HAL
open science

Profit lag and alternate network mining

Cyril Grunspan, Ricardo Pérez-Marco

To cite this version:

Cyril Grunspan, Ricardo Pérez-Marco. Profit lag and alternate network mining. MARBLE 2023, Jul 2023, London, United Kingdom. hal-02959121

HAL Id: hal-02959121

https://hal.science/hal-02959121v1

Submitted on 6 Oct 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

PROFIT LAG AND ALTERNATE NETWORK MINING

CYRIL GRUNSPAN AND RICARDO PÉREZ-MARCO

Abstract. For a mining strategy we define the notion of "profit lag" as the minimum time it takes to be profitable after that moment. We compute closed forms for the profit lag and the revenue ratio for the strategies "selfish mining" and "intermittent selfish mining". This confirms some earlier numerical simulations and clarifies misunderstandings on profitability in the literature. We also study mining pairs of PoW cryptocurrencies, often coming from a fork, with the same mining algorithm. This represents a vector of attack that can be exploited using the "alternate network mining" strategy that we define. We compute closed forms for the profit lag and the revenue ratiofor this strategy that is more profitable than selfish mining and intermittent selfish mining. It is also harder to counter since it does not rely on a flaw in the difficulty adjustment formula that is the reason for profitability of the other strategies.

*Savez-vous comment on fait son chemin ici ?**

1. Introduction

1.1. **Nakamoto consensus.** The founding paper "Bitcoin, a peer-to-peer electronic cash system" was announced at the end of October 2008 on a cryptography mailing list [21, 22]. A first version of the software implementing the new protocol developed in the article was then released on January 2009 [23].

Bitcoin gradually enjoyed worldwide success and is today the cornerstone of the new crypto-economy valuated at several hundreds of billions of dollars.

Two main reasons have contributed to its success. The decentralization of the network allows to transfer value on the internet from user to user without the assistance of a third party. Therefore, no central server, nor bank, nor jurisdiction can block payments [22]. A second reason is the invention of "smart-contracts" which made

Date: September 30, 2020.

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

Key words and phrases. Bitcoin; proof-of-work; selfish mining; intermittent selfish mining; smart mining.

*Balzac, *Le Père Goriot.*

bitcoin the first programmable currency [1, 29]. This allows to construct the "Lightning Network", on top of the Bitcoin network that offers instant and secure money transfer [6].

Bitcoin consensus, sometimes referred today as Nakamoto Consensus, is based on the use of proof of work (PoW). PoW was originally invented to fight e-mail spam [7]. It was then used by H. Finney in the design of RPOW, the first cryptocurrency based on PoW, but was not decentralized. PoW plays a crucial role to secure Bitcoin protocol and in the minting process of new bitcoins. Academically speaking it combines computer security and classical probability theory. The main problem solved by Bitcoin and its PoW design is the prevention of double spend attacks without relying on a central server. Instead of searching for a flawless deterministic distributed consensus, Nakamoto's point is that, under reasonable conditions, the probability of success of such attacks is negligible [22, 28, 13, 11] and economically non-profitable [17].

1.2. **Mining Process.** At any time, miners are working to build a new block from new transactions. This is achieved solving a laborious cryptographic puzzle which involves heavy computation and use of energy. Miners iterate the calculation of images of a cryptographic hash function of data from a new block by varying a "nonce" (and an "extra-nonce") until finding a result fulfilling certain criterion.

If successful, the miner broadcasts his discovery to the rest of the network which checks that the solution is legit. Then the new block is added to the previous known chain of blocks. The miner is then rewarded in newly minted bitcoins determined by the protocol and by the transaction fees of transactions included in the validated block. This sequence of blocks forms the "Bitcoin blockchain". This is a secured distributed ledger recording all past validated transactions [30].

1.3. **Selfish Mining.** On the early days, the community of bitcoiners met in different internet forums, among them bitcointalk.org, a forum originally created by Nakamoto in November 2009. In particular, they tried to understand in depth how Bitcoin works. Some people had doubts of the specific point in the protocol that requires miners to broadcast their blocks as soon as they are validated through the proof-of-work [26]. This fact allows the blockchain to record several thousand transactions every ten minutes in average. If a miner witholds secretly a block, he risks to loose the reward awarded to this block (coinbase and transaction fees) to a faster miner.

It is implicit in the founding paper that Nakamoto believed this to be the optimal strategy for miners [22]. Moreover, it is required in a decentralized protocol that the private economic interests of participants are in line with the protocol rules.

However, several deviant strategies were proposed in the bitcointalk.org forum and a seminal paper by Rosenfeld examined the problem in 2011 [27]. Then by the end of 2013, two articles showed that other alternative strategies than the honest one can be more profitable under suitable conditions [8, 2].

By modeling the progression of the blockchain with a Markov chain, the authors showed that a certain deviant strategy, the so-called " selfish mining " is more profitable in the long run than honest mining [8]. This is in particular the case when a mining pool detains more than 33 % of the whole network hashpower.

The assumptions underlying the result, such as the cohesion of mining pools, were challenged. In particular, the fact that the model considers only two groups of miners and only one selfish mining pool in the network [9, 3].

These objections are founded. For example, one could imagine that some miners participating in a mining pool decide to selfish mine not only against honest miners but also against their own mining pool.

Using new martingale techniques and a rigorous profitability model, it was proved in [14] that the true reason for the profitability of selfish mining resides in a flaw in the difficulty adjustment formula. An important result obtained by these new techniques is that without difficulty adjustment, the honest mining strategy is the most profitable one, hence vindicating Nakamoto's original belief. The flaw in the difficulty adjustment formula can be easily corrected. In the presence of a block-withholding attacker, the network, that does not track the production of orphan blocks, underestimates the real hash power deployed. Ultimately this makes the production of blocks easier than normal, which boosts miners income.

In particular, the selfish mining strategy only becomes profitable after a first difficulty adjustment, and after recouping all the cost employed reducing the difficulty (at the expense of regular honest mining income). At first, blocks are generated at a slower pace and all miners have their profitability reduced. Then after a first difficulty adjustment, blocks arrive faster and the selfish miner profits.

This theoretically explains why the strategy has never been implemented. No miner risks to mine at a loss for several weeks while assuming that mining conditions remain the same i.e., one selfish miner and same overall hash power over time. Moreover, if a miner has substantial hashpower, then probably double spend attacks based on 51% hashrate dominance are easier to achieve by increasing the hashpower.

The original selfish mining strategy assumes also the non-realistic hypothesis that there is no arrival of new miners to the network attracted by the lower mining difficulty.

1.4. **Smart mining.** As explained, the advantage of selfish mining is based on reducing the difficulty but continue to profit from block rewards, even before the first difficulty adjustment. Another idea presented in [12] consists in withdrawing temporarily from the network to lower the difficulty. Then come back after a difficulty adjustment to take advantage of the lower hash rate mining. The authors call such a strategy "smart mining" (sic). By considering the cost of mining (fixed cost and variable cost per time unit), it has been shown that this strategy can be more profitable than honest mining, even for low hashrates. In some sense, this attack is more serious than the selfish mining strategy because it does not exploit a particular flaw of the difficulty adjustment formula.

1.5. **Intermittent selfish mining.** Another possible strategy is the strategy of "Intermittent selfish mining" (ISM) that consists in alternating selfish and honest mining during consecutive difficulty periods. This was early discussed in social media and then numerically studied in [25].

The idea is to fully profit from the decrease of the difficulty but the downside is that the difficulty does not stabilize and recovers after each phase of honest mining.

1.6. **Alternate network mining.** In this article we introduce and study a strategy similar to "smart mining" that we name "alternate network mining" (ANM). The difference is that when the miner withdraws from the network he goes on to mine in another network with similar mining algorithm. Clearly, if honest mining is profitable, "alternate network mining" is more profitable than "smart mining", and we prove that it is also more profitable than "selfish mining" and "intermittent selfish mining". It is also effective for low hashrates. Therefore, the existence of cryptocurrencies operating with identical PoW algorithms represent a vector of attack for both networks (this was already observed in [14]). Our study focus mainly on Bitcoin and Bitcoin Cash, but can be adapted to other pairs such as BCH / BSV or ETH / ETC.

1.7. **Organization of this article.** We start briefly recalling the mathematics of Bitcoin mining and the profitability model for comparing mining strategies [14, 18]. Then, we review the selfish mining strategy and show the equivalence between the Markov chain approach and the martingale approach for the computation of profitabilities [8, 14]. Also we review how to compute time before profit with both approaches.

Then, we turn to ISM strategy and compute in closed form the apparent hashrate of the strategy as well as the time before profit. These results are new since before only numerical simulations were available. In passing we correct false claims and misunderstanding of the authors of [25]. Then, we compute the profitability of the ANM strategy. The present article is self contained.

2. Modelization

2.1. **Mining and difficulty adjustment formula.** From new transactions collected in the local database (the "mempool"), a miner builds a block B containing a trace of a previous block and a variable parameter (called a "nonce") until he finds a solution to the inequality $h(B) < \frac{1}{\delta}$ where δ is the difficulty of the hashing problem and h is a cryptographic hash function ($h = SHA_{256} \circ SHA_{256}$ for Bitcoin). This solution is the "proof-of-work" because it proves, under the usual assumptions on the pseudorandom properties of the hash function, that the miner has spend energy. For Bitcoin, the difficulty is adjusted every 2016 blocks, which corresponds on average to 2 weeks. For Bitcoin Cash and for for Bitcoin SV, the difficulty is adjusted at each block using timestamps of the previous 144 blocks (about one day). For Ethereum, the difficulty is adjusted at each block using the lapse for the discovery of the previous block. Ethereum difficulty adjustment algorithm was modified in June 2017 [4]. Since then, the adjustment takes into account the production of special orphan blocks also called "uncles". These are orphan blocks that are not too far from the "parent block". Ethereum Classic did not implement this new difficulty adjustment formula, and is more vulnerable to selfish mining [10, 19]

2.2. **Notations.** In the article, we use Satoshi's notations from his founding paper. Thus, we denote by p (resp. $q = 1 - p$) the relative hash power of the honest miners (resp. the attacker). So, at any time, q is the probability that the attacker discovers a new block before the honest miners. We also denote by $N(t)$ (resp. $N'(t)$) the counting process of blocks validated by the honest miners (resp. the attacker) during a period of t seconds from an origin date 0. The hash function SHA_{256} is assumed to be perfect, the time it takes to find a block follows an exponential distribution and hence $N(t)$ (resp. $N'(t)$) is a Poisson process. Initially (before a difficulty adjustment) the parameter of the Poisson process $N(t)$ (resp. $N'(t)$) is $\frac{p}{\tau_0}$ (resp. $\frac{q}{\tau_0}$) with $\tau_0 = 600$ seconds. We also denote by S_n (resp. S'_n) the time it takes for the honest miners (resp. the attacker) to mine n blocks, and $\tilde{S}_n$ the time it takes for the whole network to add n blocks to the official blockchain. The random variables S_n and S'_n follow Gamma distributions. The same occurs for $\tilde{S}_n$ when the attaker mines honestly (see [13]). The multiplicative difficulty adjustment parameter is denoted by δ. It is updated every $n_0 = 2016$ blocks. Thus, $\tilde{S}_{n_0}$ represents a complete mining period of n_0 official blocks and at this date, the protocol proceeds with a new difficulty.

We use the notation γ for the connectivity of the attacker. In case of a competition between two blocks, one of which has been mined and published by the attacker, γ represents the fraction of honest miners mining on top of the attacker's block.

2.3. **Profitability of a mining strategy.** We consider a miner which is active over a long period compared to the average mining time of a block. What counts for his business is its Profit and Loss (PnL) per unit of time. We denote by $R(t)$ the total income of the miner between time 0 and time $t > 0$. Similarly, we denote by $C(t)$ the total cost he incurs during this period. Note that $C(t)$ is not restricted to direct mining costs but also includes all expenditures. So, his PnL is $PnL(t) = R(t) - C(t)$ and seeks to maximize $\frac{PnL(t)}{t} = \frac{R(t)}{t} - \frac{C(t)}{t}$ for $t \to \infty$. We assume that the mining cost is independent of the mining strategy. Indeed, whether the miner broadcasts his blocks or keeps them secret, it has no impact on its mining costs. Also, whether he mines a certain cryptocurrency or another with the same hash algorithm, this does not change the mining cost per unit of time. This last quantity essentially depends on electricity costs, price of his machines, salaries of employees, etc. In particular, when the miner mines at full regime, then the cost of mining per unit of time does not depend on the strategy. In this situation, the relevant quantity is the revenue ratio $\Gamma = \lim_{t\to+\infty} \frac{R(t)}{t}$. A strategy ξ is more profitable than a strategy ξ' if and only if its revenue ratio is greater: $\Gamma\big(\xi\big) \geq \Gamma\big(\xi'\big)$ (see Corollary 2.4 below).

Note 2.1. *The literature is often obscure about profitability model and sometimes disregards mining costs without proper justification. For example, in* [25] *one can read: "We omit transaction fees and mining costs from our analysis". But, as proved in* [13]*, the only necessary assumption in order to compare strategies is to have the mining cost per unit of time to be the same for both strategies. Otherwise, the profitability analysis without considering costs does not make sense. Thus, in the smart mining strategy considered in* [12]*, the authors are naturally led to consider mining costs which depend on the miner being active or not. In the alternate network mining strategy considered in Section 5, the miner never remains inactive assuming that the mining costs are equal in both networks.*

2.4. **Attack cycles.** A strategy consists in attack cycles. The end of a cycle is determined by a *stopping time*. At the start of a cycle, the attacker and the honest miners have the same view of the official blockchain and mine on top of the same block. In general, thought this is not mandatory, during the attack cycle, the attacker mines on a fork that he keeps secret.

Example 2.2. *The sequence SSSHSHH corresponds to a particular attack cycle for the selfish mining strategy: the attacker first mines three blocks in a row that are kept secret (blocks "S"); then the honest miners mine one (block "H"); then the attacker mines another one (still secret); then the honest miners mine two blocks in a row and so the attacker decides to publish his entire fork because he only has a lead of 1 on the official blockchain. The attack cycle then ends. In this case, the attacker is victorious: all the blocks he has mined end up in the official blockchain.*

The attacker iterates attack cycles. It follows that by noting R_i the miner's income after a i-th cycle and by T_i the duration time of this cycle, the revenue ratio of the strategy is equal to $\frac{\sum_{i=1}^{n} R_i}{\sum_{i=1}^{n} T_i}$ for $n \to +\infty$. This quantity converges to $\frac{\mathbb{E}[R]}{\mathbb{E}[T]}$ by the strong law of large numbers, provided that the duration time of a cycle attack T is integrable and non-zero. Likewise, assuming that costs are integrable, the cost of mining by time unit converges to $\frac{\mathbb{E}[C]}{\mathbb{E}[T]}$. Thus we can state the following corollary.

Proposition 2.3. *Let ξ and ξ' be two mining strategies. Let R and R' (resp. C and C') be the revenue (resp. cost) of the miner by attack cycle. We denote also by T and T' the duration times of attack cycles for ξ and ξ'. Then, ξ is more profitable than ξ' if and only if $\frac{\mathbb{E}[R_\xi]}{\mathbb{E}[T]} - \frac{\mathbb{E}[C_\xi]}{\mathbb{E}[T]} > \frac{\mathbb{E}[R_{\xi'}]}{\mathbb{E}[T]} - \frac{\mathbb{E}[C_{\xi'}]}{\mathbb{E}[T]}$*

Corollary 2.4. *If moreover we assume that the cost of mining per unit of time of ξ and ξ' are equally distributed, then ξ is more profitable than ξ' if and only if $\frac{\mathbb{E}[R_\xi]}{\mathbb{E}[T]} > \frac{\mathbb{E}[R_{\xi'}]}{\mathbb{E}[T]}$.*

Example 2.5. *The attack cycles for the honest strategy are simply the time taken by the whole network to discover a block. Consequently, when the miner mines honestly, his revenue ratio is $\frac{qb}{\tau_0}$ where b is the mean value of a block (coinbase and transaction fees).*

2.5. **Performant strategy and profit lag.**

Definition 2.6. *A mining strategy is a performant strategy if its revenue ratio is greater than the revenue ratio of the honest strategy, i.e. $\frac{\mathbb{E}[R(T)]}{\mathbb{E}[T]} > \frac{qb}{\tau_0}$, where $R(T)$ is the revenue per attack cycle and T is the duration time of an attack cycle.*

For a performant strategy ξ, we have $\mathbb{E}[R(T)] \geq qb\frac{\mathbb{E}[T]}{\tau_0}$. So, at T-time, the miner is better off, on average, following strategy ξ that mining honestly. However, by continuing the strategy ξ, it is possible that the miner will incur in looses for a longer period of time. Nothing prevents having $\mathbb{E}[R(\tau)] < qb\frac{\mathbb{E}[\tau]}{\tau_0}$ for a certain stopping time $\tau > T$. This, indeed, happens for the ISM strategy. Below we define the notion of profit lag of a mining strategy.

Definition 2.7. *Let ξ be a mining strategy and τ a stopping time.*

- *ξ is profitable at date τ if $\mathbb{E}[R(\tau)] \geq qb\frac{\mathbb{E}[\tau]}{\tau_0}$.*
- *ξ is definitely a performant strategy at date τ if $\mathbb{E}[R(\tau')] \geq qb\frac{\mathbb{E}[\tau']}{\tau_0}$ for any stopping time $\tau' > \tau$ a.s.*
- *The profit lag is the smallest stopping time τ with this property.*

This definition is sound since the infimum of stopping times is a stopping time. Note that if ξ is a performant strategy with duration time T for an attack cycle, then

ξ is profitable at a date T. But in general, ξ is *not* definitely profitable at date T: the profit lag is in general longer than T. However we can prove that if ξ is a performant strategy then the profit lag is finite.

Proposition 2.8. *Let ξ be a performant mining strategy. Then, there exists τ a stopping time such that ξ is definitely a performant strategy at date τ.*

Proof. We keep the same notations as above. Let $X(t) = \mathbb{E}[R(t) - \frac{qbt}{\tau_0}]$ for $t \in \mathbb{R}_+$. Since ξ is a performant repetitive strategy, we have $X(T) > 0$ and

$$\mathrm{Inf}\,\{X(\tau)\,;\,\tau > 0 \text{ stopping time}\} = \mathrm{Inf}\,\{X(\tau)\,;\,\tau \text{ stopping time} \in [0,T]\} > -\infty$$

Let us denote by $m \in \mathbb{R}_-$ this quantity and let n be an integer with $n > \frac{|m|}{X(T)}$. Then, if $\tau > nT$ is a stopping time, we have:

$$X(\tau) = X(nT) + X(\tau - nT) \geq nX(T) + m > 0$$

Hence, we get the result. □

3. Selfish mining revisited

Selfish mining strategy can be described by the stopping time which defines the end of an attack cycle.

Definition 3.1. *The end of an attack cycle for the selfish mining strategy is given by the stopping time* $T = \mathrm{Inf}\left\{t > S_1 \,/\, N(t) = N'(t) - 1 + 2 \cdot 1_{S_1 < S_1'} + 2 \cdot 1_{S_1' < S_1 < S_2 < S_2'}\right\}$.

Example 3.2. *In the case considered above, $S_1' < S_2' < S_3' < S_1 < S_4' < S_2 < S_3$, we have $1_{S_1 < S_1'} = 1_{S_1' < S_1 < S_2 < S_2'} = 0$ and $T = \mathrm{Inf}\left\{t > S_1 \,/\, N(t) = N'(t) - 1\right\} = S_3$.*

We note also by L the number of blocks added to the official blockchain after an attack cycle. It is clear that $|N(T) - N'(T)| = 1$ so $L = \frac{N(T)+N'(T)+1}{2}$. In the previous example, we have $N(T) = 3$, $N'(T) = 4$ and $L = 4$.

Proposition 3.3. *We have:*

$$\mathbb{E}[R] = \frac{(1+pq)(p-q)+pq}{p-q}\, qb - (1-\gamma)p^2 q\, b\ ,$$
$$\mathbb{E}[T] = \frac{(1+pq)(p-q)+pq}{p-q}\, \tau_0$$
$$\mathbb{E}[L] = \frac{1+p^2q+p-q}{p-q} b$$

Proof. The counting process $N(t)$ (resp. $N'(t)$) is a Poisson process. So, T (resp. T') is an integrable stopping time. Then the result follows from Doob stopping time theorem applied to the compensated martingales $N(t) - \alpha t$ (resp. $N'(t) - \alpha' t$) with $\alpha = \frac{p}{\tau_0}$ (resp. $\alpha' = \frac{q}{\tau_0}$) and to the finite stopping time $T \wedge t$ for a fixed $t > 0$. Then, we take the limit when $t \to \infty$ using the monotone convergence theorem. We get as limits $\mathbb{E}[T]$, $\mathbb{E}[N(T)]$, $\mathbb{E}[N'(T)]$ and $\mathbb{E}[L]$. Finally, we also get $\mathbb{E}[R]$ by observing that $\mathbb{E}[R] = \mathbb{E}[N'(T)]b - (1-\gamma)p^2qb$ since the only attack cycle when $R \neq N'(T)$ happens when the attacker mines first a block, then the honest miners mine one block and the honest miners mine another block on top of the honest block. □

We set the date $t = 0$ when the selfish miner starts his attack, at the beginning of a new period of 2016 official blocks, just after a difficulty adjustment.

Corollary 3.4. *Let δ be the first difficulty parameter. We have $\mathbb{E}[\delta] = \frac{p-q+pq(p-q)+pq}{p^2q+p-q}$ and $\mathbb{E}[\tilde{S}_{n_0}] = \mathbb{E}[\delta] \cdot n_0 \cdot \tau_0$*

Proof. Let ν be the number of cycles before we get a difficulty adjustment. By Wald theorem, we get $\mathbb{E}[L] \cdot \mathbb{E}[\nu] = n_0$ and $\mathbb{E}[\tilde{S}_{n_0}] = \mathbb{E}[\nu] \cdot \mathbb{E}[T]$. So, $\mathbb{E}[\delta] = \frac{\mathbb{E}[\tilde{S}_{n_0}]}{n_0\tau_0} = \frac{n_0\mathbb{E}[T]}{\mathbb{E}[L]\tau_0}$. Hence, we get the result using Proposition 3.3 □

We can now calculate the apparent hashrate of the strategy which is defined as the fraction of blocks of the attacker in the official blockchain *on the long term.*

Corollary 3.5. *The apparent hashrate of the selfish mining strategy is*

$$q' = \frac{((1+pq)(p-q)+pq)q - (1-\gamma)p^2q(p-q)}{p^2q+p-q}$$

Proof. After a difficulty adjustment, the mining difficulty is divided by δ. So the parameters of the Poisson processes $N(t)$ and $N'(t)$ are each multiplied by δ and also the duration times of attack cycles are divided by δ. On the other hand, the revenue of the selfish miner remains constant after each attack cycle since the probability to be first to discover a new block remains the same. Therefore, his new revenue ratio after difficulty adjustment is $q'\frac{b}{\tau_0}$ with $q' = \left(\frac{\mathbb{E}[R]}{\mathbb{E}[T]} \cdot \delta\right) \cdot \frac{\tau_0}{b}$. Hence, we get the result using Proposition 3.3. □

By rearranging terms, we observe that q' is nothing but the quantity R_{pool} from [8]. But, more importantly, we have proven the following proposition.

Proposition 3.6. *Before a difficulty adjustment, official blocks are discovered on average every period of $\delta \cdot \tau_0$. The revenue ratio of the selfish miner is $\frac{q'}{\delta}\frac{b}{\tau_0}$ (lower than $q\frac{b}{\tau_0}$). After a difficulty adjustment, officials blocks are discovered on average*

every period of τ_0. The revenue ratio of the selfish miner equal $q'\frac{b}{\tau_0}$ (which is now greater than $q\frac{b}{\tau_0}$).

3.1. **Previous state-machine approach revisited.** We can model the different states taken by the network with the help of a Markov chain (X_n) as was done in [8]. The set of states is $\mathbb{N} \cup \{0'\}$. The state $\{0'\}$ corresponds to the state when there is a competition between two blocks in the network. The state $\{n\}$ corresponds to the state when the attacker has a lead of n blocks over the honest miners. In this framework, the duration time of an attack cycle is proportional to the first return time $\nu = \text{Inf}\,\{n \in \mathbb{N}^* \,/\, X_n = 0\}$. It turns out that the Markov chain is irreducible and all states are positive recurrent. It has naturally a stationary probability (π_n) and the computation shows that $\pi_0 = \frac{1-2q}{1-4q^2+2q^3}$ [8]. Therefore, a classical probability result on Markov chains shows that $\mathbb{E}[\nu] = \frac{1}{\pi_0}$ [5]. Then it is easy to check that

$$\mathbb{E}[\nu] = \frac{(1+pq)(p-q)+pq}{p-q}$$

We recover the formula obtained for $\mathbb{E}[T]$ in Proposition 3.3. Also, for r_{pool} and r_{others} defined in [8] we have

$$\frac{1}{r_{pool}+r_{others}} = \frac{1-4q^2+2q^3}{1-q-2q^2+q^3} = \mathbb{E}[\delta]$$

So we get the difficulty adjustment parameter given in Proposition 3.4. Finally, the revenue ratio before difficulty adjustment is $\frac{\mathbb{E}[R]}{\mathbb{E}[T]} = r_{pool} \cdot \frac{b}{\tau_0}$. The formula given in [8] for the apparent hashrate $q' = \frac{r_{pool}}{r_{pool}+r_{others}}$ is the same that we find: r_{pool} is the revenue ratio before difficulty adjustment and $\frac{1}{r_{pool}+r_{others}} > 1$ is the (mean) difficulty adjustment parameter δ. For the purposes of computation of the profitability there are two equivalent models: one with Poisson processes (as considered first by Satoshi Nakamoto in his founding paper and continued in [14]) and another with Markov chains. They are not equivalent for other purposes, as for example to show the optimality of honest mining when there is no difficulty adjustment. Note also that there exists a third very effective combinatorical model that captures directly the long term behavior of a deviant strategy. For the selfish mining strategy, stubborn strategy and trailing strategy, it uses the combinatorics of Dyck words and Catalan distribution (see [24, 15, 16]).

3.2. **Profit lag.** We recall that according to notations in Section 2.2, $\tilde{S}_{n_0}$ denotes the time used by the network to reach the first difficulty adjustment.

Proposition 3.7. *Set $t_0 = \frac{q\delta - q'}{q'-q} n_0 \tau_0$. On average, the selfish mining strategy is not profitable before $\tilde{S}_{n_0} + t_0$ and is profitable after this date.*

Proof. We denote by $\Delta(t)$ the difference between the revenue of the selfish miner at date t and the revenue of honest mining. As we have seen, the revenue ratio of the selfish miner before the first difficulty adjustment is lower than the revenue ratio of the honest strategy. So, the selfish mining strategy cannot be profitable before this date. Moreover, using Proposition 3.6, we get: $\mathbb{E}[\Delta(\tilde{S}_{n_0})] = (\frac{q'}{\delta} - q)n_0\delta b = -(q\delta - q')n_0 b < 0$ and for $t > 0$, we have

$$\mathbb{E}[\Delta(\tilde{S}_{n_0} + t)] = (q' - q)\frac{t}{\tau_0}b + \mathbb{E}[\Delta(\tilde{S}_{n_0})] = (q' - q)\frac{t}{\tau_0}b - (q\delta - q')n_0 b = (q' - q)(t - t_0)\frac{b}{\tau_0}$$

□

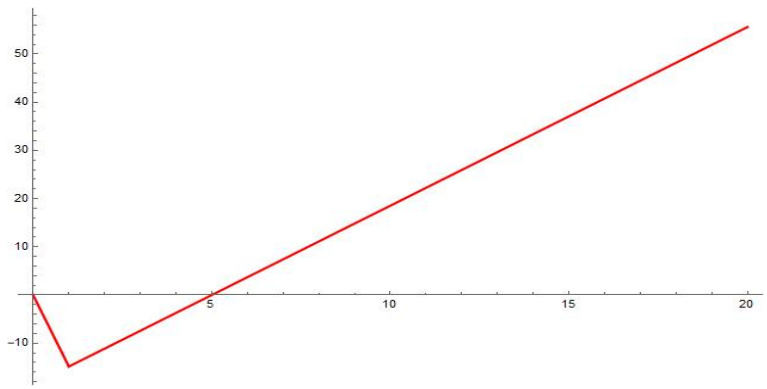

FIGURE 1. Difference of average revenue between the selfish miner and honest miner for $q = 0.1$ and $\gamma = 0.9$. X-axis: progression of the official blockchain, in difficulty adjustment units. Y-axis: revenue of the miner in coinbase units. The strategy definitely not profitable before 5 difficulty adjustments.

Example 3.8. *When $q = 0.1$ and $\gamma = 0.9$, the computation shows that we have exactly $\frac{q\delta - q'}{q' - q} = 4$. Therefore, the selfish mining strategy becomes definitely profitable at the date of the fifth difficulty adjustment which is roughly* 10 *weeks of mining (it is exactly* 10.1839 *weeks) as asserted in* [14]. *Before this date, the strategy is not profitable.*

Note 3.9. *In the introduction of* [25] *the authors claim (with no justification) that the previous computation from* [14] *is erroneous. However, in their simulations, they obtain $\mathbb{E}[\tilde{S}_{n_0}] = 15.29$ days which agrees, up to rounding, with the exact theoretical value from Corollary 3.4 which is* 15.2872. *And from here, it is not hard to get Proposition 3.7 as we did above. Obviously, our theoretical exact results agree with selfish mining simulators that one can find online* [20]. *So, apparently, the authors of* [25] *are confused, not only about this, but also about the profitability of the Intermittent Selfish Mining (ISM) that they analyze only numerically. In the next section we give rigorous analysis of ISM with closed form formulas.*

4. Intermittent selfish mining strategy

This strategy is composed by attack cycles with two distinct phases. Each phase corresponds to a complete period of mining of n_0 official blocks until the next difficulty adjustment. In the first phase, the attacker follows the selfish mining strategy. In the second phase, he mines honestly.

Proposition 4.1. *The mean duration time of the first (resp. second) phase is $n_0\,\tau_0\,\delta$ (resp. $\frac{n_0\,\tau_0}{\delta}$). During the first (resp. second) phase, the revenue ratio of the miner is $\frac{q'}{\delta}\cdot\frac{b}{\tau_0}$ (resp. $q\,\delta\cdot\frac{b}{\tau_0}$) where q' and δ are given in Corollary 3.4 and Coreollary 3.5 (denoting by δ its expected value). Moreover, denoting by R (resp. T) the revenue (resp. duration time) at the end of an attack cycle, we have:*

$$\mathbb{E}[T] = \left(\delta + \frac{1}{\delta}\right) n_0\tau_0$$
$$\mathbb{E}[R] = (q' + q)n_0 b$$

Proof. The duration time and the revenue ratio of the attacker during the selfish mining phase has been computed in the previous section. After a first difficulty adjustment, the mining difficulty is divided by δ. So, blocks arrive on an average period $\frac{\tau_0}{\delta}$ and the duration time of the second phase is $\frac{n_0\,\tau_0}{\delta}$. Each time, there is a probability q that the attacker is the first one to discover the next block. Hence, during the second phase, his revenue ratio is $\frac{q\,b}{\frac{\tau_0}{\delta}} = \frac{q\,\delta b}{\tau_0}$. Finally, the revenue of the ISM miner at the end of an attack cyle is

$$\mathbb{E}[R] = \left(\frac{q'}{\delta}\frac{b}{\tau_0}\right)\cdot n_0\,\tau_0\,\delta + \left(\frac{q\,\delta\,b}{\tau_0}\right)\cdot\frac{n_0\tau_0}{\delta}$$

and the result follows. □

Corollary 4.2. *The apparent hashrate of the Intermittent selfish strategy is $q" = \frac{q+q'}{\delta+\frac{1}{\delta}}$, and we have the closed form formula*

$$q" = \frac{q\,(1-4q^2+2q^3)\,(1+\gamma+(3-4\gamma)q+(5\gamma-11)q^2+(5-2\gamma)q^3)}{2-2q-11q^2+10q^3+18q^4-20q^5+5q^6}$$

Example 4.3. *This formula checks well with simulations from* [25]*. The threshold ISM/HM is obtained when $q" = q$. When $\gamma = 0$, we get $q" = 0.365078$. The authors from* [25] *obtain* 0.37 *with their simulations.*

Corollary 4.4. *When ISM is more profitable than honest mining, then SM is more profitable than ISM.*

Proof. By hypothesis, we have $\frac{q+q'}{\delta+\frac{1}{\delta}} > q$, which implies

$$\frac{q'}{q} > \delta + \frac{1}{\delta} - 1 > 1 > \frac{1}{\delta + \frac{1}{\delta} - 1}$$

since $\delta + \frac{1}{\delta} > 2$ for $\delta > 0$. So, $\frac{q+q'}{\delta+\frac{1}{\delta}} < q'$ and ISM is less profitable than SM. □

In Figure 2 regions in $(q, \gamma) \in [0, 0.5] \times [0, 1]$ are colored according to which strategy is more profitable (HM is the honest mining strategy).

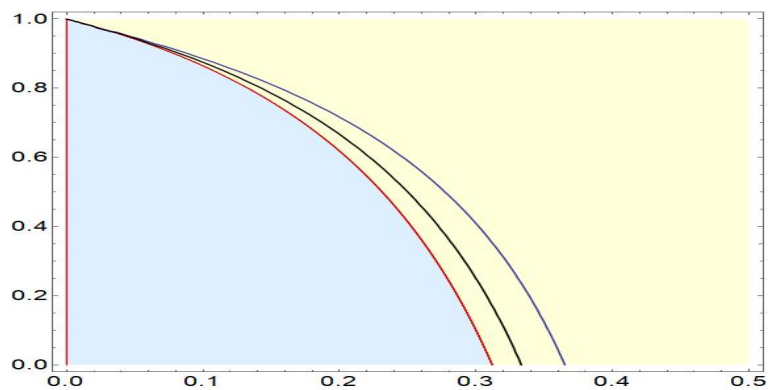

FIGURE 2. Dominance regions in parameter space (q, γ). The threshold SM/HM (resp. ISM/HM, resp. SM/ISM) in black (resp. blue, red). When ISM is more profitable than HM then SM is always more profitable than ISM.

4.1. **Profit lag.** As before, we denote by Δ the difference between the average revenue of selfish and honest mining. According to Proposition 4.1, after n attack cycles i.e., alternatively n phases of selfish mining and n phases of honest mining, the attacker earns $q" \cdot \left(\delta + \frac{1}{\delta}\right) n \cdot n_0\, b$ during $\left(\delta + \frac{1}{\delta}\right) n \cdot n_0 \tau_0$. So, at this date, we have

$$\Delta = (q" - q) \cdot \left(\delta + \frac{1}{\delta}\right) \cdot n\, n_0\, b$$

After this date, if the attacker selfish mines again, then at the end of this phase of duration $\delta n_0 \tau_0$, his revenue is $q" \cdot \left(\delta + \frac{1}{\delta}\right) n \cdot n_0\, b + \frac{q'}{\delta} \cdot n_0\, \delta\, \tau_0$. At this date, we have

$$\begin{aligned}\Delta &= (q" - q) \cdot \left(\delta + \frac{1}{\delta}\right) \cdot n\, n_0\, b + \left(\frac{q'}{\delta} - q\right) \cdot n_0\, \delta\, b \\ &= (q" - q) \cdot \left(\delta + \frac{1}{\delta}\right) \cdot n\, n_0\, b - (q\delta - q') \cdot n_0\, b\end{aligned}$$

By the end of an attack cycle, we have on average $\Delta = (q" - q)\, n_0 \left(\delta + \frac{1}{\delta}\right) b$ which is positive when ISM is more profitable than honest mining. Thus when $q" > q$, ISM is more profitable than HM before a second difficulty adjustment as noticed in [25]. This comes as no surprise since the duration time of an attack cycle of the ISM strategy corresponds to a period of 2×2016 official blocks, and any performant mining strategy is always profitable at the end of an attack cycle. See the discussion at the end of Subsection 2.5.

The authors [25] fail to understand that after another difficulty adjustment the revenue falls because the difficulty increases and ISM becomes less profitable than HM. It is only after several difficulty adjustment periods of 2016 blocks that ISM can become definitely more profitable than HM. For instance, when $q = 0.1$ and $\gamma = 0.9$ this only happens approximatively after the 13-th difficulty adjustment. Therefore, on average, it takes more than 6 months for the strategy to be definitely profitable, and this is much longer than for classical SM. Obviously the authors of [25] are overall confused with a naive notion of profitability and don't follow a rigorous profitability model.

It is important to understand what happens. First, the selfish miner invests in lowering the difficulty, and, at any moment, he can get reap immediate profits, even before the profit lag, by just mining honestly. But if the miner wants to repeat the attack cycle, he will need to burn again these profits for the purpose of lowering the difficulty. This is what happens in the first cycles of the ISM strategy. In Figure 3 we have the plot of the progression of Δ.

5. Alternate network mining strategy

The strategy is described for alternate mining between BTC and BCH networks, but obviously it applies to any pair of networks with the same mining algorithm (with frictionless switching mining operations). This time we consider three distinct types of miners: the first ones mine on Bitcoin only, the second ones on Bcash (BCH), and the third ones alternate between Bitcoin and Bcash. The ones mining alternatively between BTC and BCH are labelled "attackers" (although his strategy is legit and

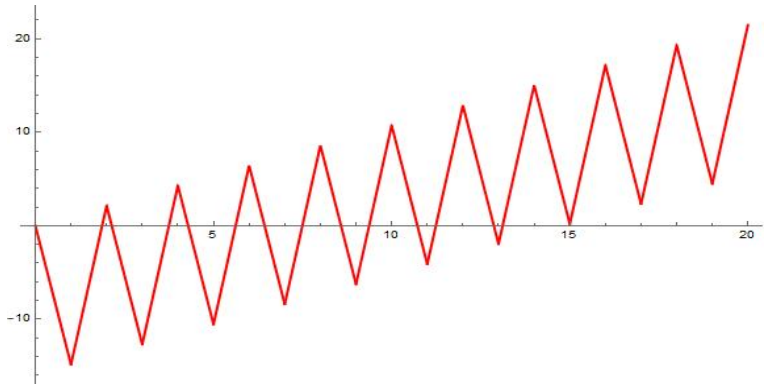

FIGURE 3. Difference of average revenue between intermittent selfish and honest mining for $q = 0.1$ and $\gamma = 0.9$. X-axis: progression of the official blockchain, in difficulty adjustment units. Y-axis: revenue of the miner in coinbase units. The strategy is not definitely profitable before 13 difficulty adjustments. By comparison, SM is definitely profitable after 5 difficulty adjustments (compare with Figure 1).

respects both network protocols). We assume that the revenue ratios of mining honestly on Bitcoin and Bcash are the same. This is approximativelly what is observed in practice since any divergence justifies a migration of hashrate from one network to the other. We denote by ρ this common value for the attacker. We assume also that there are no other miners and all miners mine with full power. In particular, the total hashrate remains constant. The attacker starts mining on Bitcoin at the beginning of a difficulty adjustment on BTC. An attack cycle is made of two phases. During Phase 1, the attacker withdraws from BTC and mines on BCH until n_0 blocks have been mined on Bitcoin. During Phase 2, the attacker comes back to mine on the BTC network until a new difficulty adjustment. We call this strategy: *Alternate Mining Strategy*. This is a variation of smart mining. The only difference being that the miner does not remain idle during Phase 1 but goes on to mine on BCH. Note that this mining strategy does not provoke periodic reorganizations of the blockchain. The main annoyance for users of both networks is that difficulty does not stabilize and for Bitcoins users blocks arrive regularly at a slower pace than in the steady regime with the miner fully dedicated to the BTC network.

We denote by δ the difficulty adjustment parameter after Phase 1. Since the miner comes back on BTC in the second phase, the second difficulty adjustment at the end of the second phase is $\frac{1}{\delta}$.

Proposition 5.1. *The duration phase of Phase 1, resp. Phase 2, is $n_0\tau_0\delta$, resp. $n_0\tau_0\frac{1}{\delta}$. During Phase 1, resp. Phase 2, the revenue ratio of the attacker is ρ, resp. $\rho\delta$. The revenue ratio of the alternate mining strategy is $\frac{1+\delta}{\delta+\frac{1}{\delta}}\rho$.*

Proof. By definition of the difficulty adjustment parameter, the duration time of Phase 1, resp. Phase 2, is $\delta n_0\,\tau_0$, resp. $\frac{1}{\delta} n_0\,\tau_0$. So the duration time of an attack cycle is $\left(\delta + \frac{1}{\delta}\right)\, n_0 \tau_0$. During Phase 1, the attacker's revenue ratio is ρ because we assume that the attacker mines honestly during this phase with the assumption that the revenue ratio is the same for BCH and BTC. During Phase 2, the mining difficulty is divided by δ. So, the revenue ratio of the attacker during this period is $\delta \cdot \rho$. Therefore, the revenue of the attacker after an attack cycle is $\rho \cdot n_0\,\tau_0\,\delta + (\rho\,\delta) \cdot \frac{n_0 \tau_0}{\delta}$. □

Corollary 5.2. *The alternate mining strategy is always more profitable than honest mining and selfish mining for all values of q.*

Proof. The first statement results from $\delta > 1$ that implies $\frac{1+\delta}{\delta+\frac{1}{\delta}} > 1$. To prove, the second statement, we remark that in Phase 1, blocks are only validated by honest miners. So this phase lasts on average $\frac{n_0 \tau_0}{p}$ and the difficulty parameter is updated accordingly: $\delta = \frac{1}{p}$ with $p = 1 - q$. So, the revenue ratio of the attacker is $\frac{2-q}{2-2q+q^2}\frac{b}{\tau_0}$ as follows by replacing δ with $\frac{1}{p}$ in the formula from Proposition 5.1. On the other hand, in the most favorable case (when $\gamma = 1$), the revenue ratio of the selfish miner is $\frac{q\left(2q^3-4q^2+1\right)}{q^3-2q^2-q+1}\frac{b}{\tau_0}$. Then, the result comes from $\frac{2-q}{2-2q+q^2} > \frac{q\left(2q^3-4q^2+1\right)}{q^3-2q^2-q+1}$ for $0 < q < 0.5$ that we prove studying the polynomial

$$(2-q)\cdot(q^3-2q^2-q+1) - (2-2q+q^2)\cdot\left(q\left(2q^3-4q^2+1\right)\right)$$

This polynomial is non-increasing on $[0, \frac{1}{2}]$ and remains positive on this interval. □

5.1. **Profit lag.** As before, we denote by Δ the difference of the average revenue between selfish mining and mining honestly from the beginning.

Proposition 5.3. *After n attack cycles, we have $\Delta = \rho \cdot \left(1 - \frac{1}{\delta}\right) n\, n_0 \tau_0 = q^2\, n\, n_0 b$ and Δ stays constant during the BCH mining phase.*

Proof. After a first phase of mining, we have $\Delta = 0$ since the revenue ratio on BCH and BTC are assumed to be equal. At the end of the second phase, we have $\Delta = \left(1 - \frac{1}{\delta}\right) \rho \cdot n_0 \tau_0$. Indeed, the second phase lasts $\frac{n_0 \tau_0}{\delta}$ and the revenue ratio of the attacker during this phase is $\rho\delta$. Now, we use that $\rho = q\frac{b}{\tau_0}$ and $\delta = \frac{1}{p}$ because only honest miners are mining during Phase 1. The strategy is then a repetion of alterning phases 1 and 2 and the result follows. □

We plot the graph of Δ in Figure 4.

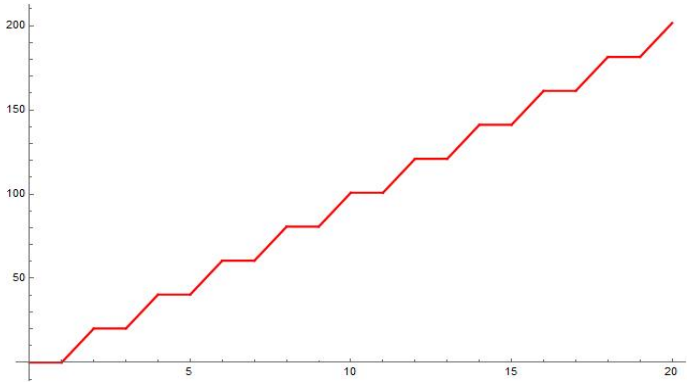

FIGURE 4. Difference of average revenue between alternate and honest mining for $q = 0.1$. X-axis: progression of the official blockchain, in difficulty adjustment units. Y-axis: revenue of the miner in coinbase units. The strategy is definitely profitable after one difficulty adjustment.

6. Conclusion

We revisit the selfish and intermitent mining strategies, and define alternate mining strategy for pairs of networks with the same PoW. We define the notion of "profit lag". We compute exact formulas for different strategies of the profit lag and the revenue ratio. We correct misunderstandings and unfounded claims in [25] by clarifying the profitability setup using the new notion of "profit lag". We show that, under natural hypothesis, the alternate mining strategy is the best one: its revenue ratio is the largest and the profit lag is the least. So, the existence of two networks sharing the same PoW algorithm can lead to instabilities of the difficulty and with blocks validated slower than normal.

References

[1] A. Antonopoulos. *Mastering Bitcoin: Unlocking Digital Cryptocurrencies* OReilly Media, Inc., 2014.

[2] L. Bahack. *Theoretical bitcoin attacks with less than half of the computational power (draft).* arXiv:1312.7013, 2013.

[3] L. Bahack, N. Courtois. *On Subversive Miner Strategies and Block Withholding Attack in Bitcoin Digital Currency.* arXiv:1402.1718, 2014.

[4] V. Buterin. *Change difficulty adjustment to target mean block time including uncles.* github.com/ethereum/EIPs/issues/100, 2017.

[5] R. Douc, E. Moulines, P Priouret, P Soulier. *Markov Chains.* Springer, 2020.

[6] T. Dryja, J. Poon. *The bitcoin Lightning Network: scalable off-chain instant payment.* lightning.network/docs/, 2016.

[7] C. Dwork, M. Naor. *Pricing via processing or combatting junk mail.* 12th Annual International Cryptology Conference, 1992.

[8] I. Eyal, E. Sirer. *Majority is not enough: bitcoin mining is vulnerable.* International Conference on Financial Cryptography and Data Security, 2014.
[9] E. Felten. *Bitcoin isn't so broken after all.* https://freedom-to-tinker.com/2013/11/07/bitcoin-isnt-so-broken-after-all, 2013.
[10] C. Feng, J. Niu. *Selfish Mining in Ethereum.* IEEE 39th International Conference on Distributed Computing Systems, 2019.
[11] G. Georgiadis, D. Zeilberger. *A combinatorial-probabilistic analysis of bitcoin attacks.* Journal of Difference Equations And Applications, 2019.
[12] C. Goren and A. Spiegelman. *Mind the mining*, Proceedings of the 2019 ACM Conference on Economics and Computation, 2019.
[13] C. Grunspan and R. Pérez-Marco. *Double spend races*, International Journal of Theoretical and Applied Finance, Vol. 21, 2018.
[14] C. Grunspan, R. Pérez-Marco. *On profitability of selfish mining.* ArXiv:1805.08281v2, 2018.
[15] C. Grunspan, R. Pérez-Marco. *On profitability of stubborn mining.*ArXiv:1808.01041, 2018.
[16] C. Grunspan, R. Pérez-Marco. *On profitability of trailing mining.* ArXiv:1811.09322, 2018.
[17] C. Grunspan, R. Pérez-Marco. *On profitability of Nakamoto double spend.* ArXiv:1912.06412, 2019.
[18] C. Grunspan, R. Pérez-Marco. *The Mathematics of Bitcoin.* European Mathematical Society Newsletter, 2020.
[19] C. Grunspan, R. Pérez-Marco. *Selfish Mining in Ethereum.* MARBLE Conference 2020.
[20] M. Khosravi. *A Full-fledged Simulation for Selfish Mining in Bitcoin.* https://armankhosravi.github.io/dirtypool2/
[21] S. Nakamoto. *Bitcoin P2P e-cash paper.* www.metzdowd.com/pipermail/cryptography/2008-October/014810.html.
[22] S. Nakamoto. *Bitcoin: a peer-to-peer electronic cash system.* Bitcoin.org/bitcoin.pdf, 2008.
[23] S. Nakamoto. *Bitcoin v0.1 released.* www.metzdowd.com/pipermail/cryptography/2009-January/014994.html.
[24] K. Nayak, E. Shi, S. Kumar, A. Miller. *Stubborn mining: generalizing selfish mining and combining with an eclipse attack.* IEEE European Symp. Security and Privacy, pages 305–320, 2016.
[25] K. Negy, P. Rizun, E. Sirer. *Selfish Mining Re-Examined.*, International Conference on Financial Cryptography and Data Security, 2020.
[26] RHorning. *Mining cartel attack.* bitcointalk.org/index.php?topic=2227.0, 2010.
[27] M. Rosenfeld. *Analysis of Bitcoin Pooled Mining Reward Systems* arXiv:1112.4980, 2011.
[28] M. Rosenfeld. *Analysis of hashrate-based double spending.* 1402.2009v1, 2014. 2018.
[29] J. Song. *Programming Bitcoin: Learn How to Program Bitcoin from Scratch.* O'Reilly Media, 2019.
[30] R. Wattenhofer. *Blockchain Science: Distributed Ledger Technology.* Independently published, 2019.

Cyril Grunspan
Léonard de Vinci Pôle Univ, Finance Lab
Paris, France,

E-mail address: cyril.grunspan@devinci.fr

Ricardo Pérez-Marco
CNRS, IMJ-PRG, Univ. Paris-Diderot
Paris, France

E-mail address: ricardo.perez.marco@gmail.com

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

*Send some anonymous and moderate satoshis to support our research at the pub.