Analysis of Selfish Miner Behavior in the Bitcoin Network

Saeideh G. Motlagh, Jelena Mišić, and Vojislav B. Mišić
Ryerson University, Toronto, ON, Canada

ICC 2021 - IEEE International Conference on Communications | 978-1-7281-7122-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICC42927.2021.9500701

***Abstract*—Selfish mining is a recognized misbehaving attack in Bitcoin. Selfish miners intentionally delay the release of newly mined blocks with the goal of gaining more revenue. While several studies have been devoted to analyzing the selfish miner behavior, the impact of selfish behavior on Bitcoin network performance has received little to no attention at all. In this work, we focus on that impact using a Markov chain that models selfish behavior both from the aspect of mining and form the aspect of block distribution time. We find that blocks mined by honest miners undergo longer distribution time compared to blocks mined by selfish miners. This delay results in intentional forking and the resulting network inconsistency provides more opportunity for selfish miners to gain unfair revenue.**

***Index Terms*—Blockchain; selfish mining**

I. Introduction

Bitcoin is the first and likely the best known cryptocurrency [10]. The high price of Bitcoin (currently in the range of tens thousand dollars[1] makes it attractive for both honest and selfish miners. One of the first and best known attacks that attempt to earn unfair revenue in Bitcoin is known as *selfish mining* [3]. Selfish miners do not follow the original consensus algorithm: they keep the blocks they mined to themselves, hoping to mine a longer chain which they will release at an opportune moment to gain revenue and waste the mining effort invested by honest nodes. In this manner, selfish miners are able to obtain disproportionate rewards, but also to potentially grow fast to become the majority in a snowball effect [3].

Most research on selfish mining has focused on analyzing the rewards that can be obtained with it, typically using Markov chains [7] or Markov Decision Processes [13]. However, there are difficulties in straightforward application of this concept, which is why a number of subsequent improvements were made [4], [15].

More recent research on selfish mining has been focused on its impact on gaining revenue [5]; on finding solutions to avoid or detect selfish mining [6], [14], and on twisting it, including different difficulty adjustment algorithms, to make even more revenue [11].

In an analysis of many selfish miners' impact on the gaining revenue presented in [4], a Bitcoin network simulator was used to compare the revenue gained by many individual selfish miners with the group of selfish miners who mine together as a group (pool). The conclusion was that, due to network complexities, it is vital to consider as many of the network's dynamic factors, like network latency, as possible to achieve the most precise result. This conclusion also indicated the need to analyze selfish behavior and its effect of gaining revenue from a network perspective.

Despite this, to the best of our knowledge, the impact of selfish mining on the Bitcoin network from a network perspective has not been investigated in depth. In this work, we develop an analytical model for the Bitcoin network with a combination of selfish and honest nodes, and analyze the impact of selfish behavior on the network performance.

The remaining part of the paper proceeds as follows: In Section II we present Markov Chain for the selfish nodes and solve the presented Markov chain and calculate different states' probability. The impact of selfish behavior on network connectivity, and data distribution will be discussed in Sections III and IV respectively. Performance results are discussed in Section V and Section VI concludes the paper.

II. Markov chain to model selfish behavior

We want to examine the impact of selfish behavior on the Bitcoin network performance and ledger inconsistency. To do so, we present the Markov Chain to model selfish behavior in the network as shown in Fig. 1. We assume selfish miners cannot be ahead of public chain more than four blocks due to their hash power. States in the Markov chain can be defined as follows:

- *Honest State* S_{-1} indicates that honest miners mine new blocks before selfish miners. In other words, selfish miners are not able to compete with honest miners for unfair rewards.
- *Steady State* S_0 indicates that there is only a single, global, public longest chain in the network. In this state, if selfish miners mine the new block prior to the honest miners, they extend their private chain by not releasing the new block to the public, and switch to the state S_1. In contrast, if honest miners mine the new block first, the chain switches to the state S_{-1}.
- *Selfish states* S_i mean that selfish miners are i blocks ahead of the public chain, where $i \geq 1$. Two scenarios are feasible in these states: first, selfish miners find the new block prior to honest miners. In this case, selfish miners do not reveal the new block private to the public, extend their private chain, and switch to state S_{i+1}. Secondly, honest miners find the new block prior to selfish miners. In this case, an intentional fork happens and chain switches to the state F_i. An exception happens

[1]https://www.google.com/search?client=firefox-b-d&q=bitcoin+worth, Last access: September 29, 2020.

if selfish miners are two blocks ahead of the public chain. In this case, if a new honest block arrives, the selfish miners release their two private blocks and win the competition. Thus, the fork will not happen since selfish miners win the competition and gain the reward for two released blocks.

- *Forking state* F_i means there is an intentional fork caused by selfish miners in the network, which results in a competition between selfish and honest miners to gain the reward by finding the next block. Three different scenarios are possible: first, selfish miners mine the new block on the chain with selfish block tip, and the chain switches to state $h_{i,a}$. Second, honest miners find the new block on the chain with a selfish block tip and chain switches to state $h_{i,b}$. Finally, honest miners find the new block on the chain with an honest block tip and chain switches to state $h_{i,c}$.
- *Heading state* $h_{i,a}$ indicates that selfish miners win the competition and gain rewards for two blocks. The network reaches consensus on a single chain after adoption time.
- *Heading state* $h_{i,b}$ shows that honest miners have found the new block on the selfish ancestor block prior to the selfish miners. As a result, both selfish and honest miners gain reward for one block and the network reach consensus after adoption time.
- *Heading state* $h_{i,c}$ indicates that honest miners have found the new block on the honest ancestor block prior to selfish miners. That is, honest miners win the competition and gain rewards for two blocks. Like previous state, the network reach consensus after adoption time.

Consequently, each state can be described in terms of parameters (Po, Ho, Hr, So, Sr, F) with the following meaning:

- Po counts the number of blocks on the public branch.
- Ho counts the number of blocks on an honest branch.
- Hr counts the rewards that honest miners gained.
- So counts the number of blocks on a selfish branch.
- Sr counts the rewards gained by selfish miners.
- F is the intentional fork indicator with values of 1 and 0.

The rate at which new blocks are released to the network is denoted with γ. Assuming that this time is exponentially distributed, γ can be used as the transition rate in the Markov chain.

We assume equal hashing power for all nodes in the network. Let N_s and N_h show the total number of selfish and honest miners, receptively, in which case the total number of nodes is $N = N_s + N_h$.

Total block arrival rate in the network is denoted with λ, which is hardwired in the Bitcoin network to approx. one block per ten minutes, or $\lambda = \frac{1}{10 \cdot 60}$ per second. As new blocks can be mined by selfish or honest nodes, the new block arrival rate can be expressed as the sum of corresponding block arrival rates, $\lambda = \lambda_h + \lambda_s$, where $\lambda_s = \lambda \frac{N_s}{N}$ and $\lambda_h = \lambda \frac{N_h}{N}$.

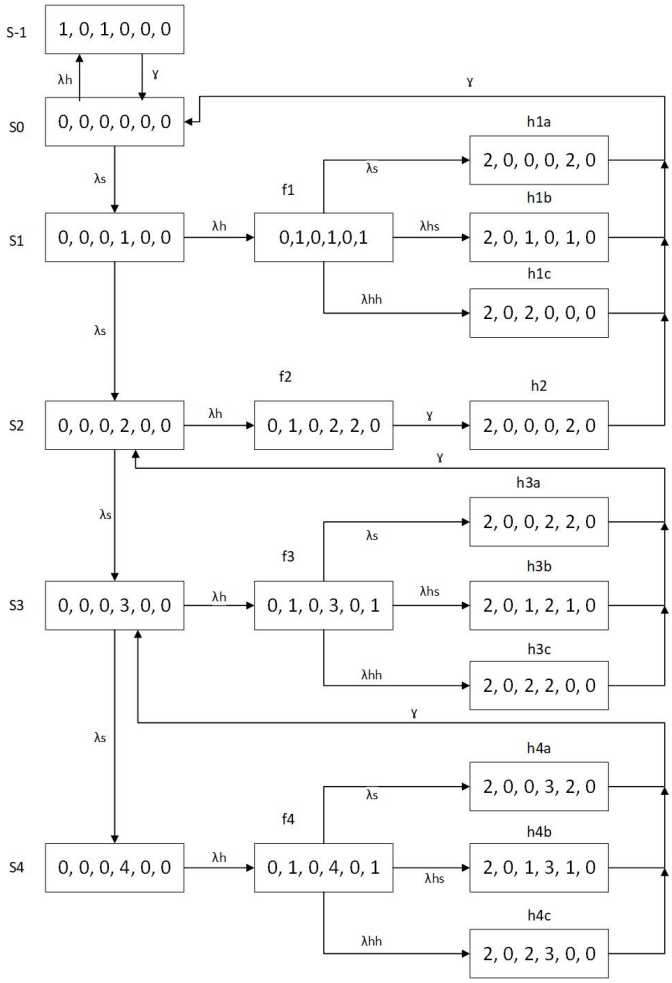


Fig. 1. Markov chain describing the behavior of a selfish node.

We can also calculate the rate of mining an honest block on top of a selfish block and, conversely, the rate of mining an honest block on top of an honest block, as

$$\lambda_{h,s} = \frac{S_{av}\lambda_s}{N} \tag{1}$$

$$\lambda_{h,h} = \frac{H_{av}\lambda_h}{N} \tag{2}$$

Where S_{av} and H_{av} denote the number of nodes that receive selfish block before honest block, or hones block before selfish block, respectively, at the time of forking in the network. Both values will be calculated in the next Section.

To solve the chain, we wrote the balance equations and law of total probability for all states. $P_{s_{-1}}$ shows the probability that honest miners mine the block.

$$P_{S_{-1}} = \frac{1}{A}\lambda_h^4(\lambda_h + \lambda_s)^2 \tag{3}$$

P_{s_0} is the steady state probability:

$$P_{S_0} = \frac{1}{A}\gamma\lambda_h^3(\lambda_s + \lambda_h)^2 \tag{4}$$

The probabilities of selfish miners having i blocks ahead of public chain, where $1 \leq i \leq 4$, are:

$$P_{S_1} = \frac{1}{A}\gamma\lambda_h^3\lambda_s(\lambda_s + \lambda_h) \tag{5}$$

$$P_{S_2} = \frac{1}{A}\gamma\lambda_h^2\lambda_s^2(\lambda_s + \lambda_h) \tag{6}$$

$$P_{S_3} = \frac{1}{A}\gamma\lambda_h\lambda_s^3(\lambda_s + \lambda_h) \tag{7}$$

$$P_{S_4} = \frac{1}{A}\gamma\lambda_s^4(\lambda_s + \lambda_h) \tag{8}$$

where A is shorthand notation for

$$\begin{aligned} A = \lambda_h\Big(\lambda_h^5 + 3\lambda_h^4\lambda_s + 4\lambda_h^3\lambda_s^2 + 2\lambda_h\lambda_s^4 + \lambda_s^5\Big) + \\ \gamma\Big(\lambda_h^5 + 4\lambda_h^4\lambda_s + 3\lambda_h^3\lambda_s^2 + 3\lambda_h^2\lambda_s^3 + 3\lambda_h\lambda_s^4 + \lambda_s^5\Big) \end{aligned} \tag{9}$$

Therefore, the total probability of selfish miners being ahead of the public chain is

$$P_{S_{tot}} = \sum_{i=1}^{4} P_{S_i} \tag{10}$$

We define selfish mode as the time a selfish node spends in any state other than steady or honest states. In this mode, selfish nodes are progressing with their selfish behavior which means that they are either in selfish states and mining selfish blocks, or in forking states competing with honest blocks to gain more revenue. The total selfish mode probability is

$$P_{S_{mode}} = 1 - P_{S_{-1}} - P_{S_0} \tag{11}$$

The intentional forking probability in states $1 \leq i \leq 4$ is

$$P_{f_1} = \frac{1}{A}\gamma\lambda_h^4\lambda_s \tag{12}$$

$$P_{f_2} = \frac{1}{A}(\lambda_h + \lambda_s)\lambda_h^3\lambda_s^2 \tag{13}$$

$$P_{f_3} = \frac{1}{A}\gamma\lambda_h^2\lambda_s^3 \tag{14}$$

$$P_{f_4} = \frac{1}{A}\gamma\lambda_h\lambda_s^4 \tag{15}$$

and the total intentional forking probability is

$$P_{f_{int,tot}} = \sum_{i=1}^{4} P_{f_i} \tag{16}$$

By the same token, the heading state probabilities for $1 \leq i \leq 4$ are

$$P_{h_1} = \frac{1}{A}(\lambda_h + \lambda_s)\lambda_h^4\lambda_s \tag{17}$$

$$P_{h_2} = \frac{1}{A}(\lambda_h + \lambda_s)\lambda_h^3\lambda_s^2 \tag{18}$$

$$P_{h_3} = \frac{1}{A}(\lambda_h + \lambda_s)\lambda_h^2\lambda_s^3 \tag{19}$$

$$P_{h_4} = \frac{1}{A}(\lambda_h + \lambda_s)\lambda_h\lambda_s^4 \tag{20}$$

Finally, we can calculate the probability of heading state substates as

$$P_{h_{i,a}} = P_{h_i}\lambda_s \tag{21}$$

$$P_{h_{i,b}} = P_{h_i}\lambda_{h,s} \tag{22}$$

$$P_{h_{i,c}} = P_{h_i}\lambda_{h,h} \tag{23}$$

III. Impact of selfish mining on network connectivity

Block transmission time is affected by selfish miners due to their selfish behavior. Selfish behavior strategy applies two main deviations from the original consensus algorithm in order to earn more revenue. First, selfish miners do not release the newly mined block immediately. Secondly, selfish miners do not participate in distributing blocks mined by honest miners. As a result, the network behaves differently for selfish blocks (i.e., blocks mined by selfish miners) as opposed to honest blocks (i.e., those mined by honest miners). To compare the manner in which the network that includes selfish and honest miners processes selfish and honest blocks, we analyze the block delivery time for each type of blocks separately.

- **Selfish blocks**: Data propagation in the Bitcoin network has been modeled in [9] using branching processes. We assume that all nodes in the network engage in mining, and that they have equal hashing power. According to [8], [2], Bitcoin network contains two types of nodes: nodes with usual number of connections in range between 5 to 13, referred to as *Ordinary* nodes, and nodes with high number of connections in range between 14 to around 60, which we call *Gateway* nodes. The number of TCP connections held by ordinary nodes is modeled using a truncated binomial probability distribution $C_n(z)$ with a mean of 8 connections. As both honest and selfish miners propagate selfish blocks in the same manner prescribed by the original Bitcoin data propagation protocol, the probability distribution function for the connectivity of selfish blocks $C_s(z)$ is the same as probability distribution function for the Bitcoin network without selfish behavior $C_n(z)$ [9]:

$$C_n(z) = C_s(z) = \sum_{k=5}^{13} p_k z^k \tag{24}$$

Furthermore, the number of TCP connections of gateway nodes follow a scale-free long-tail distribution $L_t(z)$ with parameter α and scaling factor L:

$$L_s(z) = L\sum_{k=14}^{60} \frac{1}{k^\alpha} z^k \tag{25}$$

where $L = \left(\sum_{k=14}^{60} \frac{1}{k^\alpha}\right)^{-1}$ is the normalization constant. We assume that selfish miners are combination of ordinary and gateway nodes. Thus, the final connectivity distribution mix of ordinary and gateway nodes is

$$Mix_s(z) = K_g L_s(z) + (1 - K_g)C_s(z) \tag{26}$$

where $0 \leq K_g \leq 1$ measures the portion of gateway nodes in the network.

- **Honest blocks**: Equation (26) defines the network connectivity for Bitcoin network where miners follow the original consensus algorithm. It also applies to propagation of selfish blocks. However, the network connectivity for honest blocks will change in the presence of selfish miners in the network, as selfish miners operating in selfish mode do not participate in data distribution of honest blocks so as to increase the probability of selfish block(s) being added to the main (public) chain.
To evaluate this effect, node connectivity distribution $C_n(z)$ and $L_t(z)$ should be modified. First, we calculate the probability that an honest miner has a selfish miner as peer. To do so, let $P_{selfish}$ denote the probability that a selfish peer node does not distribute received honest block:

$$P_{selfish} = \frac{N_s}{N}(1 - P_{S_{-1}}) \tag{27}$$

For one honest ordinary node, the probability of having u selfish peers is

$$P_{O_u} = \sum_{i=5}^{13} P_i \binom{i}{u} P_{selfish}^u (1 - P_{selfish})^{i-u} \tag{28}$$

By the same token, for one honest gateway node, the probability of having u selfish peers is

$$P_{G_u} = L \sum_{i=14}^{60} \frac{1}{i^\alpha} \binom{i}{u} P_{selfish}^u (1 - P_{selfish})^{i-u} \tag{29}$$

Thus, the probability generating functions (PGFs) for ordinary and gateway node connectivity in the presence of honest blocks are:

$$C_h(z) = \left(1 - \sum_{i=1}^{13} P_{O_i} C_n(z)\right) + \sum_{i=1}^{13} P_{O_i} \frac{C_n(z)}{z^i} \tag{30}$$

$$L_h(z) = \left(\sum_{k=1}^{10} P_{G_k} \frac{L_t(z)}{z^k} + (1 - \sum_{k=1}^{10} P_{G_k}) L_t(z)\right) \tag{31}$$

The final connectivity distribution for the network is

$$Mix_h(z) = K_g L_h(z) + (1 - K_g) C_h(z) \tag{32}$$

IV. Impact of selfish mining on data distribution and propagation

Let $D_{N,M_x(z)}$ be the diameter of the Bitcoin network with N nodes and connectivity expressed through (32). We use the model for data propagation using branching processes [9], however we modify it to account for the impact of selfish behavior.

- **Honest blocks**: We assign index $i = 0..D_{N,M_x(z)} - 1$ for each phase of data distribution. The PGF for the number of nodes in the first generation is

$$H_1(z) = Mix_h(z) \tag{33}$$

where $Mix_h(z)$ is connectivity PGF for honest blocks defined in (32). $\overline{H_1} = H'(1)$ is the mean number of nodes in the first generation (i.e., phase) of data distribution. The PGF for the number of nodes in subsequent generations is $H_i = H_{i-1}(Mx_i(z))$, where $Mx_i(z)$ is the PGF for the number of TCP connections available to a node in i-th generation [9], and the mean number of nodes in each generation is $\overline{H_i} = H'(1)$.
As selfish nodes will not propagate honest blocks when they are in selfish mode, the number of nodes that actually participate in honest block propagation is

$$N_{eff} = N - N_s(1 - P_{S_{-1}}) \tag{34}$$

The probability that a given node is reached in i-th generation is

$$Pt_i = \frac{\overline{H_i}}{N_{eff}} \tag{35}$$

Finally, the probability that data will not be forwarded is

$$P_{nt} = 1 - \sum_{i=0}^{D_{N,Mix_h(z)}-1} Pt_i \tag{36}$$

- **Selfish blocks:** The model for selfish blocks propagation is developed in a similar manner, except for the first generation. Using the connectivity model for selfish blocks computed in (26), the PGF for the number of nodes in the first generation is

$$H_1(z) = Mix_s(z) \tag{37}$$

For subsequent generations, the expressions obtained above for the connectivity distribution of honest nodes apply as well.

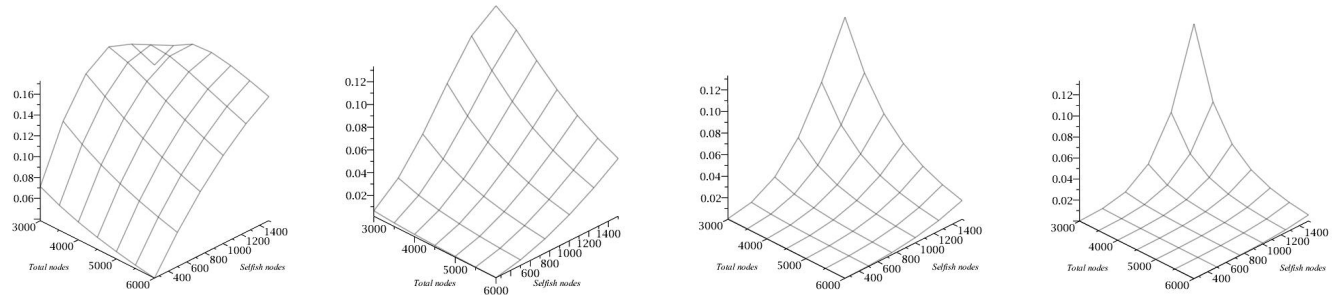
V. Performance results

We have considered the Bitcoin network with $N = 3000$ to 6000 nodes. The number of selfish miners varies between $N_s = 250$ to 1500 nodes regardless of network size. The portion of gateway nodes was set to $K_g = 0.4$ [1], [8], [12], which resulted in diameter 4 for the network. The adoption rate is set to $\gamma = 0.01$ per second.

A. Selfish node states

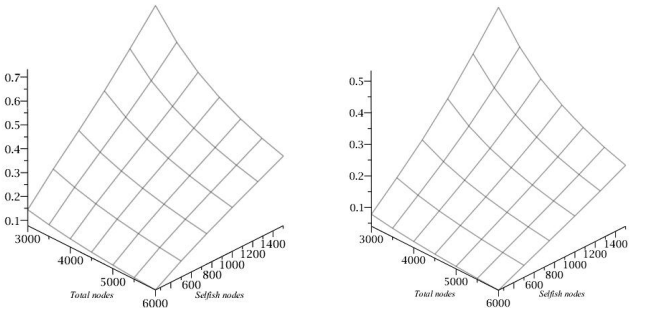
By solving the system of equations (3) to (23) we have obtained the probability distribution of selfish/honest Markov chain states. Figs. 2 shows probability distributions for selfish miners being ahead of the public chain by $1 \leq i \leq 4$ blocks, while Fig. 3 shows the total probability of selfish miners being ahead of the public chain (regardless of how many states ahead) and the probability of selfish miners being in the selfish mode.

As can be seen, there is a direct relationship between the number of selfish miners in the network and probability of selfish miners being ahead of the public chain. As shown in Fig. 2(a), this probability is highest for selfish miners to be one block ahead of the public chain; this is due to selfish miners having equal hash power as the honest ones.



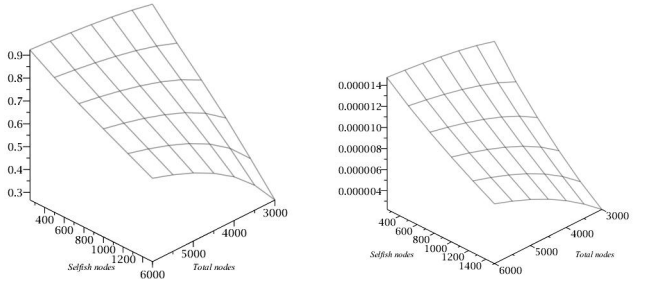
(a) $i = 1$ (b) $i = 2$ (c) $i = 3$ (d) $i = 4$

Fig. 2. Total probability of selfish miners having i blocks in their private chain.



(a) Probability of selfish mode. (b) Total probability of selfish states.

Fig. 3. Total probability of selfish miners.



(a) Probability of steady state. (b) Probability of honest miners have mined a block.

Fig. 4. Total probability for honest miners.

The probability of selfish miners being four blocks ahead of the public chain sharply increases with the number of selfish miners in the network with smaller number of nodes (N = 3000) where the total selfish miner hash power reaches half of the total network hash power at N_s = 1500, which means a higher capacity for selfish miners to compete with honest miners.

The conclusion to be drawn from Fig. 2 is that the probability of selfish miners being four blocks ahead of the public chain varies between 0.02 to 0.12 when the number of selfish miners is equal to the number of honest miners in the network.

Fig. 3(a) shows that selfish miners will be in the selfish mode for a major portion of time, as the corresponding probability reaches as much as 0.7 in the observed range of independent variables. Similar observation applies for the probability of selfish states shown in Fig. 3(b).

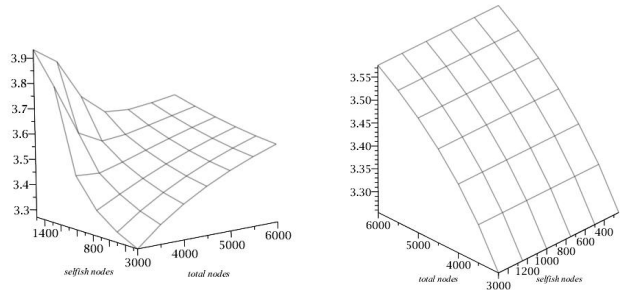
Fig. 4 shows probability distribution for honest miners. Fig. 4(a) shows that the probability of having a single public chain declines with the increase in the number of selfish miners in the network. Fortunately, the steady state P_0 has the highest probability among all other states, which shows that blockchain eventually reaches consensus and is stable despite selfish miners activity. Fig. 4(b) shows the probability that honest miners mine blocks continuously before selfish miners. This probability drops when the number of selfish miners and, consequently, their total hash power increase in the network.

B. Impact of selfish mining on network connectivity

Fig. 5 shows the mean number of hops needed for selfish and honest blocks to propagate through the entire network. As can be seen, the mean number of hops increases with network size for both selfish and honest traffic. However, the mean number of hops needed for honest blocks is higher than the mean number of hops needed for a selfish block, which is caused by selfish behavior. More hops result in longer propagation times for honest blocks than that needed for selfish blocks, which makes it easier for selfish miners to win the competition and earn more revenue.

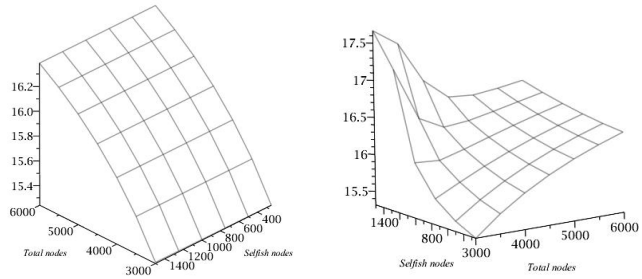
C. Impact of selfish mining on block traffic

Block delivery time for traffic of different types of blocks is shown in Fig. 6. As can be seen from Fig. 6(a), delivery time for selfish blocks shows a slight increase with the increase of the number of selfish miners. This happens because an increase in the number of selfish miners results in higher total hash power and selfish block arrival rate. Thus, node response time increases due to higher number of blocks that arrive and the corresponding increase in waiting time in the node queue which, taken together, lead to longer block delivery time. On the other hand, honest blocks delivery time also increases with the number of selfish nodes but this is caused by the selfish



(a) Honest blocks. (b) Selfish blocks.

Fig. 5. Mean number of hops needed for delivery of honest and selfish blocks respectively.



(a) Selfish block traffic. (b) Honest block traffic.

Fig. 6. Block delivery time in Bitcoin network.

node strategy that avoids distributing honest blocks, unless the node is in the steady state. In addition, when selfish nodes form a significant portion of the network, the mean number of hops for honest blocks increases. All these effects lead to block delivery time needed for honest blocks being noticeably longer than that needed for selfish blocks.

VI. Conclusion

In this paper, we have investigated the impact of selfish behavior on the Bitcoin network's performance. Our results have shown that when a large portion of nodes in the network behave selfishly, the probability of steady state in which all nodes follow the original block propagation protocol drops. In the boundary case where half of the nodes engage in selfish behavior, steady state probability drops to almost 0.3 which means that the network is not stable and the ledger is inconsistent. This, in turn, opens the possibility for double-spending and other attacks.

Our future work will focus on evaluating the revenue gained by selfish miners in more detail, and on searching for ways to detect and prevent selfish miner behavior, in particular with respect to changes in the block propagation protocol.

References

[1] S. Delgado-Segura, C. Pérez-Solà, J. Herrera-Joancomartí, G. Navarro-Arribas, and J. Borrell. Cryptocurrency networks: A new P2P paradigm. *Mobile Information Systems*, 2018.

[2] S. Delgado-Segura, C. Pérez-Sola, G. Navarro-Arribas, and J. Herrera-Joancomartí. A fair protocol for data trading based on Bitcoin transactions. *Future Generation Computer Systems*, 2017.

[3] I. Eyal and E. G. Sirer. Majority is not enough: Bitcoin mining is vulnerable. *arXiv preprint arXiv:1311.0243*, 2013.

[4] J. A. Gober. The Dynamics of a "Selfish Mining" Infested Bitcoin Network: How the Presence of Adversaries Can Alter the Profitability Framework of Bitcoin Mining, 2018. http://nrs.harvard.edu/urn-3:HUL.InstRepos:38811549.

[5] C. Grunspan and R. Perez-Marco. On profitability of selfish mining, 2018. arXiv preprint arXiv:1805.082.

[6] E. Heilman. One weird trick to stop selfish miners: Fresh bitcoins, a solution for the honest miner. In *Poster. InFC '14: Proceedings of the 18th International Conference on Financial Cryptography and Data Security.*, 2014.

[7] Q.-L. Li, Y.-X. Chang, X. Wu, and G. Zhang. A New Theoretical Framework of Pyramid Markov Processes for Blockchain Selfish Mining, 2020. arXiv:2007.01459.

[8] M. Lischke and B. Fabian. Analyzing the Bitcoin network: The first four years. *Future Internet*, 8(1):7, 2016.

[9] J. Mišić, V. B. Mišić, X. Chang, S. G. Motlagh, and M. Z. Ali. Modeling of bitcoin's blockchain delivery network. *IEEE Transactions on Network Science and Engineering*, to appear, 2019.

[10] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[11] K. Nayak, S. Kumar, A. Miller, and E. Shi. Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. In *2016 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 305–320, 2016.

[12] D. Ron and A. Shamir. Quantitative analysis of the full Bitcoin transaction graph. In *Int. Conf. Financial Cryptography and Data Security*, pages 6–24. Springer, 2013.

[13] A. Sapirshtein, Y. Sompolinsky, and A. Zohar. Optimal selfish mining strategies in bitcoin. In *Proceeding in International Conference on Financial Cryptography and Data Security*, 2016.

[14] R. Zhang and B. Prenee. Publish or perish: A backward-compatible defense against selfish mining in bitcoin. In *in Topics in Cryptology–CT-RSA 2017: The Cryptographers' Track at the RSA Conference*, pages 277–292, San Francisco, CA, USA, Feb. 2017.

[15] R. B. Zur, I. Eyal, and A. Tamar. Efficient MDP Analysis for Selfish-Mining in Blockchains, 2020. arXiv:2007.05614.