The Bitcoin Mining Game: On the Optimality of Honesty in Proof-of-work Consensus Mechanism

Juan Beccuti
Christian Jaag

Swiss Economics Working Paper 0060

August 2017

Swiss Economics SE AG
Abeggweg 15
CH-8057 Zürich

T: +41 (0)44 500 56 20
F: +41 (0)44 500 56 21

office@swiss-economics.ch
www.swiss-economics.ch

Abstract

We consider a game in which Bitcoin miners compete for a reward of each solved puzzle in a sequence of them. We model it as a sequential game with imperfect information, in which miners have to choose whether or not to report their success. We show that the game has a multiplicity of equilibria and we analyze the parameter constellations for each of them. In particular, the minimum requirement to find it optimal not to report is decreasing with the number of miners who are not reporting, and increasing the heterogeneity among players reduces the likelihood that they choose not to report.

Keywords: Bitcoin, mining game, proof-of-work, sequential game, imperfect information.
JEL:

1 Introduction

In recent years, various cryptocurrencies have been developed and introduced in the market.[1] While these currencies are increasingly gaining users, they are also starting to gain attention from governments, financial institutions, banks, and academics, among others.

In this paper we study one particular aspect of one of these decentralized cryptocurrencies, Bitcoin, from a game theoretical point of view.[2,3] Briefly speaking, in Bitcoin there is a sequence of proof-of-work puzzles that are used to order transactions (we give technical details in the next subsection). The prize for being the first in solving each of those puzzles is new bitcoins. Puzzles have the following properties: (i) finding a solution is not observable, (ii) the player who finds the solution does not get his prize unless he reports it, and (iii) a player cannot solve a puzzle unless he knows the solution of the previous one. Therefore, a player who solves a puzzle faces the trade off between reporting it and getting the prize, or starting the next puzzle secretly. In this latter case, he takes the risk that one of his rivals solves the previous puzzle, in which case there are now two competing solutions

[1] A cryptocurrency is a currency issued electronically which relies on cryptographic primitives for the security of its transactions.

[2] Nowadays, Bitcoin is the most important cryptocurrency in terms of transaction volume an total value of the circulating currency units.

[3] We follow the usual convention and we call the whole system *Bitcoin* and its unit of account *bitcoin*.

(a "fork"). However, only one of them can be considered a valid solution. Players have to decide on top of which of both solutions to solve the next puzzle and the one which finishes it first determines which of the previous solutions were valid. We study under which conditions a player has incentives to not report the solution as soon as he solves a puzzle.

This question is important for the stability of proof-of-work systems like Bitcoin because they critically depend on the publicity of solved puzzles. Previous papers have studied those incentives under a different approach to ours. For instance, Eyal & Sirer (2014) show that, under certain conditions, it is profitable for a player to follow a strategy named "selfish mining" if all the remaining players report as soon as they solve a puzzle. Additionally, Nayak, Kumar, Miller & Shi (2016) prove that selfish mining is not (in general) an optimal strategy. Indeed, they propose a different strategy (named "stubborn mining") which improves the player's payoff even more. In both cases, they specify a particular and very sophisticated strategy, comparing its payoff with the one under instantaneous reporting. Moreover, these strategies are defined assuming that all remaining players do not behave strategically and report as soon as they succeed.

In contrast, our model considers a sequential game with imperfect information. Players, after solving a puzzle, have to choose between two simple actions: report or not report. Thus, when a player finds a solution, he is uncertain whether he was the first one, or one or more of his rivals have already finished without reporting.

We start the analysis showing, by mean of an example, that our model (even though we do not propose any sophisticated strategy) gives similar results (in spirit) to Eyal & Sirer (2014) and Nayak et al. (2016); it is not necessary for a player to have a probability of success larger than 50% to find it optimal not to report it. Next, we show that there are a multiplicity of equilibria and we state conditions under which these equilibria exist. Finally, we show that the minimum requirement to find it optimal not to report is decreasing in the number of rivals not reporting, and that increasing the heterogeneity among players may be a useful tool to reduce the likelihood that they choose not to report.

Next, we give a technical introduction of how Bitcoin works. Section 2 provides the model. Section 3 analyzes it and provides the main results. Section 4 concludes.

1.1 A Brief Introduction to How Bitcoin Works

Bitcoin is a decentralized consensus mechanism and a global currency system. Böhme, Christin, Edelman & Moore (2015) provide an overview of the technology, economics and governance of Bitcoin. Technically, Bitcoin is a distributed system running on a homogeneous peer-to-peer network. Peers in the network collectively maintain a global state, the ledger. The data which are tracked by the network are the so-called outputs, i.e. tuples consisting of a value denominated in bitcoins and an output script. The output script sets up a condition that has to be satisfied in order to claim the bitcoins associated with the output. The most common case is that a signature matching an address is required. A transaction claims one or more previously unclaimed outputs and creates new outputs. By providing inputs matching the output script, the creator of the transaction proves that she is allowed to claim the output. A transaction redistributes the sum of values to new outputs and may set up arbitrary claiming conditions for them.

In order to apply a transaction to the replicas of the ledger, the transaction is flooded in the network. When a node in the network receives a transaction, it first verifies the signatures of the transaction and, if valid, the transaction input from the claiming transaction. If all scripts return true, the outputs were not claimed by a previous transaction, and the sum of new output values is smaller than or equal to the sum of claimed output values, the transaction is valid. Due to the distributed nature of the system, the order in which transactions are applied is not identical across peers, and peers may disagree about the validity of a transaction, (e.g., if two or more transactions attempt to claim the same output, the validity depends on the order they are seen by the peers.) Bitcoin eventually resolves inconsistencies by choosing one peer as leader, which imposes his changes to other peers by sending them a "block" containing all transactions it accepted since the last block. Each block contains a reference to its predecessor. Hence the ledger is a chain of blocks (the "Blockchain") with a shared history of all transactions that were applied in the past. Transactions that are included in a block of the blockchain are said to be confirmed (see Decker & Wattenhofer (2013)).

To determine which node is the leader and may impose its view on the others, the nodes attempt to find a solution to a proof-of-work puzzle with a given probability of success (see Dwork & Naor (1992)). The proof-of-work consists in finding a byte string, that combined with other data (including a hash based on all of the transactions in the block) results in a hash with a given number of leading zero bits. The number of leading zero bits is determined via consensus by all nodes and regularly adjusted to achieve an average of one result every 10 minutes in the entire network. Nodes attempting to find a solution to the proof-of-work are called miners. To incentivize miners, the node finding a block receives a reward in the form of new bitcoins, i.e. it may include a transaction in the block that has no inputs but specifies outputs for a predetermined number of coins. The first 210'000 blocks received a reward of 50 bitcoins. This reward is periodically cut in half. After 21 million bitcoins have been mined, the reward reaches zero and no further bitcoins will be created. Hence, the protocol design for Bitcoin provides for a controlled expansion of the currency and an ultimate limit to the number of bitcoins issued. Miners have a second potential source of revenue (which will become the only source of revenue once all bitcoins have been created): when submitting a transaction, the user can offer to pay a transaction fee, which is a payment to whatever miner solves the puzzle that verifies the transaction. If space in blocks is scarce, users have an incentive to offer a transaction fee in order to have their transactions included in a block (see Jaag & Haller (2017)).

2 The Model

Suppose that three players $i \in \{1, 2, 3\}$ are mining for a new block at the tip of the same blockchain (i.e., we assume that there are no forks at the starting point).

To mine a block, they compete for being the first to solve the proof-of-work puzzle. Thus, for each block there is a puzzle τ to be solved. Although all puzzles are different, we assume that the difficulty of all puzzles is the same.[4] At the beginning of the game, the first puzzle is known by all miners. However, the next puzzle depends on the solution of the previous puzzle. This is, the second puzzle cannot be solved until a miner solves first the previous puzzle and publishes its solution. Each puzzle has more than one valid solution. Thus, if two solutions for the same puzzle are published, there will be two subsequent puzzles (a fork) and miners have to choose between them. In such a case, we will say that there are two branches or tips, each one denoted by its corresponding miner (e.g., i-tip).

The prize for solving each puzzle is modelled to be constant through time and it is normalized to $v = 1$. To find the solution for each puzzle, miners spend all their *hashing* (or computational) power h_i available which is common knowledge. We make the normalization that $\sum_i h_i = 1$ and normalize the cost of using it to zero. The expected payoff for solving the puzzle τ is $E(\pi_{i,\tau}) = \gamma_{i,\tau}(\mathbf{h})v$, where $\mathbf{h} = (h_1, ..., h_n)$, and $\gamma_{i,\tau}(\mathbf{h})$ is the probability of being the first in finding a valid solution for τ. Miners are assumed to not discount the future.

Finding the solution of a puzzle is a random process which follows a

[4] Actually, the difficulty of puzzles in Bitcoin is proportional to the total effort used for solving them. For simplicity, we abstract from this issue.

Poisson distribution with parameter $\lambda_i(h_i)$, where $\lambda' > 0$ and $\lambda'' = 0$.[5] Thus, the time between two events follows an exponential distribution and the probability of being the first contestant to find the solution is given by

$$\gamma_{i,\tau}(\mathbf{h}) = Pr(i \mid t_{i,\tau} = \min\{t_{1,\tau}, ..., t_{n,\tau}\}) = \frac{\lambda_i(h_i)}{\sum_{j=1}^{n} \lambda_j(h_j)} = h_i = \gamma_i(h_i).$$

The game is a sequential game with imperfect information. The first decision is made by chance or nature (N): which miner is the winner of the first contest. Next, based on the history of events, the winner has to decide which action to take: i) to "report" (denoted by R), or ii) to "not report" (denoted by NR) and starting the next puzzle secretly. However, he is uncertain whether he is the first one to finish this puzzle a or some of his rivals have already finished it without reporting it.

In Bitcoin, the sequence of proof-of-work puzzles is infinite. If all miners were following the selfish mining strategy proposed by Eyal & Sirer (2014), there would never a report. Hence, every miner would achieve zero expected profits, giving incentives to deviate from this strategy. In other words, they should follow a stopping rule. To avoid this complication, we assume that from the second puzzle on every miner reports. Therefore, we just need to analyze a two stage game. From that point forward, the game will be as in the starting point.

To exemplify, let's assume, without loss of generality, the player 1's perspective. Suppose that miner 1 has finished the first puzzle. He believes with probability $\mu_1(h_1 \mid I_1)$ (from now on $\mu_1(h_1)$) that he was the first one in doing it. If 1 chooses R, he immediately collects the reward for the first puzzle.[6] In this case, all miners start the next puzzle simultaneously and, since it is the last puzzle, its winner announces his victory as soon as he finds the solution and gets the corresponding prize.

[5] In the Bitcoin framework, the Poisson distribution has a parameter $\lambda_i = h_i/(2^{32} D_\tau)$ where D_τ is the difficulty of the puzzle. See Rosenfeld (2011) for more details on the "Bitcoin" Poisson distribution.

[6] It is assumed that the lapse of time between solving a puzzle and deciding whether to report it or not is zero.

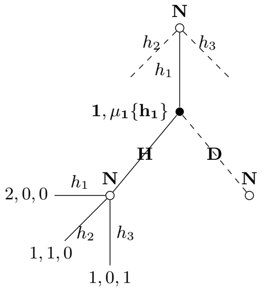


Figure 1: The graphs are a discrete time representation of a continuous time game. Subgame if miner 1 believes he is the first one to finish the first puzzle and decides to take action R.

Alternatively, 1 does not report his success on the first puzzle and starts working on the next one in secret. If 1 is the first to finish the last puzzle, which occurs with probability h_1, he is rewarded with both prizes. However, it may be the case that one of his rivals finishes the first puzzle before 1 finishes the last one. Indeed, with probability h_2 (alternatively h_3), 2 (3) finishes the first puzzle and has to choose his action.[7] In case that this rival chooses R, 1 immediately publishes his solution too.[8] Since there are two known solutions to the first puzzle, the remaining miner has to choose on top of which solution to solve the next puzzle. We consider that with probability α the remaining miner chooses the "first" solution,i.e., the 1 solution in this case.[9] Now, the reward of the first puzzle is allocated depending on the winner of the last one. If 1 is the first in finishing the latter, he is rewarded with both prizes. If it is solved by one of the miners mining in top of the 1-tip, he gets the reward of the last puzzle and 1 gets the prize of the first block. If the rival who has finished the first puzzle, or one of the miners with him, succeeds in finishing the last one, 1 does not get anything. In case that 2 (3) chooses NR, he and 1 are both mining the last block secretly, each one

[7] This rival is ignorant about whether it has already been finished because 1 did not report his solution.

[8] Notice that, by the nature of the random process, the hashing power already spent by 1 on solving the last puzzle does not increase his probability of winning it.

[9] In practice, the remaining miner chooses the solution he observes first. In our model, both reports occurs at the same time and, hence, the remaining miner has no way of knowing which was the first solution. However, to simplify and to have some symmetry for every miner we have adopted the rule that with probability α the remaining miner observes first the "first" solution.

on top of his own previous solution. If one of them solves the last puzzle, this miner will get both prizes. If the last miner finishes the first puzzle before, he will have to choose his action.

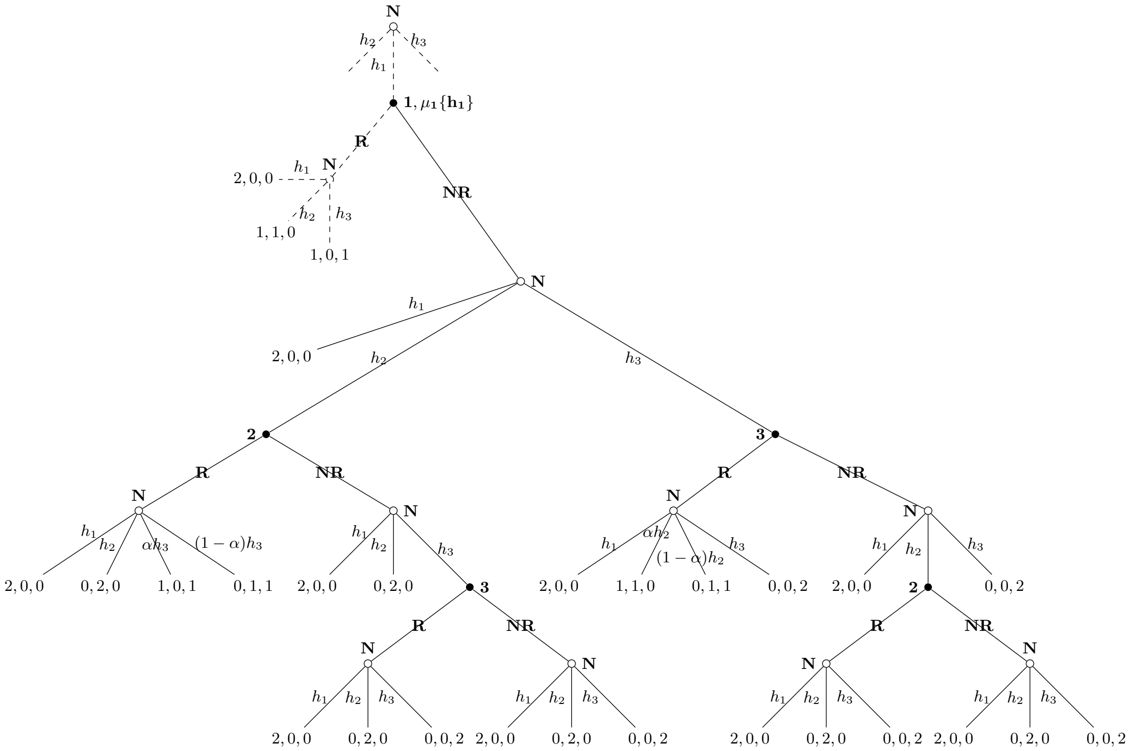


Figure 2: Subgame if miner 1 believes he is the first one to finish the first puzzle and he decides to take action NR.

Miner 1 believes with probability $\mu_1(h_2, h_1 \mid I_1)$ (from now on $\mu_1(h_2)$) that miner 2 (alternatively $\mu_1(h_3)$ and miner 3) has already finished the first puzzle without reporting it.[10] Based on the assumption that his beliefs are correct, if 1 chooses R, 2 (3) will also report immediately and both prizes are defined by the winner of the next puzzle. If he also chooses NR, there are two possible outcomes: (i) one of the two miners who are mining the last puzzle secretly finishes and the winner gets both rewards; (ii) miner 3 (2) finishes the first puzzle before than one of his rivals finishes the last one and decides his action.

Finally, miner 1 believes with probability $\mu_1(h_2, h_3, h_1 \mid I_1)$ (from now on $\mu_1(h_2, h_3)$) that he is the last miner in finishing the first puzzle: 2 (alternatively 3) was the first one and 3 (alternatively 2) the second one. Now, no

[10] Note that $\mu_1(h_2)$ refers to a joint event: solving and deciding not to report. If 2 reports, miner 1 would be sure in which node he is.

matter which action is taken by 1, he will get both prizes with probability h_1 and nothing otherwise.

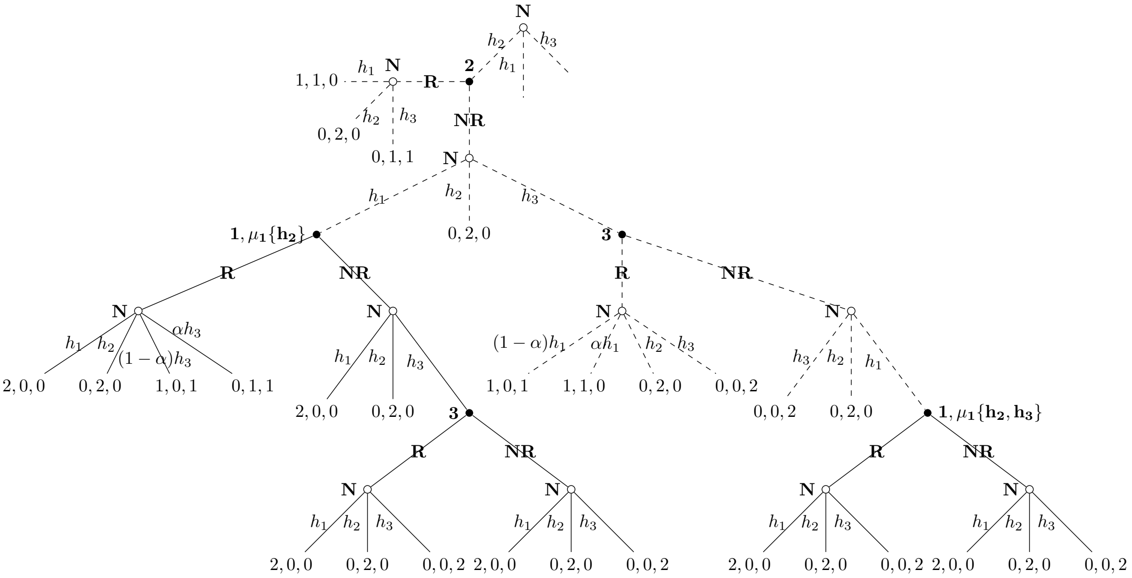
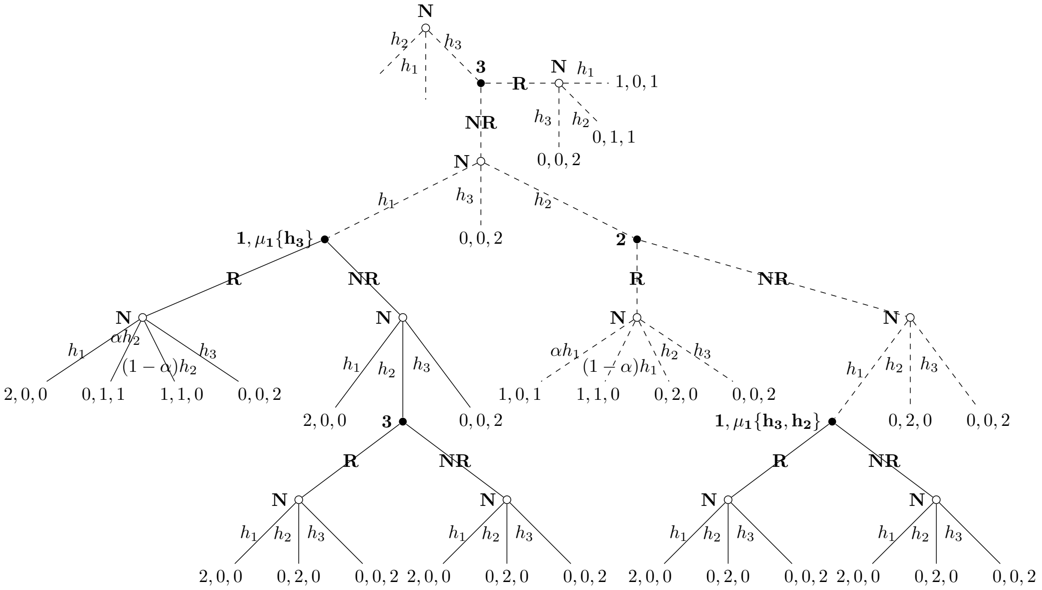


Figure 3: Above: Subgames if miner 1 believes he is the second one (after miner 2) to finish the first puzzle, and if he is the last one (2 was the first and 3 the second). **Below:** Subgames if miner 1 believes he is the second one (after miner 3) to finish the first puzzle, and if he is the last one (3 was the first and 2 the second).



Formally, we study the sequential equilibrium of the game.[11] A sequential equilibrium is an "assessment" (a pair of players' behavioral strategies and their beliefs for every information set) that is sequentially rational and consistent for every player. A sequentially rational behavioral strategy β_i for miner i specifies a probability distribution over player i's set of actions $\{R, NR\}$ (conditional of succeeding in the first puzzle, i.e., the i information set) which is a best response to the other players' strategies given i's beliefs. Conditional on succeedinf, the miner i's beliefs

$$\mu_i \equiv (\mu_i(h_i), \mu_i(h_j), \mu_i(h_k), \mu_i(h_j, h_k), \mu_i(h_k, h_i)), \tag{1}$$

are derived from the strategies using Bayes' rule.

The expected payoffs for each miner are the following:

$$\begin{aligned}
\mathbf{E}U_1(R) &= 2h_1 + \mu_1(h_1)(1 - h_1) + \mu_1(h_2)(1 - \alpha)h_3 + \mu_1(h_3)(1 - \alpha)h_2, \\
\mathbf{E}U_2(R) &= 2h_2 + \mu_2(h_2)(1 - h_2) + \mu_2(h_1)(1 - \alpha)h_3 + \mu_2(h_3)(1 - \alpha)h_1, \\
\mathbf{E}U_3(R) &= 2h_3 + \mu_3(h_3)(1 - h_3) + \mu_3(h_1)(1 - \alpha)h_2 + \mu_3(h_2)(1 - \alpha)h_1,
\end{aligned} \tag{2}$$

for action R, and

$$\begin{aligned}
\mathbf{E}U_1(NR) = 2h_1 &+ \mu_1(h_1)[2h_1h_2 + \alpha h_2(1 - \beta_2)h_3 + 2h_1h_2\beta_2h_3] \\
&+ \mu_1(h_1)[2h_1h_3 + \alpha h_3(1 - \beta_3)h_2 + 2h_1h_3\beta_3h_2]+ \\
&+ \mu_1(h_2)[2h_1h_3] + \mu_1(h_3)[2h_1h_2], \\
\mathbf{E}U_2(NR) = 2h_2 &+ \mu_2(h_2)[2h_1h_2 + \alpha h_1(1 - \beta_1)h_3 + 2h_2h_1\beta_1h_3] \\
&+ \mu_2(h_2)[2h_2h_3 + \alpha h_3(1 - \beta_3)h_1 + 2h_2h_3\beta_3h_1\}] \\
&+ \mu_2(h_1)[2h_2h_3] + \mu_2(h_3)[2h_2h_1], \\
\mathbf{E}U_3(NR) = 2h_3 &+ \mu_3(h_3)[2h_3h_1 + \alpha h_1(1 - \beta_1)h_2 + 2h_3h_1\beta_1h_2] \\
&+ \mu_3(h_3)[2h_3h_2 + \alpha h_2(1 - \beta_2)h_1 + 2h_3h_2\beta_2h_1\}] \\
&+ \mu_3(h_2)[2h_3h_1] + \mu_3(h_1)[2h_3h_2],
\end{aligned} \tag{3}$$

for action NR. Because we are not considering randomizations, from now on we assume that if any miner achieves the same profits under both actions, he will choose to follow action NR. Therefore, given μ_i, he will play $\beta_i = 1$ whenever $\mathbf{E}U_i(NR) \geq \mathbf{E}U_i(R)$, and $\beta_i = 0$ otherwise.

[11] See Osborne & Rubinstein (1994).

3 Analysis

Previous works, like Eyal & Sirer (2014) and Nayak et al. (2016), have shown that it is not necessary for a miner to have a hashing power larger than 50% of the total hashing power to find it optimal not to report. Although they show this by proposing different sophisticated strategies, a key is the assumption that, when there is a fork, a proportion of the remaining miners chooses the tip of the non-reporting miner. This feature is also captured in our setting through the parameter α.

To illustrate the importance of this parameter for that result, in the following example we show that it still holds in our simpler setting.

Example 1. *Suppose that the miner i has to decide whether to report or not, knowing that his rivals always report. Suppose that $h_j = h_k = 1/3$ and $\alpha = 1$. From equations (2) and (3), miner i does not report if $h_i \geq 1/2 - \alpha h_j h_k/(h_j + h_k)$. Therefore, it is enough for miner i to have a hashing power $h_i = 1/3$.*

The intuition is the following. Since i has already finished the first puzzle, he can take the risk of starting the next one secretly because in case that a rival solves and publishes the former before i finishes the latter, i will also report the first puzzle, creating a fork. Recall that, in that case, the allocation of the reward of the first puzzle is defined in the last puzzle and its probability of being finished on the i-tip depends on α. Hence, the larger the fraction α, the larger i's expected payoff if he is forced to report, and the lower h_i under which i finds it optimal to choose NR. Note that, if none of the remaining miners decides to mine on top of the non-reporting miner's solution (i.e., $\alpha = 0$), i will need at least half of the hashing power to find it optimal to take action NR.

The result in the previous example depends on the distribution of the hashing power and the value of the parameter α. In our model with three miners and two periods, an extreme case is one of the rivals' hashing power being close to zero. The opposite extreme is $h_j = h$ for all $j \neq i$.

Remark 1. *Depending on the distribution of the hashing power and on the parameter α, the minimum hashing power required for miner i to find it optimal to choose NR if both rivals are choosing R belongs to the range $\left[\frac{1}{3}, \frac{1}{2}\right)$.*

Proof: See the Appendix.

However, in the aforementioned works and in our analysis above, it is assumed that all miners (but the one under analysis) might not be rational and always choose to report. In what follows, we consider that all players are rational and behave strategically.

Since every miner is uncertain about the history of past events, they form beliefs about them following Bayes' rule:

$$\begin{aligned}
\mu_i(h_i) &= \frac{1}{1+h_j\beta_j+2h_j\beta_jh_k\beta_k+h_k\beta_k},\\
\mu_i(h_j) &= \frac{h_j\beta_j}{1+h_j\beta_j+2h_j\beta_jh_k\beta_k+h_k\beta_k}, \text{ for all } j \neq i,\\
\mu_i(h_j,h_k) = \mu_i(h_k,h_j) &= \frac{h_j\beta_jh_k\beta_k}{1+h_j\beta_j+2h_j\beta_jh_k\beta_k+h_k\beta_k}.
\end{aligned} \tag{4}$$

Using previous beliefs, and equations (2) and (3) we get the minimum hashing power under which $\beta_i = 1$ is sequentially rational for every miner. In particular, miner i chooses NR with probability $\beta_i = 1$ if and only if:

$$h_i \geq \frac{1}{2} - \alpha\frac{h_jh_k}{h_j+h_k} - \frac{h_jh_k[3-4(h_j+h_k)]}{2(h_j+h_k)}(\beta_j+\beta_k), \tag{5}$$

and chooses R if his hashing power is lower than the right-hand side of the equation.

The following proposition states that the game has a multiplicity of assessments which are a sequential equilibrium. It specifies under which parameters the different equilibria exist.

Proposition 1. *There is a multiplicity of sequential equilibria which depends on the parameters. In particular, there exists sets of hashing powers such that in equilibrium,*

- *all players do not report: only if $\alpha \geq 2/3$.*
- *one player reports: for any $\alpha \in (0,1)$ if the hashing power of the reporter is lower than $1/4$ or in the range $(\sqrt{2}/4, 3/5)$. For a hashing power in the range $(1/4, \sqrt{2}/4)$ there is $0 < \underline{\alpha} < \overline{\alpha} < 1$ such that the equilibrium exists for any $\alpha \in (\underline{\alpha}, \overline{\alpha})$.*
- *only one player does not report: for any $\alpha \in (0,1)$ if the hashing power of the non-reporting miner is larger or equal than $1/2$. If this hashing power is in the range $(1/4, 1/2)$, there is $\underline{\alpha} \in (0,1)$ such that the equilibrium exists for any $\alpha > \underline{\alpha}$.*
- *all players report: for any $\alpha \in (0,1)$.*

Proof: See the Appendix.

From equation (5) we can find additional results which we record in several lemmata. Since all of these results are straightforward from equation (5), the proofs are omitted.

The first, and maybe the most interesting and surprising result, is that, for any player with hashing power larger than 1/4, the minimum hashing power he requires to find it sequentially rational to not report is decreasing with the number of rivals who are also not reporting.

Lemma 1. *The minimum hashing power required to not report is decreasing with the number of miners choosing NR. This minimum hashing power is bounded below by* $1/4$.

This result is illustrated in the following example.

Example 2. *Suppose* $h_j = h_k = 1/3$ *and* $\alpha = 2/3$. *Consider the case that miner* i *has solved the first puzzle and he has to decide which action to take. If* $\beta_j = \beta_k = 0$, *miner* i *needs at least a hashing power of* $7/18$ *to find it optimal to not report his new block. If only* $\beta_j = 0$ *(or, alternatively,* $\beta_k = 0$*), miner* i *needs a hashing power* $153/432$ *to not report. Finally, if* $\beta_j = \beta_k = 1$, *it is enough for miner* i *to have* $h_i = 1/3$.

Second, if the heterogeneity among miners increases, the hashing power requirements for $\beta_i = 1$ also increases, making it less likely to find a sequentially rational assessment in which some miners play that strategy.

Lemma 2. *Given* α, *an increment in the heterogeneity among miners reduces the likelihood that an assessment in which more than one miner choose action* NR *can be a sequential equilibrium.*

However, in our three miners model, there will be always one miner choosing NR as it is shown in the following example.

Example 3. *Suppose* $\alpha = 2/3$ *and* $h_i = h_j = h_k = 1/3$. *Hence, all miners find it optimal to choose* NR. *On the other hand, suppose* $h_i = 1/3$, $h_j = 2/3$ *and* $h_k = 0$. *In this case, miners* i *and* k *choose* R *while miner* j *chooses* NR.

4 Conclusion and Discussion

In this paper we study whether proof-of-work consensus mechanisms (like the one in Bitcoin) induce an "honest" behavior (i.e., to report a new block as soon as it is mined). In contrast to previous literature, we propose a model in which all players are rational.

We find that not reporting is an optimal strategy under certain conditions. For instance, the minimum hashing power required for not reporting is decreasing in the number of rivals that find it also optimal not to report. We also show that increasing the heterogeneity among miners is a useful tool to avoid that some of them find it optimal to not report.

For future research it may be interesting whether these results are robust to an increment in the number of miners. Additionally, the degree of heterogeneity is decided by the market. Miners freely increase or decrease their hashing power, they join mining pools (i.e. aggregate their hashing power), etc., depending on the expected profits. It may be also interesting to study whether there is an economic mechanism to induce miners to keep a certain degree of heterogeneity among them.

Appendix - Proofs

Proof of Remark 1: Consider that j and k are reporting. Then, miner i does not report if $h_i \geq 1/2 - \alpha h_j h_k/(h_j + h_k)$. In the limit, if h_j (or h_k) goes to *zero*, the left hand side goes to 1/2. The lowest threshold is all the remaining miners having the same hashing power, i.e., $h_j = h_k$.[12] Since $h_j + h_k = 1 - h_i$, after some algebraic manipulations, i finds it optimal to choose NR if and only if

$$2\alpha h_j^2 \geq 1 - h_i - 2h_i(1 - h_i).$$

Because $2h_j = 1 - h_i$ if $h_j = h_k$, the previous inequality is equivalent to

$$h_i \geq \frac{2-\alpha}{4-\alpha}, \text{ with } \alpha \in (0,1).$$

Thus, if there are three miners, the minimum hashing power needed by i to find it optimal to take action NR is in the range $[1/3, 1/2)$.

■

Proof of Proposition 1: We look for the conditions on the hashing power and on the parameter α that make every possible assessment sequentially rational. We do this by studying the different strategy profiles one by one. Since all the information sets are reached under the sequentially rational strategy profile under the set of beliefs, we just need to use Bayes' rule to check consistency.

1- Suppose $\beta_i = 1$ for every miner. Then, it must be that condition

[12] To see this, note that we can replace $h_j + h_k$ by $1 - h_i$. Next, $h_j h_k$ has a maximum if $h_j = h_k$ for a given h_i.

(5) holds for every miner while $h_1 + h_2 + h_3 = 1$. After some algebraic manipulations, those are satisfied only if

$$\alpha \geq 1 + \frac{h_1^2 + h_2^2 + h_3^2 - 12h_1h_2h_3}{h_2h_3 + h_1h_3 + h_1h_2}.$$

The previous inequality relaxes (i.e., the right hand side is minimized) if $h_i = h_j = h_k = 1/3$, yielding $\alpha \geq 2/3$. Any other distribution of the hashing powers will require a larger α to satisfy (5) for every miner.

The beliefs for every miner i sustaining this equilibrium are given by:

$$\mu_i(h_i) = \frac{1}{1 + h_j + 2h_jh_k + h_k},$$
$$\mu_i(h_j) = \frac{h_j}{1 + h_j + 2h_jh_k + h_k},$$
$$\mu_i(h_j, h_k) = \frac{h_jh_k}{1 + h_j + 2h_jh_k + h_k}.$$

2- Without lost of generality, let's consider $\beta_i = \beta_j = 1$ and $\beta_k = 0$. Therefore, condition (5) must be satisfied for miners i and j,

$$h_i \geq \frac{1}{2} - \alpha\frac{h_jh_k}{h_j + h_k} + \frac{h_jh_k[1 - 4h_i]}{2(h_j + h_k)},$$
$$h_j \geq \frac{1}{2} - \alpha\frac{h_ih_k}{h_i + h_k} + \frac{h_ih_k[1 - 4h_j]}{2(h_i + h_k)},$$

while for miner k,

$$h_k < \frac{1}{2} - \alpha\frac{h_ih_j}{h_i + h_j} + \frac{h_ih_j[1 - 4h_k]}{(h_i + h_j)}.$$

The set of the previous inequalities holds if and only if,

$$\begin{aligned}&2h_i(1 - h_i) + 2h_j(1 - h_j) - 2h_k(1 - h_k) > \\ &\qquad 1 - (h_i + h_j) + h_k + 2\alpha(h_ih_j - h_ih_k - h_jh_k) + h_k(h_i + h_j) - 2h_ih_j,\end{aligned}$$

and after some manipulations and using $h_i + h_j = 1 - h_k$, if and only if

$$h_k^2 - 3h_k + 6h_ih_j > 2\alpha(h_ih_j - h_k(h_i + h_j)). \tag{6}$$

Since

$$\begin{aligned} h_k^2 - 3h_k &= h_k(h_k - 1) - 2h_k, \\ &= -h_k(h_i + h_j) - 2h_k, \end{aligned}$$

equation (6) can be written as

$$h_i h_j - h_k(h_i + h_j) - 2h_k + 5h_i h_j > 2\alpha(h_i h_j - h_k(h_i + h_j)).$$

We have two subcases: (i) $h_i h_j < h_k(h_i + h_j)$, and (ii) $h_i h_j > h_k(h_i + h_j)$. (i) If $h_i h_j < h_k(h_i + h_j)$,

$$\alpha > \frac{2h_k - 5h_i h_j}{2(h_k(h_i + h_j) - h_i h_j)} + \frac{1}{2}. \tag{7}$$

In case that the right hand side (rhs) is lower than zero, the inequality is satisfied by any $\alpha \in (0, 1)$. This happens if

$$\frac{2h_k - 5h_i h_j}{2(h_k(h_i + h_j) - h_i h_j)} + \frac{1}{2} < 0 \quad \Leftrightarrow \quad h_k(3 - h_k) < 6h_i h_j.$$

Because $h_i h_j < h_k(1 - h_k)$, we have

$$h_k(3 - h_k) < 6h_i h_j < 6h_k(1 - h_k),$$

and is necessary (but not sufficient) to have $h_k < 3/5$.

On the other hand, it must be that

$$\frac{h_i h_j}{1 - h_k} < h_k < \frac{1}{2} - \alpha\frac{h_i h_j}{h_i + h_j} + \frac{h_i h_j[1 - 4h_k]}{(h_i + h_j)}. \tag{8}$$

Thus,

$$\frac{h_i h_j(4h_k + \alpha)}{1 - h_k} < \frac{1}{2} \quad \Leftrightarrow \quad \alpha < \frac{1 - h_k}{2h_i h_j} - 4h_k.$$

Since, by assumption $(1 - h_k)h_k > h_i h_j$, it is sufficient to ask,

$$\alpha < \frac{1}{2h_k} - 4h_k.$$

The right-hand side is larger than 1 if $h_k < 1/4$. Thus, with any $h_k < 1/4$ there exists $\alpha \in (0, 1)$ which satisfies equation (7) and (8).

(ii) If $h_ih_j > h_k(h_i+h_j)$,

$$\alpha < \frac{1}{2} - \frac{2h_k - 5h_ih_j}{2(h_ih_j - h_k(h_i+h_j))}. \tag{9}$$

If the right-hand side is lower than zero, there is no α satisfying the set of inequalities. This happens if

$$\begin{aligned} &\frac{1}{2} - \frac{2h_k - 5h_ih_j}{2(h_ih_j - h_k(h_i+h_j))} < 0, \\ \Leftrightarrow \quad & 6h_ih_j < h_k(3-h_k). \end{aligned}$$

By assumption $h_k(1-h_k) < h_ih_j$, therefore, it must be that $6h_k(1-h_k) < h_k(3-h_k)$, which holds if $h_k > 3/5$. Hence, $h_k < 3/5$ in order for a range of $\alpha \in (0,1)$ to exist satisfying the inequalities.

To have that any $\alpha \in (0,1)$, the right-hand side must be larger or equal than one. In such a case

$$-\frac{1}{2} > \frac{2h_k - 5h_ih_j}{2(h_ih_j - h_k(h_i+h_j))} \quad \Leftrightarrow \quad h_k(1+h_k) < 4h_ih_j.$$

Since $h_ih_j > h_k(h_i+h_j)$, either $4h_ih_j > h_k(1+h_k) > 4h_k(h_i+h_j)$ or $4h_ih_j > 4h_k(h_i+h_j) > h_k(1+h_k)$. The first case is satisfied if $h_k > 3/5$, yielding to a contradiction. The second case is satisfied for any $h_k < 3/5$. If

$$\frac{h_ih_j}{1-h_k} < \frac{1}{2} - \alpha\frac{h_ih_j}{h_i+h_j} + \frac{h_ih_j[1-4h_k]}{(h_i+h_j)},$$

then

$$\alpha < \frac{1-h_k}{2h_ih_j} - 4h_k.$$

The right-hand side is larger than one if $(1-h_k)/(2h_ih_j) > 1+4h_k$. Because by assumption $1/(2h_k) > (1-h_k)/(2h_ih_j)$, it is necessary $h_k < 1/4$. Finally, if

$$\frac{h_ih_j}{1-h_k} > \frac{1}{2} - \alpha\frac{h_ih_j}{h_i+h_j} + \frac{h_ih_j[1-4h_k]}{(h_i+h_j)},$$

then

$$\alpha > \frac{1-h_k}{2h_ih_j} - 4h_k.$$

If the right-hand side is below 0, then any $\alpha \in (0,1)$ satisfies the requirements for this equilibrium. This happens if $(1-h_k)/(2h_ih_j) < 4h_k$. Since $1/h_k > (1-h_k)/(h_ih_j)$, it is sufficient to ask $\sqrt{2}/4 < h_k$.

The beliefs sustaining this equilibrium are given by:

$$\begin{aligned}
&\mu_i(h_i) = \frac{1}{1+h_j}, \quad \mu_i(h_j) = \frac{h_j}{1+h_j}, \quad \mu_i(h_j, h_k) = 0,\\
&\mu_j(h_j) = \frac{1}{1+h_i}, \quad \mu_j(h_i) = \frac{h_i}{1+h_i}, \quad \mu_j(h_i, h_k) = 0,\\
&\mu_k(h_k) = \frac{1}{1+h_j+2h_jh_i+h_i}, \quad \mu_k(h_i) = \frac{h_k}{1+h_j+2h_jh_i+h_i}, \quad \mu_k(h_j, h_i) = \frac{h_jh_i}{1+h_j+2h_jh_i+}
\end{aligned}$$

3- Suppose that $\beta_i = 1$ and $\beta_j = \beta_k = 0$. Now, we need

$$\begin{aligned}
&h_i \geq \frac{1}{2} - \alpha\frac{h_jh_k}{h_j+h_k},\\
&h_j < \frac{1}{2} - \alpha\frac{h_ih_k}{h_i+h_k} - \frac{h_ih_k[3-4(h_i+h_k)]}{2(h_i+h_k)},\\
&h_k < \frac{1}{2} - \alpha\frac{h_jh_i}{h_j+h_i} - \frac{h_jh_i[3-4(h_j+h_i)]}{2(h_j+h_i)}.
\end{aligned}$$

These conditions are satisfied if and only if

$$\alpha \begin{cases} > \frac{(1-h_i)(1-3h_i)-(1-h_j)(1-2h_j)-(1-h_k)(1-2h_k)+8h_ih_jh_k}{2(h_jh_k-h_ih_k-h_ih_j)} & \text{if } h_i < \frac{h_jh_k}{h_j+h_k},\\ < \frac{(1-h_i)(1-3h_i)-(1-h_j)(1-2h_j)-(1-h_k)(1-2h_k)+8h_ih_jh_k}{2(h_jh_k-h_ih_k-h_ih_j)} & \text{if } h_i > \frac{h_jh_k}{h_j+h_k},\\ \text{indeterminate} & \text{if } h_i = \frac{h_jh_k}{h_j+h_k}. \end{cases}$$

Take the first case. It must be that

$$\frac{h_jh_k}{h_j+h_k} > h_i \geq \frac{1}{2} - \alpha\frac{h_jh_k}{h_j+h_k},$$

which is satisfied if and only if

$$\alpha > \frac{h_j+h_k}{2h_jh_k} - 1.$$

If the right-hand side is lower than zero, any $\alpha \in (0,1)$ sustains the equilibrium. This is,

$$\frac{h_j+h_k}{2h_jh_k} - 1 < 0 \quad \Leftrightarrow \quad \frac{h_j+h_k}{h_jh_k} < 2.$$

Since $(h_j + h_k)/(h_j h_k) < 1/h_i$ by assumption, if $1/h_i < 2$ (or $1/2 > h_i$) then $(h_j + h_k)/(h_j h_k) < 2$. If $1/2 > h_i$, there is a right-hand side larger than zero and α must be larger than it to sustain the equilibrium.

In the extreme case, if the right-hand side is larger than one, there is not $\alpha \in (0, 1)$ sustaining all the inequalities. Therefore,

$$\frac{h_j + h_k}{2h_j h_k} - 1 < 1 \quad \Leftrightarrow \quad \frac{h_j + h_k}{h_j h_k} < 4.$$

Since $(h_j + h_k)/(h_j h_k) < 1/h_i$ by assumption, if $1/h_i < 4$ (or $h_i > 1/4$) then $(h_j + h_k)/(h_j h_k) < 4$. Thus, for any $h_i > 1/4$, there is $\underline{\alpha} < 1$ such that the three inequalities hold for any $\alpha > \underline{\alpha}$.

Consider $h_i > h_j h_k/(h_j + h_k)$, and let's check if

$$\frac{(1 - h_i)(1 - 3h_i) - (1 - h_j)(1 - 2h_j) - (1 - h_k)(1 - 2h_k) + 8h_i h_j h_k}{2(h_j h_k - h_i h_k - h_i h_j)} \in (0, 1).(10)$$

Since $h_i > h_j h_k/(h_j + h_k)$, the denominator is lower than zero. As a consequence, to have the right-hand side larger than zero, it must be that the numerator is also lower than zero. This is, after some operations,

$$h_i^2 - 3h_i + 4h_j h_k(1 + 2h_i) < 0,$$

for all $h_i(1 - h_i) > h_j h_k$. Hence,

$$h_i(h_i - 3) + 4h_i(1 - h_i)(1 + 2h_i) < 0 \quad \Leftrightarrow \quad -8h_i^2 + 5h_i + 1 < 0,$$

which holds for all $h_i > 5/16 + \sqrt{57}/16$.

If the numerator is larger than the denominator,

$$h_i^2 - 3h_i + 4h_j h_k(1 + 2h_i) > 2h_j h_k - 2h_i(1 - h_i) \quad \Leftrightarrow \quad 2h_j h_k(1 + 4h_i) > h_i(1 + h_i),$$

for all $h_i(1 - h_i) > h_j h_k$, implying

$$2h_i(1 - h_i)(1 + 4h_i) > h_i(1 + h_i) \quad \Leftrightarrow \quad -8h_2 + 5h_i + 1 > 0,$$

yielding a contradiction.

Therefore, the numerator of equation (10) must be lower than the denominator with $h_i > 5/16 + \sqrt{57}/16 > 1/2$ and, as consequence, any $\alpha \in (0, 1)$ sustains the equilibrium.

The set of beliefs sustaining this equilibrium is:

$$\mu_i(h_i) = 1, \quad \mu_i(h_j) = 0, \quad \mu_i(h_j, h_k) = 0,$$
$$\mu_j(h_j) = \frac{1}{1+h_i}, \quad \mu_j(h_i) = \frac{h_i}{1+h_i}, \quad \mu_j(h_i, h_k) = 0,$$
$$\mu_k(h_k) = \frac{1}{1+h_i}, \quad \mu_k(h_i) = \frac{h_i}{1+h_i}, \quad \mu_k(h_i, h_j) = 0.$$

4- Suppose that $\beta_i = 0$ for every i. Hence, for every miner i,

$$h_i < \frac{1}{2} - \alpha\frac{h_j h_k}{h_j + h_k}. \tag{11}$$

After some manipulations, all conditions on the hashing powers are satisfied if and only if

$$\alpha < \frac{h_i^2 + h_j^2 + h_k^2}{h_j h_k + h_i h_k + h_i h_j}.$$

The previous inequality becomes more strict if miners are homogeneous (i.e., $h_i = 1/3$ for every i), requiring $\alpha \leq 1$. Therefore, any $\alpha \in [0, 1)$ satisfies (11) for all miners and for any distribution of the hashing powers.

The beliefs for every miner i sustaining this equilibrium are given by:

$$\mu_i(h_i) = 1, \quad \mu_i(h_j) = 0, \quad \mu_i(h_j, h_k) = 0.$$

■

References

Böhme, R., Christin, N., Edelman, B. & Moore, T. (2015), ‘Bitcoin: Economics, technology, and governance’, *The Journal of Economic Perspectives* **29**(2), 213–238.

Decker, C. & Wattenhofer, R. (2013), ‘Information propagation in the bitcoin network’, *Peer-to-Peer Computing (P2P), 2013 IEEE Thirteenth International Conference on* .

Dwork, C. & Naor, M. (1992), ‘Pricing via processing or combatting junk mail’, *Annual International Cryptology Conference* .

Eyal, I. & Sirer, E. (2014), ‘Majority is not enough: Bitcoin mining is vulnerable’, *International Conference on Financial Cryptography and Data Security* pp. 436–454.

Jaag, C. & Haller, A. (2017), ‘Economic mechanism design in bitcoin: Mining, block formation and transaction fees’, *Swiss Economics Working Paper* .

Nayak, K., Kumar, S., Miller, A. & Shi, E. (2016), ‘Stubborn mining: Generalizing selfish mining and combining with an eclipse attack’, *IEEE European Symposium on Security and Privacy* pp. 305–320.

Osborne, M. & Rubinstein, A. (1994), *A Course in Game Theory.*

Rosenfeld, M. (2011), ‘Analysis of bitcoin pooled mining reward systems’, *arXiv preprint arXiv: 1112.4980* .