Selfish Mining in Ethereum

Ricardo Pérez-Marco, Cyril Grunspan

SELFISH MINING IN ETHEREUM

CYRIL GRUNSPAN AND RICARDO PÉREZ-MARCO

Abstract. We study selfish mining in Ethereum. The problem is combinatorially more complex than in Bitcoin because of major differences in the reward system and a different difficulty adjustment formula. Equivalent strategies in Bitcoin do have different profitabilities in Ethereum. The attacker can either broadcast his fork one block by one, or keep them secret as long as possible and publish them all at once at the end of an attack cycle. The first strategy is damaging for substantial hashrates, and we show that the second strategy is even worse. This confirms what we already proved for Bitcoin: Selfish mining is most of all an attack on the difficulty adjustment formula. We show that the current reward for signaling uncle blocks is a weak incentive for the attacker to signal blocks. We compute the profitabilities of different strategies and find out that for a large parameter space values, strategies that do not signal blocks are the best ones. We compute closed-form formulas for the apparent hashrates for these strategies and compare them. We use a direct combinatorics analysis with Dyck words to find these closed-form formulas.

1. Introduction

1.1. **Selfish mining strategies in Ethereum.** Research on selfish mining (in short SM) in Ethereum is quite recent. We can mention as recent contributions [1] (numerical study) and [3].

The authors of [3] use a Markov chain model and compute the stationary probability. Then they study what they call the "absolute revenue" of the attacker which corresponds to the apparent hashrate after a difficulty adjustment as explained in our articles on blockwithholding attacks in the Bitcoin network (see [4], [5], [6]). Their theoretical analysis seems also confirmed by their numerical simulations. They do not provide closed-form formulas (for example Formulas (8) and (9) in Section 3-E involve double infinite sums). But more importantly, their study is limited to the following strategy of the attacker:

(1) The attacker refers to all possible orphan blocks;
(2) When new blocks are validated by the honest miners, the attacker makes public the part of his fork sharing the same height as the "honest" blockchain.

(See Algorithm 1 in [3], Lines 1 and 19 from Section 3-C)

We label this strategy as "Strategy 1" or SM1. The procedure of a Bitcoin selfish miner to release his secret fork is irrelevant for the profitability of the classical selfish mining attack. However, this is not so in Ethereum. In particular, the precise algorithm presented in [3] is not the most profitable as we will prove. An

Date: April 30th, 2019.

2010 *Mathematics Subject Classification.* 68M01, 60G40, 91A60.

Key words and phrases. Bitcoin, Ethereum, blockchain, proof-of-work, selfish mining, Catalan numbers, Dyck path, random walk.

alternative strategy for the attacker would be to keep secret all his fork until he is on the edge of being caught-up by the honest miners. Then, and only at this critical moment, he would release his complete fork and override the public blockchain. We label this second strategy as "Strategy 2" or SM2. In Bitcoin, both strategies have the same effect since only matters the number of blocks mined by the attacker and added to the official blockchain. But in Ethereum, this is not so because of the different reward incentives that gives rewards to "nephew" blocks who refer to "uncle" blocks. "Uncle" blocks are orphan blocks with a parent in the official blockchain, and the descendants of this parent in the official blockchain are its "nephew" blocks. Also uncle blocks get rewards when referred by nephews.

1.2. **Performance of Ethereum selfish mining strategies.** To understand what the best strategy for the attacker is, we need an in-deep knowledge of the nature of the selfish mining attack. In [4] we give a correct economic modeling with a model of repetition game, and we consider the time element that is absent from older Markov chain models. What is important for the attacker is to maximize the number of validated blocks in the official blockchain *per unit of time*, which is different from the percentage of blocks he validates. With this correct modeling, it becomes then clear that the attack is an exploit on Bitcoin's difficulty adjustment formula, that does include the orphan blocks. Then the attacker lowers artificially the difficulty, at the expense of orphaned honest blocks, and succeeds to validate more blocks per unit of time.

Point (2) in "Strategy 1" creates numerous competitions between the attacker's fork and the honest blockchain. This increases the production of orphan blocks that becomes important for a substantial hashrate of the attacker. Signaling these orphan blocks yields additional rewards to the attacker, but it goes against its main goal to lower the difficulty. Indeed, the difficulty's adjustment formula in Ethereum counts for "uncles", that are the orphan blocks directly attached to the main chain. Therefore, increasing the number of uncles by Point 2 has the following contradictory effects: On one hand, the attacker's revenue increases because of the new "inclusion rewards", but on the other hand, the difficulty is not lowered, so the attacker ends up mining less official blocks per unit of time in Strategy 1 compared to Strategy 2.

On the contrary, if the attacker decides to avoid competitions with honest miners as much as possible, he will earn less inclusion rewards (he can even decide to ignore totally these rewards) but his speed of validation of blocks will increase. So, what is the best strategy will depend very sensitively on the parameters of the reward system.

As explained in [5], the correct benchmark to compare profitabilities of two strategies is the revenue ratio

$$\Gamma = \frac{\mathbb{E}[R]}{\mathbb{E}[T]}$$

where R is the revenue of the miner per attack cycle and T is the duration of an attack cycle. In Bitcoin, after a difficulty adjustment, this quantity becomes in the long run proportional to

$$\tilde{\Gamma} = \frac{\mathbb{E}[R_s]}{\mathbb{E}[L]}$$

where L (resp. R_s) is the number of new blocks (resp. new blocks mined by the attacker) added to the official blockchain per attack cycle. The difficulty adjustment is not continuous in Bitcoin as it is updated every 2016 official new blocks. With the martingale tools introduced in [4], we computed how long it takes for the attack to become profitable (this computation is not possible with the old Markov chain model).

In Ethereum, the difficulty adjustment formula is different. The revenue ratio is proportional to

$$\tilde{\Gamma} = \frac{\mathbb{E}[R]}{\mathbb{E}[L] + \mathbb{E}[U]}$$

where U is the number of referred uncles and R is the total revenue of the attacker in the attack cycle. Moreover, the revenue R per attack cycle has three different contributions :

(1) The revenue R_s coming from "static" blocks.
(2) The revenue R_u coming from "uncles" blocks.
(3) The revenue R_n coming from "nephews" blocks.

In Bitcoin's revenue analysis only R_s is present. Therefore, for Ethereum we have

$$\tilde{\Gamma} = \frac{\mathbb{E}[R]}{\mathbb{E}[L] + \mathbb{E}[U]} = \frac{\mathbb{E}[R_s] + \mathbb{E}[R_u] + \mathbb{E}[R_n]}{\mathbb{E}[L] + \mathbb{E}[U]}$$

The new terms on the numerator $\mathbb{E}[R_u]$ and $\mathbb{E}[R_n]$ increase the revenue of the attacker and are incentives for block withholding attacks. On the other hand, the new term $E[U]$ in the denominator plays against the profitability of the attack and tends to mitigate the attack. Only an exact computation of these terms can show which one is the most profitable strategy. Another particularity of Ethereum is the continuous adjustment of the difficulty. Thus a block-witholding attack is very quickly profitable.

There are other selfish mining strategies in Ethereum. For instance, the attacker can publish his secret blocks slowly, two by two, instead of one by one. In this article we limit our study to Strategy 1 and Strategy 2. The main result are the closed-form formulas for the apparent hashrates in Strategy 1 and 2. The main conclusion is that the effect on the difficulty adjustment is prevalent, so that Strategy 2 outperforms Strategy 1.

2. A combinatorics approach

In this section we present a general setup that is common for all strategies. We apply our combinatorics approach to selfish mining as done previously for Bitcoin [6]. Dyck words and Catalan numbers are a powerful tool to compute the revenue ratio of a selfish miner in Bitcoin. In [6] we proved the following Theorem and Corollary:

Theorem 2.1. *Let L be the number of official new blocks added to the official blockchain after an attack cycle. We have*

$$\mathbb{P}[L = 1] = p \ ,$$
$$\mathbb{P}[L = 2] = pq + pq^2 \ ,$$

and for $n \geq 3$,

$$\mathbb{P}[L = n] = pq^2(pq)^{n-2}C_{n-2}$$

where $C_n = \frac{(2n)!}{n!(n+1)!}$ *is the n-th Catalan number.*

Corollary 2.2. *We have* $\mathbb{E}[L] = 1 + \frac{p^2q}{p-q}$.

We can represent the combinatorics information of an attack cycle ω by the chronological sequence of blocks, S (for Selfish) and H (for Honest). The relation between selfish mining and Dyck words is the following (see [6]),

Proposition 2.3. *Let* ω *be an attack cycle starting with SS. Then,* ω *ends with H and the intermediate sequence w defined by* $\omega = SSwH$ *is a Dyck word.*

Definition 2.4. *For* $n \geq 0$, *we denote by* $C_n(x) = \sum_{k=0}^{n} C_k x^k$, *the n-th partial sum of the generating series of the Catalan number.*

Example 2.5. We have $C_4(x) = 1 + x + 2x^2 + 5x^3 + 14x^4$.

Definition 2.6. *We define* $\pi_0 = \pi_1 = 0$ *and for* $k \geq 2$,

$$\pi_k = pq^2(\mathbf{1}_{k=2} + \mathbf{1}_{k\geq 2} \cdot (pq)^{k-2}C_{k-2}) \ .$$

The following lemma results from Theorem 2.1.

Lemma 2.7. *Let* ω *be an attack cycle.*

- *For* $k \geq 0$, *the probability that* ω *is won by the attacker and* $L(\omega) = k$ *is* π_k.
- *For* $k \geq 2$, *the probability that* ω *is won by the attacker and* $L(\omega) \leq k$ *is* $pq^2 + pq^2C_{k-2}(pq)$.

Proof. We have either $\omega =$ SHS or ω starts with SS. The result then follows from Lemma 6.2 in the Appendix. □

For Ethereum, the "static" part R_s of the revenue of the selfish miner coming from rewards for validated blocks is the same as for Bitcoin. However, we need to add the new terms R_s and R_n coming from uncle and nephew rewards.

Definition 2.8. *If* ω *is an attack cycle, we denote by* $U(\omega)$ *(resp.* $U_s(\omega)$, $U_h(\omega)$*) the random variable counting the number of uncles created during the cycle* ω *which are referred by nephew blocks (resp. nephew blocks mined by the selfish miner, nephew blocks mined by the honest miners) in the cycle* ω *or in a later attack cycle.*

We denote by $V(\omega)$ *the random variable counting the number of uncles created during the cycle* ω *and are referred by nephew blocks (honest or not) in an attack cycle strictly after* ω.

We take from [3] the notation K_u for the uncles reward function, and we denote by π the inclusion reward (see the glossary at the end).

For a general block withholding strategy, the random variables from Definition 2.8 do not contain all the information for the computation of the attacker's revenue. It depends not only on the number of uncles mined by the attacker but also on their distance d to its corresponding nephews.

However, for a miner following a selfish mining strategy, the part of his revenue coming from uncle rewards are easy to compute, as shown in the next Proposition, because only the case $d = 1$ is possible. This observation was already made in [3].

Proposition 2.9. *Let $R_u(\omega)$ be the total amount of uncle rewards of the selfish miner during an attack cycle ω. We have:*

$$\mathbb{E}[R_u] = p^2 q(1-\gamma)K_u(1) \ .$$

Currently on Ethereum we have $K_u(1) = \frac{7}{8}b$.

Proof. Let ω be an attack cycle. If $\omega = \text{SHH}$ with a second honest block mined on top of another honest block after a competition, the attacker has an uncle which is referred by the second honest block of the honest miners in the cycle ω. Otherwise, if $\omega \neq \text{SHH}$ then the attacker has no uncle in the cycle ω (the only uncle blocks are those mined by the honest miners). □

The *apparent hashrate* is the long term apparent hashrate of the attacker after the manipulation of the difficulty by the attacker.

Definition 2.10. *We denote by $\widetilde{q}_B$, resp. $\widetilde{q}_E$, the long term apparent hashrate of the selfish miner in Bitcoin, resp. Ethereum, defined by*

$$\widetilde{q}_B = \frac{\mathbb{E}[R_s]}{\mathbb{E}[L]}$$
$$\widetilde{q}_E = \frac{\mathbb{E}[R_s] + \mathbb{E}[R_u] + \mathbb{E}[R_n]}{\mathbb{E}[L] + \mathbb{E}[U]}$$

For Bitcoin we have the following formula (see [2] and [4]),

$$\widetilde{q}_B = \frac{[(p-q)(1+pq) + pq]q - (p-q)p^2q(1-\gamma)}{pq^2 + p - q}$$

For Ethereum only $\mathbb{E}[U]$ and $\mathbb{E}[U_s]$ are relevant for the computation of the apparent hashrate of the selfish miner:

Theorem 2.11. *We have*

$$\widetilde{q}_E = \widetilde{q}_B \cdot \frac{\mathbb{E}[L]}{\mathbb{E}[L] + \mathbb{E}[U]} + \frac{p^2q(1-\gamma)K_u(1)}{\mathbb{E}[L] + \mathbb{E}[U]} + \frac{\mathbb{E}[U_s]}{\mathbb{E}[L] + \mathbb{E}[U]}\,\pi \ .$$

Currently on Ethereum we have $K_u(1) = \frac{7}{8}$ and $\pi = \frac{1}{32}$.

Proof. Using Proposition 2.9, we have:

$$\begin{aligned}\widetilde{q}_E &= \frac{\mathbb{E}[R_s]+\mathbb{E}[R_u]+\mathbb{E}[R_n]}{\mathbb{E}[L]+\mathbb{E}[U]}\\ &= \frac{\mathbb{E}[R_s]}{\mathbb{E}[L]}\cdot\frac{\mathbb{E}[L]}{\mathbb{E}[L]+\mathbb{E}[U]}+\frac{\mathbb{E}[R_u]}{\mathbb{E}[L]+\mathbb{E}[U]}+\frac{\mathbb{E}[U_s]}{\mathbb{E}[L]+\mathbb{E}[U]}\,\pi\\ &= \tilde{q}_B\cdot\frac{\mathbb{E}[L]}{\mathbb{E}[L]+\mathbb{E}[U]}+\frac{p^2q(1-\gamma)K_u(1)}{\mathbb{E}[L]+\mathbb{E}[U]}+\frac{\mathbb{E}[U_s]}{\mathbb{E}[L]+\mathbb{E}[U]}\,\pi\end{aligned}$$

□

In next sections we compute $\mathbb{E}[U_s]$ and $\mathbb{E}[U]$ for different selfish mining strategies.

3. Strategy 1: Maximum belligerence signalling all uncles.

We consider here the strategy described in [3] where the attacker engages in competition with the honest miners as often as possible, and signals all possible "uncles'.

3.1. General definitions and basic results.

Definition 3.1. *The relative height of an orphan block $\mathfrak{b}$ validated by the honest miners is the difference between the height of the secret fork of the attacker at the time of creation of $\mathfrak{b}$ and the height of $\mathfrak{b}$. We denote it $h(\mathfrak{b})$.*

Example 3.2. For $\omega =$ SSSHSHSHH, the first three "honest" blocks have relative height equal to 2 and the last "honest" block has a relative height equal to 1.

Proposition 3.3. *Let $\mathfrak{b}$ be an uncle block mined by an honest miner and signaled by a nephew block which is at a distance d of $\mathfrak{b}$. Then, we have $h(\mathfrak{b}) < d$.*

Proof. Let $\mathfrak{b}'$ be the last block mined by the selfish miner at the date of creation of $\mathfrak{b}$. Notice that $h(\mathfrak{b})$ is also the number of blocks between $\mathfrak{b}$'s parent and $\mathfrak{b}'$. Thus the distance between $\mathfrak{b}$ and a possible nephew is necessarily strictly greater than $h(\mathfrak{b})$. □

Note 3.4. Let $n \geq 0$ and $\omega = \mathrm{SS}w$ be an attack cycle with $w = w_1 \dots w_{2n+1}$, $w_i \in \{S, H\}$ and $w_{2n+1} = H$. Then, w can be identified with a simple finite path $(X_i)_{0\leqslant i\leqslant 2n+1}$ starting from 0, satisfying: $\forall i \leqslant 2n+1, X_i = X_{i-1}+1$ (resp. $X_i = X_{i-1}-1$) if $w_i = S$ (resp. $w_i = H$) and ending at $X_{2n+1} = -1$ (see the Appendix). The index i indicates the $(i+2)$-th block validated during ω. It has been mined by the attacker (resp. honest miners) if $X_i = X_{i-1}+1$ (resp. $X_i = X_{i-1}-1$).

Proposition 3.5. *Let $\omega = SSw$ an attack cycle starting with two S with $w = w_1 \dots w_{2n+1}$, $w_i \in \{S, H\}$ and $w_{2n+1} = H$. We denote by $X : [0, 2n+1] \longrightarrow [-1, +\infty]$ the path associated with w as in Note 3.4. For $i \leqslant 2n+1$, let $\mathfrak{b}_i$ denote the i-th validated block in w. Then we have:*

$$X_i < X_{i-1} \Longrightarrow h(\mathfrak{b}_i) = X_i + 2$$

Proof. By induction on i, we show that $X_i + 2$ represents the advance of the fork created by the attacker over the official blockchain at the time of creation of the i-th block in w. Now, if $X_i < X_{i-1}$ then by Note 3.4, $\mathfrak{b}_i$ is a block validated by the honest miners. So $h(\mathfrak{b}_i)$ is well defined, and we get the result using Definition 3.1. □

Proposition 3.6. *Let $\omega = SSw$ be an attack cycle starting with two S and let $\mathfrak{b}_i$ be the i-th block validated in w. We denote by X the associated path according to Note 3.4. If $\mathfrak{b}_i$ is an uncle then we have:*

(1) $X_i < n_1 - 2$
(2) $X_i < X_{i-1}$

Proof. This follows from Proposition 3.3 and Proposition 3.5. □

Definition 3.7. *If $\omega = SSw$ is an attack cycle starting with two blocks S, then we denote by $H(\omega)$ the random variable counting the number of blocks in the cycle ω fulfilling (1) and (2) from Proposition 3.6.*

If w is an attack cycle, the condition $\omega = \mathrm{SS}\ldots$ means that ω starts with two S.

Proposition 3.8. *We have:*

$$\mathbb{E}[H(\omega)|\omega = SS\ldots] = \frac{p}{p-q}\left(1 - \left(\frac{q}{p}\right)^{n_1-1}\right)$$

Proof. See Lemma 6.1 in the Appendix. □

3.2. **Expected number of referred uncles by attack cycle.** We can be more precise in Proposition 3.6.

Lemma 3.9. *Let $\omega = SSw$ and X be the associated path from Note 3.4. We denote by $\mathfrak{b}_i$ the i-th block in w and suppose that conditions (1) and (2) from Proposition 3.6 are satisfied. The probability for $\mathfrak{b}_i$ to be an uncle is equal to γ, except when $\mathfrak{b}_i$ is the first block validated by the honest miners, then this probability is* 1.

Example 3.10. Suppose that $n_1 = 4$ and let $\omega = \mathrm{SS}w$ with $w = \mathrm{SHSSSHHHH}$. The blocks validated by the honest miners correspond to an index $i \in E = \{2, 6, 7, 8, 9\}$. We have $X_6 = 2$ and $X_i < 2$ for $i \in E$ and $i \neq 6$. The first block validated by the honest miners is an uncle with probability 1. The second block validated by the honest miners is a stale block which cannot be referred by a nephew block. All other blocks validated by the honest miners in ω can be uncles with probability γ. Note also that the last three blocks of the honest miners are not referred in ω and will be referred by the first future official block of the next attack cycle.

Using these observations, we can now compute $\mathbb{E}[U]$.

Proposition 3.11. *We have:*

$$\mathbb{E}[U] = q + \frac{q^3\gamma}{p-q} - \frac{p^3}{p-q}\left(\frac{q}{p}\right)^{n_1+1}\gamma - q^{n_1+1}(1-\gamma)$$

Proof. If $\omega = H$, then $U(\omega) = 0$. If $\omega \in \{\text{SHS}, \text{SHH}\}$, then, $U = 1$. Otherwise, ω starts with two consecutive S. Then, by Proposition 3.8 and Lemma 3.9, we have,

$$\mathbb{E}[U] = (0\cdot p)+1\cdot(pq^2+p^2q)+(\mathbb{E}[H(\omega)|\omega = \text{SS}\ldots]\gamma+(1-\gamma)(p+pq+\ldots+pq^{n_1-2}))\cdot q^2$$

The last term comes from the following fact: When the first honest block present in ω corresponds to an index i satisfying $X_i < n_1 + 2$, then its contribution to $\mathbb{E}[U]$ is underestimated by $\mathbb{E}[H(\omega)|\omega = \text{SS}\ldots]\gamma$ because it has probability 1 to be an uncle. This only occurs when ω starts with SS...SH with the first k blocks validated by the selfish miner with $k \leqslant n_1$, from where we get the last term. In conclusion we have:

$$\mathbb{E}[U] = pq + \left(\frac{p}{p-q}\left(1 - \left(\frac{q}{p}\right)^{n_1-1}\right)\gamma\right)\cdot q^2 + (1-\gamma)(1-q^{n_1-1})\cdot q^2$$

and we get the result by rearranging this last equation. □

Note 3.12. In particular, we obtain $\lim_{n_1\to\infty}\mathbb{E}[U] = q + \frac{q^3\gamma}{p-q}$. This limit can also be derived by observing that if $n_1 = \infty$, then $\mathbb{E}[U|L = n] = 1 + \gamma(n-2)$ and using Theorem 2.1.

Now, we compute the expected number of uncles per attack cycle which are referred by nephews (honest or not) belonging to the next attack cycle.

Lemma 3.13. *The probability for an attack cycle to end with exactly k consecutive appearances of "H" with $k \geq 1$, conditional that it starts with SS, is pq^{k-1}.*

Proof. Let $k \geq 1$. An attack cycle ω ends with exactly k consecutive appearances of "H" if and only if $\omega = \text{SSw}\overline{\text{H}}$ where w is a Dyck word that ends with exactly $k-1$ "H". The result then follows from Appendix, Lemma 6.4. □

Proposition 3.14. *We have:*

$$\mathbb{E}[V] = \frac{q^2}{p}(1-q^{n_1-1})\gamma + (1-\gamma)pq^2\,\frac{1-(pq)^{n_1-1}}{1-pq}$$

Proof. If an attack cycle ω does not start with two S, then $V(\omega) = 0$. If ω starts with two "S" and ends with exactly k "H" in a row ($k \geq 1$), then only the last n_1-1 blocks can be uncles signaled by future blocks. This happens with probability γ for each block H in this sequence, except for the first block validated by the honest miners if it belongs to this sequence. In this last case, $\omega = \text{SS}\ldots\text{SH}\ldots\text{HH}$ with at most n_1 letters S and $n_1 - 1$ letters "H". So, by Lemma 3.13, we have

$$\mathbb{E}[V] = q^2\sum_{k\geq 1}\inf(k, n_1-1)pq^{k-1}\gamma + (1-\gamma)q\sum_{k=1}^{n_1-1}(pq)^k$$

□

3.3. **Expected revenue of the selfish miner from inclusion rewards.** We compute now $\mathbb{E}[U_h]$.

Proposition 3.15. *We have:*

$$\mathbb{E}[U_h] = p^2q + (p + (1-\gamma)p^2q)\left(\frac{q^2}{p}(1-q^{n_1-1})\gamma + (1-\gamma)pq^2\,\frac{1-(pq)^{n_1-1}}{1-pq}\right)$$

Proof. We have $U_h(\omega) = U_h^{(1)}(\omega) + U_h^{(2)}(\omega)$ where $U_h^{(1)}(\omega)$ (resp. $U_h^{(2)}(\omega)$) counts the number of uncles referred by honest nephews only present in ω (resp. in the next attack cycle after ω). It is clear that $U_h^{(1)}(\mathrm{SHH}) = 1$ and $U_h^{(1)}(\omega) = 0$ if $\omega \neq \mathrm{SHH}$. So,

$$\mathbb{E}[U_h^{(1)}] = p^2q \tag{1}$$

Moreover, given ω, the probability that H is the next official block after ω is $p + (1-\gamma)p^2q$. This happens if and only if the next attack cycle is either H or SHH. If this event occurs, then the first honest block in the next attack cycle will signal the previous uncles created in ω. Therefore, we have

$$\mathbb{E}[U_h^{(2)}] = (p + (1-\gamma)p^2q)\cdot\mathbb{E}[V] \tag{2}$$

Hence, we get the result by (1), (2) and Proposition 3.14. □

Corollary 3.16. *We have*

$$\begin{aligned}\mathbb{E}[U_s] =& q + \frac{q^3\gamma}{p-q} - \frac{pq^2}{p-q}\left(\frac{q}{p}\right)^{n_1-1}\gamma - q^{n_1+1}(1-\gamma)\\ &- \left[p^2q + (p+(1-\gamma)p^2q)\left(\frac{q^2}{p}(1-q^{n_1-1})\gamma + (1-\gamma)pq^2\frac{1-(pq)^{n_1-1}}{1-pq}\right)\right]\end{aligned}$$

Proof. With the same notations as above, we have: $U(\omega) = U_s(\omega) + U_h(\omega)$ and we use Proposition 3.11 and Proposition 3.15. □

3.4. **Apparent hashrate of Strategy 1.** Using Theorem 2.11, Proposition 3.11 and Corollary 3.16 we can plot the region of $(q,\gamma) \in [0, 0.5] \times [0, 1]$ of dominance of the selfish mining Strategy 1 (SM1) over the honest strategy. This corresponds to $\widetilde{q}_E > q$. We obtain Figure 1.

We compute now the expected revenue of the honest miners by attack cycle. We compute first the expected distance between uncles and nephews by attack cycle.

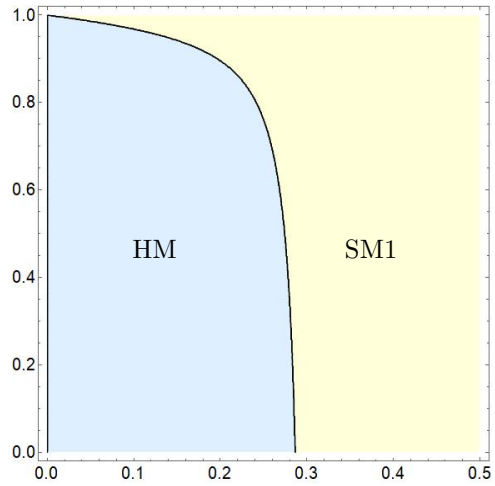

FIGURE 1. Comparing HM and SM1 strategies.

3.5. **Expected distance between uncles and nephews by attack cycle.** If $\mathfrak{b}$ is an uncle, we denote by $\delta(\mathfrak{b})$ the distance between $\mathfrak{b}$ and its nephew. We start by a remark.

Remark 3.17. *Let $\mathfrak{b}$ be an orphan block validated by the honest miners as in Definition 3.1. If $\mathfrak{b}$ is an uncle then $\delta(\mathfrak{b}) = h(\mathfrak{b}) + 1$.*

Definition 3.18. *If $\omega = SSw$ is an attack cycle starting with two blocks S, we set*

$$D(\omega) = \sum_{\mathfrak{b}} (h(\mathfrak{b}) + 1)$$

where the sum is taken over all honest blocks $\mathfrak{b}$ in ω fulfilling Conditions (1) and (2) from Proposition 3.6.

Proposition 3.19. *We have:*

$$\mathbb{E}[D(\omega)|\omega = SS\ldots] = \frac{p}{(p-q)^2}\left(2p - q - \big(p + n_1(p-q)\big)\cdot\left(\frac{q}{p}\right)^{n_1-1}\right)$$

Proof. Let $\omega = \mathrm{SS}w$ be an attack cycle starting with two S with $w = w_1 \ldots w_\nu$ and let X be the associated path according to Note 3.4. In particular, we have $X_\nu = -1$ and $X_i \geq 0$ for $i < \nu$. By Proposition 3.5 and Lemma 6.1 in the Appendix, we

have:

$$\begin{aligned}\mathbb{E}[D(\omega)|\omega = \mathrm{SS}\ldots] &= \mathbb{E}\left[\sum_{i=1}^{\nu}(X_i+3)\cdot\mathbf{1}_{(X_i<n_1-2)\wedge(X_i<X_{i-1})}\right]\\ &= \mathbb{E}\left[\sum_{i=1}^{\nu}X_i\cdot\mathbf{1}_{(X_i<n_1-2)\wedge(X_i<X_{i-1})}\right] + 3\mathbb{E}\left[\sum_{i=1}^{\nu}\cdot\mathbf{1}_{(X_i<n_1-2)\wedge(X_i<X_{i-1})}\right]\\ &= \frac{p}{(p-q)^2}\left(2q-p-\big(q+(n_1-2)(p-q)\big)\cdot\left(\frac{q}{p}\right)^{n_1-1}\right)+\frac{3p}{p-q}\left(1-\left(\frac{q}{p}\right)^{n_1-1}\right)\\ &= \frac{p}{(p-q)^2}\left(2q-p+3(p-q)-\big(q-2(p-q)+n_1\big(p-q\big)+3\big(p-q\big)\big)\cdot\left(\frac{q}{p}\right)^{n_1-1}\right)\end{aligned}$$

Hence, we get the result. □

Definition 3.20. *Let ω be an attack cycle. We set*

$$\Delta(\omega) = \sum_{\mathfrak{b}}\delta(\mathfrak{b})$$

The last sum being taken over all refered uncles in ω.

Proposition 3.21. *We have*

$$\begin{aligned}\mathbb{E}[\Delta] =& pq + \frac{pq^2\gamma}{(p-q)^2}\left(2p-q-\big(p+n_1(p-q)\big)\cdot\big(\frac{q}{p}\big)^{n_1-1}\right)\\ &+\frac{(1-\gamma)q}{p}\left(q(1+p)-(1+n_1p)q^{n_1}\right)\end{aligned}$$

Proof. We proceed as in the proof of Proposition 3.11. If $\omega = H$, then $\Delta(\omega) = 0$. If $\omega \in \{\mathrm{SHS}, \mathrm{SHH}\}$, then, $\Delta(\omega) = 1$. Otherwise, ω starts with two consecutive S. Then, using Lemma 3.9, we get

$$\mathbb{E}[\Delta] = pq^2 + p^2q + (\mathbb{E}[D(\omega)|\omega = \mathrm{SS}\ldots]\gamma + (1-\gamma)(2p+3pq+\ldots+n_1pq^{n_1-2}))\cdot q^2$$

The last term comes from the following fact: when the first honest block present in ω corresponds to an index i satisfying $X_i < n_1+2$, then its contribution to $\mathbb{E}[\Delta]$ is underestimated by $\mathbb{E}[D(\omega)|\omega = \mathrm{SS}\ldots]\gamma$ because it has probability 1 to be an uncle. This only occurs when ω starts with SS...SH with the first k blocks validated by the selfish miner with $k \leqslant n_1$, from where we get the last term. We have:

$$\begin{aligned}2q+3q^2+\ldots+n_1q^{n_1-1} &= -1+\left(\frac{q^{n_1+1}-1}{q-1}\right)' = -1+\left(\frac{q^{n_1+1}}{q-1}\right)'-\left(\frac{1}{q-1}\right)'\\ &= -1+\frac{(n_1+1)q^{n_1}}{q-1}-\frac{q^{n_1+1}}{(q-1)^2}+\frac{1}{(q-1)^2}\\ &= -1+\frac{1}{p^2}+\frac{q^{n_1}}{(q-1)^2}\left((n_1+1)(q-1)-q\right)\\ &= \frac{1-p^2}{p^2}-\frac{q^{n_1}}{p^2}\left(q+(n_1+1)p\right)\\ &= \frac{q(1+p)}{p^2}-\frac{q^{n_1}}{p^2}\left(1+n_1p\right)\end{aligned}$$

So,

$$\left(2p + 3pq + \ldots + n_1 pq^{n_1-2}\right) q^2 = \frac{q}{p}\left(q(1+p) - \left(1 + n_1 p\right) q^{n_1}\right) \tag{3}$$

Hence we get the result using Proposition 3.19. □

3.6. **Deflation.** With the new difficulty adjustment formula, the duration time of an attack cycle in Ethereum is $(\mathbb{E}[L] + \mathbb{E}[U])\tau_1$ where τ_1 is the mean interblock time in Ethereum (which is currently 15 seconds). The number of coins created in an attack cycle is $\left(\mathbb{E}[L] + \frac{7}{8}\mathbb{E}[U] - \frac{1}{8}\mathbb{E}[\Delta] + \mathbb{E}[U]\pi\right) b$ where b is the coinbase in Ethereum. Thus, on average, there is a monetary creation of

$$\frac{\mathbb{E}[L] + \left(\frac{7}{8} + \pi\right)\mathbb{E}[U] - \frac{\mathbb{E}[\Delta]}{8}}{\mathbb{E}[L] + \mathbb{E}[U]}\, b$$

for every inter-block time τ_1, whereas without selfish miner, it is only b on average. So, selfish mining leads to a deflation index

$$\iota = \frac{\left(\frac{1}{8} - \pi\right)\mathbb{E}[U] + \frac{\mathbb{E}[\Delta]}{8}}{\mathbb{E}[L] + \mathbb{E}[U]} \tag{4}$$

Currently we have $\pi = \frac{1}{32}$, thus $\iota > 0$.

3.7. **Apparent hashrate of the honest miners.** Let $\tilde{p}$ be the apparent hashrate of the honest miners in presence of a selfish miner. We have

$$\tilde{p} + \tilde{q} = 1 - \iota \tag{5}$$

where $\tilde{q}$ is the apparent hashrate of the selfish miner. We observe numerically that $\tilde{q} > q - \iota$ for any values of (q, γ). So, even if the attack is not profitable for the selfish miner (case $\tilde{q} < q$) we have $\tilde{p} < p$ which means that the honest miners are impacted by the presence of a selfish miner in the network.

4. Strategy 2A: Brutal Fork signaling all uncles.

We study now another Selfish Mining Strategy (Strategy 2 or SM2): Brutal fork. In this case, the attacker keeps secret his blocks as long as possible and only releases its fork, all at once, at the end of the attack cycle. We call this strategy "brutal fork" because this leads, periodically, to deep reorganizations of the official blockchain. Strategy 2A (or SM2A) corresponds to the case when also the attacker refers all possible uncles.

Proposition 4.1. *We have* $\mathbb{E}[U] = q - q^{n_1+1}$.

Proof. We have $U = 0$ if and only if the attack cycle is H or if it starts with $n_1 + 1$ blocks of type S. Otherwise, we have $U = 1$. So,

$$\mathbb{E}[U] = \mathbb{P}[U > 0] = 1 - (p + q^{n_1+1}) = q - q^{n_1+1}$$

□

We compute now $\mathbb{E}[V]$

Proposition 4.2. *We have* $\mathbb{E}[V] = pq^2 \cdot \frac{1-(pq)^{n_1-1}}{1-pq}$.

Proof. We have $V = 1$ if and only if the attack cycle ω is SS..SH..H with $2 \leq k \leq n_1$ S. In that case, the first H is an uncle signaled by the first future official block in the attack cycle after ω. Otherwise, $V = 0$. So, $\mathbb{E}[V] = pq^2 + \ldots + p^{n_1-1}q^{n_1}$, and we get the result. □

Proposition 4.3. *We have* $\mathbb{E}[U_h] = p^2q + \big(p + (1-\gamma)p^2q\big)\, pq^2 \cdot \frac{1-(pq)^{n_1-1}}{1-pq}$.

Proof. The proof is almost identical as the proof of Proposition 3.15. If $U_h^{(1)}(\omega)$ (resp. $U_h^{(2)}(\omega)$) counts for the number of uncles referred by honest nephews only present in ω (resp. in the attack cycle just after ω), then we have $\mathbb{E}[U_h^{(1)}] = p^2q$, $\mathbb{E}[U_h^{(2)}] = \big(p + (1-\gamma)p^2q\big) \cdot \mathbb{E}[V]$ and $U_h = U_h^{(1)} + U_h^{(2)}$. The only difference is the value of $\mathbb{E}[V]$ which this time is given by Proposition 4.2, and we get the result. □

Corollary 4.4. *We have*

$$\mathbb{E}[U_s] = \mathbb{E}[U] - \mathbb{E}[U_h] = q - q^{n_1+1} - \left(p^2q + \big(p + (1-\gamma)p^2q\big)\, pq^2 \cdot \frac{1-(pq)^{n_1-1}}{1-pq}\right)$$

Note 4.5. When $\gamma = 0$, the two strategies 1 and 2A are identical: in both cases, the honest miners always build blocks on top of honest blocks. So, $\mathbb{E}[U]$, $\mathbb{E}[U_h]$ and $\mathbb{E}[U_s]$ must coincide for $\gamma = 0$. We can check in the different formulas that this is the case. See Propositions 3.11, 3.15,4.1, 4.3 and Corollaries 3.16, 4.4.

4.1. **Apparent hashrate of Strategy 2A.** We use again Theorem 2.11 and we plot in parameter space in Figure 2 the region of $(q, \gamma) \in [0, 0.5] \times [0, 1]$ comparing Selfish Mining Strategy 2A to the honest strategy.

We observe that if $\gamma = 0$ then we have SM2A is superior to honest mining when $q > 28.65\%$. Also we have for all values of q and γ that SM2A is superior to SM1. Therefore it is never profitable for the attacker to engage in competitions with the honest miners.

4.2. **Apparent hashrate of the honest miners.** We compute first the expected distance between an uncle and its nephew. We keep the same notation for Δ as in Definition 3.20.

Proposition 4.6. *We have* $\mathbb{E}[\Delta] = \frac{q}{p}\big(q(1+p) - \big(1 + n_1p\big)q^{n_1}\big)$

Proof. If an attack cycle ω starts with S...SH with k S, $k \leq n_1$, then there is exactly one uncle in ω and its distance to its nephew is k. In any other cases, there is no uncle in ω. Therefore, $\mathbb{E}[\Delta] = \sum_{k=1}^{n_1} kpq^k$ Hence we get the result by (3). □

The apparent hashrate $\tilde{p}$ of the honest miners is $\tilde{p} = 1 - \tilde{q} - \iota$ with ι given by (4). Numerically, we observe that we have always $\tilde{p} < p$ except in a tiny region when q and γ is small ($q < 6\%$ and $\gamma < 22\%$).

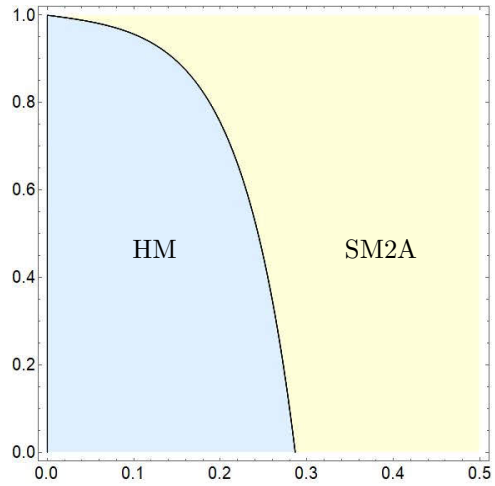

FIGURE 2. Comparing HM and SM2A strategies.

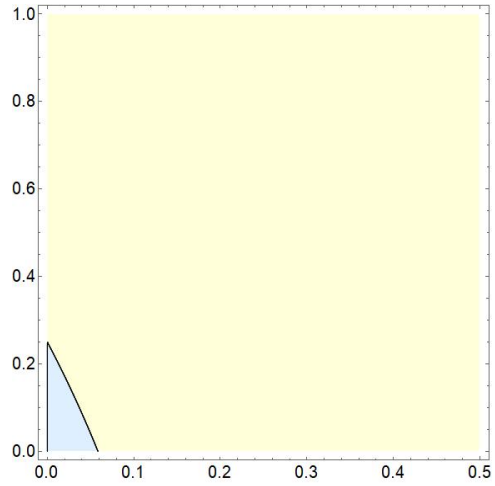

FIGURE 3. Comparing $\tilde{p}$ and p: The honest miners are negatively affected even when the attack is not profitable for the selfish miner except for a tiny region around $(0, 0)$ (case SM2A).

5. Strategy 2B: Brutal Fork without signaling uncles.

In this strategy, the attacker signals no uncles in order to maximize the impact on the difficulty adjustment formula. In that case we have $U_s = 0$. In our analysis of the profitability of the strategy, we need to consider another important rule of Ethereum's protocol: a nephew can only signal at most two uncles. Instead of computing $\mathbb{E}[U]$, it is simpler to compute $\mathbb{E}[U']$ where $U'(\omega)$ is defined as the number of signaled uncles with nephews in ω. We have,

$$\mathbb{E}[U] = \mathbb{E}[U'] \tag{6}$$

Since the attacker does not signal uncles, we have $U'(\omega) = 0$ if $\omega \notin \{H, \mathrm{SHH}\}$.

To ease notations, we set $U'(H)$ for $U'(\{H\})$.

Lemma 5.1. *We have:*

$$\begin{array}{rcl}\mathbb{P}[U'(H)=1] & = & \sum_{i=2}^{n_1-2}\left(1-pq^2-pq^2C_{n_1-2-i}(pq)\right)\pi_i+\pi_{n_1-1}+\pi_{n_1}\\ \mathbb{P}[U'(H)=2] & = & \sum_{i+j\leq n_1}\pi_i\pi_j\\ \mathbb{P}[U'(H)\geq 3] & = & 0\end{array}$$

Proof. We have $U'(H)=1$ if and only if the two last attack cycles before H are in the following order from the oldest to the most recent one: ω' and ω such that:

(1) ω won by the attacker with $L(\omega)\leqslant n_1$.
(2) ω' won by the honest miners or by the attacker but with $L(\omega')>n_1-L(\omega)$.

Note that if $L(\omega)\geqslant n_1-1$ then (2) is automatically satisfied. So,

$$\mathbb{P}[U'(H)=1]=\sum_{i=2}^{n_1-2}\left(1-pq^2-pq^2C_{n_1-2-i}(pq)\right)\pi_i+\pi_{n_1-1}+\pi_{n_1}$$

In the same way, we have $U'(H)=2$ if and only if the two last attack cycles before H are ω' and ω such that ω' and ω are both won by the attacker with $L(\omega)+L(\omega')\leq n_1$. Indeed, a block can only refer at most two uncles. Hence, we get the result. □

Example 5.2. For $n_1=6$, we have using Example 2.5:

$$\begin{array}{rl}\mathbb{P}[U'(H)=1]= & \pi_5+\pi_6+\sum_{i\leqslant 4}(1-pq^2-pq^2C_{4-i}(pq))\pi_i\\ & =pq^2\left(14p^4q^4+p^3(5-9q)q^3+2p^2(1-2q)q^2+p\left(q-4q^2\right)+2\right)\\ \mathbb{P}[U'(H)=2]= & \pi_2^2+2\pi_2\pi_3+2\pi_2\pi_4+\pi_3^3\\ & =p^2q^4\left(5p^2q^2+2pq+4\right)\\ \mathbb{P}[U'(H)\geq 3]= & 0\end{array}$$

Definition 5.3. *We define* $P_{n_1}(p,q)=\mathbb{E}[U'(H)]$.

Example 5.4. When $n_1=6$, we have by Example 5.2:

$$P_6(p,q)=pq^2\left(14p^4q^4+p^3(q+5)q^3+2p^2q^2+p(4q+1)q+2\right)$$

Lemma 5.5. *We have*

$$\mathbb{E}[U'(\omega)|\omega=SHH]=(P_{n_1}(p,q)+1)\cdot(1-\gamma)+(pq^2+pq^2C_{n_1-3}(pq)+1)\cdot\gamma$$

Proof. Suppose that $\omega=$SHH. We have two cases: The second honest block can be built on top of a block validated by the selfish miner or not. If the first official block of ω is honest, then it signals any uncle which is at distance less or equal than n_1, like in the previous situation. Moreover, the first block mined by the selfish miner is an uncle signaled by the second block mined by the honest miners. This gives the first term of the right hand side. If the first official block of ω is a block mined by the attacker, then the first block validated by the honest miners is an

uncle signaled by the second block mined by the honest miners. This last block will also signal another uncle which is at distance less than $n_1 - 1$ of the first official block of ω. There is such an uncle if and only if the attack cycle ω' before ω is an attack cycle won by the attacker with $L(\omega') \leq n_1 - 1$. This gives the second term of the right hand side. Hence, we get the result. □

Theorem 5.6. *We have*

$$\mathbb{E}[U] = (p + (1-\gamma)p^2q)P_{n_1}(p,q) + \gamma p^2 q\left(pq^2 + pq^2 C_{n_1-3}(pq)\right) + p^2 q$$

Proof. We have $\mathbb{E}[U] = \mathbb{E}[U']$ and

$$\begin{aligned}\mathbb{E}[U'] &= \mathbb{E}[U'(\omega)|\omega = H]\mathbb{P}[\omega = H] + \mathbb{E}[U'(\omega)|\omega = \text{SHH}]\mathbb{P}[\omega = \text{SHH}] \\ &= P_{n_1}(p,q)p + (P_{n_1}(p,q)+1)\cdot(1-\gamma)p^2q + (pq^2 + pq^2C_{n_1-3}(pq)+1)\cdot\gamma p^2 q\end{aligned}$$

□

5.1. **Apparent hashrate of Strategy 2B.** The computation of $\mathbb{E}[U]$ is a polynomial expression in p and q that can be carried out with the help of a computer algebra system. We plot in parameter space in Figure 3 the region of $(q,\gamma) \in [0, 0.5] \times [0, 1]$ comparing Selfish Mining Strategies 2A and 2B, and honest mining. We also compare SM1, SM2A and SM2B in Figure 4.

We observe that if $\gamma = 0$ then we have SM2B is superior to honest mining when $q > 28.80\%$. Also, for $q > 30.13\%$ we have that SM2B is even better than SM2A (whathever γ is). Thus, in this case, the attacker does not even need to bother to signal blocks.

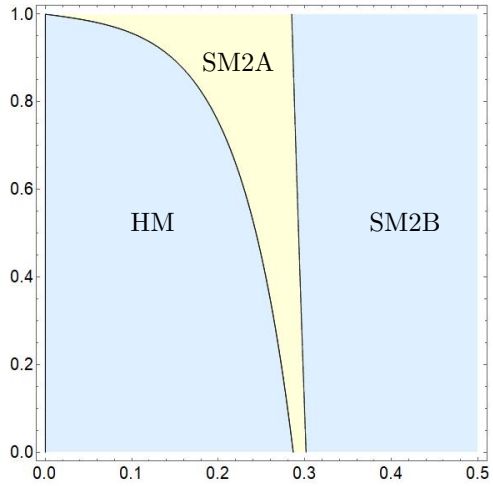

FIGURE 4. Comparing the strategies HM, SM2A and SM2B.

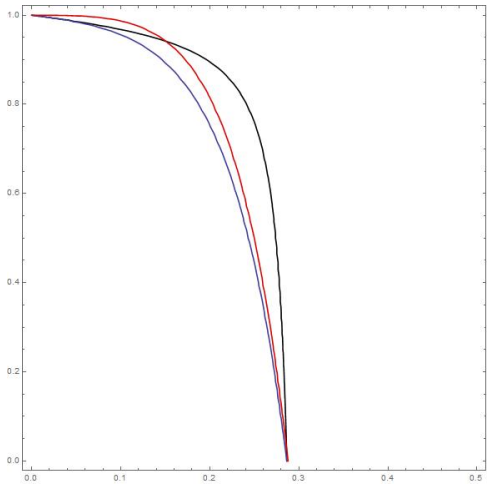

FIGURE 5. Comparing the strategies SM1 (black), SM2A (blue) and SM2B (red).

5.2. **Apparent hashrate of the honest miners.**

Definition 5.7. *If ω is an attack cycle, we denote by $\Delta'(\omega)$ the average number of the distance between a nephew belonging to ω and an uncle which does not necessarily belong to ω.*

In a similar way as before, we prove:

Lemma 5.8. *We have*

$$\mathbb{E}[\Delta'(\omega)|\omega = H] = \sum_{|\mathbf{i}|\leqslant n_1} \left(\sum_j j \cdot i_j\right) \left(1 - pq^2 - pq^2 C_{n_1-2-|\mathbf{i}|}(pq)\right) \prod_j \pi_{i_j}$$

Definition 5.9. *We define $Q_{n_1}(p,q) = \mathbb{E}[\Delta'(\omega)|\omega = H]$*

The same computations as in the previous section leads to

$$Q_5(p,q) = pq^2\left(25p^3q^3 + 20p^2q^3 + 8p^2q^2 + 16pq^2 + 3pq + 4\right)$$
$$Q_6(p,q) = pq^2\left(84p^4q^4 + 54p^3q^4 + 25p^3q^3 + 96p^2q^4 + 20p^2q^3 + 8p^2q^2 + 16pq^2 + 3pq + 4\right)$$

This enables us to compute $\mathbb{E}[\Delta']$ using the following result with $n_1 = 6$.

$$\mathbb{E}[\Delta'] = (p + (1-\gamma)p^2q)Q_{n_1}(p,q) + \gamma p^2 q Q_{n_1-1}(p,q) + p^2 q \ .$$

Finally, we note that $\mathbb{E}[\Delta] = \mathbb{E}[\Delta']$. From here, we get the apparent hashrate of the honest miners using (4) and (5). We observe numerically that we have always $\tilde{p} < p$.

6. Conclusions

We have given closed-form formulas for the long term profitability of different selfish mining strategies in the Ethereum network. This is combinatorially more complex than in Bitcoin network which has a simpler reward system. Precisely, the particular reward system that incentives signaling blocks is an effective countermeasure to Selfish mining but only when the count of uncle blocks are incorporated into the difficulty adjustment formula (this is the case for the current implementation of the difficulty adjustment formula). This analysis provides a good illustration of the fact that selfish mining is an attack on the difficulty adjustment formula. We study, for the first time, selfish mining strategies that do not signal any blocks. We prove that they are the most profitable ones in the long run. It may appear counter-intuitive that refusing the signaling fees is the most profitable strategy with the current reward parameters when q is larger than 30%. But this is explained again because selfish mining is an attack on the difficulty adjustment formula.

Appendix

6.1. **Random walk.**

We compute the expected numbers of descents in a biased random walk conditional to be bounded by a fixed bound.

Lemma 6.1. *Let (X_k) be a biased random walk starting from $X_0 = 0$ with $\mathbb{P}[X_{k+1} = X_k + 1] = q$ and $\mathbb{P}[X_{k+1} = X_k - 1] = p$ for $k \in \mathbb{N}$, with $p + q = 1$ and $q < p$. Let $\nu(X)$ be the stopping time defined by $\nu(X) = \inf\{i \geq 0; X_i = -1\}$, and for $n \geq 0$, let*

$$u_n(X) = \sum_{i=1}^{\nu} \mathbf{1}_{(X_i<n)\wedge(X_i<X_{i-1})}$$
$$v_n(X) = \sum_{i=1}^{\nu} X_i \cdot \mathbf{1}_{(X_i<n)\wedge(X_i<X_{i-1})}$$

Then we have

$$u_n = \mathbb{E}[u_n(X)] = \frac{p}{p-q}\left(1 - \left(\frac{q}{p}\right)^{n+1}\right) \tag{7}$$

$$v_n = \mathbb{E}[v_n(X)] = \frac{p}{(p-q)^2}\left(2q - p - \big(q + n(p-q)\big) \cdot \left(\frac{q}{p}\right)^{n+1}\right) \tag{8}$$

Proof. We have $u_0 = 1$ (resp. $v_0 = -1$). If $X_1 = -1$, then we have $u_n(X) = 1$ (resp. $v_n(X) = -1$). If $X_1 = 1$, then

$$\begin{aligned} u_n(X) =& \sum_{i=1}^{\nu'} \mathbf{1}_{(X'_i<n-1)\wedge(X'_i<X'_{i-1})} + \sum_{i=1}^{\nu''} \mathbf{1}_{(X''_i<n)\wedge(X''_i<X''_{i-1})} \\ =& u_{n-1}(X') + u_n(X'') \end{aligned}$$

with

$$\begin{aligned} X'_i =& X_{i+1} - 1 \\ \nu' =& \inf\{i > 0; X'_i = -1\} \\ X_i^{''} =& X'_{i+\nu'} - X'_{\nu'} \\ \nu^{''} =& \inf\{i > 0; X_i^{''} = -1\} \end{aligned}$$

By the Markov property, X' and $X^{''}$ are two independant simple biased random walk with a probability p (resp. q) to move to the left (resp. right). So, taking expectations, we get:

$$u_n = p \cdot 1 + q \cdot (u_{n-1} + u_n)$$

which is equivalent to

$$u_n - \frac{p}{p-q} = \left(\frac{q}{p}\right)\left(u_{n-1} - \frac{p}{p-q}\right)$$

So we get (7) by induction on n. In the same way, we have:

$$\begin{aligned} v_n(X) =& \sum_{i=1}^{\nu'} (X'_i + 1) \cdot \mathbf{1}_{(X'_i < n-1) \wedge (X'_i < X'_{i-1})} + \sum_{i=1}^{\nu"} X_i^{''} \cdot \mathbf{1}_{(X_i^{''} < n) \wedge (X_i^{''} < X_{i-1}^{''})} \\ =& u_{n-1}(X') + v_{n-1}(X') + v_n(X^{''}) \end{aligned}$$

Taking expectations again, we get

$$v_n = p \cdot (-1) + q \cdot (u_{n-1} + v_{n-1} + v_n) \tag{9}$$

Set $c_n = \left(\frac{p}{q}\right)^n v_n$. Then, (9) leads to

$$\begin{aligned} c_n =& c_{n-1} + \left(\frac{p}{q}\right)^{n-1} u_{n-1} - \left(\frac{p}{q}\right)^n \\ =& c_{n-1} + \left(\frac{2q-p}{p-q}\right) \cdot \left(\frac{p}{q}\right)^n - \frac{q}{p-q} \end{aligned}$$

So, by induction, we get

$$c_n = c_0 + \left(\frac{2q-p}{p-q}\right) \cdot \left(\frac{p}{q}\right) \cdot \frac{\left(\frac{p}{q}\right)^n - 1}{\left(\frac{p}{q}\right) - 1} - \frac{nq}{p-q}$$

After rearranging terms, we get (8). □

6.2. **Dyck words.** Let $\mathcal{D}$ be the space of Dyck words based on the alphabet $\{S, H\}$. If $w = w_1 \dots w_{2k}$ with $k \in \mathbb{N}$, then we define $|w| = k$. We have proved in [6] that we can endow $\mathcal{D}$ with a probability measure $\bar{\mathbb{P}}$ given by $\bar{\mathbb{P}}[w] = p(pq)^{|w|}$ for $w \in \mathcal{D}$. Note that $\bar{\mathbb{P}}[w]$ can be interpreted as the probability that a simple biased random walk X starting from 0 and stopping at -1 follows exactly the path given by w i.e., $X_i = X_{i-1} + 1$ (resp. $X_i = X_{i-1} - 1$) if $w_i = S$ (resp. $w_i = H$) for $i \leqslant 2|w|$ and $X_{2|w|+1} = -1$.

Lemma 6.2. *Let $n \geq 0$ and $\mathcal{D}_n = \{w; |w| \leqslant n\}$. Then, $\bar{\mathbb{P}}[\mathcal{D}_n] = pC_n(pq)$ where $C_n(x)$ is the n-th partial sum of the generating series $C(x)$ of the Catalan numbers.*

Proof. We have

$$\bar{\mathbb{P}}[\mathcal{D}_n] = \sum_{w\in\mathcal{D}_n} p(pq)^{|w|} = p\sum_{k=0}^{n}\sum_{|w|=k}(pq)^k = p\sum_{k=0}^{n} C_k(pq)^k = pC_n(pq)$$

□

We can make more precise Proposition 2.3.

Proposition 6.3. *Let $\omega = SSwH$ be an attack cycle starting with SS. Then, $w \in \mathcal{D}$ and $\mathbb{P}[\omega] = q^2\bar{\mathbb{P}}[w]$*

Lemma 6.4. *The probability that a Dyck word ends with the subsequence SHH..H with n letters H at the end is pq^n.*

Proof. Consider the "reversal" map $\sigma : \quad \mathcal{D} \longrightarrow \mathcal{D}$ given by

$$w = w_1 \dots w_{2|w|} \longmapsto \sigma(w) = \tilde{w} = \tilde{w}_{2|w|} \dots \tilde{w}_1$$

with $\tilde{w}_i = S$ (resp. H) if $w_i = H$ (resp. S). Then σ is one to one and preserves $\bar{\mathbb{P}}$ i.e., for $w \in \mathcal{D}$, we have $\bar{\mathbb{P}}[\sigma(w)] = \bar{\mathbb{P}}[w]$. So, the probability that a Dyck word ends exactly with n letter(s) H is the same as the probability that a Dyck word starts with n letter(s) S and then is followed by a letter H. Thus this probability is pq^n. □

For $w \in \mathcal{D}$, we define $f(w) = \inf\{i \geq 0; w_i = H\}$.

Lemma 6.5. *Let $n \geq 0$ and $E = \{w \in \mathcal{D}; f(w) \leqslant \inf\{|w|, n\}\}$. Then we have*

$$\bar{\mathbb{P}}[E] = (1-q^n) - \frac{p(1-(pq)^n)}{1-pq}$$

Proof. Let $w \in \mathcal{D}$. To have $f(w) \leqslant |w|$ means that at least one H is followed by an S i.e., w is not of the form SS...SHH...H. For all integer $k \leqslant n$, we have

$$\Sigma_{w;(f(w)=k)\wedge(f(w)\leqslant|w|)}(pq)^{|w|} = pq^{k-1}\cdot\sum_{j=0}^{k-2} qp^j$$

So, if we consider a biased random walk starting from 0 with a probability p to move to the left (resp. right) then both terms represent the probability of the following event: We have $k-1$ first step(s) to the right, then $j+1$ steps to the left with $0 \leqslant j \leqslant k-2$ and then at least one step to the right before reaching 0. So, we have

$$\begin{aligned}
\bar{\mathbb{P}}[E] &= \sum_{k=1}^{n} pq^{k-1}\cdot\sum_{j=0}^{k-2} qp^j \\
&= p\sum_{k=1}^{n} q^{k-1}\cdot(1-p^{k-1}) \\
&= p\sum_{k=1}^{n} q^{k-1} - p\sum_{k=1}^{n}(pq)^{k-1} \\
&= (1-q^n) - \frac{p(1-(pq)^n)}{1-pq}
\end{aligned}$$

□

6.3. **Glossary.**

6.3.1. *Revenue ratio and apparent hashrate.* The revenue ratio $\tilde{\Gamma}$ of a miner following a strategy with repetitions of attack cycles like selfish mining is given by $\tilde{\Gamma} = \frac{\mathbb{E}[R]}{\mathbb{E}[T]}$ where R (resp. T) is the revenue of the miner after an attack cycle (resp. the duration time of an attack cycle). The apparent hashrate $\tilde{q}$ is defined by $\tilde{q} = \tilde{\Gamma}\frac{\tau}{b}$ where b (resp. τ) is the coinbase (resp. interblock time).

6.3.2. *Terminology.* Ethereum has a special terminology that we summarize.

Uncle	orphan block whose parent belongs to the official blockchain
Nephew	regular block that refers to an "uncle" which is at a distance less than n_1
Distance	number of official blocks between a nephew N and a parent's uncle U.

6.3.3. *Mining reward.* If an uncle U is referred by a nephew N which is at a distance d, then U earns an "uncle reward" which is worth $K_u(d)b$ and N gets an additional reward of $K_n(d)b$, where b is the coinbase. Also, a nephew can refer at most two uncles. Today, on Ethereum, we have $b = 2$ ETH, $K_u(d) = \frac{8-d}{8} \cdot \mathbf{1}_{d \leqslant n_1}$ with $n_1 = 6$ and $K_n(d) = \pi = \frac{1}{32}$.

Uncle reward	reward granted to an uncle block referred by a nephew
inclusion reward	additional reward granted to a nephew that refers an uncle

References

[1] Alf Zugenmaier, Fabian Ritz. *The impact of uncle rewards on selfish mining in ethereum*, IEEE Symposium on Security and Privacy, p.50-57, 2018.

[2] Emin Gun Sirer, Ittay Eyal. *Majority is not enough: bitcoin mining is vulnerable*, International Conference on Financial Cryptography and Data Security, p.436-454, 2014.

[3] Chen Feng, Jianyu Niu. *Selfish mining in ethereum*, ArXiv:1901.04620, 2019.

[4] Cyril Grunspan, Ricardo Pérez-Marco. *On profitability of selfish mining*, ArXiv:1805.08281v2, 2018.

[5] Cyril Grunspan, Ricardo Pérez-Marco. *On profitability of trailing mining*, ArXiv:1811.09322, 2018.

[6] Cyril Grunspan, Ricardo Pérez-Marco. *Bitcoin selfish mining and Dyck words*, ArXiv:1902.01513, 2019.

[7] Cyril Grunspan, Ricardo Pérez-Marco. *Selfish mining and Dyck words in Bitcoin and Ethereum networks*, To appear in Tokenomics Conf. Proceedings, ArXiv:1904.07675, 2019.

Léonard de Vinci, Pôle Univ., Research Center, Paris-La Défense, Labex Réfi, France
E-mail address: cyril.grunspan@devinci.fr

CNRS, IMJ-PRG, Labex Réfi, Paris, France
E-mail address: ricardo.perez.marco@gmail.com

Author's Bitcoin Beer Address (ABBA)[1]: 1KrqVxqQFyUY9WuWcR5EHGVvhCS841LPLn

[1] Send some bitcoins to support our research at the pub.