NBER WORKING PAPER SERIES

MARKET STRUCTURE IN BITCOIN MINING

June Ma
Joshua S. Gans
Rabee Tourky

Working Paper 24242
http://www.nber.org/papers/w24242

NATIONAL BUREAU OF ECONOMIC RESEARCH
1050 Massachusetts Avenue
Cambridge, MA 02138
January 2018

Market Structure in Bitcoin Mining
June Ma, Joshua S. Gans, and Rabee Tourky
NBER Working Paper No. 24242
January 2018
JEL No. E42,L1

ABSTRACT

We analyze the Bitcoin protocol for electronic peer-to-peer payments and the operations that support the "blockchain" that underpins it. It is shown that that protocol maps formally into a dynamic game that is an extension of standard models of R&D racing. The model provides a technical foundation for any economic analysis of 'proof of work' protocols. Using the model, we demonstrate that free entry is solely responsible for determining resource usage by the system for a given reward to mining. The endogenous level of computational difficulty built into the Bitcoin protocol does not mitigate this usage and serves only to determine the time taken to process transactions. Regulating market structure will mitigate resource use highlighting the importance of identifying the benefits of competition for the operation of the blockchain.

June Ma
Research School of Economics
Australian National University
Canberra, ACT, 0200,
Australia
june.ma@anu.edu.au

Rabee Tourky
Research School of Economics
Australian National University
Canberra, ACT, 0200,
Australia
rabeetourky@gmail.com

Joshua S. Gans
Rotman School of Management
University of Toronto
105 St. George Street
Toronto ON M5S 3E6
CANADA
and NBER
joshua.gans@gmail.com

1 Introduction

At the core of Bitcoin (the first peer-to-peer electronic payments network) is the "blockchain", a decentralized public ledger on which transactions are recorded (Nakamoto, 2008). Decentralization prevents control of the blockchain by an individual agent making manipulation of the ledger virtually impossible (à la Kocherlakota (1998)). This relies on ensuring that there are sufficient independent nodes participating that incur the costs of storing and verifying the blockchain.

The incentive for decentralized nodes to do these tasks arises from their participation in what is essentially a "mining" game to add new transactions (grouped into blocks) to the blockchain. This competition between users on the network known as "miners" involves solving a computational puzzle and is embedded into the Bitcoin protocol. Transactions are recorded on the blockchain each time a puzzle is solved, with the solver being rewarded with newly issued bitcoins and any transaction fees offered. Finding the puzzle's solution does not require strategy but brute force in the form of guessing. The more guesses are made, the more likely a miner will be the first to solve the puzzle. In other words, it is computational strength – increasing the speed of guessing – that drives who is likely to win the game. However, the process is not deterministic so there is no guarantee that the miner in the network with the most computational power will solve the puzzle first.

Having participants play this game resolves two important issues. First, it ensures that only one suggested block of transactions will be sent to the network for verification. Second, it reduces the probability that a bad actor can suggest a block of transactions that, say, takes back currency that they have already spent. The only way one might conceivably distort the blockchain is to control the game, but to do that requires proof-of-work – solving the puzzle – which is costly. The expenditure of real resources is, therefore, key to the integrity of the network.

A feature of the Bitcoin protocol is to allow anyone to expend those resources and, indeed, encourage it. Each time the game is run, it is essentially an all-pay tournament to win. Not surprisingly, concern has developed regarding the consumption of real resources. Computational effort in mining relies upon sophisticated processors and the energy to power them. As the value of bitcoins has risen, the reward from the mining game has become larger, with consequent entry increasing energy usage. As of January 2018, it is estimated that the Bitcoin network uses 40.64 TWh of energy to operate annually, slightly more than Hungary, the world's 57th largest consumer

of energy.[1] This is 75 times higher than the annual energy consumption of the centralized Visa network, which processed an average 150 million transactions per day in 2016, compared to the 44 million transactions processed on the Bitcoin network for the whole of 2017.[2] This has caught the attention of policy-makers with China, which controls 75 per cent of total mining power, moving in 2018 to tax bitcoin miners' energy usage.[3] Consequently, it is important to understand what drives both competition and the flow of resources into the Bitcoin network.

This paper provides the first model of Bitcoin mining with a view to understanding how it can lead to intense resource usage. Using the stochastic process described by the cryptographic function, we derive the full game being played by miners showing that it is essentially an extension of the R&D racing game analyzed by Loury (1979). The extensions reflect the occurrence of multiple events before a reward is attained in Bitcoin mining and the endogenous determination of computational difficulty in the Bitcoin protocol. Our aim is to investigate the elements of the protocol that have intensified resource usage and identify ways by which a decentralized proof-of-work system can operate without incurring such costs.

Our contribution is, primarily, technical. We show how the Bitcoin protocol maps into formal game played between miners, that an equilibrium exists and that free entry determines key outcomes. Importantly, we show that the difficulty of the computational puzzle is not an instrument that can regulate resource usage and that, instead, direct targeting of a desired number of competing nodes would allow the costs of the system to be aligned with potential benefits. We believe that this provides an important foundation for both further theoretical and empirical work on blockchains and cryptocurrency networks.

To date, there exist a few formal economic models to understand the operation of digital cryptocurrency systems. Gans and Halaburda (2015) examine how digital currencies can support platform operations while Athey et.al. (2016) examine the use of digital currencies to improve the efficiency of payments. Huberman et.al. (2017) provide an analysis of Bitcoin focusing on the protocols surrounding the determination of transaction fees that

[1]Digiconomist. "Bitcoin Energy Consumption Index." 2018. Accessed January 13. `https://digiconomist.net/bitcoin-energy-consumption`.

[2] Visa."Visa acceptance for retailers." 2018. Accessed January 13. `https://usa.visa.com/run-your-business/small-business-tools/retail.html`; Blockchain. "Confirmed Transactions Per Day." 2018. Accessed January 13 `https://blockchain.info/charts/n-transactions`.

[3]Wildau, Gabriel. 2018. "China moves to shutter bitcoin mines." *Financial Times*, January 9, 2018.

are part of the rewards miners receive. They show that the Bitcoin protocol provides an equilibrium outcomes that mimics a Vickery-Groves-Clark mechanism for the determination of transaction fees when there is competition amongst miners. While they rely on miner free entry, they do not model the impact of miners on the "proof of work" protocol at the heart of the Bitcoin system. Our paper is focused upon miner competition as mediated by the Bitcoin protocol while keeping the reward (from both newly minted coins and transaction fees) as exogenous.

In the following section, we provide an overview of Bitcoin mining and how its market structure has evolved in practice. We describe the model in section 3, and show that the mining game has a unique and stable equilibrium in section 4 when the number of miners is fixed. The proof is technically challenging because the difficulty of the computational puzzle changes with the choices of miners. In section 5, we allow for free entry and examine the long-run outcomes of the game. We discuss the implications of free entry on social welfare in section 6, and show that it requires more intensive resource usage than a regulated monopoly outcome.

2 An Overview of Bitcoin Mining

The puzzle that Bitcoin miners work on is a cryptographic proof-of-work function which requires significant effort in computational time and power to solve. The proof-of-work puzzle is based on the *Hashcash proof-of-work* proposed in Back (2002), the details of which we return to below. Solving the puzzle requires a set number of computations to be completed. This threshold is determined by the difficulty level of the puzzle, which is dynamically set by the network. The level of difficulty is adjusted every 2016 blocks according to an algorithm, such that a new block of transactions is added to the network every 10 minutes on average. If the mean time for the addition of previous 2016 blocks falls below this, the difficulty level is increased for the next 2016 blocks. In practice, this means that difficulty is adjusted approximately every two weeks.

Each miner chooses a computing technology (notably computer hardware) which they use in their attempt to solve the puzzle. The greater their computational power, the greater the number of calculations they can compute within a given time interval. Therefore, the probability of being the first to solve the puzzle increases with the amount of computing technology a miner possesses. As the proof-of-work function is a random process, solving the puzzle involves brute force trial and error computations. As a result, there is no guarantee that the miner in the network with the most computational

power will be the first to solve the puzzle. Once a miner proposes that they have solved the puzzle, it only requires one calculation to verify if they are correct. The work is expensive while proving that it has been done is cheap.

There is no inherent value is solving the computational puzzle itself. The miners' incentives to engage in the proof-of-work exercise and process transactions is driven by an allocation of newly issued bitcoins along with any transaction fees that might be submitted by users. Bitcoins are issued for each block at a diminishing rate over time. The reward of newly minted bitcoins prescribed by the network halves for every 210,000 blocks until the total supply of 21 million bitcoins has been exhausted. Thus, mining has an incidental role in issuing new currency in the absence of a centralized authority.

Transaction fees provide a similar incentive and are intended to become more important as the rate of bitcoin issuance falls. Such fees are offered by users when they want to impact upon the speed at which their transaction is processed. Miners will observe transaction fees when determining the set of transactions that will comprise a block they process for the network. Their incentive is to prioritize the highest value transactions according to fees and it is this that determines users' price for priority. Transaction fees have become more common in recent times as demand for transactions on the network has rapidly increased, causing delays in the processing of transactions. These delays are a result of the 1 megabyte limit on block size. When demand for transactions on the network increases above the 1 megabyte block size, the time in between the proposal of a transaction by a user and its assembly into a block lengthens. Thus, these delays occur independently of the Bitcoin mining network, since the protocol mandates that a transaction should take an average of 10 minutes to be processed once it has been assembled into a block through its dynamic adjustments to difficulty. Accordingly, there is no incentive for miners to deliberately slow mining activity to create delay and gain additional transaction fees in the long run due to the dynamic adjustment of the puzzle's difficulty.[4]

Why does the Bitcoin protocol target a 10 minute block processing time? The rationale for having some period of time between when a miner wins the game and the processing of the next block is to ensure that there is time to communicate the winning block to the network and network participants to support adding it to the blockchain. The 10 minutes appeared to be arbitrarily listed as an example in Nakamoto (2008), which had the virtue of being far quicker than the day or more it takes for ordinary bank-to-

[4]Huberman et al. (2017) presents a detailed examination of the determination of transaction fees.

bank transfers. Other cryptocurrencies such as Ethereum now have block processing times of around 20 seconds.

In summary, the net payoff to the winning miner is the total reward (newly minted coins plus transaction fees) less the cost of their computing technology. Miners who are unsuccessful at solving a computational puzzle incur the cost of their computing technology. Once a computational puzzle has been solved and the associated block of transactions recorded on the network, the miners move on to compete on processing the next block of transactions.

2.1 Hashcash Cryptographic Proof-of-Work

We now turn to focus on the puzzle at the heart of Bitcoin, which is based on the Hashcash cryptographic proof-of-work proposed by Back (2002). Hashcash was initially proposed as a solution to prevent bulk email spam. To send an email, the sender must first complete a proof-of-work to show the email is legitimate as they have exerted some effort. The computing power and cost required to complete the proof-of-work is negligible for a single email. However, these costs become a deterrent when multiple proofs are required to be completed in order to send bulk email spam. Bitcoin mining uses a variation of the Hashcash proof-of-work to verify transactions and to secure the blockchain.

The Hashcash proof-of-work is a cost function that describes the amount of effort required to solve a puzzle. In Bitcoin mining, it gives the number of computations a miner must complete before they find the solution of the puzzle. The function takes an input of some arbitrary length and maps it into an output of a fixed length, which is referred to as a "hash". The cost function has the following properties:

1. The solution is difficult and costly to compute, but easily verifiable once found.

2. The output of the function is random.

3. It is a one-to-one function.

4. It is practically impossible to invert.

Since the function is difficult to invert, it is often referred to as a one way function. We now describe the proof-of-work cost function.

Let $h : S \to S$ be a cryptographic cost function, where S represents a sequence of alphanumeric characters. Given an output h, the proof-of-work

puzzle requires an input m to be found, such that $h(m) = h$. It becomes evident from this why the cost function should not be invertible, since the solution m to the proof-of-work would be straightforward to find it if were.

Each block of the Bitcoin blockchain contains the "challenge" string of the puzzle, which includes the output h of the previous block. This feature links all previously verified blocks in historical order. The miners in the network compete to be the first to find the "proof" string, which when concatenated with the challenge string, gives the required input m of the cost function such that $h(m) = h$.

Bitcoin uses the SHA-256 function, which gives an output that is 256 bits in length. The output h is a string of numbers that is preceded by K zeros. The string of zeros represents the difficulty of the proof-of-work, and represents the computational cost required to solve the puzzle. Since the output of the cost function is random, the only way to find the proof string to m is through brute force trial and error computations.

The cost function returns a random number between 0 and the 256 bit number for each computation. Once a miner achieves an output preceded by the required K zeros, they have solved the proof. This is simple to verify by others on the network once they have the proposed input m of the cost function. As K increases, the computational cost required to solve the puzzle increases exponentially (Nakamoto, 2008). Specifically, to find an output preceded by K zeros requires 2^K computations on average.

Once a solution is proposed to the network, it requires at least 50 per cent of the network to agree that the proposed solution is correct. The longest chain in the blockchain represents the majority decision since it represents the greatest proof-of-work effort. As long as the majority of computing power is controlled by honest users, the honest chain forms the blockchain.

The cost function between blocks on the blockchain are independent. Thus, solving one puzzle does not affect one's ability to solve subsequent puzzles. The proof-of-work function prevents double spending by making it difficult and impractical for transactions to be reversed once they are recorded on the network. It is designed such that the incentive to verify transactions on the network is greater than that to attack the network since it is costly to redo a proof-of-work and difficult to then keep up with the rest of the honest network.

2.2 Mining Pools

Nakamoto (2008) appeared to envisage a purely peer-to-peer system in which mining could be performed on off-the-shelf computers. However, as Bitcoin gained popularity and its value appreciated, miners began to invest in more

powerful computers to increase their computational power, and hence their probability of winning the Bitcoin mining race. Although miners could perform a higher number of computations, the average time taken to complete each mining race did not change due to the dynamic adjustment of difficulty by the protocol in response to changes in technology. As the difficulty of these puzzles increased, mining became infeasible and unprofitable for individual miners on regular computers, whose probability of winning effectively fell to zero. Bitcoin mining is now essentially an arms race, with the majority of mining activity taking place in large purpose built warehouses using dedicated mining equipment. This gave rise to the formation of mining pools, in which individual miners pool their processing power to increase their probability of winning. The method of sharing mining revenue amongst a mining pool differs between pools, but rewards are generally distributed proportionately to the effort contributed by each miner. Some pools charge fees to miners in order to participate in the pool.

The hashing power controlled by each mining pool is highly volatile as it depends on the number of active nodes in the system at a given point in time. However, the largest mining pools by active hashing power have been fairly consistent in recent times. The largest pools are based in China and include AntPool, BTC.com, BTC.top, and ViaBTC.[5]

In 2014, the Ghash.io mining pool approached 51 per cent of the overall hash rate in the Bitcoin network, raising concerns amongst users. This is because a miner or mining pool who holds 51 per cent or more of computing power in the network can potentially double spend if they are able to solve consecutive blocks and prevent transfers between other users, effectively "attacking" the network. However, the nature of the proof-of-work as a random process mitigates this concern to a certain extent, as controlling the majority of hashing power does not guarantee winning.

3 The Model

We now take this underlying computational process and use it to derive, formally, the Bitcoin mining game. Consider a network with N identical miners, indexed $i \in \{1, 2, \ldots, N\}$, competing to be the first to solve a computational puzzle. Each miner i chooses a computing technology $x_i \in \mathbb{R}_+$ at a cost of $c(x_i)$ at time $t = 0$ to compete in the race. The network determines a difficulty level $K \in \mathbb{R}_{++}$, representing the threshold number of computations required to solve the puzzle, and sets the target puzzle solution time as

[5]Blockchain. "Hashrate Distribution." 2018. Accessed January 13. `https://blockchain.info/pools`.

$\delta^* \in \mathbb{R}_+$. The difficulty of the puzzle adjusts dynamically to ensure that δ^* is realized on average. All miners face the same cost function $c : \mathbb{R}_+ \to \mathbb{R}_+$, where $c(x_i)$ is an analytic strictly increasing convex function. A miner i chooses some technology $x_i > 0$ only if they can meet the minimum effort cost required to complete K computations, otherwise they do not compete and choose $x_i = 0$.

The technology for solving computational puzzles is formally equivalent to a stochastic Poisson process

$$X_i(t) \sim \text{Poisson}(x_i)\,,$$

where x_i gives the expected number of computations miner i will complete in a time interval t given their choice of technology. We assume that the Poisson processes over miners are independent and operate in parallel.[6]

The Poisson process $X_i(t)$ gives a random time $t_i \in \mathbb{R}_+$ at which K computations are completed by miner i that is independent between the miners. The distribution of the random variable t_i is

$$t_i \sim \text{Gamma}(K, x_i)\,.$$

We denote the probability density function of this random variable by

$$\gamma_{K,x_i}(t_i) = \frac{t_i^{K-1}}{\Gamma(K)} x_i^K e^{-x_i t_i}\,,$$

where $\Gamma(\cdot)$ is the Gamma function. The probability that miner i completes K computations within time t_i increases with their technology x_i.

The network determines a reward of newly minted bitcoins $B \in \mathbb{R}_+$ won by the first miner to complete the threshold of K computations. Additionally, the winning miner receives the aggregate of the transaction fees $f \in \mathbb{R}_+$ offered with the transactions in the associated block.[7] If $t = (t_1, t_2, \ldots, t_N)$ is a realization of the success times of miners and $t_i < t_j$ for all $j \neq i$, then miner i receives a payoff

$$P - c(x_i)\,,$$

[6]If miners are organized into pools, they could potentially coordinate their computations and so there would be an economy between them. Nonetheless, so long as mining pools are of equal size, the independence assumption would still hold.

[7]Each user determines their transaction fee in a first price sealed bid auction. As miners are profit maximizing, they assemble transactions into blocks to maximize f, which is the sum of the highest transaction fees offered at that fit within maximum block size, and is also an equilibrium in the first price sealed bid auction amongst users. This fee f is fixed in bitcoin, but varies in dollar terms due to exchange rate fluctuations when the network is in disequilibrium. See Huberman et al. (2017) for more details.

where $P = B + f$.

All other miners $j \neq i$ receive a payoff of $-c(x_j)$. Let $W_i \subseteq \mathbb{R}_+^N$ be the set of time realizations for which miner i is the first to solve the puzzle

$$W_i = \{t \in \mathbb{R}_+^N : t_i < t_j, \text{ for all } j \neq i\}.$$

The probability of miner i winning by realizing a time profile $t \in W_i$ given the strategy profile (x_i, x_{-i}) is given by the cumulative density function of the minimum order statistic of the gamma random variable t_i, across all N miners

$$\pi(W_i; K, x_i, x_{-i}) := \mathbb{P}(t \in W_i) = \prod_{-i} \left[1 - \int_0^{t_i} \gamma_{K,x_{-i}}(t_i)\ dt_i\right].$$

We note $\pi(W_i; K, x_i, x_{-i})$ is an infinitely differentiable function.

The expected payoff of a strategy profile $x = (x_1, x_2, \ldots, x_N)$ for miner i, is therefore

$$U_i(x_i) = P\pi(W_i; K, x_i, x_{-i}) - c(x_i) = \mathbb{E}(P) - c(x_i). \tag{1}$$

Each miner i chooses the technology x_i that maximizes their expected payoff.

Definition 3.1. A *Nash equilibrium* is an action profile $x^* = (x_1^*, x_2^*, \ldots, x_N^*)$ representing the technology for each miner $i = 1, 2, \ldots, N$ given a fixed difficulty level K of the computational puzzle if

1. $U_i(x_i^*, x_{-i}^*) \geq U_i(x_i, x_{-i}^*)$ for all miners $i = 1, 2, \ldots, N$, where $x_i^* \neq x_i$, and

2. The expected time required to solve the computational puzzle given (K, x^*) is $\delta_K \in \mathbb{R}_+$, where

$$\delta_K := \mathbb{E}_K(t) = \mathbb{E}_K(\min\{t_1, t_2, \ldots, t_N\}). \tag{2}$$

It is worth emphasizing that this definition has an implicit assumption that each miner has no impact on K. In effect, it is assumed that N is sufficiently large that no such impact arises. Below we discuss the implications of relaxing this assumption.

Definition 3.2. A *symmetric equilibrium* is a pair $(K^*, x^*) \in \mathbb{R}_+^2, K \geq 1$ such that

1. (K^*, x^*) is a *Nash equilibrium* for all i by Definition 3.1, and

2. The expected time of completion is δ^*, where δ^* is the target solution time set by the network.

4 Results

We now analyze the game described in the previous section.

4.1 Stage game

First, we show that an equilibrium of the stage game exists. This we do in steps with two important lemmas.

Lemma 4.1. *Holding the choices of other miners fixed at x_{-i}, the probability of miner i winning is increasing in x_i.*

Proof. The expected time for miner i to successfully complete the puzzle is $\mathbb{E}(t_i) = \frac{K}{x_i}$. The first derivative of the probability of winning with respect to x_i is

$$\begin{aligned}\frac{\partial \pi(W_i; K, x_i, x_{-i})}{\partial x_i} &= \frac{\partial \pi(W_i; K, x_i, x_{-i})}{\partial \mathbb{E}(t_i)} \frac{\partial \mathbb{E}(t_i)}{\partial x_i} \\ &= \frac{\partial \pi(W_i; K, x_i, x_{-i})}{\partial \mathbb{E}(t_i)} \left(\frac{-K}{x_i^2}\right) \\ &> 0\,,\end{aligned}$$

since we know the first term is negative from (11). □

However, if $N > 1$, there is no choice of technology for which winning is guaranteed, since the proof-of-work is a random process

$$\pi(W_i; K, x_i, x_{-i}) < 1 \quad \text{for all } x_i\,.$$

Although greater computing technology increases the rate at which computations can be completed and the probability that a miner will win, it does not guarantee that the miner with the most technology will be first to reach K computations.

Lemma 4.2. *Holding the choices of other miners fixed at x_{-i}, the probability of miner i winning increases in x_i at a decreasing rate.*

Proof. The second derivative of $\pi(W_i; K, x_i, x_{-i})$ with respect to x_i is

$$\begin{aligned}\frac{\partial^2 \pi}{\partial x_i^2} &= \frac{\partial^2 \pi}{\partial \mathbb{E}(t_i)^2} \left(\frac{\partial \mathbb{E}(t_i)}{\partial x_i}\right)^2 + \frac{\partial \pi}{\partial \mathbb{E}(t_i)} \frac{\partial^2 \mathbb{E}(t_i)}{\partial x_i^2} \\ &= \frac{\partial^2 \pi}{\partial \mathbb{E}(t_i)^2} \left(\frac{-K}{x_i^2}\right)^2 + \frac{\partial \pi}{\partial \mathbb{E}(t_i)} \left(\frac{2K}{x_i^3}\right).\end{aligned}$$

The second term is always negative by (11). By (12), we know that $\frac{\partial^2 \pi}{\partial \mathbb{E}(t_i)^2}$ is negative on $\hat{x}_i < \frac{K}{\mathbb{E}(\hat{t}_i)}$. Therefore, the sign of the second derivative is negative on $x_i < \hat{x}_i$, noting that $\hat{x}_i \geq 0$.

For $x_i > \hat{x}_i$, the first term is always positive. Since $\pi(W_i; K, x_i, x_{-i}) < 1$ for all x_i, it must be the case that

$$\frac{\partial^2 \pi}{\partial \mathbb{E}(t_i)^2} \left(\frac{-K}{x_i^2} \right)^2 < \frac{\partial \pi}{\partial \mathbb{E}(t_i)} \left(\frac{2K}{x_i^3} \right),$$

for $x_i > \hat{x}_i$. If not, then this implies that $\pi(W_i; K, x_i, x_{-i})$ is increasing and convex in the region. This contradicts the fact that probability can never exceed 1. Therefore, it must be the case that $\pi(W_i; K, x_i, x_{-i})$ is strictly concave on x_i

$$\frac{\partial^2 \pi(W_i; K, x_i, x_{-i})}{\partial x_i^2} < 0\,.$$

□

Using these lemmas we can now prove the following:

Proposition 4.1. *There exists a unique interior Nash equilibrium solution $x^* > 0$ to (1) if $U_i(x) \geq 0$ for some $x > 0$.*

Proof. The necessary conditions for x^* to be a unique interior Nash equilibrium solution to (1) are

$$\frac{\partial U_i(x^*)}{\partial x_i} = P \frac{\partial \pi(W_i; K, x^*)}{\partial x_i} - \frac{\partial c(x^*)}{\partial x_i} = 0 \tag{3}$$

$$\frac{\partial^2 U_i(x^*)}{\partial {x_i}^2} = P \frac{\partial^2 \pi(W_i; K, x^*)}{\partial {x_i}^2} - \frac{\partial^2 c(x^*)}{\partial {x_i}^2} < 0\,. \tag{4}$$

The first term of (3) is positive by Lemma 4.1 and is concave in x_i by Lemma 4.2. The second term is positive and convex in x_i by assumption. Therefore, there must exist some point x^* for which the first order condition in (3) holds.

To show that x^* is a unique solution, we note that the first term of (4) is negative by Lemma 4.2, and the second term is positive by assumption for all x_i. Therefore, the second order condition in (4) is satisfied. Since $U_i(x_i)$ is strictly concave on x_i, the interior Nash equilibrium solution x^* is a unique maximum. □

We have shown that there exists a unique interior Nash equilibrium solution $x^* = (x_1^*, x_2^*, \ldots, x_N^*)$ for every difficulty level K, where the number of miners in the network is fixed. Each Nash equilibrium (K, x^*) gives a unique expected time for completion of the puzzle δ_K. Therefore, at each stage of the Bitcoin mining game, there exists a unique Nash equilibrium.

4.2 Dynamic game

To achieve the target time of δ^* on average, the network periodically adjusts the difficulty level K of the computational puzzle in each stage of the extensive game. If the average time at which the computational puzzles have been solved in the previous period is below the target δ^*, then the difficulty level K increases, and vice versa.

Proposition 4.2. *Ceteris paribus, the expected time required to solve a puzzle δ_K is strictly monotonically increasing in the difficulty level K.*

Proof. The expected time for the puzzle to be solved by the network for the same K is

$$\mathbb{E}_K(t) = \mathbb{E}_K(\min\{t_1, t_2, \ldots, t_N\}) = \int_0^\infty \left[1 - \int_0^t \gamma_{K,x^*}(t)\ dt\right]^N\ dt\,.$$

The cumulative density function of the Gamma distribution decreases in its shape parameter, which is given by the difficulty level K. Thus, the term in the square brackets becomes larger as K increases, such that the expected time for the puzzle to be solved by the network increases. □

As miners need to complete more computations before finding a solution for the puzzle, the expected time taken for the network to solve the puzzle increases. This is intuitive since the expected time for a given miner i to solve the puzzle is $\mathbb{E}_K(t_i) = \frac{K}{x_i}$.

Proposition 4.3. *The Nash equilibrium technology x^* is increasing in the difficulty level K.*

Proof. Suppose we are initially at some Nash equilibrium (K, x^*) which has an expected solution time of δ_K. Each miner i has an equal probability of winning since they all choose the same technology x^* at the Nash equilibrium

$$\pi(W_i; K, x^*) = \left[1 - \int_0^{t_i} \gamma_{K,x^*(t_i)}\ dt_i\right]^{N-1} = \frac{1}{N}\,. \tag{5}$$

Now suppose we increase the difficulty level from K to $K + \varepsilon$, where $\varepsilon > 0$. In the short run, miners cannot re-optimize immediately, so their technology remains as x^*. The miners still have an equal probability of winning at the new difficulty level $K + \varepsilon$. Therefore, the integrand of (5) must be the same for $\gamma_{K,x^*}(t)$ and $\gamma_{K+\varepsilon,x^*}(t)$.

Solving for the Nash equilibrium technology x^*,

$$\begin{aligned}
\gamma_{K,x^*}(t) &= \gamma_{K+\varepsilon,x^*}(t) \\
\frac{t^{K-1}}{\Gamma(K)}(x^*)^K e^{-x^*t} &= \frac{t^{K+\varepsilon-1}}{\Gamma(K+\varepsilon)}(x^*)^{K+\varepsilon}e^{-x^*t} \\
\frac{1}{\Gamma(K)} &= \frac{t^\varepsilon}{\Gamma(K+\varepsilon)}(x^*)^\varepsilon \qquad (6) \\
\therefore x^* &= \left[\frac{1}{t^\varepsilon}\frac{\Gamma(K+\varepsilon)}{\Gamma(K)}\right]^{\frac{1}{\varepsilon}}.
\end{aligned}$$

Since $\Gamma(\cdot)$ is convex on $\mathbb{R}_{++}$, $\frac{\Gamma(K+\varepsilon)}{\Gamma(K)}$ is increasing in K. Furthermore, as $\varepsilon \to 1$, $x^* \to \frac{K+1}{t}$. As the target puzzle solution time is fixed by the protocol at δ^*, given K, the equilibrium technology is given by $x^* = \frac{K}{\delta^*}$, which increases linearly in K. □

With a fixed number of miners, miners compete away part of their profits by increasing their mining power as difficulty adjusts to maintain the target puzzle solution time. This is a competitive externality that, if they could, miners would want to internalize so as to operate with a lower technology (and cost). Individual and aggregate technology increases until the difficulty level which gives the target solution time for the puzzle is reached.[8]

Proposition 4.4. *There exists a unique symmetric equilibrium* (K^*, x^*) *given a fixed number of miners* N *and fixed target solution time* δ^*.

Proof. Fix δ^*. By Proposition 4.2, $\mathbb{E}_K(t)$ is strictly monotonically increasing in K. Hence, the K which is associated with δ^* must be unique. Fixing the $K = K^*$ where K^* gives δ^*, there exists some unique Nash equilibrium level of technology x^* by Proposition 4.1.

Hence, (K^*, x^*) is a unique symmetric equilibrium with an average target solution time of δ^*. □

Allowing K to vary over time in order to maintain a target solution time δ^* on average, we showed that for each N and P, there exists a difficulty level K^* and a Nash equilibrium x^* given K^* that form a unique equilibrium of the dynamic game. Therefore, with a fixed number of miners N, there is exists a symmetric subgame perfect equilibrium in which each miner plays their Nash equilibrium strategy at every stage of the game.

[8] Dimitri (2017) also notes the existence of this competitive externality albeit for a game with a more abstract and simplified structure than the one presented here.

With a fixed prize and a fixed market structure, the gross expected payoff is constant at every Nash equilibrium due to the symmetry of the game. Since the Nash equilibrium technology x^* is increasing in the difficulty level K, the cost of mining increases in K. Miners earn positive profits in equilibrium if given K^*, the equilibrium technology satisfies

$$x^* < c^{-1}\left(\frac{P}{N}\right).$$

This will be true for N low or P high.

5 Free entry

A key feature of the Bitcoin protocol is that anyone can become a miner. Thus, there is free entry and here we examine the long-run outcomes of the game when N is endogenous. In equilibrium, the symmetry of the game means that all miners must choose the same technology x^*. Therefore, the probability of a miner i being the first to solve the puzzle and win the prize is

$$\pi(W_i; K^*, x^*) = \frac{1}{N}.$$

Allowing free entry into the game drives expected profits to zero in equilibrium, with miners indifferent between entering and exiting

$$U_i(x^*) = \frac{P}{N} - c(x^*) = 0,$$

where x^* satisfies (3) for all i. There are zero aggregate profits in equilibrium, since competition drives up the social cost of mining. Social costs are independent of market structure, and solely dependent on the prize.

$$Nc(x^*) = P. \tag{7}$$

The equilibrium technology when there is free entry is characterized by

$$x^* = c^{-1}\left(\frac{P}{N}\right). \tag{8}$$

This gives a useful comparative static result.

Proposition 5.1. *The equilibrium technology decreases with N.*

Proof. Recall that $c'(\cdot) > 0$ by assumption, which implies $(c^{-1})'(\cdot) > 0$. Differentiating (8) with respect to N

$$\frac{\partial x^*}{\partial N} = \frac{\partial c^{-1}\left(\frac{P}{N}\right)}{\partial \frac{P}{N}} \frac{\partial \frac{P}{N}}{\partial N} = \frac{\partial c^{-1}\left(\frac{P}{N}\right)}{\partial \frac{P}{N}} \left(\frac{-P}{N^2}\right) < 0\,,$$

for $N, \frac{P}{N} \neq 0$. □

The incentive to rent technology decreases with increased competition in the network. As more miners enter the race, the probability that a miner i will be the first to solve the puzzle falls, decreasing the expected value of the winning.

Proposition 5.2. *Under free entry, the equilibrium social cost of mining $Nc(x^*)$ equals P.*

Proof. Miners enter if $P > Nc(x)$, and will choose the same optimal technology as the existing miners did in the previous period since they are identical. That is, $x_{i,t+1} = x^*_{i,t}$ for all i. Therefore, $N_{t+1}c(x^*_t) > N_t c(x^*_t)$.

Since aggregate technology has increased in period $t+1$, the average time taken to solve the puzzle falls by Proposition 4.2, such that $\delta_{t+1} < \delta^*$. The average solution time falls in subsequent periods as miners continue entering, until the protocol increases the difficulty of the puzzle K to regulate the network back towards δ^*. By Proposition 4.3, miners respond to the increased level of difficulty by increasing their technology, again raising the aggregate cost of technology. This adjustment continues until the free entry number of miners is reached. □

This result shows that free entry causes rents from mining to dissipate in terms of increased resource costs. Interestingly, it also means that the level computational difficulty does not impact on resource usage. That is, those costs are driven solely by P so that changes in K are solely for the purposes of determining the targeted block processing time. Indeed, if the targeted block time were reduced, K would be lower but resource use per unit of time would increase as the relevant period for the stage game would be compressed. That is, reducing the Bitcoin protocol target from 600 seconds to 300 seconds would double expected resource usage.

Once the network is at some long run equilibrium (K^*, x^*), miners are indifferent between entering and exiting the mining race since there are zero expected profits. Therefore, any movements away from an equilibrium will be due to changes in the prize P. We can now investigate what happens when the reward, P, from mining changes.

Proposition 5.3. *The equilibrium technology, x^*, and the equilibrium difficulty level, K^*, are increasing in P.*

Proof. For x^*, differentiating (8) with respect to P

$$\frac{\partial x^*}{\partial P} = \frac{\partial c^{-1}\left(\frac{P}{N}\right)}{\partial \frac{P}{N}} \frac{\partial \frac{P}{N}}{\partial P} = \frac{\partial c^{-1}\left(\frac{P}{N}\right)}{\partial \frac{P}{N}} \frac{1}{N} > 0$$

since $(c^{-1})'(\cdot) > 0$ for $N, \frac{P}{N} \neq 0$. For K, by Proposition 5.3, the equilibrium technology x^* is increasing in P, and by Lemma 4.3, x^* is increasing in the difficulty level K given a fixed N. Therefore, it must be the case that the unique equilibrium difficulty level K^* is higher for a larger prize P, holding N constant. □

This proposition shows that investment in computational power increases as the expected value of the prize increases, raising the cost of mining. The part of the prize prescribed by the network of newly minted bitcoins halves for every 210,000 blocks that are verified until the supply of all 21 million bitcoins has been exhausted. When the prize next halves, short-run profits will fall below zero and we would expect to see, in the absence of an increase in transaction fees or an appreciation of the exchange rate to compensate for the difference, miners respond by decreasing their computing technology and others exiting the mining race. Since social costs are equal to the prize in equilibrium, the network becomes less resource consuming as block reward falls to zero.

Absent supply changes, appreciation in the bitcoin exchange rate causes the prize to increase. When this happens it is likely that technology cannot increase immediately in the short run and there are positive profits. In the long run, by Proposition 5.3, an increase in the prize, causes the equilibrium technology and cost to increase. Subsequently, the rate at which puzzles are solved increases until the next period when the difficulty level increases to maintain the solution target δ^*. The difficulty level in the network has consistently trended upward since it began. As can be seen in Figure 1, this is highly positively correlated with the bitcoin exchange rate. That figure demonstrates that the mechanism by which this occurs is an increase in the hash rate induced by the stronger economic incentives to mine bitcoin.

6 Welfare analysis

Even though the endogenous computational difficulty changes with economic conditions in the Bitcoin network, the analysis has demonstrated that total

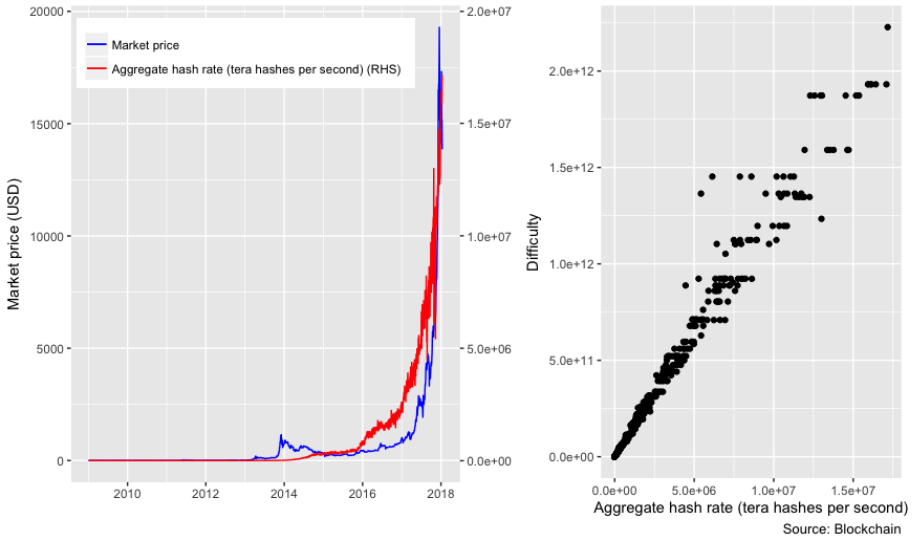

Figure 1: Bitcoin market price and mining activity

resource usage is determined by those economic conditions with computational difficulty only playing a role in setting the processing time for individual blocks. The Bitcoin protocol was designed to be open in that anybody could become a node and/or miner with more competition seen as a virtue in increasing the security and robustness of the network.

The problem is that this very openness also contributes to the overall social cost of the network. As the following proposition demonstrates, an increase in the number of miners raises both the technological and overall costs of the network.

Proposition 6.1. *As the number of miners increases, the aggregate equilibrium technology, Nx^*, and the aggregate equilibrium cost, $Nc(x^*)$, increase.*

Proof. By (8), the aggregate level of technology is given by $Nx^* = Nc^{-1}\left(\frac{P}{N}\right)$.

$$\frac{\partial Nx^*}{\partial N} = c^{-1}\left(\frac{P}{N}\right) - \frac{P}{N}\frac{\partial c^{-1}\left(\frac{P}{N}\right)}{\partial N},$$

which is positive since $\frac{\partial c^{-1}\left(\frac{P}{N}\right)}{\partial N} = \frac{\partial x^*}{\partial N} < 0$ by Proposition 5.1. □

By Proposition (4.4), there exists a unique equilibrium (K^*, x^*) for every N. If at any stage of the game, there is entry or exit, the long run equilibrium of Bitcoin mining game changes. For instance, if the number of miners increases to N', then the network moves towards a new equilibrium (K'^*, x'^*). Although entry leads to each miner reducing their technology, the aggregate

level of technology Nx'^* increases. Since $c(\cdot)$ is convex and increasing in x, this implies that the social cost of mining increases with the number of miners in the network. As noted above, the endogeneity of K does not prevent this outcome.

By contrast, when there is a single miner, the costs of operating the network are minimized. While this is seemingly an implication of Proposition 6.1, when there is a single miner, that agent will understand the impact of their own choices on K. To see this, note that when $N = 1$, that miner receives the prize, P, with probability 1. Therefore, it is guaranteed the payoff

$$U_m(x_m) = P - c(x_m)\,.$$

To maximize its payoff, the monopolist miner chooses the minimum amount of technology required to solve the puzzle. Since the monopolist's choice of technology influences the dynamic difficulty of the puzzle, it chooses the minimum technology required to perform a single computation, such that $K = 1$. If the target time is δ^*, then

$$\delta_1^* = \mathbb{E}_1(t) = \frac{1}{x_m}\,.$$

Hence, the monopolist's optimal technology is

$$x_m^* = \frac{1}{\delta_1^*}\,.$$

Free entry prevents this outcome. However, it is worthwhile noting that rather than the cost of running the network being in the billions of dollars per year, a monopoly miner could operate the network on a laptop. In effect, a monopolist provides a centralized public ledger, something that was only possible privately through trusted third parties prior to the invention of Bitcoin. The benefits of the blockchain are said to arise from its decentralized nature. But as more competition has clear costs in the protocol, our analysis highlights the need to be more precise about the benefits of that competition, which are as yet unmodeled. This can be nuanced. For instance, Nakamoto (2008) argued that the ability of a monopoly miner to undermine the network's operation in their own interests was limited. A nefarious miner would not be able to appropriate currency owned by others, but would only be able to double or multi-spend currency they already owned. Moreover, any instances of multi-spending would be visible on the blockchain, and could undermine confidence in the network. From an economics perspective, therefore, it is far from clear that a monopolist would have an incentive to attack

its own network as this would deny it a stream of future returns. Our focus here has been on the mining operation of the Bitcoin protocol and to highlight potential welfare losses arising from competition. How to properly model and consider the security and stability of the network as it relates to competition is something we leave for future research.

7 Conclusion

The fact that the Bitcoin network consumes a large amount of energy (and computational) resources, highlights its significance as a phenomenon of interest for further economic research. This paper has provided a foundation for that research by mapping the Bitcoin protocol itself into a game between miners and how these relate to key parameters that are potentially observable by researchers. In doing so, it highlights the role of competition in driving those costs and suggests that absent clear benefits to more competition, the resources miners are expending in the Bitcoin system are likely to be wasteful. This highlights the importance of modeling and examining those benefits something that we have not done in this paper. Such an analysis would require understanding the importance of decentralization in the integrity of peer-to-peer payment systems as well as other benefits blockchain technologies might bring.[9]

Our model highlights that free entry is the cause of resource usage by the Bitcoin that is related to the prize in the game in terms of the value of newly minted bitcoins. In terms of what might reduce that resource usage we can point to several opportunities. First, a network could provide a means of ensuring that only a limited number of miners play the game at any stage. Rather than having free entry determine that number, the number could be the minimum necessary to achieve integrity and other goals. However, if this number were small, one would have to investigate whether individual miners had an incentive and ability to influence the computational difficulty of the puzzle in equilibrium. Second, a network could issue its currency upfront and rely exclusively on user-generated transaction fees as the reward for running the network. Presumably, the rationale for a slower release of currency was stabilization of the currency's value. However, experience with bitcoin suggests that rationale is been undermined.

[9]For instance, in terms of verification (Catalini and Gans (2017)).

A Appendix

We derive the minimum order statistic for the independent random variables $t_1, t_2, \ldots, t_N$, where $t_i \sim \text{Gamma}(K, x_i)$.

We note the cumulative density function of the gamma distribution is

$$F(t; K, x) = \int_0^t f(t; K, x)\ dt = \int_0^t \frac{t^{K-1}}{\Gamma(K)} x^K e^{-xt}\ dt\,.$$

Without loss of generality, suppose we wish to find the probability that $i = 1$ will reach the threshold of K Poisson events in the fastest time. Let $T = min(t_2, t_3, \ldots, t_N)$. Then, the cumulative density function of t_1 is

$$\begin{aligned}
\mathbb{P}(T > t_1) &= \mathbb{P}(T_2 > t_1, T_3 > t_1, \ldots, T_N > t_1) \\
&= \mathbb{P}(T_2 > t_1)\mathbb{P}(T_3 > t_1) \cdots \mathbb{P}(T_N > t_1) \\
&= [1 - F_2(t_1)]\,[1 - F_3(t_1)] \cdots [1 - F_N(t_1)] \\
&= \prod_{i=2}^{N} [1 - F_i(t_1)]\ .
\end{aligned} \tag{9}$$

This gives the probability that the t_1 is the fastest realized time for all t_i.

The probability density function of t_1 is

$$\begin{aligned}
\mathbb{P}(T = t_1) &= \frac{d}{dt_i}[1 - \mathbb{P}(T > t_1)] \\
&= \sum_{i=2}^{N} f_i(t_1) \prod_{\substack{j=2 \\ j \neq i}}^{N} [1 - F_j(t_1)]\ .
\end{aligned}$$

Now, suppose that the random variables t_i are i.i.d. to $\text{Gamma}(K, x)$, (9) becomes

$$\mathbb{P}(T > t_1) = [1 - F(t_1)]^{N-1}\ . \tag{10}$$

Taking the first order derivative of (10) with respect to the realized time t_1

$$\frac{\partial \mathbb{P}(T > t_1)}{\partial t_1} = -(N-1) f(t_1)[1 - F(t_1)]^{N-2} < 0\,, \tag{11}$$

for all $t_1 > 0$. The probability that $i = 1$ realizes the fastest time for all i is decreasing in the actual time realized.

The second derivative with respect to t_1 is

$$\begin{aligned}
&\frac{\partial^2 \mathbb{P}(T > t_1)}{\partial t_1^2} \\
&\quad = -(N-1)\left[f'(t_1)[1 - F(t_1)]^{N-2} - (N-2) f(t_1)^2 [1 - F(t_1)]^{N-3}\right]\ .
\end{aligned}$$

We note that $f'(t_1) > 0$ for $t_1 < \frac{K-1}{x}$ and $f'(t_1) < 0$ for $t_1 > \frac{K-1}{x}$, where $\frac{K-1}{x}$ is the mode of $f(t_1; K, x)$. Furthermore, $f''(t_1) < 0$. All other terms are positive for all t_1.

The sign of the second derivative on $t_1 < \frac{K-1}{x}$ depends on the relative size of the terms in the square bracket. If

$$|f'(t_1)[1 - F(t_1)]^{N-2}| > |(N-2)f(t_1)^2[1 - F(t_1)]^{N-3}|\,,$$

then the second derivative will be negative for some $\hat{t}_1 \leq \frac{K-1}{x}$. If not, then the second derivative is positive on $t_1 < \frac{K-1}{x}$.

For $t_1 > \frac{K-1}{x}$, the term in square brackets is always negative, so the sign of the second derivative is always positive in this region.

In summary, the cumulative density function is concave when $t_1 < \hat{t}_1$, and convex when $t_1 > \hat{t}_1$, where $\hat{t}_1 \geq 0$.

$$\begin{cases} \frac{\partial^2 \mathbb{P}(T>t_1)}{\partial t_1^2} < 0\,, & t_1 < \hat{t}_1 \\ \frac{\partial^2 \mathbb{P}(T>t_1)}{\partial t_1^2} = 0\,, & t_1 = \hat{t}_1 \\ \frac{\partial^2 \mathbb{P}(T>t_1)}{\partial t_1^2} > 0\,, & t_1 > \hat{t}_1\,. \end{cases} \tag{12}$$

References

Athey, S., I. Parashkevov, V. Sarukkai, and J. Xia (2016). Bitcoin Pricing, Adoption, and Usage: Theory and Evidence. Available at `https://ssrn.com/abstract=2826674`.

Back, A. (2002). Hashcash – A Denial of Service Counter-Measure. Available at `http://www.hashcash.org/papers/hashcash.pdf`.

Catalini, C., J.S. Gans, (2017). Some simple economics of the blockchain. Available at `https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2874598`.

Dimitri, N. (2017). Bitcoin Mining as a Contest. *Ledger 2*, 31-37.

Gans, J. S. and H. Halaburda (2015). Some Economics of Private Digital Currency. In *Economic Analysis of the Digital Economy*, pp. 257–276. University of Chicago Press.

Huberman, G., J. D. Leshno, and C. C. Moallemi (2017). Monopoly without a Monopolist: An Economic Analysis of the Bitcoin Payment System. *Columbia Business School Research Paper*. Available at `https://ssrn.com/abstract=3025604`.

Kocherlakota, N. (1998). Money is Memory. *Journal of Economic Theory 81*, 232–251.

Loury, G. C. (1979). Market Structure and Innovation. *The Quarterly Journal of Economics 93*(3), 395–410.

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Available at `https://bitcoin.org/bitcoin.pdf`.