Computational Economics
https://doi.org/10.1007/s10614-023-10503-2

A Note on the Non-proportionality of Winning Probabilities in Bitcoin

José Parra-Moyano[1] · **Gregor Reich**[2] · **Karl Schmedders**[1]

Accepted: 13 October 2023

Abstract
The security of any proof-of-work blockchain protocol is based upon the assumption that the probability of a miner finding the next valid block is proportional to that miner's hashing power and constant throughout the process of mining that block (i.e., that the mining process is a "memoryless" process). While the literature assumes that the mining process is indeed memoryless, in this paper we use deductive reasoning to show how, given the finiteness of hashing functions' domains, this is not the case. This implies that the Bitcoin protocol induces a centralization of miners' hashing power, which in turn threatens the long-term viability of Bitcoin and of other cryptocurrencies based on similar protocols. The novelty of this paper stems from our documenting of a previously unrecognized flaw in the incentive system sustaining Bitcoin's security.

Keywords Bitcoin · Negative hypergeometric · Poisson distribution · Quantum computing

1 Introduction

The key differential characteristic of a blockchain is its protocol. A blockchain protocol is an economic mechanism that incentivizes miners (i.e., the agents participating in the blockchain-based system) to correctly process information in a decentralized manner. The first example of an application based on a blockchain protocol was Bitcoin, a decentralied peer-to-peer currency first conceived and described by Nakamoto (2008). The protocol that gave rise to Bitcoin was the first of its kind, and belongs to the family of proof-of-work protocols.

✉ José Parra-Moyano
jose.parramoyano@imd.org

[1] IMD Lausanne, Chemin de Bellerive 23, Lausanne, Switzerland

[2] Tsumcor Research AG, Sonnenbergstrasse 74, Schwerzenbach, Switzerland

Published online: 31 October 2023

In the Bitcoin protocol (just like in every other proof-of-work protocol), miners compete with one another in a trial-and-error contest to find the next valid block of the blockchain. Miners pass potentially valid blocks (i.e., blocks that respect the structure defined by the protocol and only contain correct transactions) through a hash function in the hope that the resulting hash is below a predefined threshold called "the target." A valid block is a block that not only respects the structure of the protocol and only contains correct transactions, but whose hash is below "the target.". If a potentially valid block results in a non-valid block once it has been hashed (i.e., if it is potentially valid but its hash is above the target), the miner changes a small component of the block called the "nonce" (an integer) and hashes the resulting, new version of the originally potentially valid block again. Every miner hashes a series of potentially valid blocks until one finds a valid block and wins the contest. The contest then starts anew. Winning this contest (i.e., finding a valid block) is rewarded with newly created bitcoins.

The properties that blockchain protocols need to possess in order to enable a decentralized system that relies on anonymous, profit-driven miners who can freely join the system—i.e., a system fulfilling the promise made by Nakamoto (2008)—have been formalized in three axioms (Leshno & Strack, 2020). One of these states that miners can have no incentive to centralize their resources (i.e., their hashing power). In the case of Bitcoin, this axiom implies that the change in any miner's winning probability resulting from a change in that miner's hashing power needs to be directly proportional to that change in hashing power. Should this axiom not be fulfilled, the Bitcoin protocol would incentivize the centralization of the miners' hashing power, and thus, Bitcoin would not anymore be the decentralized system that it aims to be. In other words, Bitcoin's value proposition (the decentralization that avoids the double-spending of peer-to-peer digital currencies) would fall apart.

So far, the vast majority of the literature has assumed that a miner's probability of finding a new block in the Bitcoin blockchain (and in Bitcoin-like blockchains) follows a negative binomial distribution, which implies that the axiom whereby miners have no incentive to centralize their resources is fulfilled. Hence, scholars consider the process of bitcoin mining to be a memoryless process (memoryless in the sense that miners forget the nonces that have previously resulted in hashes below the target, such that every newly tried nonce has the same success probability than the previous one, no matter how many nonces have been tried before). However, parts of this assumption are increasingly being questioned. The work conducted by Grunspan and Pérez-Marco (2017) corrects the analysis given in Nakamoto (2008) regarding the success of double-spending attacks on the Bitcoin blockchain. Specifically, Grunspan and Pérez-Marco (2017) demonstrate that the success of double-spending attacks is not consistent with assuming that the winning probabilities of bitcoin miners follow a negative binomial distribution. Along these same lines, Bowden et al. (2018) challenge the assumption of the block arrival rate following a negative binomial distribution. But while they demonstrate that the block arrival rate does not in fact follow such a distribution, they leave the question of *"How does the arrival rate of blocks really behave?"* open.

This is the question that we anser with our work, in which we show how, due to the—often overlooked—finiteness of the domain of hashing functions,

proof-of-work–based cryptocurrencies (including Bitcoin and Bitcoin-like cryptocurrencies) follow a negative hypergeometric distribution.

This result has an important implication for scholarship and practice, as it implies that the winning probability for a miner mining a block are non-proportional to that miner's hashing power and non-constant throughout the time spent mining that block. From this result it emerges that, at a theoretical level, the axiom whereby miners can have no incentive to centralize their resources is not fulfilled by Bitcoin, or by any Bitcoin-like protocol. This reveals a misconception in the understanding of proof-of-work protocols, and a flaw in the economic incentive structure that sustains proof-of-work protocols such as that of Bitcoin. Given that the viability of Bitcoin (and of any Bitcoin-like protocol) relies on the economic incentives that its protocol offers (Ciaian et al., 2021), our findings reveal an unknown threat to Bitcoin's and similar systems' long-term viability. Moreover, as mining pools base the way in which they divide their earnings among the miners on the assumption that miners have no incentive to centralize their resources, our result force such pools to search for alternative ways to compensate miners for their work.

The rest of the paper is organized as follows. In Sect. 2 we review the most relevant related literature. In Sect. 3 we conduct a theoretical analysis around proof-of-work mining and illustrate the non-proportionality of the selection rule for Bitcoin. In Sect. 4 we run a simulation to compare the relative probabilities for miners of different sizes when the memoryless property is fulfilled, and when it is not. Additionally, we describe the challenges that mining pools face when assessing the actual hashing power of miners, something that complicates the solution to the problem we are describing. In Sect. 5 we discuss the implications of our work, and in Sect. 6 we conclude.

2 Literature and Background

It is broadly accepted that the process of finding a valid block in proof-of-work protocols follows a negative binomial distribution (as a limiting case of the negative binomial distribution). This is assumed by authors such as John et al. (2022), Nakamoto (2008), Rosenfeld (2011), and Cocco and Marchesi (2016) when introducing and studying the properties of bitcoin mining, and by authors such as Li et al. (2023), Eyal and Sirer (2018), Houy (2016), Dimitri (2017), and Wang et al. (2019) when studying the incentive structure for miners. Authors including Rosenfeld (2014), Solat and Potop-Butucaru (2016), Beccuti and Jaag (2017), and Aggarwal et al. (2018) also make similar assumptions when studying the security aspects of the Bitcoin protocol. When studying information propagation in the Bitcoin network, Miller and La Viola (2014), Göbel et al. (2015), and Lewenberg et al. (2015) repeat this assumption, and assumption that is also made by Decker and Wattenhofer (2013) when studying the creation of forks in the Bitcoin network, by Cong et al. (2018) and Hayes (2019) when creating models for bitcoin valuation, by Cong et al. (2019) when studying mining pool centralization, by Easley et al. (2019) when analyzing the evolution of Bitcoin transaction fees, and by Chiu and Koeppl (2017) when studying the optimal design of cryptocurrencies.

Other papers that assume proportional winning probabilities include that of Halaburda et al. (2022), which studies the microeconomics of cryptocurrencies, that of Easley et al. (2019), which studies the evolution of transaction fees in Bitcoin, that of Böhme et al. (2015), which points out risks and regulatory issues as Bitcoin interacts with the conventional financial system and the real economy, that of Benigno et al. (2019), which studies the classic "impossible trinity" in a two-country economy with complete markets, that of Schilling et al. (2020), which studies some elements of central bank digital currencies (CBDCs), and that of Athey et al. (2015), which develops a model of user adoption and use of virtual currencies.

The assumption that finding a block in a proof-of-work blockchain follows a negative binomial distribution implies that miners' probabilities of winning are directly proportional to their hashing power, and therefore remain constant throughout the mining process (i.e., given a constant hashing power, the winning probability remains constant throughout the time that elapses between the moment at which the miner starts trying to find the next valid block and the moment at which any miner finds and broadcasts that valid block). This implies that for every attempt (i.e., for every newly generated hash) the winning probability is independent of the number of previously tried and failed attempts. In other words, this assumption implies that there is no *within-block learning* when mining blocks and that thus.

This lack of *within-block learning* is what Cong et al. (2019) call the "well-known 'memoryless' property" of proof-of-work mining, which "implies that the event of finding a solution is captured by a negative binomial process with the arrival rate proportional to a miner's share of hash rates globally" as described by Eyal and Sirer (2018) and Sapirshtein et al. (2015). By assuming that the arrival rate of blocks follows a Poisson distribution (a limiting case of the negative binomial distribution), all these authors are assuming (as stated by Cong et al., 2019) that mining is a memoryless process, and therefore that the probability of finding a valid block is independent of previous attempts to do so.

The idea behind proof-of-work mining being a memoryless process is captured by one of the axioms that blockchain protocols need to fulfill in order to enable a decentralized system that relies on anonymous, profit-driven miners who can freely join the system (Leshno & Strack, 2020)—namely, the axiom stating that miners can have no incentive to centralize their resources. From this axiomatic formalization results that any protocol in which a miner i devotes an amount of work x_i (hashing power in the case of Bitcoin) to finding the next valid block of the blockchain will only be anonymous, robust to Sybil attacks (such that miners cannot split their performance to increase their probabilities of winning), and robust to merging (such that miners cannot increase their selection probability by merging) if the selection mechanism is given by a proportional selection rule (Halaburda et al., 2022). This proportional selection rule is the same rule that determines the winning probability of drawing a winning ball from an urn, assuming that non-winning balls are returned to the urn (i.e., drawing from an urn *with* replacement). The rule implies that regardless of the number of versions of a particular potentially valid block that a miner has

previously generated, the probability that the next version of the block (the same block with a different nonce) results in a hash below the threshold remains constant.[1]

While this assumption about proportional winning probabilities in proof-of-work protocols is widespread, parts of it are increasingly being questioned. The work conducted by Grunspan and Pérez-Marco (2017) corrects the analysis given in Nakamoto (2008) regarding the success of double-spending attacks on the Bitcoin blockchain. Grunspan and Pérez-Marco (2017) give a closed-form formula for the probability of success of a double-spending attack, and in doing so assume that the number of blocks $N(t)$ mined at time t follows the negative binomial distribution. Grunspan and Pérez-Marco (2017) do not revise or study the arrival rate of blocks, but do demonstrate that one of the characteristics of the Bitcoin blockchain that emerges from this arrival rate–namely, the probability of the success of double-spending attacks–differs from what was previously assumed in the literature. Their work is especially important for us since while it still accepts that the arrival rate of blocks follows the negative binomial distribution, it challenges the assumption implied by Nakamoto (2008) that the probability of the success of a double-spending attack also follows a negative binomial distribution.

Along these same lines, Ciaian et al. (2021) question the memoryless property by stating that "the probability of winning a mining contest increases with the miner size." Bowden et al. (2018), meanwhile, challenge the assumption of the block arrival rate following the negative binomial distribution, and based on a stochastic analysis of the block arrival process demonstrate that this is indeed not the case. They present a refined mathematical model for block arrivals, focusing on both block arrivals during a period of constant difficulty and how the difficulty level evolves over time. Their work, however, leaves the question of "How does the arrival rate of blocks really behave?" open. This question motivates our work.

3 Analysis

In this section we briefly describe the concept of hashing, illustrate how the Bitcoin protocol works, and by using deductive reasoning derive why the selection rule in proof-of-work protocols is based on non-proportional probabilities.

3.1 Fundamentals of the SHA-256 Function

The SHA-256 (Secure Hash Algorithm 256) function is a cryptographic, one-way compression function. The domain of the SHA-256 function is composed of any string of a length of up to 2^{64} bits. This implies that the domain of the SHA-256 function, while being colossal, is finite.

The result of a hashing function is called a "hash." The SHA-256 function is a one-way (non-invertible) function. Hence, the original data can not be retrieved

[1] From this point on we will use the terms "selection rule" and "winning probability" in an interchangeable manner.

from the resulting data (i.e., only by conducting a trial-and-error process can one find the input that yields a particular output).

3.2 Bitcoin Mining as an Urn Problem

The Bitcoin protocol defines how bitcoins come into existence and how bitcoin transactions are validated. Miners write blocks (pieces of information that contain information about previous transactions) and compete to position their block as the next valid block of the blockchain. Every block has a block header. A block's header has a size of 80 bytes (Antonopoulos, 2014) and, as explained by Courtois et al. (2014), contains the following information: The version number of the protocol (with a size of 4 bytes); the hash of the previous block's header (with a size of 32 bytes); the Merkle root (with a size of 32 bytes); the timestamp (with a size of 4 bytes); the target (with a size of 4 bytes); the padding + len (with a size of 4 bytes); and the nonce (with a size of 4 bytes). Miners fix all the components of the block, and for that particular block try different nonces.

A block is accepted by other miners as "the next valid block" of the blockchain only if it fulfills two conditions. First, the information contained in the block respects the rules determined by the Bitcoin protocol: it can only contain new and legitimate transactions made by the blockchain's users and it contains a (hashed) reference to the last valid block. We refer to blocks that fulfill this condition as "potentially valid blocks." Second, the hash of this block's header (i.e., the result of passing the block's header through the SHA-256 function) results in a number that is below a certain threshold, called the "target."[2]

Miners write potentially valid blocks, expecting the hash of their block's header to be smaller than the target. Whenever the hashing of a potentially valid block's header results in a non-valid hash (i.e., a hash that is greater than the target), miners change a small component of the block's header (an integer called the "nonce") that does not affect the correctness of the information contained in the block (such that this block still respects the first condition) but that makes this new version of the block's header result in a new, different hash. Once a block is accepted as valid, a new problem using the hash of this newly accepted block as the one of the inputs for the next block starts for all the miners. It is essential to note that a one-bit change in the hashed input (the computationally smallest unit that can be changed) results in a completely new hash that has no relationship whatsoever with the previous one. Most importantly, since the hashing function is non-invertible it is impossible to anticipate which input (which nonce) is going to yield a particular output. This is the cornerstone of proof-of-work mining: for a potentially valid block, miners have to try many different 32-bit random nonces to potentially find the next valid block. This design makes mining under this protocol a binary stochastic process.

[2] In fact, miners pass the block header through the SHA-256 function and hash the resulting 256-bit string again using the same SHA-256 function. It is the second hashing that needs to result in a number below the target. Since the SHA-256 function is deterministic, this does not alter our calculations.

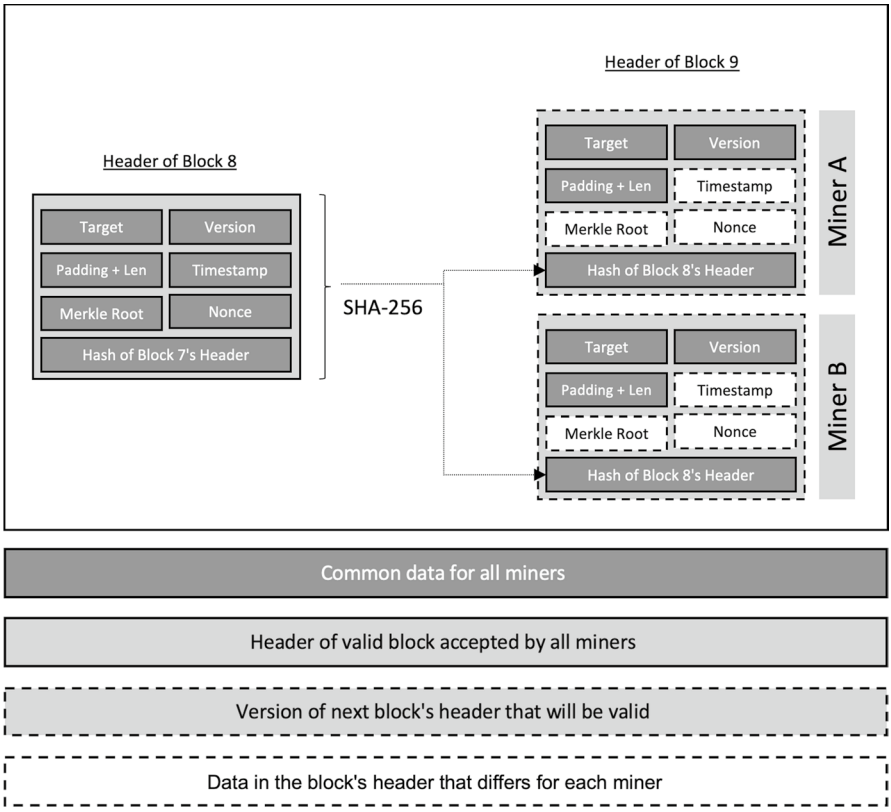

Fig. 1 Illustration of a blockchain and the different versions of a block that can become the next valid block. Own illustration

Since the difficulty of the mining process has increased so much, sometimes miners do not find a valid block when iterating over all the nonces for their block. In such a case, miners change the coinbase (a part of the Merkle tree with enough space to add some free text), such that they increase the space they can explore in their search for the next valid block. The size of the coinbase transaction is 8 bytes. This implies that miners have 4 bytes of the nonce plus 8 bytes of the coinbase in terms of space in which to make combinations (Antonopoulos, 2014).

Figure 1 illustrates the creation of a block. In Fig. 1, the hash of Block 8's header serves as the basis for two different, potentially valid blocks, each from one of two competing miners (Miner A and Miner B).

Since generating a new hash requires electricity, miners have an incentive to only hash blocks that have the potential to be valid. Hashing a block that contains transactions that contravene the rules of the protocol will result in the rest of the miners rejecting that block, even if the hash of the block is below the target. With this mechanism, the Bitcoin protocol prevents the double-spending problem. In other words, systems based on blockchain protocols are reliable because the expected

value of writing wrong transactions in the ledger is negative, and miners are therefore deterred from such illicit behavior. This, however, is only the case as long as the winning probabilities in proof-of-work protocols are proportional to the effort (electricity) spent on mining. Should larger miners gain a non-proportional (positive) wining probability relative to their size, then miners would have an incentive to pool their resources, which would result, from a strategic point of view, in a very large mining pool controlling the totality of the hashing power. Such a situation would turn Bitcoin (and any Bitcoin-like system) centralized and reliant on one central authority. Thus, the whole purpose of Bitcoin (and of any system based on a Bitcoin-like blockchain) would disappear.

3.3 Statistical Properties of Proof-of-Work Mining

Given the finite set of combinations that a miner can try in order to find the next valid block of a proof-of-work blockchain, the process of proof-of-work mining has the same structure as the classical urn problem. The simplest urn problem that one can think of consists of an urn containing balls of two different colors. One person draws balls from the urn. The problem can be set such that drawing a ball of one color (green) represent a success, whereas drawing a ball of the other color (red) represents a failure. In the Bitcoin setting, each miner draws balls from their personal urn. The urn contains all the versions of the potentially valid block that a miner is aiming to mine (i.e., the different versions of the same block's header with different nonces). Hashing the potentially valid block with one of the nonces is equivalent to drawing a ball from the urn, and hence it is a process that can result in either success (a hash below the target) or failure (a hash above the target). Each miner is confronted with such an "urn" and successively tries different inputs (nonces) until it or another miner finds a truly valid block (success). Note that the urn of each miner is different from that of each other miner, as the blocks differ in at least one respect: the address to which the newly created bitcoins need to be sent should that block become the next valid block of the blockchain.

Urn problems can be of two types: urns with or without replacement. In an urn problem with replacement, the balls are returned to the urn once they are drawn. Hence, in an urn with balls of two colors, red and green, in which balls are drawn until a green ball is drawn, the probability of drawing a green ball from the urn remains constant, no matter how many red balls have been previously drawn. In an urn problem without replacement however, the probability of drawing a green ball from the urn increases every time that a red ball is drawn. The fact that nonces known to yield a non-valid hash for a particular potentially valid block's header are not tried twice implies that this problem has the structure of the urn problem "without replacement," in which balls representing a failure event are not returned to the urn.

Schemes of sampling from a finite population without replacement like the one that we are describing here are governed by the negative hypergeometric distribution. Consider a total population of N elements, of which M elements are labeled a "success" and the remaining $N - M$ elements are considered a "failure." Suppose we

select elements out of the population without replacement until the number of "successes" reaches a fixed number m. Define the random variable X as the total number of draws (without replacement) in the sample until (and including) the mth success. Then, X follows the negative hypergeometric distribution $X \sim \mathrm{NHG}(N, M, m)$. The probability mass function (PMF) of the negative hypergeometric distribution is given by

$$\Pr(X = x) = \frac{\binom{x-1}{m-1}\binom{N-x}{M-m}}{\binom{N}{M}}.$$

For the special case of a single success, $m = 1$, the PMF simplifies to

$$\Pr(X = x) = \frac{\binom{N-x}{M-1}}{\binom{N}{M}}.$$

As the number of hashes made by a miner can be expressed in relation to units of time (i.e., in hash *rate*), this implies that the probability that a hash results in a valid block increases with the time a miner spends attempting to solve a particular block. Therefore, the winning probability in proof-of-work protocols is non-proportional, a fact that is beneficial for relatively larger miners. This is at odds with the axiomatic formalization derived by Leshno and Strack (2020) and more importantly, implies that proof-of-work protocols like the Bitcoin protocol are not *memoryless* processes that induce no centralization of the hashing power.

4 Scenario Comparison

In this section we compare the differences in the relative winning probabilities of miners when assuming the mining process follows the negative binomial distribution vs. the negative hypergeometric distribution. Moreover, as the negative binomial distribution can be used to approximate the negative hypergeometric distribution, we derive the size of the total hashing power that must be reached for that approximation to result in significant errors. Finally, we comment on the difficulty of measuring the actual hashing power of a network.

4.1 Comparison of Winning Probabilities

For illustration purposes, let us assume a small, simplified domain of a hash function that only accepts 100,000 possible inputs. This means that the domain of this function contains 100,000 elements. Moreover, let us assume that we have two miners, A and B, each of which has written a potentially valid block whose respective

Table 1 Relationship of winning probabilities

Relationship of winning probabilities							
Distribution	Metric	1 s	2 s	5 s	10 s	100 s	200 s
Neg. Bin	Winning Prob. A	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001
	Winning Prob. B	0.0010	0.0010	0.0010	0.0010	0.0010	0.0010
	Winning Prob. B/A	10.000	10.000	10.000	10.000	10.000	10.000
Neg. Hyp	Winning Prob. A	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001
	Winning Prob. B	0.0010	0.0010	0.0010	0.0010	0.1011	0.0120
	Winning Prob. B/A	10.000	10.009	10.036	10.082	10.989	12.236

headers have space for 10,000 possible different nonces each. Let us continue by assuming that Miner A has a hash rate of 1 hash per second (i.e., it can compute one hash per second) whereas Miner B has a hash rate of 10 hashes per second (i.e., it can compute 10 hashes per second). Finally, let us assume that out of these 10,000 combinations of block + nonce, the difficulty level is defined such that exactly one of the 10,000 combinations yields a valid block for each of the miners. While this is an oversimplified version of mining, it serves the purpose of illustrating the differences between using the negative binomial distribution and the negative hypergeometric distribution.

Table 1 presents the winning probabilities and relative winning odds of Miner B with respect to Miner A, under each of the different distributions and at different points in time given that no valid block has previously been found. In this example, under the negative binomial distribution assumption (urn with replacement) the probability of each miner finding the next valid block remains constant during the time they spend mining. Therefore, the probability of Miner B winning is 10 times larger than that of Miner A winning. The value of 10 is consistent with the fact that Miner B has a hash rate that is ten times larger than the hash rate of Miner A. This construct has the same structure as the urn problem with replacement (the probabilities of winning are independent of the number of previously tried and failed attempts, such that there is no learning).

However, under the assumption of a negative hypergeometric distribution the winning probability of each miner increases across the time that the miners spend mining (i.e., across the time that a miner spends trying and failing). This construct has the same structure as the urn problem without replacement (the probabilities of winning are dependent on the number of previously tried and failed attempts, such that there is learning). It is interesting to see how the winning probability of Miner A, while increasing across time, is the same across time once it is rounded to four significant digits. This illustrates how the increase in the winning probability is very sensitive to the number of possible combinations (i.e., the increase in the winning probability is very sensitive to the number of balls in the urn). Comparing the winning probability of Miner B with that of Miner A, we see an increase over time. This is the crucial aspect of the whole analysis. Bigger miners (in terms of their hash rate) explore the space faster than smaller miners and by doing so increase their

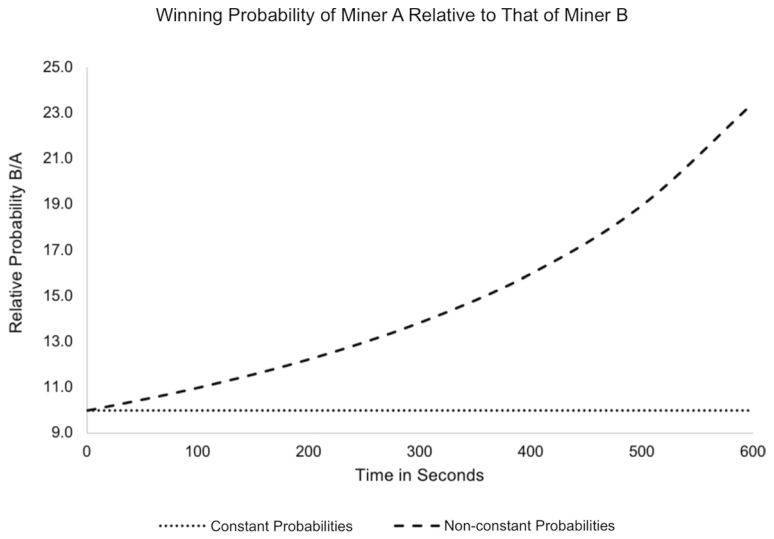

Fig. 2 Illustration of the relationship between the winning probabilities of each miner under different assumptions about the distributions. Own illustration

winning probabilities faster than do smaller miners. This feature of the mining process provides an economic incentive for miners to pool together in order to benefit from this *within-block learning*. Such pooling would be something entirely different to the innocuous pooling observed thus far, in which miners pool together simply to smooth their income over time.

Figure 2 illustrates the relation of the relative odds of Miner B to those of Miner A under both scenarios. The distance between the two lines represents the effect of the *within-block learning* feature, which leads to non-proportional winning probabilities in proof-of-work protocols.

4.2 On the Differences Between Distributions

Probability events that follow the Poisson distribution can occur n times in an interval. The average number of events in an interval is designated by λ, which is also called the rate parameter. In the Poisson distribution the probability of observing k events in an interval is given by

$$P(k) = e^{-\lambda}\frac{\lambda^k}{k!}.$$

As described by Rosenfeld (2011), the bitcoin protocol sets the target value (the minimum number of zeroes with which a hash will be considered valid) such that every hash has a probability $\frac{2^{16}-1}{D2^{48}}$ of yielding a valid block, beeing D the Difficulty level in the system, which is autamoatically derived from the network metrics and can hence be autonomously calculated by each miner. For the sake of simplicity,

we– like other scholars– approximate this value by $\frac{1}{D2^{32}}$. This implies that the Poisson parameter that determines the probability of winning of miner i over time t expressed in seconds (the probability of finding a valid block after mining for time t) can be written as

$$\lambda_i = \frac{h_i t}{D2^{32}},$$

$h_i t$ being the hash rate of miner i and D the Difficulty parameter. Therefore, a miner i mining at a rate of h_i hashes per second for time t (time expressed in seconds) has an expected rate of finding a valid hash that is given by $\frac{h_i t}{D2^{32}}$. This is the same as assuming that the probability of finding a valid block follows the negative binomial distribution.

To approximate the negative hypergeometric distribution with the Poisson dsitribution (as a limiting case of the negative binomial distribution) requires the following conditions to be fulfilled: n is large, p is small, and λ is of moderate size. However, since drawing from an urn without replacement isn't a binomial process, this approximation induces an error $E(k) = |P_{\text{NHG}}(k) - P_{\text{Poisson}}(k)|$. This error is the distance between the dotted lines in Fig. 2. If the number of balls is very large, and the number of draws is relatively small compared to the total number of balls, then the probability p doesn't change much with each draw. However, when p is not small, then error increases and becomes significant.

4.3 Conditions for the Memoryless Property to Vanish

As suggested by Devore and Berk (2012), the negative hypergeometric distribution should not be estimated by the negative binomial distribution when the sample size, n, is bigger than 5% of the population size, N.[3] In our case this means that once a miner has explored around 5% of all possible nonces that miner shall start experiencing the effect of non-proportional probabilities. The reason is that in that case, the errpr $E(k) = |P_{\text{NHG}}(k) - P_{\text{Poisson}}(k)|$ is non-negligible.

The nonce has a size of 4 bytes. Additionally, the coinbase has a size of 8 bytes. Given that a byte contains 8 bits, the domain that a miner explore contains 2^{96} possible inputs. This implies approximately $8 * 10^{28}$ possible inputs that the miner can change in order to try different hashes for a given block. We refer to this space as the *restricted domain*, which in the analogy of the urn problem represents the number of balls contained in the urn. Given that the restricted domain contains approximately $8 * 10^{28}$ possible inputs, 5% of that domain lies at $4 * 10^{27}$.

As of September 2023, the global hash rate of Bitcoin was approximately 400 million terahashes per second (i.e., $4 * 10^{20}$ hashes per second).[4] This indicates that

[3] Note that this 5% is an approximate figure indicating the area of the sample size around which this difference starts to matter (rather than a dichotomous threshold that should serve as a hard line).

[4] The global hash rate of Bitcoin can be consulted at any time at sites such as https://www.blockchain.com/explorer/charts/hash-rate.

by that date the effect of the non-proportionality of winning probabilities in Bitcoin had not yet become significant. However, with every increase in the global hash rate[5] this effect increases in significance.

From this result it emerges that the higher the hash rate, the larger the risk of the memoryless property of the Bitcoin protocol vanishing. This is paradoxical, because thus far, it was assumed that increases in the global hash rate increase the security of Bitcoin by reducing the probability of a double-spending attack. However, while this remains true, increases in the global hash rate also bring the point in which Bitcoin will cease the be decentralized, closer. This is a counter-intuitive outcome: increases in the global hash rate increase the security of Bitcoin and bitcoin-like protocols, until they don't.

4.4 On the Impossibility of Correctly Estimating Actual Hashing Power

Due to the nature of the Bitcoin protocol, only the miner that wins each block can be observed (and not the miners that mine it but do not find a valid version of it). We can observe neither the hashing power of the successful miner nor that of the miners that competed to find a valid version of each block but without success. The impossibility of observing the work actually done, is of particular importance for mining pools. Mining pools are groups of miners who share their computational resources in order to parallelize the search for the next valid block of the blockchain, and increase the frequency with which they collectively find blocks. Miners participate in a mining pool to smooth their earnings across time (not to increase their total expected earnings). Mining pools split their bitcoin earnings among their members in a manner that is proportional to the shares of each member of the pool (Can et al., 2022). A share is a hash that while not being below the target is low enough to be considered close enough to the target (it is usually a hash starting with enough zeroes). Since generating shares is related to actually performing hashing operations, it makes sense that mining pools estimate the hashing power of their miners by observing the number of shares that these miners supply to the pool. Based on these shares, different mining pools use different reward schemes to distribute the bitcoins earned among their members. All these reward schemes, however, assume that mining is a memoryless process, thus establishing linear relationships between shares and rewards.

5 Discussion

Many important aspects of proof-of-work protocols are based on the assumption that the probability of finding the next valid block of a proof-of-work blockchain is governed by a Poisson distribution as a limiting case of the negative binomial

[5] Note that the average annual growth rate of the hashpower of Bitcoin between September 2019 and September 2023 lies at 266%. See https://ycharts.com/indicators/bitcoin_network_hash_rate for the reference.

distribution. These important aspects include that no double-spending attack has a positive expected outcome as long as the miner conducting it possesses less than 50% of the total hash rate, that the winning probabilities of miners are proportional to their hash rates, that the winning probabilities of miners are constant throughout the process of mining a particular block, and that miners have no incentive to pool other than to smooth their income across time. Moreover, mining pools use this same assumption as the foundation on which they base the distribution of their earnings among member miners.

By studying the Bitcoin proof-of-work protocol from a probabilistic perspective, we find that in fact, and due to the finiteness of the hash function, the probability of finding the next valid block of a proof-of-work blockchain is governed by a negative hypergeometric distribution. Hence, the probabilities of finding valid blocks of blockchains using proof-of-work protocols are non-proportional and non-constant throughout time. This is the key insight from our study, and answers the question that motivated it.

In theory, this result reveals that miners generating more hashes per second increase their winning probabilities faster than those (smaller) miners that generate fewer hashes per second. Consequently, it emerges that at a theoretical level not even Bitcoin fulfills the three axioms derived by Leshno and Strack (2020). This implies that the Bitcoin protocol induces centralization, and that the hashing power required for a double-spending attack to succeed is smaller than expected. Hence, the Bitcoin system, and any system based on a proof-of-work protocol whose block size is limited, is more fragile than is currently believed. Therefore, the aspects of proof-of-work protocols that assume a negative binomial distribution should be revised.

In practice, at the current hash rate, and given the huge domain of the hash functions used by proof-of-work protocols, approximating the negative hypergeometric distribution by the negative binomial distribution yields no significant error. Yet the fact that the incentive structure associated with proof-of-work protocols is flawed, and that at a certain hash rate systems based on proof-of-work protocols shall collapse, remains.

Given the result of our paper, what becomes particularly challenging and should be further and thoroughly discussed by scholars and practitioners is the difficulty (impossibility?) of actually calculating the hashing power of miners. If we cannot trust the proportional relationship between miners' hashing power and their probability of finding the next valid block of the blockchain, we cannot correctly calculate the hashing power of the network or of the miners, and thus are blind with regard to the point at which the memoryless property of the Bitcoin protocol and Bitcoin-like protocols will vanish.

5.1 Implications

Our results imply that some aspects of the assumptions made with regard to the properties of bitcoin mining (Nakamoto, 2008; Rosenfeld, 2011; Cocco & Marchesi, 2016) should be revised. Similarly, the assumptions made when modelling the incentive structure for miners to participate in the mining process should be refined

(Eyal and Sirer, 2018; Houy, 2016; Dimitri, 2017; and Wang et al., 2019). Likewise the security aspects of the Bitcoin protocol that rely on its memoryless property—including those described by Rosenfeld (2014), Solat and Potop-Butucaru (2016), Beccuti and Jaag (2017), Aggarwal et al. (2018), Miller and La Viola (2014), Göbel et al. (2015), Lewenberg et al. (2015), and Decker and Wattenhofer (2013)—need to be reconsidered in order to incorporate the insights derived in this paper. Our results also imply that adjustments are required in those models that study the microeconomics of cryptocurrencies (Halaburda et al., 2022), mining pool centralization (Easley et al., 2019), bitcoin valuation Hayes (2019), the evolution of transaction fees (Cong et al., 2019), the optimal design of cryptocurrencies (Chiu & Koeppl, 2017), CBDCs (Benigno et al., 2019), and user adoption and use of virtual currencies (Athey et al., 2015).

In light of the results presented in this paper, it seems imperative to develop more advanced protocols whose incentive structures keep the behavior of miners constant at different hash rate levels. Proof-of-stake protocols may be one solution, although further economic analysis is needed.

Additionally, future research should continue studying the long-term viability of blockchain protocols at increased hash rates. Anticipating changes in miners' behavior is important since blockchain technology (ultimately governed by protocols) is playing an increasing role in the operations of organizations of all kinds.

Furthermore, future research should also extend the study made in this paper to blockchains other than the Bitcoin blockchain, in order to discover if, at their current respective hash rates, miners are already experiencing *within-block learning*.

Moreover, the results presented in this paper imply that the development of quantum computing may alter the viability of blockchain protocols in ways thus far not considered. To understand this statement, it is important to understand that the straightforward threat that quantum computing poses to blockchain protocols comes from the fact that quantum computers could easily make hashing functions invertible. Experts in quantum computing have proposed cryptographic methods of developing hashing functions that would remain non-invertible even in the presence of quantum computing (Li et al., 2019). However, what this type of advance in cryptography does not take into consideration is the fact that quantum computers would enable miners to generate hashes much faster than they do with currently available computers. Using a quantum computer, a miner could conduct "a quadratic speed-up in the number of operations compared to a classical computer, which should lead to an increased hash rate" (Stewart et al., 2018). This increase in hashing capacity, together with the finiteness of the blocks, would make the non-proportionality of wining probabilities very clear at an applied level. Therefore, even if advances in cryptography were to keep hashing functions non-invertible in the presence of quantum computing, these technical advances would not be able to address the fact that the probability of winning is non-proportional. This reveals more exactly how quantum computing threatens this type of protocol.

Finally, and most importantly, scholars and practitioners alike should study alternative reward schemes for mining pools—schemes that offset the advantage gained by larger pools once the memoryless property of the Bitcoin protocol vanishes. In fact, having alternative reward schemes that offset the potential advantage of larger

pools could increase the decentralization of cryptocurrencies like Bitcoin and play a pivotal role keeping the system secure.

5.2 Limitations

Naturally there are a number of limitations to our study that need to be considered. First, we study the proof-of-work protocol used by Bitcoin. While all proof-of-work protocols work in a similar manner, they use different hash functions with different domain sizes. Proof-of-work protocols using other hashing functions are subject to different domains; for these protocols, the hash rates at which the negative binomial distribution can be approximated by the binomial distribution are different than for the Bitcoin protocol. Second, we are assuming that miners fix some parts of a potentially valid block and then explore a relatively small area of that block's header to find a valid hash. It could, however, be that miners also change the timestamp when they exhaust the nonce (and not only the coinbase). For miners that do this, the memoryless property will remain for much longer than it will for those who only change the coinbase.

6 Conclusion

In this paper we have proven that given the finiteness of the SHA-256 function the probability of finding the next block in the Bitcoin blockchain is governed by a negative hypergeometric distribution and not by a negative binomial distribution. This implies that winning probabilities in proof-of-work protocols are non-proportional and non-constant within a block, which is equivalent to stating that there is *within-block learning* in the mining process of proof-of-work protocols and that the act of mining currencies in such protocols is not a "memoryless process." While at the current hash rate approximating the negative hypergeometric distribution by the negative binomial distribution yields no significant error, our results reveal a misconception present in the understanding of proof-of-work protocols.

The major implication of our findings is that, given a certain total hash rate, miners of proof-of-work protocols will have an incentive to concentrate as much hashing power as they can to benefit from non-proportional winning probabilities. This would result in the proof-of-work system collapsing. This gives rise to a contradiction, since having a higher total hash rate increases the security of a proof-of-work–based system by making it harder for a miner to conduct double-spending attacks, but at the same time brings the point of collapse closer. The implementation of quantum computing would bring that point of collapse dramatically closer. To avoid such a collapse, proof-of-work protocols need to be revised.

Based on the results presented here and the implications of this paper, we would like, finally, to reinforce the notion that blockchain protocols are economically, and not cryptographically, secured (Ciaian et al., 2021). Hence, an economic analysis of the incentives behind any protocol is crucial to assessing the viability of blockchains and the applications based on them. Therefore, we argue, for blockchain technology

to be fully understood the economics behind any blockchain protocol should be studied with care.

Funding The authors have not disclosed any funding.

Declarations

Conflict of interest The authors have not disclosed any competing interests.

References

Aggarwal, D., Brennen, G., Lee, T., Santha, M., & Tomamichel, M. (2018). Quantum attacks on bitcoin, and how to protect against them. *Ledger, 3*(0).

Antonopoulos, A. M. (2014). *Mastering bitcoin: Unlocking digital crypto-currencies* (1st ed.). O'Reilly Media Inc.

Athey, S., Parashkevov, I., Sarukkai, V., & Xia, J. (2015). *Bitcoin pricing, adoption, and usage: Theory and evidence*. Stanford University Graduate School of Business. (Working Paper No. 3469).

Beccuti, J., & Jaag, C. (2017). The bitcoin mining game: On the optimality of honesty in proof-of-work consensus mechanism (Working Papers No. 0060). Swiss Economics. Retrieved from https://EconPapers.repec.org/RePEc:chc:wpaper:0060

Benigno, P., Schilling, L. M., & Uhlig, H. (2019). *Cryptocurrencies, currency competition, and the impossible trinity*. National Bureau of Economic Research. (Working Paper No. 26214).

Böhme, R., Christin, N., Edelman, B., & Moore, T. (2015). Bitcoin: Economics, technology, and governance. *Journal of Economic Perspectives, 29*(2), 213–38.

Bowden, R. , Keeler, H.P., Krzesinski, A.E., & Taylor, P.G. (2018). Block arrivals in the bitcoin blockchain. *CoRR*.

Can, B., Hougaard, J. L., & Pourpouneh, M. (2022). On reward sharing in blockchain mining pools. *Games and Economic Behavior, 136*, 274–298.

Chiu, J., & Koeppl, T. (2017). *The Economics Of Cryptocurrencies - Bitcoin And Beyond* (Working Paper No. 1389). Economics Department, Queen's University.

Ciaian, P., Kancs, d'Artis, & Rajcaniova, M. (2021). The economic dependency of bitcoin security. *Applied Economics, 53*(49), 5738–5755.

Cocco, L., & Marchesi, M. (2016). Modeling and simulation of the economics of mining in the bitcoin market. *PLOS ONE, 11*(10), 1–31.

Cong, L. , He, Z. , & Li, J. (2019). Decentralized mining in centralized pools (Working Paper No. 25592). NBER.

Cong, L. , Li, Y. , & Wang, N. (2018). Tokenomics: Dynamic adoption and valuation (Working Paper No. 63). Columbia Business School Research Paper.

Courtois, N. T. , Grajek, M. , & Naik, R. (2014). The unreasonable fundamental incertitudes behind bitcoin mining. *CoRR*.

Decker, C., & Wattenhofer, R. (2013). Information Propagation in the Bitcoin Network. In: IEEE P2P 2013 Proceedings, pp. 1–10.

Devore, J. L., & Berk, K. N. (2012). Discrete Random Variables and Probability Distributions. In *Modern mathematical statistics with applications* (pp. 96–157). Springer New York.

Dimitri, N. (2017). Bitcoin mining as a contest. *Ledger, 2*, 31–37.

Easley, D., O'Hara, M., & Basu, S. (2019). From mining to markets: The evolution of bitcoin transaction fees. *Journal of Financial Economics, 134*(1), 91–109.

Eyal, I., & Sirer, E. G. (2018). Majority is not enough: Bitcoin mining is vulnerable. *Communications of the ACM, 61*(7), 95–102.

Göbel, J. , Keeler, H. P. , Krzesinski, A. E. , & Taylor, P. G. (2015). Bitcoin blockchain dynamics: The selfish-mine strategy in the presence of propagation delay. *CoRR*.

Grunspan, C., & Pérez-Marco, R. (2017). Double Spend Races. *International Journal of Theoretical and Applied Finance*.

Halaburda, H., Haeringer, G., Gans, J., & Gandal, N. (2022). The microeconomics of cryptocurrencies. *Journal of Economic Literature, 6*(3), 971–1013.
Hayes, A. S. (2019). Bitcoin price and its marginal cost of production: Support for a fundamental value. *Applied Economics Letters, 26*(7), 554–560.
Houy, N. (2016). The bitcoin mining game. *Ledger, 1*, 53–68.
John, K., O'Hara, M., & Saleh, F. (2022). Bitcoin and beyond. *Annual Review of Financial Economics, 14*, 95–115. First published as a Review in Advance on March 22, 2022.
Leshno, J. D., & Strack, P. (2020). Bitcoin: An axiomatic approach and an impossibility theorem. *American Economic Review: Insights, 2*(3), 269–86.
Lewenberg, Y. , Bachrach, Y. , Sompolinsky, Y. , Zohar, A. , & Rosenschein, J. S. (2015). Bitcoin mining pools: A cooperative game theoretic analysis. In: *Proceedings of the 2015 international conference on autonomous agents and multiagent systems* (pp. 919–927).
Li, C., Xu, Y., Tang, J., & Liu, W. (2019). Quantum blockchain: A decentralized, encrypted and distributed database based on quantum mechanics. *Journal of Quantum Computing, 1*(2), 49–63.
Li, Z. , Reppen, A. M. , & Sircar, R. (2023). A mean field games model for cryptocurrency mining. *Management Science*0(0)Ahead of print.
Miller, A. K., & La Viola, J. (2014). Byzantine consensus from moderately-hard puzzles: A model for bitcoin (Technical Report). University of Central Florida.
Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*
Rosenfeld, M. (2011). Analysis of bitcoin pooled mining reward systems. *CoRR*arXiv: 1112.4980.
Rosenfeld, M. (2014). Analysis of hashrate-based double spending. *CoRR*arxiv:abs/1402.2009.
Sapirshtein, A. , Sompolinsky, Y. , & Zohar, A. (2015). Optimal selfish mining strategies in bitcoin. *CoRR*arxiv:abs/1507.06183.
Schilling, L., Fernández-Villaverde, J., & Uhlig, H. (2020). *Central bank digital currency: When price and bank stability collide*. National Bureau of Economic Research. (Working Paper No. 28237).
Solat, S., & Potop-Butucaru, M. (2016). ZeroBlock: Preventing selfish mining in bitcoin. *CoRR*.
Stewart, I., Ilie, D., Zamyatin, A., Werner, S., Torshizi, M., & Knottenbelt, W. (2018). Committing to quantum resistance: A slow defence for bitcoin against a fast quantum computing attack. *Royal Society Open Science, 5*, 180410.
Wang, W., Hoang, D. T., Hu, P., Xiong, Z., Niyato, D., Wang, P., & Kim, D. I. (2019). A survey on consensus mechanisms and mining strategy management in blockchain networks. *IEEE Access, 7*, 22328–22370.