Reward Analysis in Proof-of-Work Blockchains with Multiple Selfish Miners

Sheng-Wei Wang

Department of Electronic Engineering
National United University
Miaoli, TAIWAN

swwang@nuu.edu.tw

2025-11-20 @ CSE.NTOU

About Me

- Education
  - B.S. in Information Management, National Taiwan University
  - Ph.D. in Communications Engineering, National Tsing Hua University
- Experiences
  - Dept. of Applied Informatics, Fo Guang University, Yilan
  - Dept. of Computer Science and Information Engineering, Tamkang University, New Taipei City
  - Dept. of Electronic Engineering, National United University, Miaoli
- Research Interests
  - Optical Networks
  - Blockchain and Distributed Ledger Technology (DLT)
  - Byzantine Fault Tolerance (BFT) Protocols

Outlines

1. **Introductions**
   - Blockchains
   - Selfish Mining Strategy
     - One Selfish Miner
     - Multiple Selfish Mining Strategy
2. **Research Works on Multiple Selfish Miners**
   - Simulation Based Observations (ICBC2024)
   - An Accurate Analytical Model (ICC2024)
   - Multiple Selfish Miners Extension (IEEE TNSM)
   - Our Works and Conclusions

Bitcoins

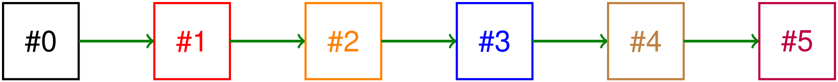
Bitcoin: A Peer-to-Peer Electronic Cash System

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 blocks
- Consecutive blocks form a blockchain using cryptography
- Hash value of previous block is stored



Mining & Miners

- Node creating the block earns rewards (Miner)
- Mining is the process to create a valid block in order to get rewards

Let's talk about mining.....

Mining & Proof-of-Works

- Who will earn the reward?

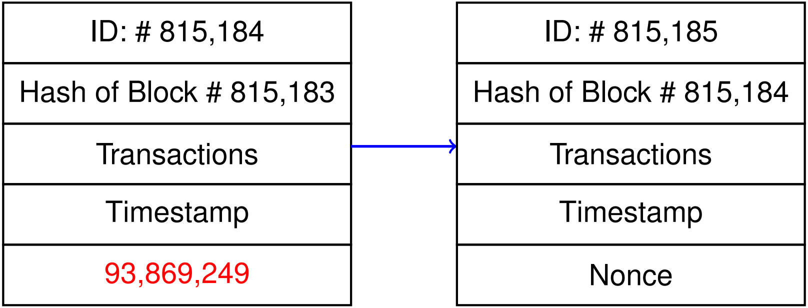
ID: # 815,184
Hash of Block # 815,183
Transactions
Timestamp
Nonce

- Hash of Block #815,183:

  00000000000000000003d09220e85bbdbb832b86e3dc711c5cda888b1daf5985

Mining & Proof-of-Works

- Who will earn the reward?



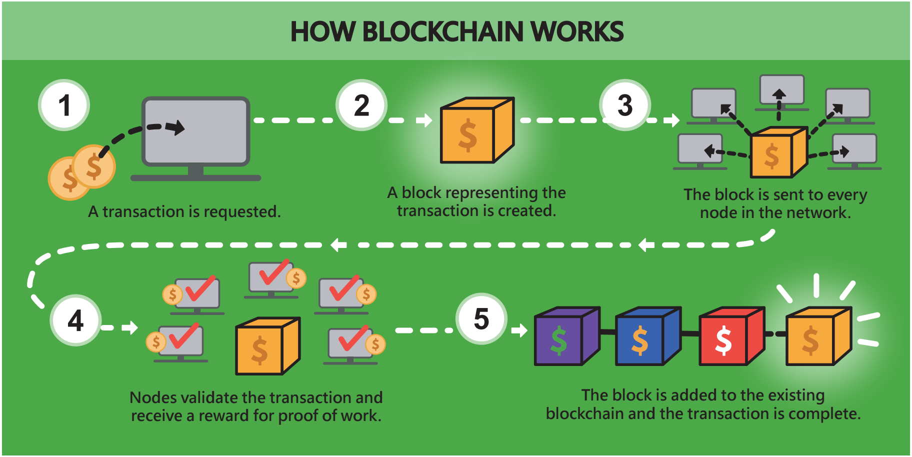
- Hash of Block #815,183:

  00000000000000000003d09220e85bbdbb832b86e3dc711c5cda888b1daf5985

- Hash of Block #815,184:

  00000000000000000000cc167c107a24883b34c16aad188aaa72412cc0ef437a

How A PoW Blockchain Works?



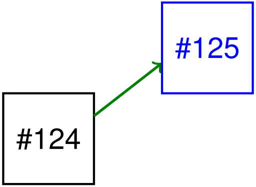
Selfish Mining & Rewards

- How many rewards can a miner earn?
- The number of nonces attempted by a miner 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

Selfish Mining & Rewards

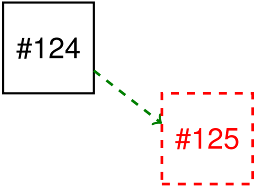
#124

Selfish Mining & Rewards



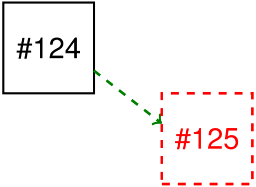
- If honest miner mined the next block first, he announces the block immediately
- All other miners validate the block and start to mine the next one

Selfish Mining & Rewards



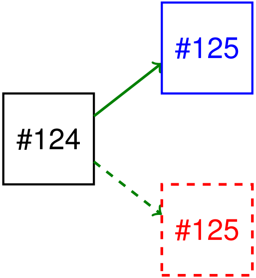
- If honest miner mined the next block first, he announces the block immediately
- All other miners validate the block and start to mine the next one
- If selfish miner mined the next block first, he hides the block in his private branch and starts to mine the next one

Selfish Mining & Rewards



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

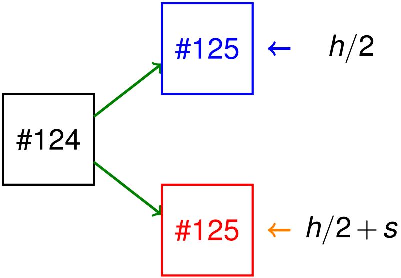
Selfish Mining & Rewards



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

- The honest miner announces the block immediately

Selfish Mining & Rewards

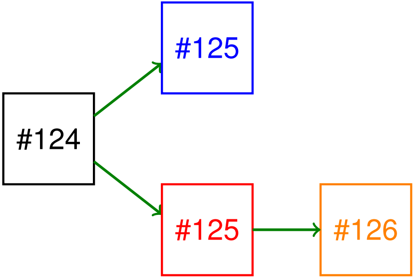


If then honest miner mined the next block first under the above condition:

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

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

Selfish Mining & Rewards

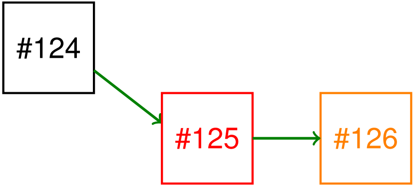


If then honest miner mined the next block first under the above condition:

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

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

Selfish Mining & Rewards

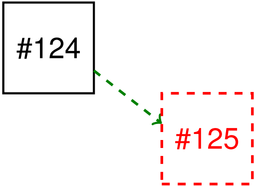


If then honest miner mined the next block first under the above condition:

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

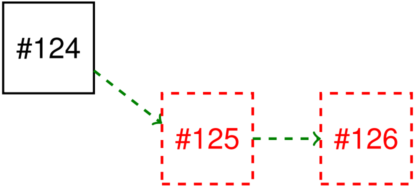
Longest Chain Rule: The branch on which the next block is mined first (longest chain) becomes the valid chain

Selfish Mining & Rewards



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

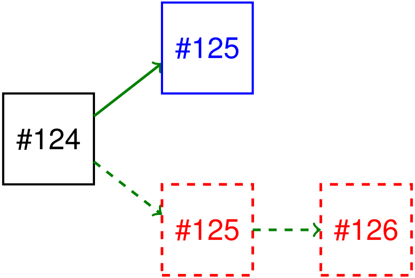
Selfish Mining & Rewards



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

- The selfish miner hides the blocks in his branch

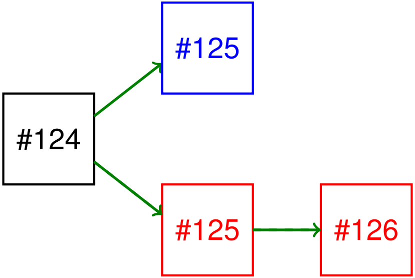
Selfish Mining & Rewards



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

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

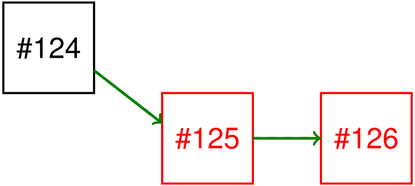
Selfish Mining & Rewards



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

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

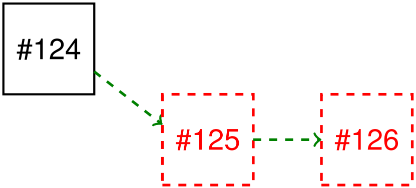
Selfish Mining & Rewards



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

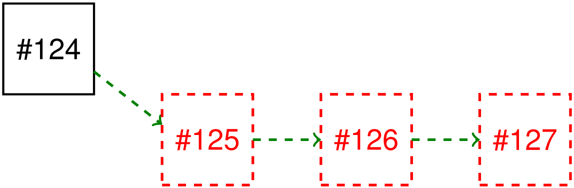
- The selfish miner hides the blocks in his branch
- If the honest miner mined the block now, the honest miner releases the block immediately
- The selfish miner releases his all blocks immediately
- Longest chain rule is applied

Selfish Mining & Rewards



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

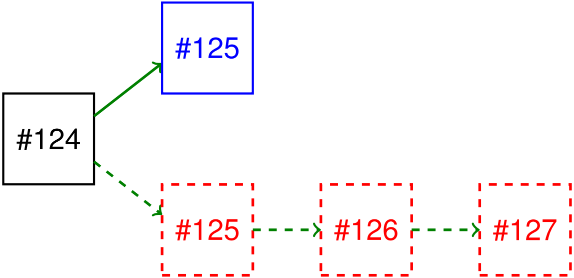
Selfish Mining & Rewards



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

- The selfish miner hides the blocks in his branch

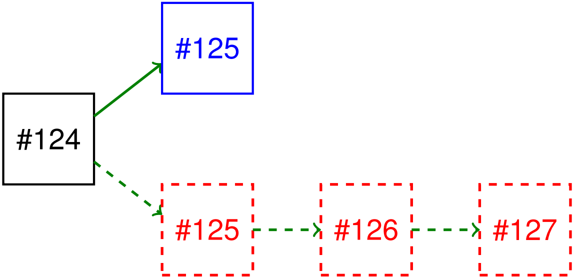
Selfish Mining & Rewards



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

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

Selfish Mining & Rewards



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

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

Reward Earned by Selfish Miner

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

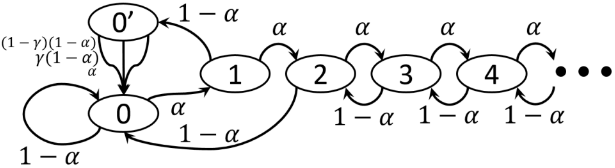
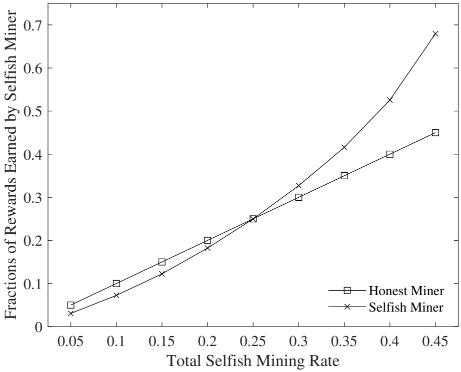


Fig. 1: State machine with transition frequencies.

- Fraction of reward earned by the selfish miner $RW(\alpha)$ can be calculated by a closed-form function of his mining rate α

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

Rewards & Profitable Threshold (25%)



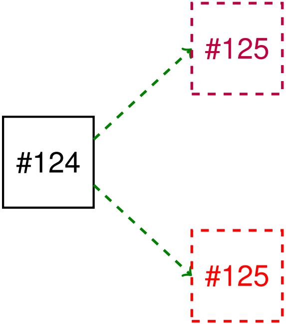
- Profitable: Earns more than those earned if he is honest
- Profitable threshold: the smallest mining rate making a miner profitable

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
- We consider a blockchain with TWO selfish miners first

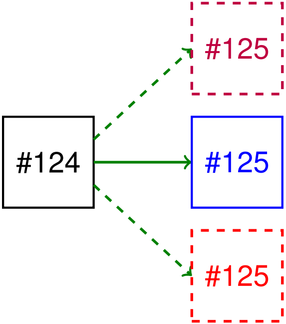
Two Selfish Miners (Case 1)

An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



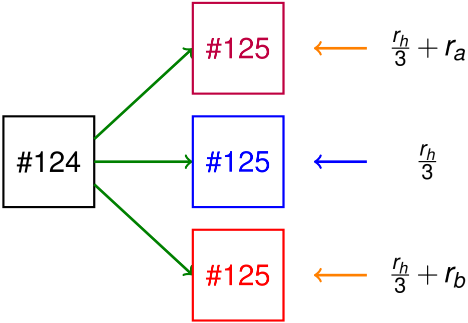
Two Selfish Miners (Case 1)

An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



Two Selfish Miners (Case 1)

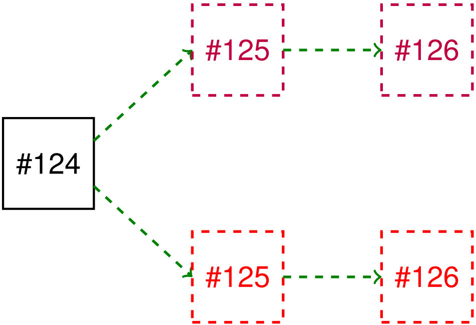
An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



Longest chain rule shall be applied.

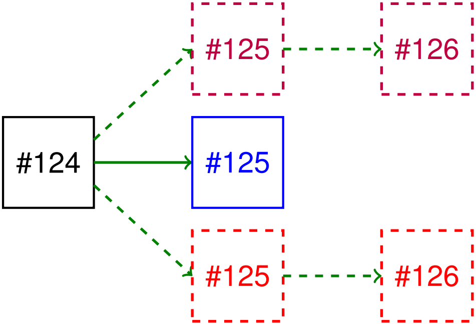
Two Selfish Miners (Case 2)

An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



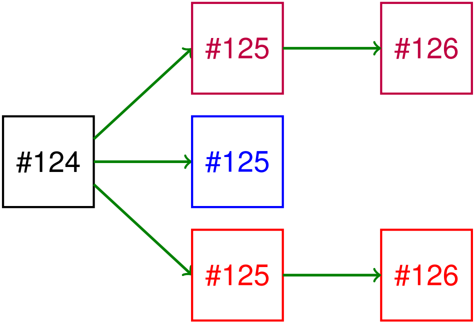
Two Selfish Miners (Case 2)

An honest miner *Henry* (r_h) and two selfish miners *Alice*(r_a) and *Bob*(r_b)



Two Selfish Miners (Case 2)

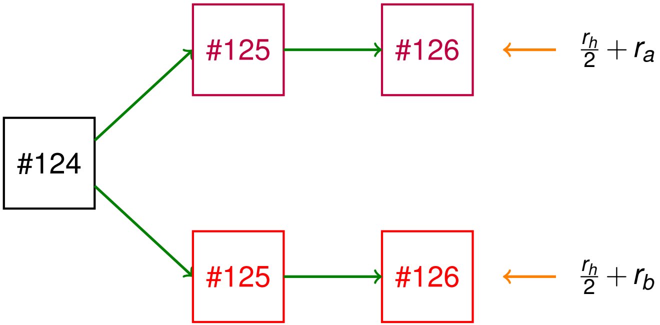
An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



Longest chain rule shall be applied.

Two Selfish Miners (Case 2)

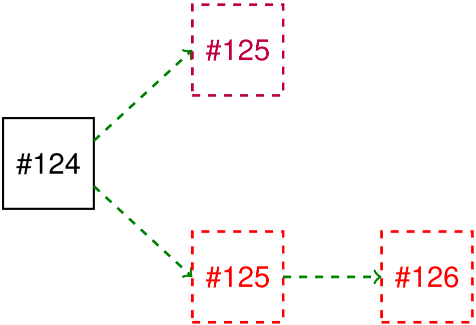
An honest miner *Henry* (r_h) and two selfish miners *Alice*(r_a) and *Bob*(r_b)



Longest chain rule shall be applied.

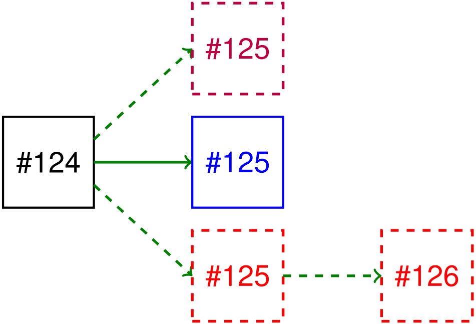
Two Selfish Miners (Case 3)

An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



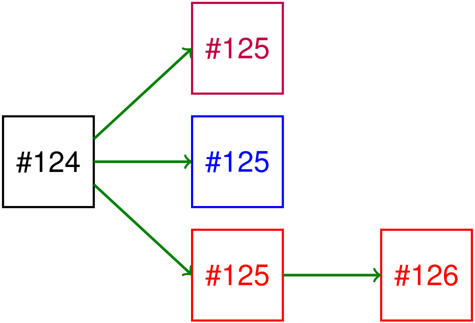
Two Selfish Miners (Case 3)

An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



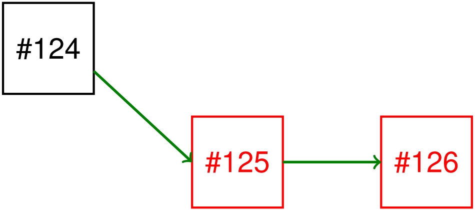
Two Selfish Miners (Case 3)

An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



Two Selfish Miners (Case 3)

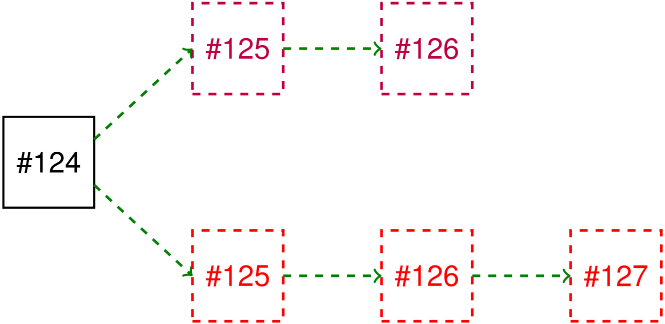
An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



Longest chain rule shall be applied.

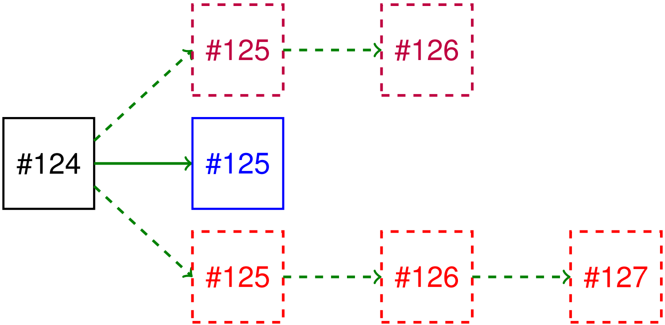
Two Selfish Miners (Case 4)

An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



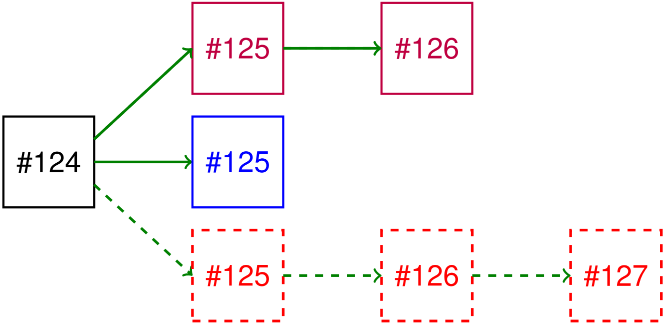
Two Selfish Miners (Case 4)

An honest miner *Henry* (r_h) and two selfish miners *Alice*(r_a) and *Bob*(r_b)



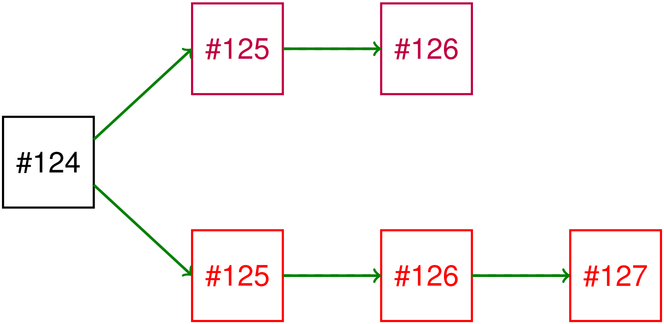
Two Selfish Miners (Case 4)

An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



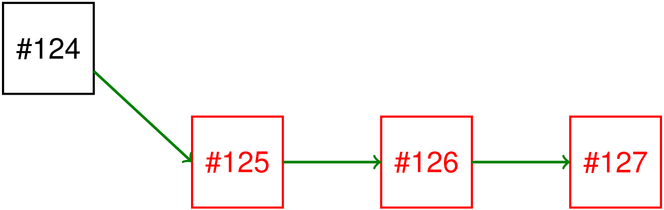
Two Selfish Miners (Case 4)

An honest miner *Henry (r_h)* and two selfish miners *Alice(r_a)* and *Bob(r_b)*



Two Selfish Miners (Case 4)

An honest miner *Henry* (r_h) and two selfish miners *Alice*(r_a) and *Bob*(r_b)

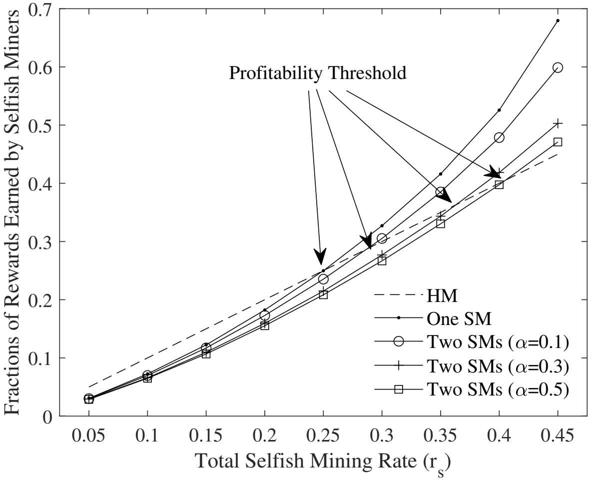


Longest chain rule shall be applied.

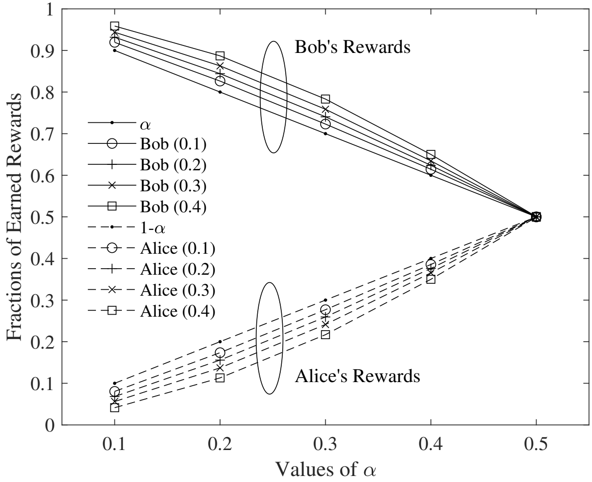
Simulation Based Observations

- S.W. Wang, ”Analysis of Earned Rewards In A Blockchain with Two Selfish Miners,” in *2024 IEEE International Conference on Blockchain and Cryptocurrency (ICBC 2024)*, Dublin, Ireland, May 27-31, 2024.

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. Tzeng, ”An Accurate Analytical Model for A Proof-of-Work Blockchain with Multiple Selfish Miners,” in *2024 IEEE International Conference on Communications (ICC)* Denver, Colorado, USA, June 9-13, 2024

Motivations & Contributions

- Previous works use simulations to study the interesting properties of earned rewards
  - Time consuming
  - Lack of theoretical contributions
- An analytical model to calculate the rewards earned by different miners is much more desirable

Motivations & Contributions

- Previous works use simulations to study the interesting properties of earned rewards
  - Time consuming
  - Lack of theoretical contributions
- An analytical model to calculate the rewards earned by different miners is much more desirable

The Question

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

Motivations & Contributions

- Previous works use simulations to study the interesting properties of earned rewards
  - Time consuming
  - Lack of theoretical contributions
- An analytical model to calculate the rewards earned by different miners is much more desirable

The Question

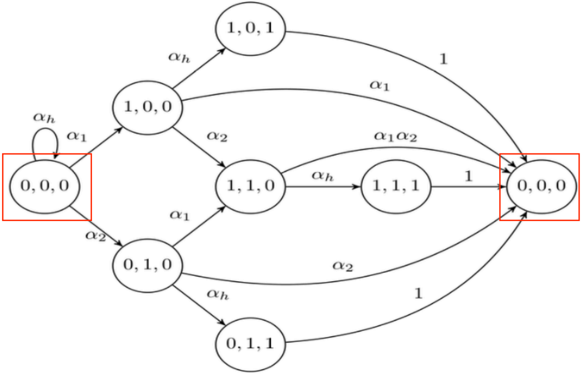
Can we efficiently and accurately calculate the reward earned by each miner in a blockchain with two selfish miners?

The Answer & Our Contribution

Yes. A closed-form expression with high accuracy is derived.

Previous Work: Two Selfish Miners

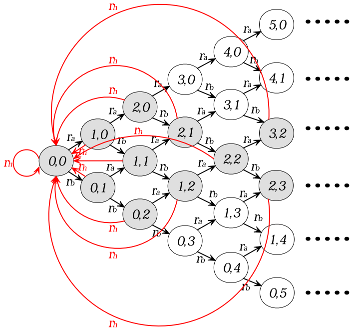
- Analytical Model Proposed by *Q. Bai*, and *et al.*



  - Two states with the same definition
  - Not very accurate because some states are ignored

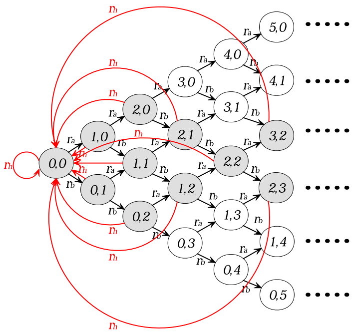
The Proposed Analytical Model

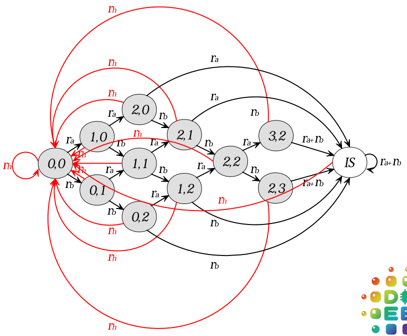
- State (n_a, n_b): Alice and Bob have their private branches with n_a and n_b blocks respectively
- End-of-Selfish (ES) states and In-Selfish (IS) states



The Proposed Analytical Model

- State (n_a, n_b): Alice and Bob have their private branches with n_a and n_b blocks respectively
- End-of-Selfish (ES) states and In-Selfish (IS) states





Steady-State Probabilities

- Steady state probability of an ES state (n_a, n_b):

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

- Steady state probability of an IS state:

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

- Sum of the probabilities equals to 1 where $\pi_{0,0}$ can be easily obtained.

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

- Closed-form expressions are obtained

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

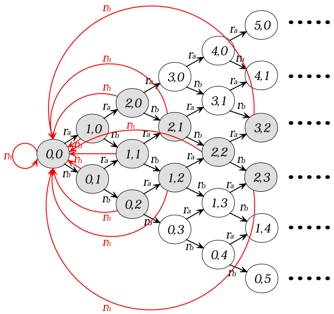


Figure 1: Exact Model

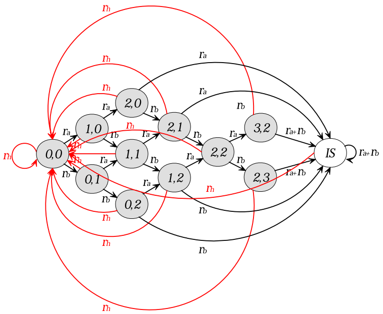
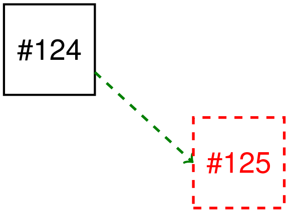
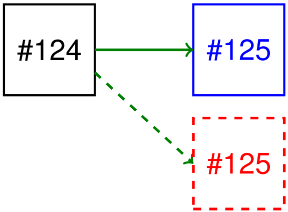


Figure 2: Approximate Model

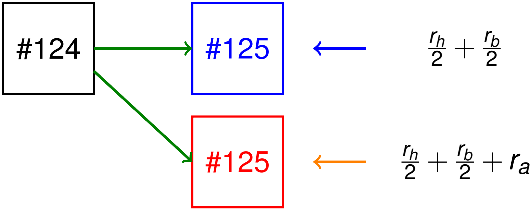
State $(0,1)$ and $(1,0)$



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

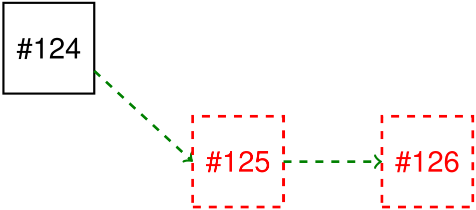


State $(0,1)$ and $(1,0)$

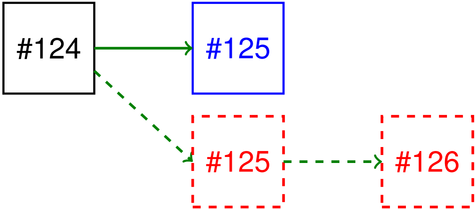


State s	Alice $R_a(s)$	Bob $R_b(s)$	Henry $R_h(s)$
(1,0)	$2r_a+r_b/2+r_h/2$	r_b	$3r_h/2+r_b/2$
(0,1)	r_a	$r_a/2+2r_b+r_h/2$	$3r_h/2+r_a/2$

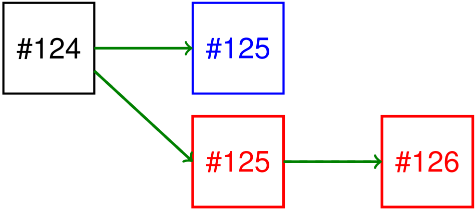
State $(0, 2)$ and $(2, 0)$



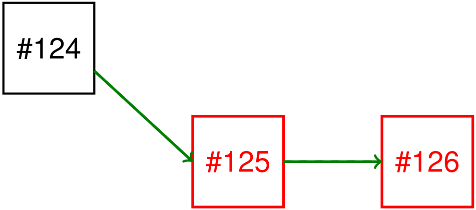
State $(0, 2)$ and $(2, 0)$



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



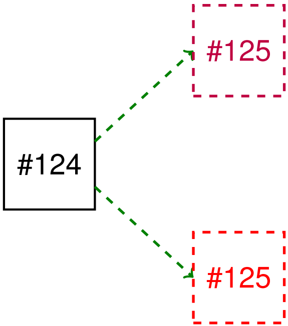
State $(0,2)$ and $(2,0)$



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

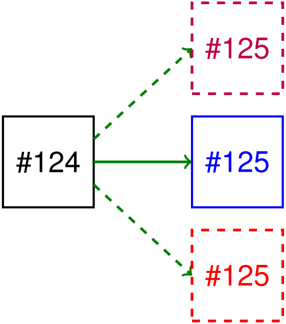
State $(1, 1)$

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



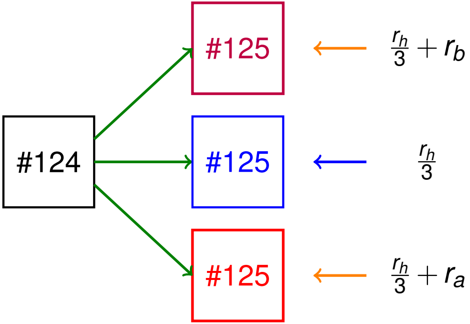
State $(1, 1)$

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



State $(1,1)$

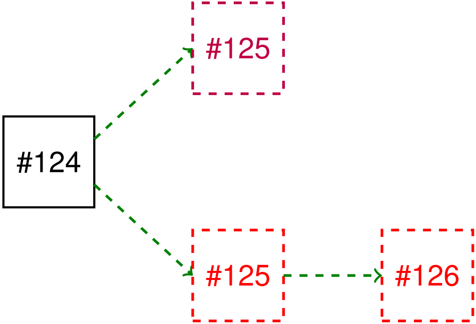
Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



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

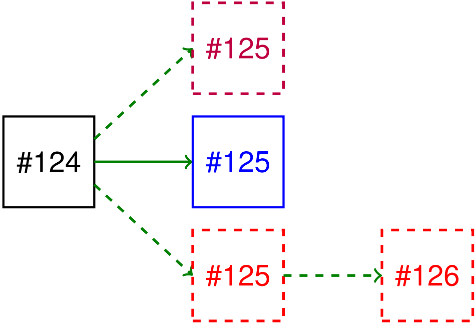
State $(1, 2)$ and $(2, 1)$

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



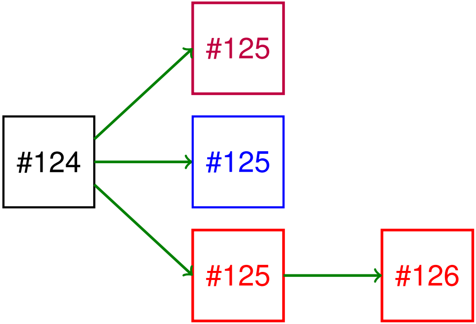
State $(1,2)$ and $(2,1)$

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



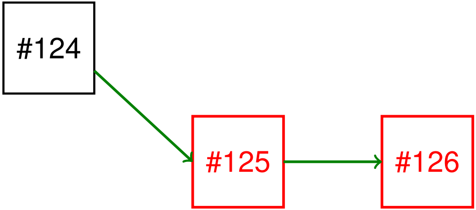
State $(1,2)$ and $(2,1)$

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



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

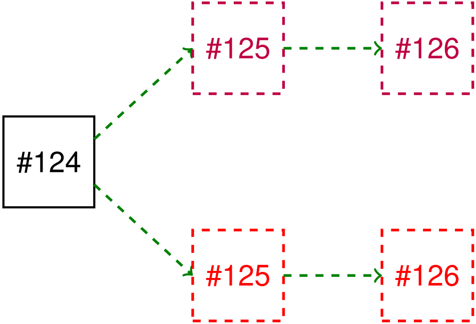
Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



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

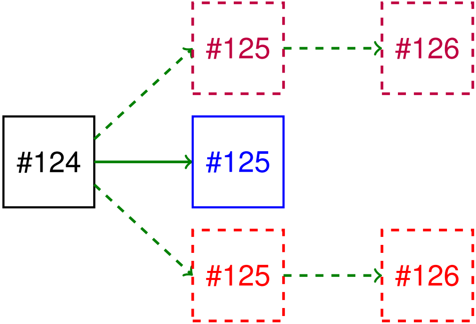
State $(2, 2)$

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



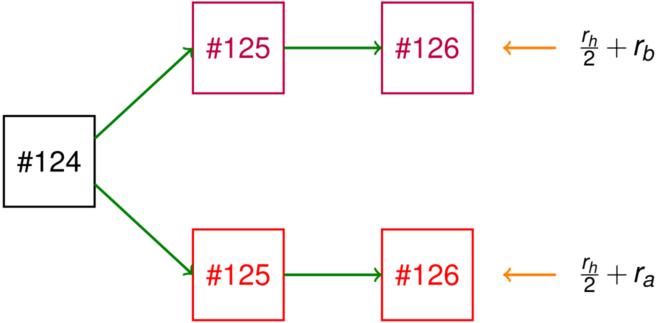
State $(2, 2)$

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



State $(2,2)$

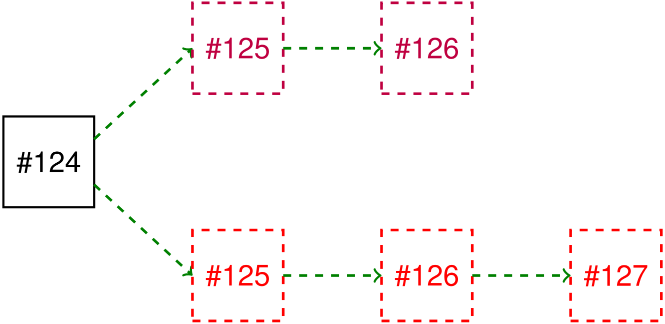
Honest miner *Henry (r_h)* and Selfish Miners *Alice(r_a)* and *Bob(r_b)*



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

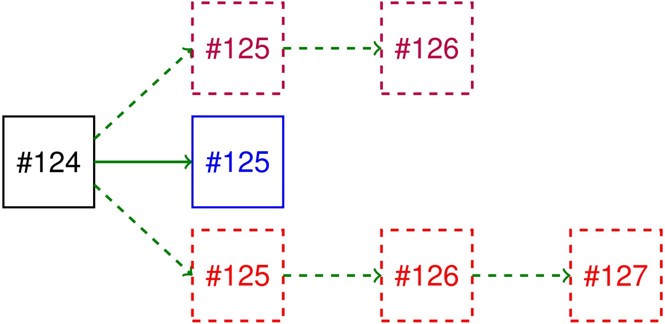
State $(2, 3)$ and $(3, 2)$

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



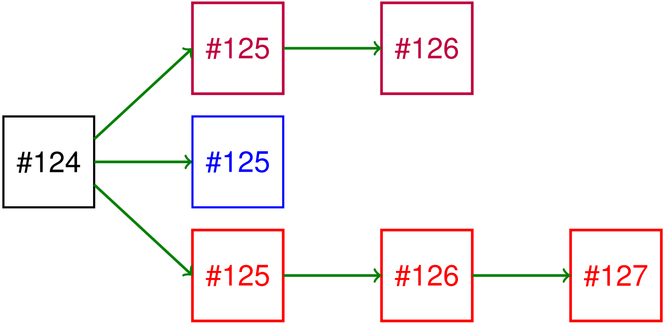
State $(2,3)$ and $(3,2)$

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



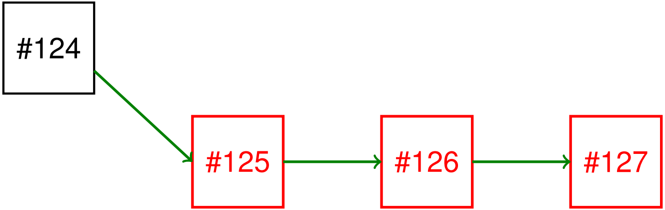
State $(2,3)$ and $(3,2)$

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

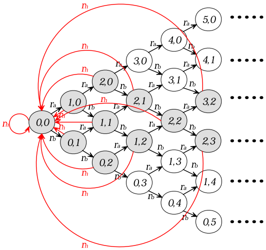


Figure 3: Exact Model

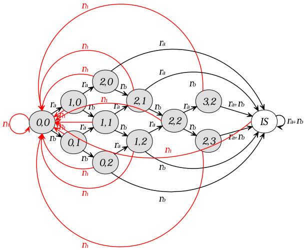


Figure 4: Approximate Model

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	$3r_h/2 + r_b/2$
(0,1)	r_a	$r_a/2 + 2r_b + r_h/2$	$3r_h/2 + r_a/2$
(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.

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

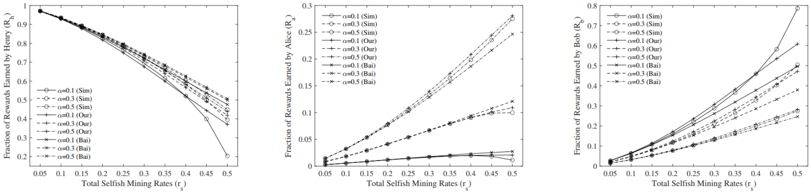
- Sum of the steady-state probabilities equals to 1.

$$\pi_{IS} + \sum_{s \in ES} \pi_s = 1 \tag{1}$$

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

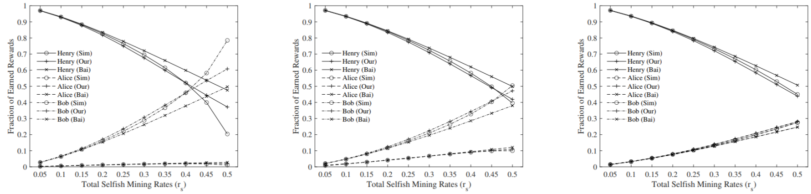
- The steady-state probability and expected earned rewards can be expressed in a closed-form of r_a, r_b, and r_h.

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

Conclusions

- An accurate analytical model for Proof-of-Work blockchain with two selfish miners is proposed
- Except the situation when there is a selfish miner with dominant mining rate, the maximum percentage of differences is 4.98%
- Our proposed analytical model performs closer to the simulation results than previous approach

Multiple Selfish Miners Extension

- S.W. Wang and S.S. Tzeng, ”An Accurate and Efficient Analytical Model for Security Evaluation of PoW Blockchains with Multiple Independent Selfish Miners,” under review after Major Revision in *IEEE Transactions on Network and Service Management*

Our Model: Extension to Multiple Selfish Miners

Step. 1 Describe a state for M selfish miners

Step. 2 Identify the ES states and IS states

Step. 3 Construct the Markov Chain

Step. 4 Calculate the reward in each state

Step. 5 Calculate the steady-state probability

Step. 6 Calculate the fraction of earned rewards

[Step. 1] Describe a state for M selfish miners

- Let M be the number of independent selfish miners.
- There are totally $M+1$ miners, including one honest miner.
- Let state $(n_1, n_2, \cdots, n_M)$ represents the state that selfish miner i has n_i blocks in his private branch.
- State $s_0 = (0, 0, \cdots, 0)$ is the state from which all miners start to mine the first block after the previous selfish mining phase ends.
- The mining rate of the honest miner is denoted as r_h and the mining rate of selfish miner i is denoted as r_i.

$$r_h + r_s = r_h + \sum_{i=1}^{M} r_i = 1. \tag{2}$$

[Step. 2] Identify the ES states and IS states

- An algorithm to identify ES or IS state is proposed.
- Take $M = 8$ for example.
  - ES state: $(0, 1, 3, 4, 2, 2, 5, 6)$
  - IS state: $(0, 1, \textcolor{red}{4}, 4, 2, 2, 5, 6)$

[Step. 3] Construct the Markov Chain

- Enumerate all ES states
  - For each ES state $s = (n_1, n_2, \cdots, n_M)$, we generate M new states $s_i^+ = (n_1, n_2, \cdots, n_i + 1, \cdots, n_M)$
  - If state s_i^+ is an ES state, the state is pushed into the ES list and put into the Markov chain
  - The transition rate from s to s_i^+ is set to r_i.
  - If state s_i^+ is not an ES state, we add an transition rate r_i from s to IS.
  - Finally, the Markov chain is obtained.

[Step. 3] Construct the Markov Chain

- Number of ES states:

$$N_{ES} = (M+2)^M - \sum_{k=3}^{M+1} \{ \sum_{m=1}^{M} \binom{M}{m} [\sum_{s=1}^{k-2} (-1)^{s-1} \binom{k-2}{s} (k-s)^{M-m}] \}$$

Table 1: Number of *ES* states under different numbers of selfish miners *M*.

M	2	3	4	5
N_{ES}	11	51	299	2,163
M	6	7	8	9
N_{ES}	18,731	189,171	2,183,339	28,349,043

[Step. 4] Calculate the reward in each state

- Three cases:
  1. $(0,1,3,4,2,2,5,6)$: Only one winer
  2. $(0,1,3,6,2,2,5,6)$: Multiple winners without honest miner
  3. $(0,1,1,0,0,0,1,0)$: Multiple winners with honest miner
- An algorithm to calculate the earned reward of each miner (honest/selfish) is proposed

[Step. 5] Calculate the steady-state probability

- Let the steady-probability of state $s_0 = (0, 0, \cdots, 0)$ as π_{s_0}.
- Assuming $N = n_1 + n_2 + \cdots + n_M$, we have

$$\pi_s = \binom{N}{n_1} r_1^{n_1} \binom{N-n_1}{n_2} r_2^{n_2} \binom{N-n_1-n_2}{n_3} r_3^{n_3} \cdots \binom{n_M}{M} r_M^{n_M} \pi_{s_0}.$$

- IS state

$$\pi_{IS} = \sum_{s \in ES \text{ and } s_i^+ \notin ES} \pi_s \times r_i.$$

- Normalization

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

[Step. 6] Calculate the fraction of earned rewards

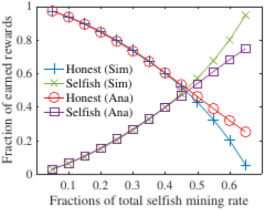
- For each miner, he can earn R_x units of rewards where $x = h$ or $x = i$.

$$R_x = \sum_{s \in ES \cup \Omega} R_x(s) \times \pi_s \quad \text{where} \quad x = h, 1, 2, \cdots, M$$

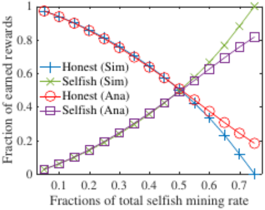
- The fraction of rewards earned by a miner:

$$ER_x = \frac{R_x}{R_h + \sum_{i=1}^{M} R_i} \quad \text{where} \quad x = h, 1, 2, \cdots, M$$

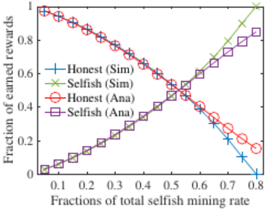
Accuracy of the proposed model



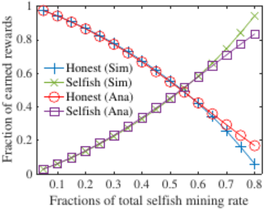
(a) Fractions of rewards earned by honest and all $M = 2$ selfish miners



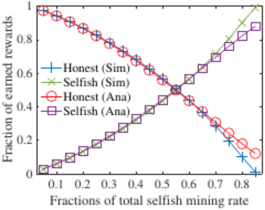
(b) Fractions of rewards earned by honest and all $M = 3$ selfish miners



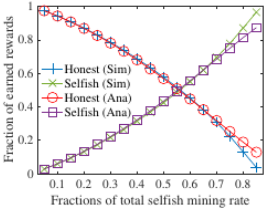
(c) Fractions of rewards earned by honest and all $M = 4$ selfish miners



(d) Fractions of rewards earned by honest and all $M = 5$ selfish miners

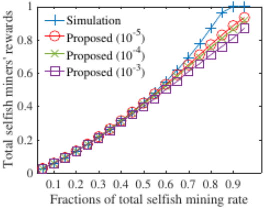


(e) Fractions of rewards earned by honest and all $M = 6$ selfish miners

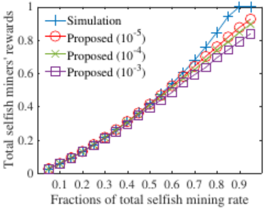


(f) Fractions of rewards earned by honest and all $M = 7$ selfish miners

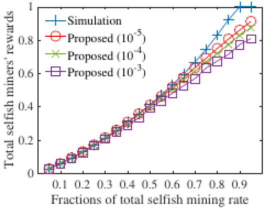
ES States Truncation: Accuracy vs. States



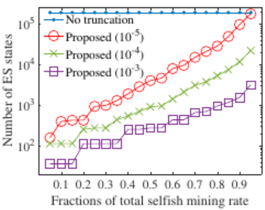
(a) Fractions of rewards earned by all $M = 7$ selfish miners with truncation



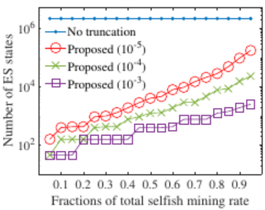
(b) Fractions of rewards earned by all $M = 8$ selfish miners with truncation



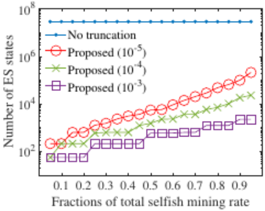
(c) Fractions of rewards earned by all $M = 9$ selfish miners with truncation



(d) Number of ES states with truncation when $M = 7$ selfish miners exist



(e) Number of ES states with truncation when $M = 8$ selfish miners exist



(f) Number of ES states with truncation when $M = 9$ selfish miners exist

ES States Truncation: Execution Time

M	2	3	4	5
N_{ES}	11	51	299	2,163
M	6	7	8	9
N_{ES}	18,731	189,171	2,183,339	28,349,043

M	2	3	4	5	6	7	8	9
Analytical model without truncation	< 0.01	< 0.01	0.01	0.17	7.28	331.62	11,366.82	Extremely long
Analytical model with truncation ($T_{ES} = 10^{-5}$)	Unnecessary					1.54	4.35	9.49
Analytical model with truncation ($T_{ES} = 10^{-4}$)	Unnecessary					0.24	0.47	0.94
Analytical model with truncation ($T_{ES} = 10^{-3}$)	Unnecessary					0.01	0.09	0.19
Simulation with 10^7 **confirmed blocks**	1.75	1.89	2.01	2.12	2.22	2.31	2.41	2.49
Simulation with 10^8 **confirmed blocks**	17.33	18.55	19.89	20.98	22.05	22.72	23.80	24.68
Simulation with 10^9 **confirmed blocks**	173.51	187.78	200.46	211.52	220.83	231.04	239.84	250.26

Our Works - Journal Papers

Journal Papers

1. S.W. Wang and S.S. Tzeng, "Accurate Estimation of Selfish Mining Rate by Stale Block Ratio in a Proof-of-Work Blockchain," accepted by IEEE Transactions on Network and Service Management, 2025.
2. S.S. Tzeng, and L.C. Wang, "SLChain: A Stochastic Lightweight Blockchain with Selfish Mining Attack Mitigation," under review after rejection (revise & resubmit) by IEEE Internet of Things Journal.
3. S.W. Wang and S.S. Tzeng, "An Accurate and Efficient Analytical Model for Security Evaluation of PoW Blockchains with Multiple Independent Selfish Miners," under review after major revision decision made by IEEE Transactions on Network and Service Management.

Our Works - Conference Papers

1. S.W. Wang, W.L. Chen, and S.S. Tzeng, ”Nonce Distribution in Bitcoin Blockchain,” in The 25th Asia-Pacific Network Operations and Management Symposium (APNOMS 2025), Kaohsiung, Taiwan, September 22-24, 2025.
2. S.W. Wang and S.S. Tzeng, ”Security Analysis of Majority and Selfish Mining Attacks in A Blockchain with Sharding,” in 2025 IEEE International Conference on Communications (ICC 2025), Montreal, Canada, June 8-12, 2025.
3. S.W. Wang and S.S. Tzeng, ”An Accurate Analytical Model for A Proof-of-Work Blockchain with Multiple Selfish Miners,” in 2024 IEEE International Conference on Communications (ICC 2024), Denver, Colorado, USA, June 9-13, 2024
4. S.W. Wang, ”Analysis of Earned Rewards In A Blockchain with Two Selfish Miners,” in 2024 IEEE International Conference on Blockchain and Cryptocurrency (ICBC 2024), Dublin, Ireland, May 27-31, 2024.
5. S.W. Wang, ”A Game Theory Based Rational Mining Strategy in Blockchains With Multiple Rational Miners,” in International Conference on Computing, Networking and Communications (ICNC 2024), Big Island, Hawaii, USA, Feburary 19-22.
6. S.W. Wang, ”Selfish Mining Attacks in Sharded Blockchains,” in International Conference on Computing, Networking and Communications (ICNC 2024), Big Island, Hawaii, USA, Feburary 19-22.

Conclusions

- Some research topics can be further surveyed by using the proposed analytical model
- We stopped the study in selfish mining and switch our focus to
  - Directed Acyclic Graph (DAG) based DLT
  - Consensus in Permissioned Blockchains
  - Reputation-based PBFT
  - Federated Learning in Blockchains
- Read, think, and challenge

Conclusions

- Some research topics can be further surveyed by using the proposed analytical model
- We stopped the study in selfish mining and switch our focus to
  - Directed Acyclic Graph (DAG) based DLT
  - Consensus in Permissioned Blockchains
  - Reputation-based PBFT
  - Federated Learning in Blockchains
- Read, think, and challenge

Thank you!