\alpha[1+(2-\alpha)\alpha]} & \text{otherwise} . \end{cases}$$

Note: Function $RW(\alpha)$ is strictly increasing and convex in [0,1/2].

Selfish Mining in Sharded Blockchains

- S.W. Wang, "Selfish Mining Attacks in Sharded Blockchains," in *2024 International Conference on Computing Networking and Communications (ICNC) (ICNC 2024)*, Big Island, Hawaii, USA, 2024.

Will Selfish Mining Work?

- Criticism: It is impossible to have a miner with more than 25% of mining rates.
- Answer: Doesn’t work in Large Public Blockchains such as Bitcoin or Ethereum.

Will Selfish Mining Work?

- Criticism: It is impossible to have a miner with more than 25% of mining rates.
- Answer: Doesn't work in Large Public Blockchains such as Bitcoin or Ethereum.
- However,
  - Mining pools with multiple miners
  - Small scale blockchains
  - Sharding

Sharding

- Main problem of blockchain technology is its scalability
- Main idea of sharding is to partition the nodes (miners) into a number of subsets each of which maintains a sub-blockchain (shard)
- Advantage: Throughput is improved
- Disadvantage:
  - Large cross-shard transactions
  - Security issues (The number of nodes decreased)
    - Consensus
    - Selfish Mining

Single vs. Sharded Blockchain

- Single Blockchain

#0 → #1 → #2 → #3 → #4 → #5

- Sharded Blockchain (3-Shard Blockchain)

#1-0 → #1-1 → #1-2 → #1-3 → #1-4 → #1-5

Single vs. Sharded Blockchain

- Single Blockchain

#0 → #1 → #2 → #3 → #4 → #5

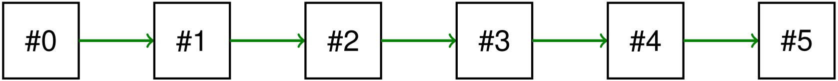
- Sharded Blockchain (3-Shard Blockchain)

#1-0 → #1-1 → #1-2 → #1-3 → #1-4 → #1-5

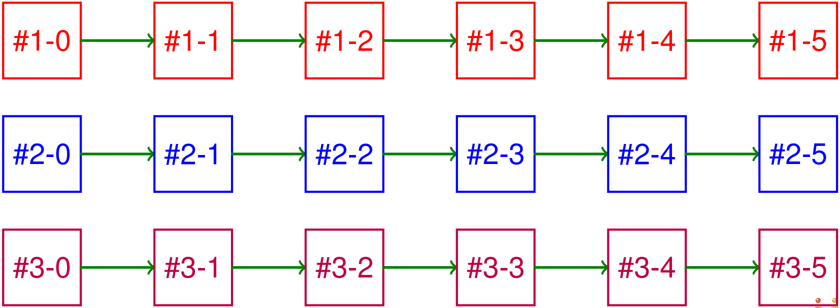
#2-0 → #2-1 → #2-2 → #2-3 → #2-4 → #2-5

Single vs. Sharded Blockchain

- Single Blockchain



- Sharded Blockchain (3-Shard Blockchain)



Fraction of Overall Rewards Earned by Selfish Miners

- Fraction of rewards earned in a single blockchain

$$R_{single} = RW(s/(s+h))$$

- Honest and Selfish mining rates in shard i are denoted as h_i and s_i in K-shard blockchain where $h = \sum_{i=1}^{K} h_i$ and $s = \sum_{i=1}^{K} s_i$
- Fraction of rewards earned in shard i

$$R_i = RW(s_i/(h_i+s_i))$$

- Fraction of rewards earned in shard i

$$R_{shard} = \frac{1}{K}\sum_{k=1}^{K} R_k$$

Optimization Problem

Given the number of shards K, the honest mining rates h_i for all shards i,

$$\textbf{Maximize} \quad R_{shard} = \frac{1}{K}\sum_{i=1}^{K} R_i$$

$$\textbf{With respect to} \quad s_i\ ,\ \ \forall\, i = 1, 2, \cdots, K$$

Subject to constraints

$$\sum_{i=1}^{K} s_i = s$$

$$R_i = RW\left(\frac{s_i}{h_i + s_i}\right)$$

$$0 \le s_i \le s, \quad i = 1, \cdots, K$$

Proposed Algorithm

Main idea: to dominate as more shards as possible.

Step. 1 Sort the shards in ascending order according to the values of h_i. Let remaining selfish rate $\bar{s} = s$.

Step. 2 Repeat assigning $s_i = h_i$ and then $\bar{s} = \bar{s} - s_i$ until $\bar{s} < h_{m+1}$. The number of shards dominated by selfish miners is denoted as m.

Step. 3 Assign $s_{m+1} = \bar{s}$. Set $\bar{s} = 0$ now.

Step. 4 Assign $s_{m+2}, s_{m+3}, \cdots, s_K$ to 0s.

Proposed Algorithm

Main idea: to dominate as more shards as possible.

Step. 1 Sort the shards in ascending order according to the values of h_i. Let remaining selfish rate $\bar{s} = s$.

Step. 2 Repeat assigning $s_i = h_i$ and then $\bar{s} = \bar{s} - s_i$ until $\bar{s} < h_{m+1}$. The number of shards dominated by selfish miners is denoted as m.

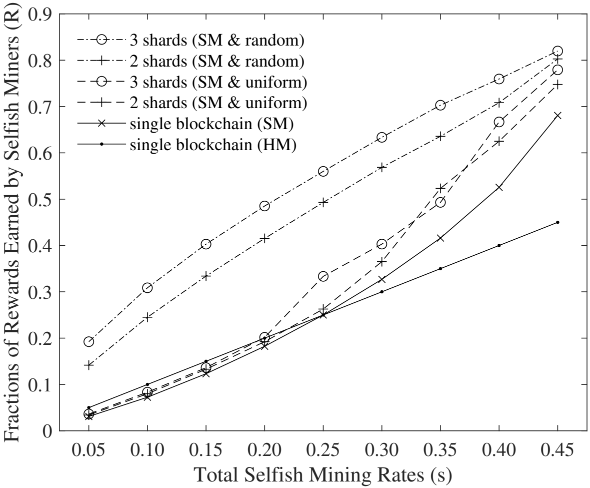
Step. 3 Assign $s_{m+1} = \bar{s}$. Set $\bar{s} = 0$ now.

Step. 4 Assign $s_{m+2}, s_{m+3}, \cdots, s_K$ to 0s.

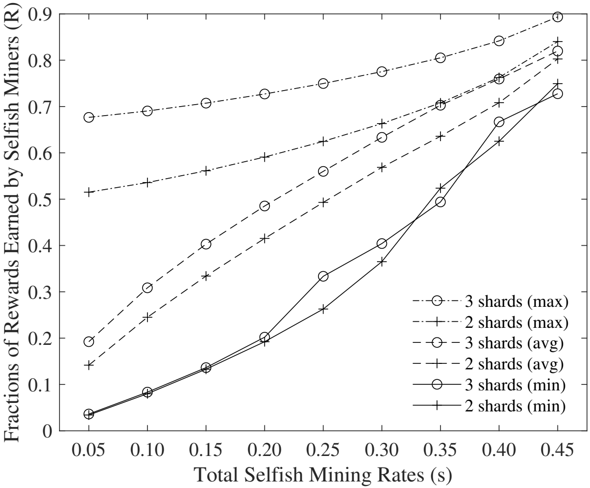
Rewards Analysis:

$$R_{shard} = \frac{1}{K}[m \times 1 + 1 \times RW(\frac{s_{m+1}}{s_{m+1} + h_{m+1}})]$$

Numerical Results: Profitable Threshold & Number of Shards



Defending The Selfish Mining Attacks



Multiple Selfish Miners in A Blockchain

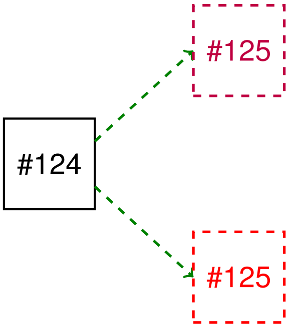
- S.W. Wang and Y.L. Tsou, ”Rewards Analysis in Proof-Of-Work Blockchains With Two Selfish Miners: From Network and Miner 's Perspectives,” in *2023 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom) (IEEE BlackSeaCom 2023)*, Istanbul, Turkey, 2023.

Multiple Selfish Miners

- Selfish mining strategy enables a miner to be profitable
- Multiple miners with sufficient mining rates may choose to employ selfish mining strategy in order to earn more rewards
- There will be multiple independent selfish miners in the blockchain without knowing each other

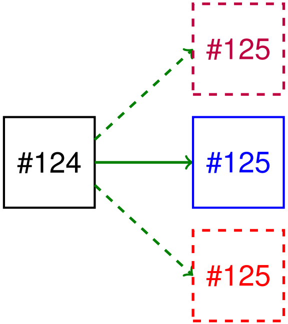
Two Selfish Miners (Case 1)

Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)



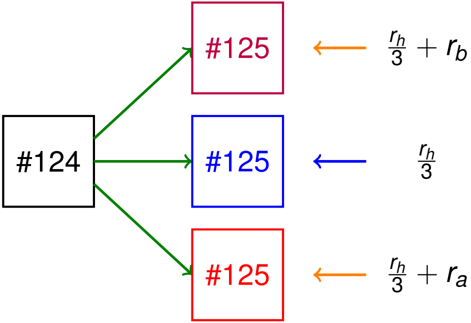
Two Selfish Miners (Case 1)

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



Two Selfish Miners (Case 1)

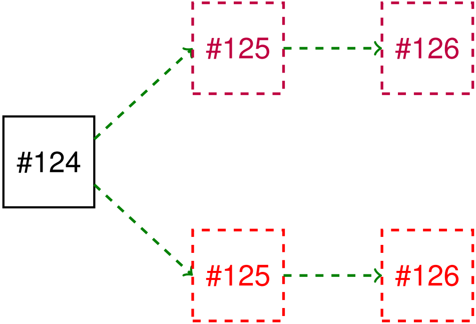
Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



Longest chain rule shall be applied.

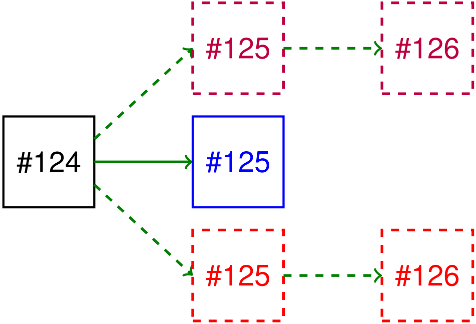
Two Selfish Miners (Case 2)

Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)



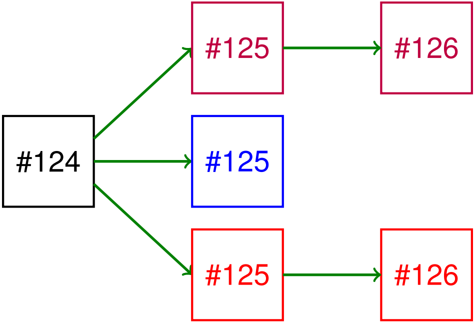
Two Selfish Miners (Case 2)

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



Two Selfish Miners (Case 2)

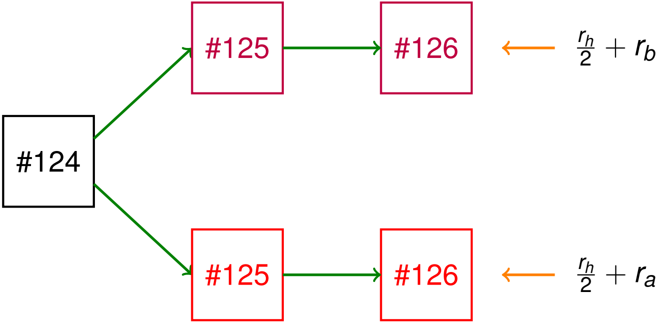
Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



Longest chain rule shall be applied.

Two Selfish Miners (Case 2)

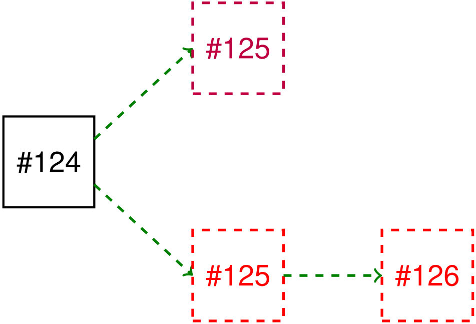
Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)



Longest chain rule shall be applied.

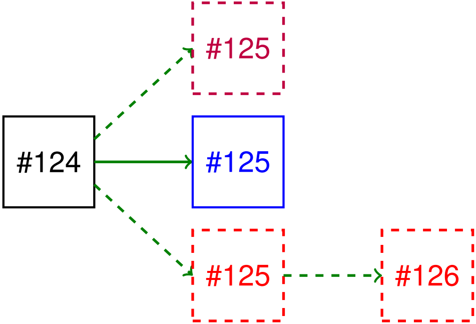
Two Selfish Miners (Case 3)

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



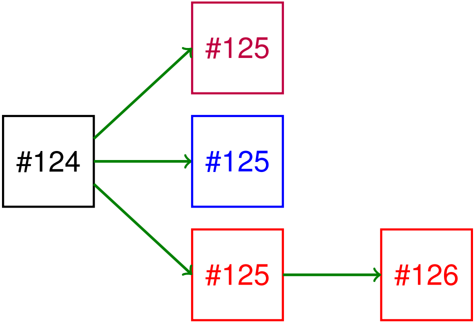
Two Selfish Miners (Case 3)

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



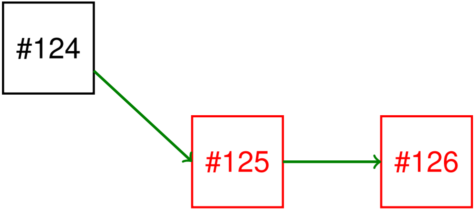
Two Selfish Miners (Case 3)

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



Two Selfish Miners (Case 3)

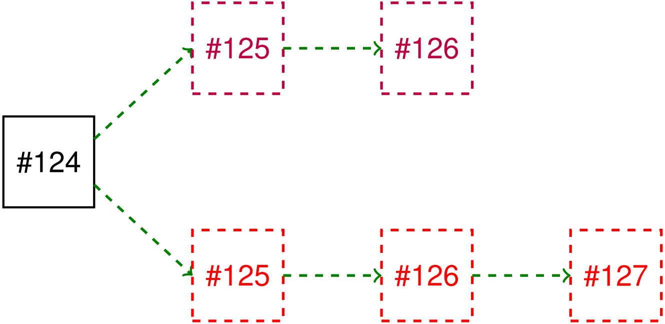
Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



Longest chain rule shall be applied.

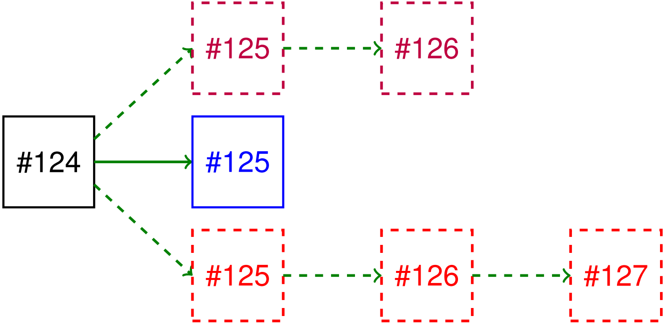
Two Selfish Miners (Case 4)

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



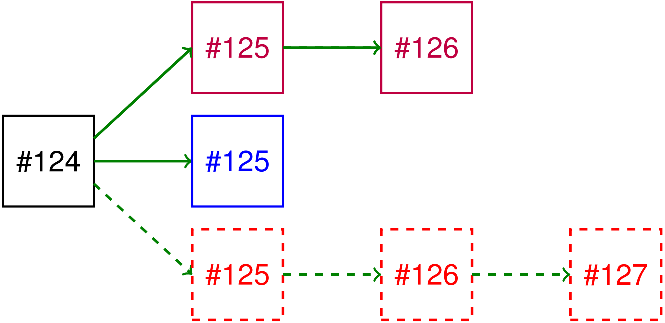
Two Selfish Miners (Case 4)

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



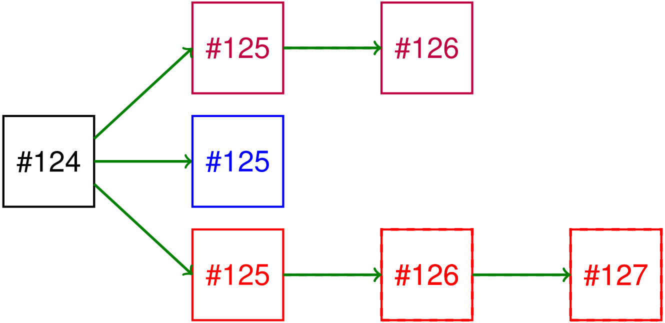
Two Selfish Miners (Case 4)

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



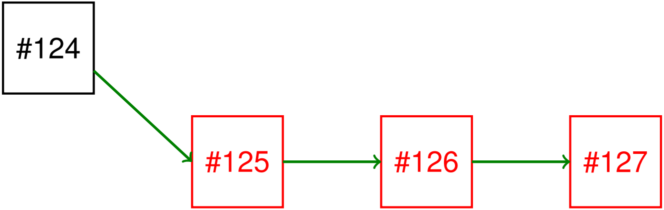
Two Selfish Miners (Case 4)

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



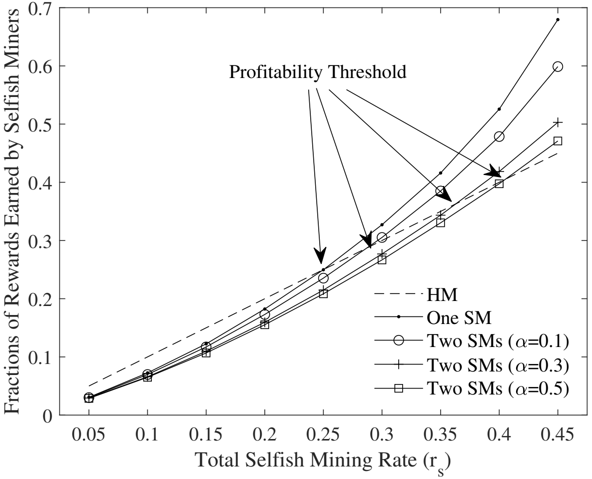
Two Selfish Miners (Case 4)

Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)

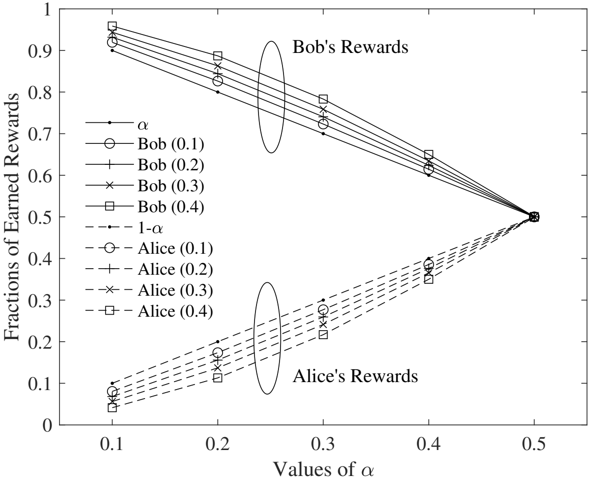


Longest chain rule shall be applied.

Simulations: Rewards Earned by Multiple Selfish Miners



Simulations: Rewards Earned by Strong/Weak Selfish Miners



An Accurate Analytical Model

- S.W. Wang and S.S. Tseng, ”An Accurate Analytical Model for A Proof-of-Work Blockchain with Multiple Selfish Miners,” Submitted to *2024 IEEE International Conference on Communications (ICC) (IEEE ICC 2024)*, Denver, USA, 2024.

Motivations for Analytical Model

The problem

Can we efficiently and accurately calculate the reward earned by each miner in a blockchain with multiple selfish miners?

- Using simulations is time consuming and lacks of theoretical contributions
- An analytical model, especially the derived closed-form expressions, to calculate the rewards earned by different miners is much more desirable

First Analytical Model Proposed by *Q. Bai*, and *et al.*

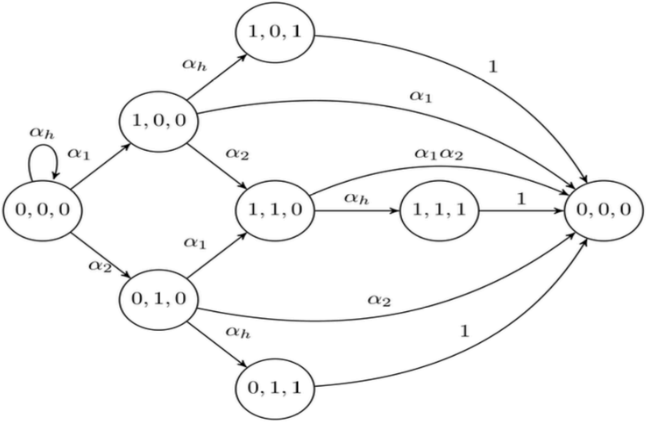


Fig. 5: State machine with N=2.

Our Model: End-of-Selfish (ES) and In-Selfish (IS) States

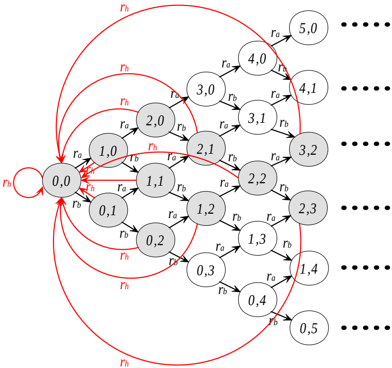


Figure 1: Exact Model

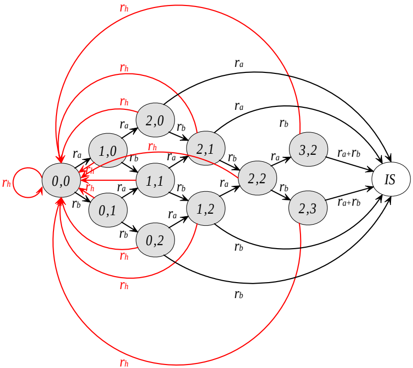
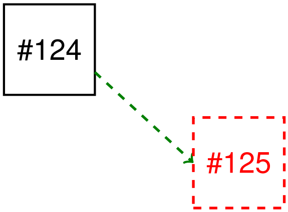
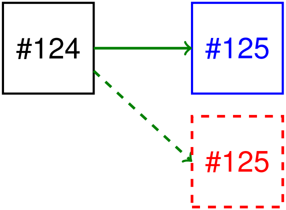


Figure 2: Approximate Model

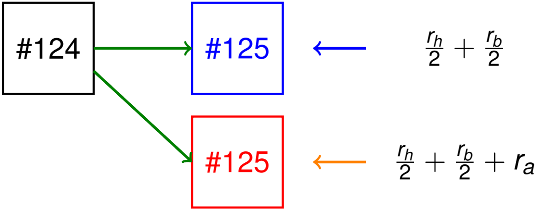
State $(0, 1)$ and $(1, 0)$



State $(0,1)$ and $(1,0)$

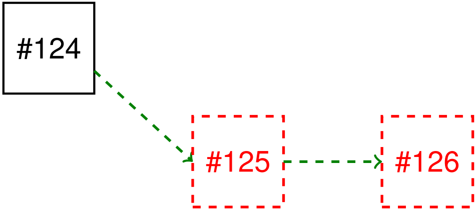


State $(0,1)$ and $(1,0)$

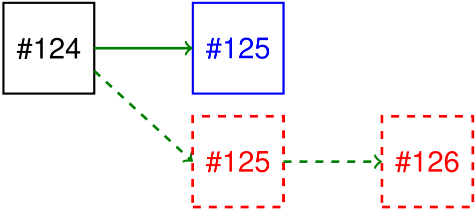


State s	Alice $R_a(s)$	Bob $R_b(s)$	Henry $R_h(s)$
(1,0)	$2r_a + r_b/2 + r_h/2$	r_b	$r_h + r_b/2$
(0,1)	r_a	$r_a/2 + 2r_b + r_h/2$	$r_h + r_b/2$

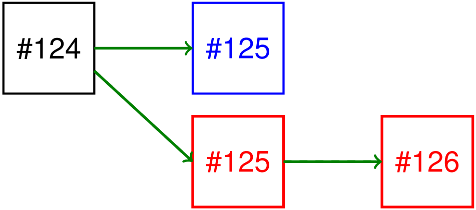
State $(0, 2)$ and $(2, 0)$



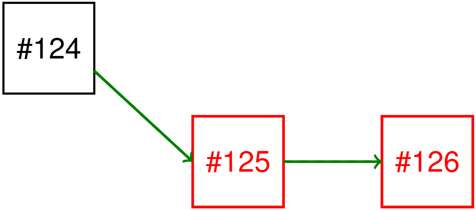
State $(0, 2)$ and $(2, 0)$



State $(0, 2)$ and $(2, 0)$



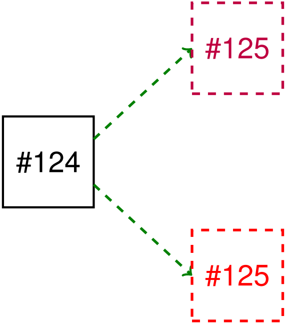
State $(0, 2)$ and $(2, 0)$



State s	Alice $R_a(s)$	Bob $R_b(s)$	Henry $R_h(s)$
(2,0)	2	0	0
(0,2)	0	2	0

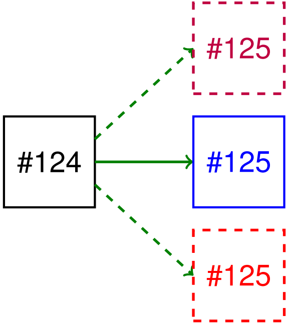
State $(1, 1)$

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



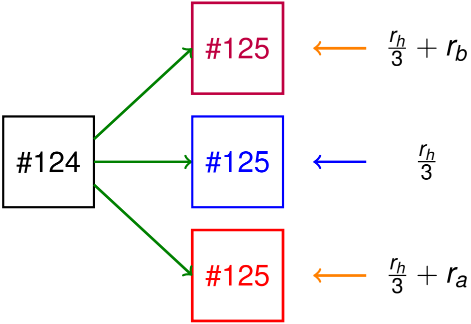
State $(1, 1)$

Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)



State $(1, 1)$

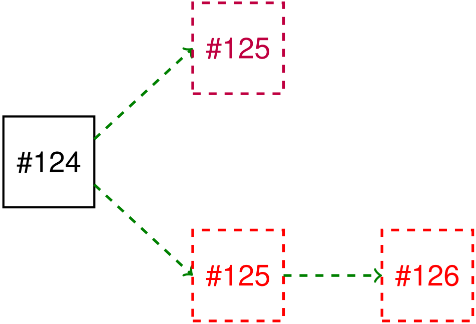
Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)



State s	Alice $R_a(s)$	Bob $R_b(s)$	Henry $R_h(s)$
(1,1)	$2r_a + r_h/3$	$2r_b + r_h/3$	$4r_h/3$

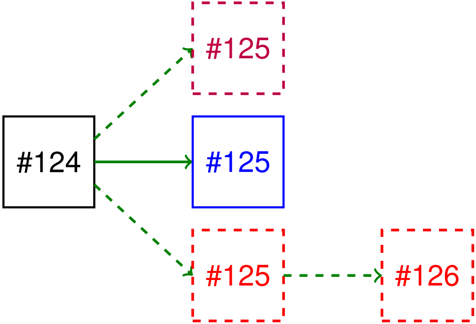
State $(1,2)$ and $(2,1)$

Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)



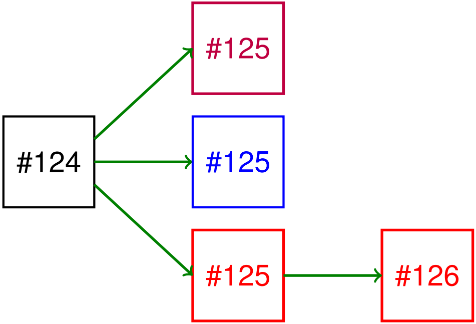
State $(1, 2)$ and $(2, 1)$

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



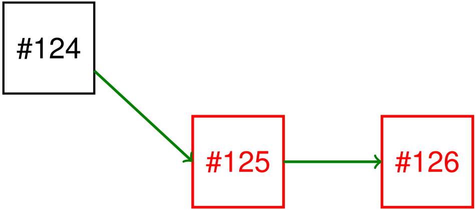
State $(1, 2)$ and $(2, 1)$

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



State $(1,2)$ and $(2,1)$

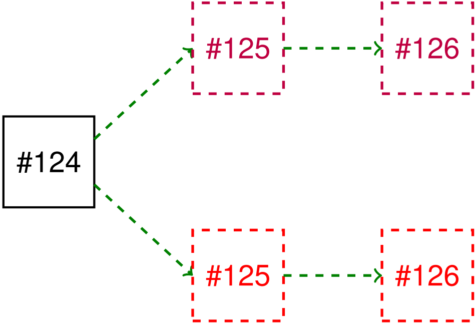
Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



State s	Alice $R_a(s)$	Bob $R_b(s)$	Henry $R_h(s)$
(2,1)	2	0	0
(1,2)	0	2	0

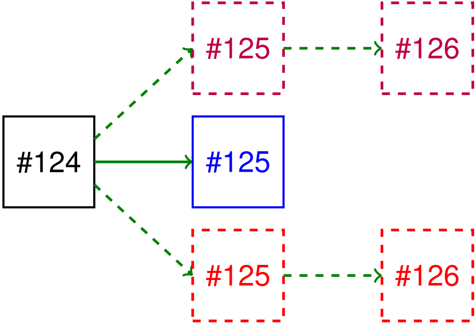
State $(2, 2)$

Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)



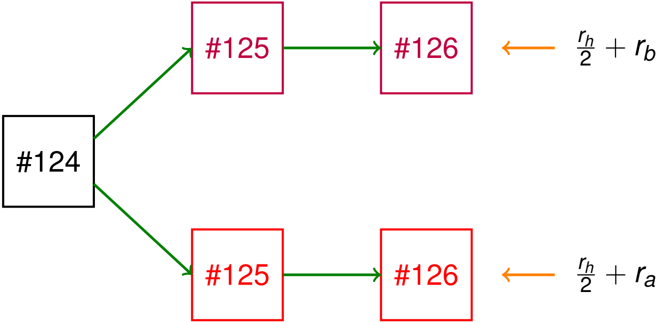
State $(2, 2)$

Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)



State $(2, 2)$

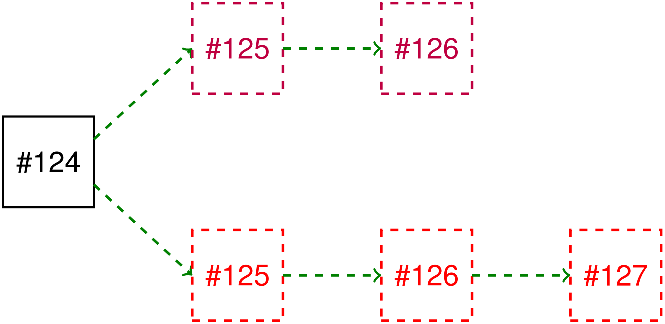
Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)



State s	Alice $R_a(s)$	Bob $R_b(s)$	Henry $R_h(s)$
(2,2)	$3r_a + r_h$	$3r_b + r_h$	r_h

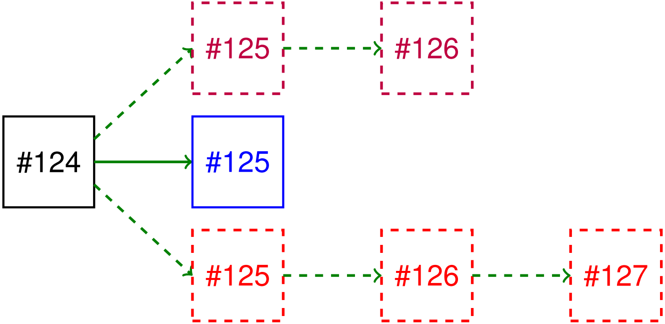
State $(2, 3)$ and $(3, 2)$

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



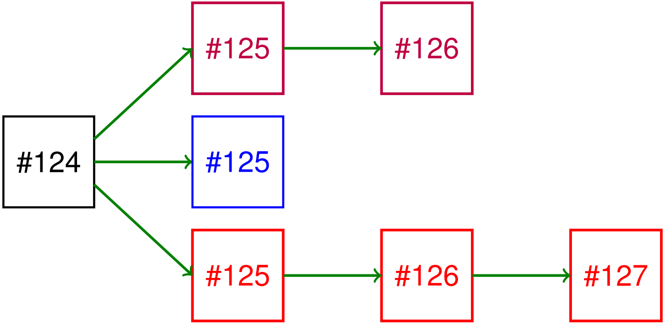
State $(2, 3)$ and $(3, 2)$

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



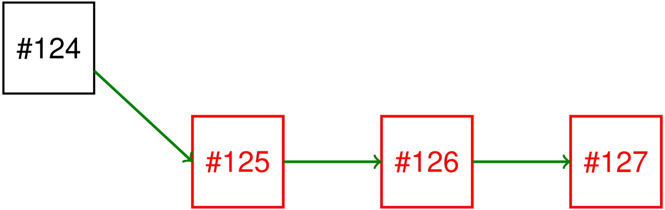
State $(2, 3)$ and $(3, 2)$

Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



State $(2,3)$ and $(3,2)$

Honest miner *Henry* (r_h) and Selfish Miners *Alice*(r_a) and *Bob*(r_b)



State s	Alice $R_a(s)$	Bob $R_b(s)$	Henry $R_h(s)$
(3,2)	3	0	0
(2,3)	0	3	0

IS State

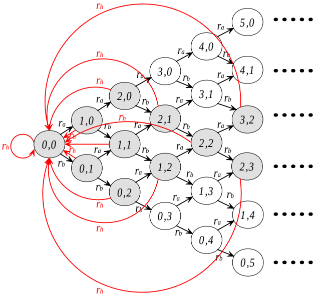


Figure 3: Exact Model

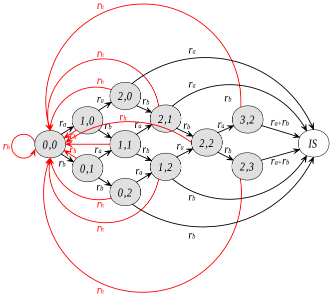


Figure 4: Approximate Model

State s	Alice $R_a(s)$	Bob $R_b(s)$	Henry $R_h(s)$
IS	$3r_a^3/(r_a^3+r_b^3)$	$3r_b^3/(r_a^3+r_b^3)$	0

Our Model: Expected Earned Rewards

State s	Alice $R_a(s)$	Bob $R_b(s)$	Henry $R_h(s)$
(0,0)	0	0	1
(1,0)	$2r_a + r_b/2 + r_h/2$	r_b	$r_h + r_b/2$
(0,1)	r_a	$r_a/2 + 2r_b + r_h/2$	$r_h + r_b/2$
(2,0)	2	0	0
(0,2)	0	2	0
(1,1)	$2r_a + r_h/3$	$2r_b + r_h/3$	$4r_h/3$
(2,1)	2	0	0
(1,2)	0	2	0
(2,2)	$3r_a + r_h$	$3r_b + r_h$	r_h
(3,2)	3	0	0
(2,3)	0	3	0
IS	$3{r_a}^3/({r_a}^3 + {r_b}^3)$	$3{r_b}^3/({r_a}^3 + {r_b}^3)$	0

Our Model: Steady-State Probability

- Let π_{n_a,n_b} be the steady-state probability of state (n_a, n_b).

$$\pi_{n_a,n_b} = \binom{n_a + n_b}{n_b} {r_a}^{n_a} {r_b}^{n_b} \pi_{0,0}$$

- π_{IS} can be calculated as follows.

$$\pi_{IS} = r_a(\pi_{2,0} + \pi_{2,1} + \pi_{3,2} + \pi_{2,3}) + r_b(\pi_{0,2} + \pi_{1,2} + \pi_{3,2} + \pi_{2,3})$$

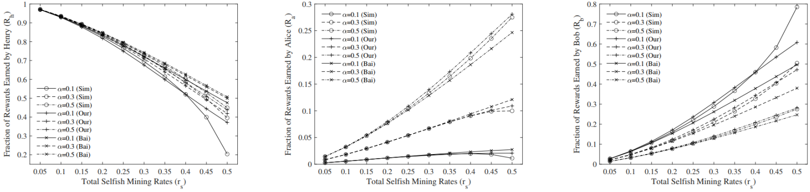
- Sum of the steady-state probabilities equals to 1.

$$\pi_{IS} + \sum_{s \in ES} \pi_s = 1 \quad (1)$$

where $\pi_{0,0}$ can be easily obtained.

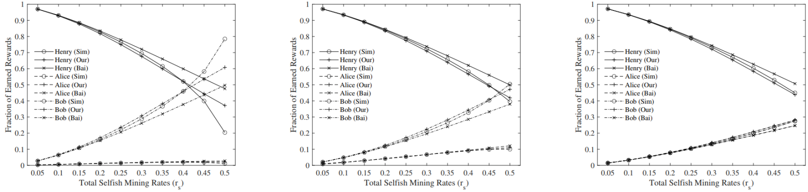
- The steady-state probability and expected earned rewards can be expressed in a closed-form of r_a, r_b, and r_h.

Numerical Results



(a) Henry's Rewards (b) Alice's Rewards (c) Bob's Rewards

Fig. 3: Fractions of rewards earned by Henry, Alice, and Bob



(a) $\alpha = 0.1$ (b) $\alpha = 0.3$ (c) $\alpha = 0.5$

Fig. 4: Fractions of rewards earned with different values of α

Our Model: Extension to Multiple Selfish Miners

Step. 1 Use n-tuple states to describe the blockchain with n selfish miners

Step. 2 Identify the ES states and IS states

Step. 3 For each ES states, calculate the expected rewards

Step. 4 For IS states, merge them into one single state and approximate the expected rewards

Step. 5 Calculate steady-state probability

Step. 6 Calculate the fractions of earned rewards

Rational Mining Strategy

- S.W. Wang, ”A Game Theory Based Rational Mining Strategy in Blockchains With Multiple Rational Miners,” in *2024 International Conference on Computing Networking and Communications (ICNC) (ICNC 2024)*, Big Island, Hawaii, USA, 2024.

Rational Miners

- If a miner is rational, he may choose honest rather than selfish mining strategy in order to earn more rewards if his mining rate is not large enough
- In a blockchain with a single rational miner and all others are honest miners, it has been shown that the miner can be profitable if the fraction of his mining rate is larger than 25%
- Rational Mining in a blockchain with a single rational miner:
  - fraction of mining rate > 0.25: selfish mining
  - fraction of mining rate < 0.25: honest mining

Rational Miners

- Blockchains with two (2) rational miners are investigated
- Analytical models are employed
- Two selfish miners *Alice* and *Bob* are independent without knowing each other
- Payoff matrices with mining rates between 0.1 and 0.5

Rewards (Alice, Bob)		Bob's Strategy: Honest	Bob's Strategy: Selfish
Alice's Strategy	Honest	R_a^{HH}, R_b^{HH}	R_a^{HS}, R_b^{HS}
	Selfish	R_a^{SH}, R_b^{SH}	R_a^{SS}, R_b^{SS}

Calculations of Earned Rewards

Rewards (Alice, Bob)		Bob's Strategy: Honest	Bob's Strategy: Selfish
Alice's Strategy	Honest	R_a^{HH}, R_b^{HH}	R_a^{HS}, R_b^{HS}
	Selfish	R_a^{SH}, R_b^{SH}	R_a^{SS}, R_b^{SS}

[1] Q. Bai, X. Zhou, X. Wang, Y. Xu, X. Wang and Q. Kong, "A Deep Dive Into Blockchain Selfish Mining," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, Shanghai, China, 2019, pp. 1-6.

Calculations of Earned Rewards

Rewards (Alice, Bob)		Bob's Strategy: Honest	Bob's Strategy: Selfish
Alice's Strategy	Honest	R_a^{HH}, R_b^{HH}	R_a^{HS}, R_b^{HS}
	Selfish	R_a^{SH}, R_b^{SH}	R_a^{SS}, R_b^{SS}

- R_a^{HH} and R_b^{HH}: Proportional to their mining rates

[1] Q. Bai, X. Zhou, X. Wang, Y. Xu, X. Wang and Q. Kong, "A Deep Dive Into Blockchain Selfish Mining," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, Shanghai, China, 2019, pp. 1-6.

Calculations of Earned Rewards

Rewards (Alice, Bob)		Bob's Strategy: Honest	Bob's Strategy: Selfish
Alice's Strategy	Honest	R_a^{HH}, R_b^{HH}	R_a^{HS}, R_b^{HS}
	Selfish	R_a^{SH}, R_b^{SH}	R_a^{SS}, R_b^{SS}

- R_a^{HH} and R_b^{HH}: Proportional to their mining rates
- R_a^{HS},R_b^{HS}, R_a^{SH},R_b^{SH}: Only one single selfish miner

[1] Q. Bai, X. Zhou, X. Wang, Y. Xu, X. Wang and Q. Kong, "A Deep Dive Into Blockchain Selfish Mining," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, Shanghai, China, 2019, pp. 1-6.

Calculations of Earned Rewards

Rewards (Alice, Bob)		Bob's Strategy Honest	Bob's Strategy Selfish
Alice's Strategy	Honest	R_a^{HH}, R_b^{HH}	R_a^{HS}, R_b^{HS}
	Selfish	R_a^{SH}, R_b^{SH}	R_a^{SS}, R_b^{SS}

- R_a^{HH} and R_b^{HH}: Proportional to their mining rates
- R_a^{HS},R_b^{HS}, R_a^{SH},R_b^{SH}: Only one single selfish miner
  - Selfish miner: Earns rewards by *RW* function

[1] Q. Bai, X. Zhou, X. Wang, Y. Xu, X. Wang and Q. Kong, "A Deep Dive Into Blockchain Selfish Mining," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, Shanghai, China, 2019, pp. 1-6.

Calculations of Earned Rewards

Rewards (Alice, Bob)		Bob's Strategy: Honest	Bob's Strategy: Selfish
Alice's Strategy	Honest	R_a^{HH}, R_b^{HH}	R_a^{HS}, R_b^{HS}
	Selfish	R_a^{SH}, R_b^{SH}	R_a^{SS}, R_b^{SS}

- R_a^{HH} and R_b^{HH}: Proportional to their mining rates
- R_a^{HS},R_b^{HS}, R_a^{SH},R_b^{SH}: Only one single selfish miner
  - Selfish miner: Earns rewards by *RW* function
  - Honest miner: Shares the remaining rewards with Henry

[1] Q. Bai, X. Zhou, X. Wang, Y. Xu, X. Wang and Q. Kong, "A Deep Dive Into Blockchain Selfish Mining," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, Shanghai, China, 2019, pp. 1-6.

Calculations of Earned Rewards

Rewards (Alice, Bob)		Bob's Strategy: Honest	Bob's Strategy: Selfish
Alice's Strategy	Honest	R_a^{HH}, R_b^{HH}	R_a^{HS}, R_b^{HS}
	Selfish	R_a^{SH}, R_b^{SH}	R_a^{SS}, R_b^{SS}

- R_a^{HH} and R_b^{HH}: Proportional to their mining rates
- R_a^{HS},R_b^{HS}, R_a^{SH},R_b^{SH}: Only one single selfish miner
  - Selfish miner: Earns rewards by *RW* function
  - Honest miner: Shares the remaining rewards with Henry
- R_a^{SS},R_b^{SS}: By an analytical model proposed by *Bai, et. al.*[1]

[1] Q. Bai, X. Zhou, X. Wang, Y. Xu, X. Wang and Q. Kong, "A Deep Dive Into Blockchain Selfish Mining," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, Shanghai, China, 2019, pp. 1-6.

Payoff Matrices

Payoff Matrices			Bob's Strategy							
		r_b	0.1		0.2		0.3		0.4	
	r_a		Honest	Selfish	Honest	Selfish	Honest	Selfish	Honest	Selfish
Alice's Strategy	0.1	Honest	**0.100,0.100**	**0.103**,0.072	**0.100,0.200**	**0.102**,0.182	**0.100**,0.300	**0.096,0.327**	**0.100**,0.400	**0.079,0.526**
		Selfish	0.072,**0.103**	0.078,0.078	0.072,**0.206**	0.079,0.199	0.072,0.309	0.068,**0.359**	0.072,0.412	0.054,**0.550**
	0.2	Honest	**0.200,0.100**	**0.206**,0.072	**0.200,0.200**	**0.204**,0.182	**0.200**,0.300	**0.192,0.327**	**0.200**,0.400	**0.158,0.526**
		Selfish	0.182,**0.102**	0.199,0.076	0.182,**0.204**	0.199,0.199	0.182,0.307	0.177,**0.369**	0.182,0.409	0.136,**0.574**
	0.3	Honest	0.300,**0.100**	0.309,0.072	0.300,**0.200**	0.307,0.182	0.300,0.300	0.289,**0.327**	0.300,0.400	0.237,**0.526**
		Selfish	**0.327,0.096**	**0.359**,0.068	**0.327,0.192**	**0.369**,0.177	**0.327**,0.289	**0.341,0.341**	**0.327**,0.385	**0.262,0.555**
	0.4	Honest	0.400,**0.100**	0.412,0.072	0.400,**0.200**	0.409,0.182	0.400,0.300	0.385,**0.327**	0.400,0.400	0.316,**0.526**
		Selfish	**0.526,0.079**	**0.550**,0.054	**0.526,0.158**	**0.574**,0.136	**0.526**,0.237	**0.555,0.262**	**0.526**,0.316	**0.455,0.455**

- All above payoff matrices have only one Nash equilibrium
- Find more details when mining rate is between 0.2 and 0.3

Payoff Matrices

Payoff Matrices			Bob's Strategy							
		r_b	0.20		0.21		0.22		0.23	
	r_a		Honest	Selfish	Honest	Selfish	Honest	Selfish	Honest	Selfish
Alice's Strategy	0.20	Honest	**0.200,0.200**	**0.204**,0.182	**0.200,0.210**	**0.204**,0.195	**0.200,0.220**	**0.203**,0.209	**0.200,0.230**	**0.202**,0.222
		Selfish	0.182,**0.204**	0.199,0.199	0.182,**0.215**	0.198,0.214	0.182,0.225	0.198,**0.230**	0.182,0.235	0.196,**0.246**
	0.21	Honest	**0.210,0.200**	**0.215**,0.182	**0.210,0.210**	**0.214**,0.195	**0.210,0.220**	**0.213**,0.209	**0.210,0.230**	**0.212**,0.222
		Selfish	0.195,**0.204**	0.214,0.198	0.195,**0.214**	0.213,0.213	0.195,0.224	0.212,**0.229**	0.195,0.234	0.211,**0.245**
	0.22	Honest	**0.220,0.200**	0.225,0.182	**0.220,0.210**	0.224,0.195	**0.220,0.220**	0.223,0.209	**0.220,0.230**	0.222,0.222
		Selfish	0.209,**0.203**	**0.229**,0.197	0.209,**0.213**	**0.229**,0.212	0.208,0.223	**0.228,0.228**	0.208,0.233	**0.227,0.245**
	0.23	Honest	**0.230,0.200**	0.235,0.182	**0.230,0.210**	0.234,0.195	**0.230,0.220**	0.233,0.209	**0.230,0.230**	0.232,0.222
		Selfish	0.222,**0.202**	**0.246**,0.196	0.222,**0.212**	**0.245**,0.211	0.222,0.222	**0.244,0.227**	0.222,0.232	**0.243,0.243**
	0.24	Honest	**0.240,0.200**	0.245,0.182	**0.240,0.210**	0.244,0.195	**0.240,0.220**	0.244,0.209	**0.240,0.230**	0.243,0.222
		Selfish	0.236,**0.201**	**0.263**,0.195	0.236,**0.211**	**0.262**,0.210	0.236,0.221	**0.262,0.225**	0.236,0.231	**0.261,0.242**
	0.25	Honest	**0.250,0.200**	0.256,0.182	**0.250,0.210**	0.255,0.195	**0.250,0.220**	0.254,0.209	**0.250,0.230**	0.253,0.222
		Selfish	**0.250,0.200**	**0.281**,0.193	**0.250,0.210**	**0.281**,0.208	**0.250**,0.220	**0.280,0.223**	**0.250**,0.230	**0.279,0.240**
	0.26	Honest	0.260,**0.200**	0.266,0.182	0.260,**0.210**	0.265,0.195	0.260,**0.220**	0.264,0.209	0.260,**0.230**	0.263,0.222
		Selfish	**0.265,0.299**	**0.300**,0.191	**0.265,0.209**	**0.300**,0.205	**0.265**,0.219	**0.299,0.221**	**0.265**,0.229	**0.299,0.237**
	0.27	Honest	0.270,**0.200**	0.276,0.182	0.270,**0.210**	0.275,0.195	0.270,**0.220**	0.274,0.209	0.270,**0.230**	0.273,0.222
		Selfish	**0.280,0.197**	**0.319**,0.189	**0.280,0.207**	**0.320**,0.203	**0.280**,0.217	**0.320,0.218**	**0.280**,0.227	**0.320,0.234**

Payoff Matrices

Payoff Matrices		r_b	0.24		0.25		0.26		0.27	
	r_a		Honest	Selfish	Honest	Selfish	Honest	Selfish	Honest	Selfish
Alice's Strategy	0.20	Honest	**0.200**,**0.240**	**0.201**,0.236	**0.200**,**0.250**	**0.200**,**0.250**	**0.200**,0.260	**0.199**,**0.265**	**0.200**,0.270	**0.197**,**0.280**
		Selfish	0.182,0.245	0.195,**0.263**	0.182,0.256	0.193,**0.281**	0.182,0.266	0.191,**0.300**	0.182,0.279	0.188,**0.319**
	0.21	Honest	**0.210**,**0.240**	**0.211**,0.236	**0.210**,**0.250**	**0.210**,**0.250**	**0.210**,0.260	**0.209**,**0.265**	**0.210**,0.270	**0.207**,**0.280**
		Selfish	0.195,0.244	0.210,**0.263**	0.195,0.255	0.208,**0.281**	0.195,0.265	0.206,**0.300**	0.195,0.275	0.203,**0.320**
	0.22	Honest	**0.220**,**0.240**	0.221,0.236	**0.220**,**0.250**	0.220,**0.250**	**0.220**,0.260	0.219,**0.265**	**0.220**,0.270	0.217,**0.280**
		Selfish	0.209,0.244	**0.225**,**0.262**	0.209,0.254	**0.224**,**0.280**	0.209,0.264	**0.221**,**0.299**	0.209,0.274	**0.218**,**0.320**
	0.23	Honest	**0.230**,**0.240**	0.231,0.236	**0.230**,**0.250**	0.230,**0.250**	**0.230**,0.260	0.229,**0.265**	**0.230**,0.270	0.227,**0.280**
		Selfish	0.222,0.243	**0.242**,**0.261**	0.222,0.253	**0.240**,**0.279**	0.222,0.263	**0.238**,**0.299**	0.222,0.273	**0.234**,**0.320**
	0.24	Honest	**0.240**,**0.240**	0.241,0.236	**0.240**,**0.250**	0.240,**0.250**	**0.240**,0.260	0.238,**0.265**	**0.240**,0.270	0.237,**0.280**
		Selfish	0.236,0.241	**0.259**,**0.250**	0.236,0.251	**0.257**,**0.278**	0.236,0.261	**0.255**,**0.297**	0.236,0.272	**0.252**,**0.319**
	0.25	Honest	**0.250**,**0.240**	0.251,0.236	**0.250**,**0.250**	0.250,**0.250**	**0.250**,0.260	0.248,**0.265**	**0.250**,0.270	0.247,**0.280**
		Selfish	**0.250**,0.240	**0.278**,**0.257**	**0.250**,0,250	**0.276**,**0,276**	**0.250**,0.260	**0.274**,**0.295**	**0.250**,0.270	**0.270**,**0.318**
	0.26	Honest	0.260,**0.240**	0.261,0.236	0.260,**0.250**	0.260,**0.250**	0.260,0.260	0.258,**0.265**	0.260,0.270	0.257,**0.280**
		Selfish	**0.265**,0.238	**0.298**,**0.255**	**0.265**,0.248	**0.296**,**0.274**	**0.265**,0.258	**0.293**,**0.293**	**0.265**,0.268	**0.290**,**0.315**
	0.27	Honest	0.270,**0.240**	0.272,0.236	0.270,**0.250**	0.270,**0.250**	0.270,0.260	0.268,**0.265**	0.270,0.270	0.266,**0.280**
		Selfish	**0.280**,0.237	**0.319**,**0.252**	**0.280**,0.247	**0.317**,**0.270**	**0.280**,0.257	**0.315**,**0.290**	**0.280**,0.266	**0.312**,**0.312**

Bob's Strategy spans columns 0.24–0.27.

Payoff Matrices: Two miners both have dominant strategies

Rewards (Alice, Bob)		Bob $r_b = 0.1$ Honest	Bob $r_b = 0.1$ Selfish
Alice $r_a = 0.2$	Honest	**0.200**,**0.100**	**0.206**,0.072
	Selfish	0.182,**0.102**	0.199,0.076

Rewards (Alice, Bob)		Bob $r_b = 0.4$ Honest	Bob $r_b = 0.4$ Selfish
Alice $r_a = 0.3$	Honest	0.300,0.400	0.237,**0.526**
	Selfish	**0.327**,0.385	**0.262**,**0.555**

- Mining rate > 0.25: Selfish mining strategy
- Mining rate < 0.22: Honest mining strategy

Payoff Matrices: Only one miner has dominant strategy

Rewards (Alice, Bob)		Bob $r_b = 0.21$ Honest	Bob $r_b = 0.21$ Selfish
Alice $r_a = 0.24$	Honest	**0.240**,**0.210**	0.244,0.195
	Selfish	0.236,**0.211**	**0.262**,0.210

Rewards (Alice, Bob)		Bob $r_b = 0.27$ Honest	Bob $r_b = 0.27$ Selfish
Alice $r_a = 0.23$	Honest	**0.230**,0.270	0.227,**0.280**
	Selfish	0.222,0.273	**0.234**,**0.320**

- Mining rate between $[0.22, 0.25]$: Follow the other rational miner's strategy if he has dominant strategy

Payoff Matrices: No miner has dominant strategy

Rewards (Alice, Bob)		Bob $r_b = 0.24$ Honest	Bob $r_b = 0.24$ Selfish
Alice $r_a = 0.23$	Honest	**0.230,0.240**	0.231,0.236
	Selfish	0.222,0.243	**0.242,0.261**

- Two Nash Equilibria exist
- Mixed strategy can be applied
- Select a strategy according to a probability distribution

Mixed Strategy

Rewards (Alice, Bob)		Bob's Strategy	
		Honest(q)	Selfish $(1-q)$
Alice's Strategy	Honest (p)	R_a^{HH}, R_b^{HH}	R_a^{HS}, R_b^{HS}
	Selfish $(1-p)$	R_a^{SH}, R_b^{SH}	R_a^{SS}, R_b^{SS}

Main idea: to make the other miner earn *indifferent* rewards no matter which strategy the other miner uses.

- Using honest mining, Bob earns $p \times R_b^{HH} + (1-p) \times R_b^{SH}$.
- Using selfish mining, Bob earns $p \times R_b^{HS} + (1-p) \times R_b^{SS}$.
- Solve equation $p \times R_b^{HH} + (1-p) \times R_b^{SH} = p \times R_b^{HS} + (1-p) \times R_b^{SS}$

Mixed Strategy

Rewards (Alice, Bob)		Bob's Strategy	
		Honest(q)	Selfish ($1-q$)
Alice's Strategy	Honest (p)	R_a^{HH}, R_b^{HH}	R_a^{HS}, R_b^{HS}
	Selfish ($1-p$)	R_a^{SH}, R_b^{SH}	R_a^{SS}, R_b^{SS}

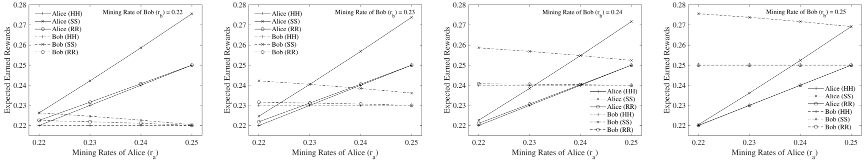
Main idea: to make the other miner earn *indifferent* rewards no matter which strategy the other miner uses.

- $p = \frac{R_b^{SS} - R_b^{SH}}{R_b^{HH} + R_b^{SS} - R_b^{SH} - R_b^{HS}}$
- $q = \frac{R_a^{SS} - R_a^{SH}}{R_a^{HH} + R_a^{SS} - R_a^{SH} - R_a^{HS}}$

Rational Mining Strategy with Two Rational Miners

- If mining rate is < 0.22, use Honest Mining
- If mining rate is > 0.25, use Selfish Mining
- If mining rate ranges from 0.22 to 0.25,
  - If the other miner has dominant strategy, follow his dominant mining strategy
  - If the other miner has no dominant strategy, solve the payoff matrices according to the probability distribution

Numerical Results



(a) $r_b = 0.22$ (b) $r_b = 0.23$ (c) $r_b = 0.24$ (d) $r_b = 0.25$

Fig. 1: Expected rewards earned by Alice and Bob under both honest, selfish, or rational mining strategies

- Both honest (HH) $\leq$ both rational (RR) $\leq$ both selfish (SS)
- Mixed strategy performs close to honest strategy

Ongoing & Future Works

Ongoing Works

- Find optimal number of shards in a sharded blockchain to defend the selfish mining attacks
- Using the analytical model to explain the observations in blockchains with two selfish miners
- Verify the accuracy of the proposed analytical mode in blockchains with more than two selfish miners
- Solve the game with two Nash equilibria (Stag Hunt game)

Ongoing Works

- Find optimal number of shards in a sharded blockchain to defend the selfish mining attacks
- Using the analytical model to explain the observations in blockchains with two selfish miners
- Verify the accuracy of the proposed analytical mode in blockchains with more than two selfish miners
- Solve the game with two Nash equilibria (Stag Hunt game)

Thanks for your listening!