2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC) | 978-1-6654-3578-9/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICBC51069.2021.9461110

# A Predicament in Securing Blockchain Consensus via Controlling Cryptopuzzle Difficulty

Venkata Sriram Siddhardh Nadendla
Department of Computer Science
Missouri University of Science and Technology
Rolla, Missouri 65401.
Email: nadendla@mst.edu

Lav R. Varshney
Department of Electrical and Computer Engineering,
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801.
Email: varshney@illinois.edu

***Abstract*—Blockchain systems often employ proof-of-work consensus protocols to validate and add transactions into hashchains. These protocols involve competition among miners in solving cryptopuzzles (e.g. SHA-256 hash computation in Bitcoin) in exchange for a monetary reward. Here, we model mining as an all-pay auction, where miners' computational efforts are interpreted as bids, and the allocation function is the probability of solving the cryptopuzzle in a single attempt with unit (normalized) computational capability. Such an allocation function captures how blockchain systems control the difficulty of the cryptopuzzle as a function of miners' computational abilities (bids). In an attempt to reduce mining costs, we investigate designing a mining auction mechanism which induces a logit equilibrium amongst the miners with choice distributions that are unilaterally decreasing with costs at each miner. We show it is impossible to design a lenient allocation function that does this. Specifically, we show that there exists no allocation function that discourages miners to bid higher costs at logit equilibrium, if the rate of change of difficulty with respect to each miner's cost is bounded by the inverse of the sum of costs of all the miners. Additionally, we also show that it is necessary to have allocation functions that decrease with increasing number of players. As a result, it is difficult to achieve decentralization and accomplish secure blockchain systems with global block difficulty which relies only on total hash rate.**

## I. Motivation

Permissionless blockchain systems including the Bitcoin cryptocurrency rely on proof-of-work consensus protocols that involve competitions among participants to solve difficult computational problems (*cryptopuzzles*). These participants, called *miners*, are bounded by the costs of resources needed for computation, such as energy. The consensus protocols, however, are subject to so-called forking attacks (51% attacks), where a miner or pool of miners having a large fraction of the computation power in the system can asymptotically almost surely fork the blockchain to prevent new transactions from being verified, double-spend coins, or destroy the system via dramatic loss of confidence [1], [2]. As such, an implicit assumption in ensuring the security of the distributed trust system is that there are a large number of independent miners with incentives to follow the protocol. In current practice, though, a small number of participants perform the majority of mining, often concentrated in locales such as in China where energy costs are low [3].

978-1-6654-3578-9/21/$31.00 ©2021 IEEE

One can view mining as participating in an all-pay auction [4], where the bidding strategy captures heterogeneity amongst miners due to non-identical computational abilities and diverse electricity costs at different geographic locations. Recall that in an all-pay auction, the bid is forfeited whether win or lose [5]. In Bitcoin, mining involves computing the SHA-256 hash function over and over as quickly as possible, and so the bid can be thought of as the hash rate; likewise in other blockchain systems. Equal hash rates (bids) incur varying costs to different miners, depending on the basic cost of resources in different locales. The Nash equilibrium strategies for all-pay auctions under complete information are such that only the two strongest players (lowest costs of bidding) should actively participate and all others should bid zero [6]; this is exactly a concentration of participants that is problematic from a security perspective.

On the other hand, there is over-participation in many practical settings of all-pay auctions where there are many more than two active participants. Several explanations for over-bidding behavior have been suggested in the literature, including bounded rationality and prospect-theoretic explanations [7].

Even with rational agents, overbidding behavior can emerge. In the context of crowdsourcing contests, previous work in designing auction systems has demonstrated that reducing information about competitors can increase participation [8]–[10]. When players have incomplete information about other players' strengths, the Bayesian equilibrium strategies involve participation by more than two players [11]. Alternatively, when bids do not directly translate into winning or losing, but rather only into increased chances of winning or losing, quantal response equilibrium (QRE) strategies also promote greater participation than that of Nash equilibrium strategies [12]. (Note that QRE is often used to model bounded rationality of human agents, but in blockchain mining, the auction itself has inherent uncertainty due to the randomness in correctly finding the correct hash in a mining contest.)

Since 2017, game theory has been used by several researchers in the design of secure blockchain systems [4], [13]–[21]. Most of these efforts investigate various economic reasons behind the centralization of Bitcoin mining. For example, Budish showed that the necessary conditions for miners to be at equilibrium are very expensive, which promotes miners to sabotage via pooling their resources [14]. On the other hand, Huberman *et al.* have shown that both the block reward and the transaction fees in Bitcoin do not reflect miners' preferences, causing temporal fluctuations in miners' investments [15]. Another interesting perspective on the centralization of Bitcoin miners was given by Leonardos *et al.*, where miners are assumed to play oceanic games, as opposed to non-cooperative games, which model interactions between small numbers of dominant players and large numbers of individually insignificant players, as in the case of Bitcoin mining [16]. It was shown that oceanic games in Bitcoin mining incentivize miners to join forces and form coalitions that increase the concentration of mining power. Sun *et al.* model mining as a game where each miner has limited mining power and competes for multiple tokens. By finding closed-form expressions of both Nash equilibrium and Stackelberg equilibrium of the game, they find that heterogeneous participation emerges. These findings complement similar findings that heterogeneous participation emerges from other game-theoretic models and equilibria [19]–[21]. Such heterogeneous participation has similar security implications as we find here.

Like these efforts, our work also contributes further to the game theory of blockchain systems. Specifically, we model blockchain mining as an all-pay auction to design cryptopuzzles that discourage miners to adopt higher computational costs at logit equilibrium (QRE with logit responses) with all miners actively participating. We show that it is not possible to design such a trustworthy distributed protocol, if the blockchain system does not react sharply to the increasing miner costs.

## II. Quantal Response Equilibrium

Traditionally, interaction between players in a game is analyzed using a solution concept called Nash equilibrium, where heterogeneous players make decisions in a selfish manner so as to maximize their respective expected utilities. This stems from the assumption that every player knows the expected utilities of every other player in the game. However, this is not true in the real world due to our inability to comprehend human rationality perfectly from limited/sparse/repetitive choice observations. Therefore, a natural extension is to fit choice observations to a noisy utility model (as in regression problems). In other words, the utility obtained by Player $i$ for choosing the $j$th strategy is denoted as

$$u_{i,j} = v_{i,j} + \epsilon_{i,j}, \tag{1}$$

where $v_{i,j}$ is a representative utility that is measurable by the observer, and $\epsilon_{i,j}$ is a random noise term that accounts for all the unobservable/unknown attributes. Since the players are assumed to be utility maximizers, the probability of choosing the $j$th alternative (*a.k.a.* logit choice probability) can be computed as

$$Pr(u_{i,j} > u_{i,k}, \ \forall\ k \neq j) = \frac{\exp(v_{i,j})}{\displaystyle\sum_{k \in \mathcal{S}} \exp(v_{i,k})}, \tag{2}$$

when the random noise $\epsilon_{i,j}$ follows Gumbel-distribution (an extreme value distribution) [22].

Quantal response equilibrium (QRE) is a natural extension of Nash equilibrium in a discrete (or 'quantal') choice setting when all the players are assumed to maximize random utilities that are consistent with respect to observed decisions in the past. Logit equilibrium is a special kind of QRE where the random noise in the utility model follows the Gumbel distribution. Let $\pi_{i,j}$ denote the logit choice probability for Player $i$ for choosing the $j$th alternative in the choice set $\mathcal{C}$. In a game-theoretic setting with a player set $\mathcal{N} = \{1, \ldots, N\}$, the representative utility $v_{i,j}$ is a function of logit probabilities of all players $\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_N$. Therefore, the logit equilibrium is formally defined as follows [23].

**Definition 1.** *Given a normal-form game $\Gamma = \{\mathcal{N}, \mathcal{S}, \boldsymbol{u}\}$, a logit equilibrium is any tuple of logit probabilities $\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_N$ which satisfies the set of equations*

$$\pi_{i,j} = \frac{\exp\left[v_{i,j}(\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_N)\right]}{\displaystyle\sum_{k \in \mathcal{S}} \exp\left[v_{i,k}(\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_N)\right]}, \tag{3}$$

*for all $i \in \mathcal{N}$ and $j \in \mathcal{S}$.*

While the above definition is designed to handle discrete choices, a similar definition has also been extended to continuous choice models with random utility models. In this case, the denominator term in Equation (3) reduces to a constant that ensures that the density integrates to one. As a result, we have the following definition for the continuous logit equilibrium [12].

**Definition 2.** *Given a normal-form game $\Gamma = \{\mathcal{N}, \mathcal{S}, \boldsymbol{u}\}$ where $\mathcal{S}$ is a continuous strategy set, then a logit equilibrium is any tuple of logit density functions $\pi_1, \ldots, \pi_N$ which satisfies the set of equations*

$$\pi_i(s) = \delta_i \exp\left[v_i(\pi_1(s), \ldots, \pi_N(s))\right], \tag{4}$$

*for all $i \in \mathcal{N}$ and $s \in \mathcal{S}$, where $\delta_i$ is a constant that ensures that the above density function integrates to one.*

Since QRE imposes the requirement that the beliefs match the equilibrium choice probabilities, it requires solving for a fixed point in choice probabilities, which is similar to that of finding a Nash equilibrium. Furthermore, QRE admits a quantal response function (analogous to best response functions in the case of Nash equilibrium), which smooths out discontinuities in best response functions, while simultaneously preserving monotonicity. In other words, as quantal response functions become steeper, they have the potential to approximate best response functions. Consequently, QRE is an effective generalization of Nash equilibrium.

## III. Modeling Blockchain Mining as All-Pay Auctions with Quantal Responses

Let $\mathcal{M} = \{1, \ldots, N\}$ denote the set of $N$ blockchain miners, who compete against each other in solving a given cryptographic puzzle (*i.e.* computing a target hash) and win a prize of value $A > 0$. During this competition, each miner makes multiple attempts sequentially to solve the crypto-puzzle. Let the outcome of the $k$th attempt made by the $i$th miner be denoted as $a_{i,k} \in \{0, 1\}$, where $a_{i,k} = 1$ denotes the puzzle being solved successfully. Since a crypto-puzzle can only be solved using random guesses, it is natural to model the outcome of the $i$th miner at time $k$, i.e. $a_{i,k}$, as a Bernoulli random variable with probability $P(a_{i,k} = 1) = p_i$. Note that this probability $p_i$ characterizes the difficulty-level of the crypto-puzzle at the $i$th agent, since smaller values of $p_i$ needs several Bernoulli trials to obtain the outcome of $a_{i,k} = 1$. Note that modeling blockchain mining as a sequence of Bernoulli trials is not new. For example, Bagaria *et al.* have modeled Bitcoin mining as a Poisson process [24].

In this paper, we assume that each player employs a different hash rate in computing the hash function in the crypto-puzzle. Let $K$ denote the total number of random guesses after which one of the miners solves the cryto-puzzle successfully. Then, the $i$th miner wins the prize $A$, if

$$\sum_{k=1}^{K} a_{i,k} = 1. \tag{5}$$

In practical settings, mining agents have non-identical computational capabilities. For example, a miner with larger computational resources can complete the task in less effort per attempt (e.g. average run-time to execute a pseudorandom generator), as opposed to a less resourceful miner who needs more effort per attempt to complete the same task. This effort cost could be based on the cost of energy or specialized hardware availability [25]. We model this miner heterogeneity (in terms of computational abilities and/or geo-location based disparities in electricity prices) using a non-negative cost-bid $c_i \in \mathbb{R}_+$ per attempt at the $i$th miner, for all $i = 1, \ldots, N$. Furthermore, if we assume that the joint belief about the other agents' cost-bids are denoted as $\phi(\boldsymbol{c}_{-i})$, the average probability with which the $i$th miner solves the puzzle before other miners is given by

$$\begin{aligned} Q_i(c_i) &= \mathbb{E}_{\phi(c_{-i})}\left[q_i(c_i, \boldsymbol{c}_{-i})\right] \\ &= \mathbb{E}_{\phi(c_{-i})}\left[P\left(\sum_{k=1}^{K} a_{i,k} = 1, \sum_{k=1}^{K} a_{-i,k} = 0 \;\middle|\; c_i, \boldsymbol{c}_{-i}\right)\right], \\ &= \int_{\mathbb{R}^{N-1}} p_i(1-p_i)^{K-1} \cdot \prod_{j \neq i} \left[(1-p_j)^K\right] \cdot \phi(\boldsymbol{c}_{-i})\ d\boldsymbol{c}_{-i}. \end{aligned} \tag{6}$$

where $q_i(c_i, \boldsymbol{c}_{-i})$ is the conditional probability with which the $i$th miner solves the puzzle before other miners, given the bid profile $\boldsymbol{c} = (c_i, \boldsymbol{c}_{-i})$.

Then, the expected utility of the $i$th miner choosing a cost-bid $c_i$ is given by

$$U_i(c_i) \;\;=\;\; A \cdot Q_i(c_i) - K \cdot c_i, \tag{7}$$

where the first term represents the average reward obtained by the $i$th miner, and the second term represents the total effort invested by the $i$th miner over $K$ attempts. Since the competition ends whenever a miner finds the target hash within the given crypto-puzzle, the *individual rationality* of each miner is satiated only when $U_i \geq 0$ for all $i = 1, \ldots, n$.

Furthermore, since blockchain is known to automatically choose the difficulty of the crypto-puzzle depending on miners' ability profile $\boldsymbol{c} = \{c_1, \ldots, c_N\}$, we model this allocation as a probability $f(\boldsymbol{c})$ with which the puzzle can be solved in one attempt per unit cost (computational ability). In the rest of this paper, we refer to $f(\boldsymbol{c})$ as the *allocation function*. For example, in the context of *Bitcoin* (which uses hashcash algorithm with SHA-256), given that the previous 2016 blocks would have been found at the rate of one every 10 minutes (i.e. 600 seconds), a difficulty offset of $D$ effectively requires an average of $D \times \frac{2^{32}}{600}$ hashes per second to find a valid block [26]. In other words, if the cost-bid $c_i$ represents the hash rate at the $i^{th}$ miner, Bitcoin's allocation function is given by

$$f(\boldsymbol{c}) = \frac{600}{D \times 2^{32}} \sum_{j \in \mathcal{M}} c_j. \tag{8}$$

However, this formula should be taken with a grain of salt, as it will change depending on how the cost-bid is defined in our auction model. For example, if the cost-bid represents the cost of guessing the hash, an appropriate transformation to the cost-bids should be used to compute the allocation function, as shown in [27].

In the presence of multiple agents (miners) with a cost-bid profile $\boldsymbol{c} = \{c_1, \ldots, c_N\}$, we can compute the success probability $p_i$ at the $i^{th}$ miner as

$$p_i = f(\boldsymbol{c}) \cdot \frac{c_i}{\displaystyle\sum_{j \in \mathcal{M}} c_j}. \tag{9}$$

In other words, the strategies available at the blockchain system (auctioneer) is to generate an appropriate crypto-puzzle via choosing a difficulty-level that specifies $p_i$ at its miners accordingly. On the other hand, the miners' strategies include choosing effort-costs, which are revealed to the blockchain system. Therefore, it is natural to model this interaction between the blockchain system and its miners as a *mining auction*, where the miners' effort-costs are their bids $\boldsymbol{c} = \{c_1, \cdots, c_N\}$ within the auction and the blockchain system (auctioneer) designs prize $A$ and allocation function $f$.

In a traditional game-theoretic setting, the equilibrium of the mechanism is defined as a strategy profile where all the miners employ best responses to all the other miners' responses. In other words, for any $i \in \mathcal{M}$, given a belief $\phi(\boldsymbol{c}_{-i})$ from all the other players, the best response employed by the $i$th miner at Nash equilibrium satisfies the following conditions:

$$U_i(c_i, \phi(\boldsymbol{c}_{-i})) \;\geq\; U_i(c_i', \phi(\boldsymbol{c}_{-i})), \tag{10}$$

for all $c_i' \in \mathbb{R}$ and for all $i \in \mathcal{M}$.

Although a blockchain system usually reveals its allocation function publicly to its miners (e.g. Bitcoin), agents may not know[1] the type of other players since miners (or miner pools)

[1] Although it is possible to estimate miners' bids from historical interactions, it is impossible to know if other miners have updated their computational capabilities in this auction round.

may not necessarily reveal their bids to other agents. Consequently, the miners can potentially violate their *individual rationality* conditions and not necessarily follow Nash equilibrium stated in Equation (10). As noted previously, similar behavior is also observed in several auction settings where human agents over-dissipate their bids and seemingly violate their *individual rationality* due to incomplete information [8]. An alternate method to account for the overdissipation of bids (efforts) is to justify decision errors using random utility models at the players [12]. More specifically, the uncertainty in the utility of $i^{th}$ miner in Equation (7) comes from the lack of knowledge of $K$ in advance and is fundamental to the blockchain setting, rather than a manifestation of bounded rationality. Therefore, in this paper, we assume that the miners choose strategies so that the mining auction converges to quantal response equilibrium (QRE), as opposed to NE.

Given that the blockchain costs can be interpreted as a continuous-valued choice, we adopt the continuous logit equilibrium definition stated in Definition 2 in our all-pay auction model for blockchain mining, as shown below.

$$\begin{aligned} \pi_i(c_i) &= \delta_i \exp\left(\frac{U_i(c_i)}{\mu}\right) \\ &= \delta_i \exp\left(\frac{A \cdot Q_i(c_i) - K \cdot c_i}{\mu}\right) \end{aligned} \tag{11}$$

for all $i = 1, \ldots, N$, where $\pi_i(c_i)$ denotes the probability of $i^{th}$ miner solving the cryptopuzzle before any other agent, $U_i$ is the expected utility at the $i$th agent as given in Equation (7), $\mu$ is the error parameter, and $\delta_i$ is a constant that ensures that the density integrates to one. Obviously, when $c_i = 0$, we have $p_i = 0$. Therefore, $\delta_i = \pi_i(c_i = 0)$.

In typical Blockchain systems, miners typically join together as mining pools to gather large amounts of computational resources, which results in a large computational cost $c_i$ at the $i$th miner. Therefore, our goal is to investigate how $\pi_i(c_i)$ change with the computational cost $c_i$, at logit equilibrium. In this regard, we present the necessary condition for the $i$th miner to be discouraged to have a large $c_i$ in the following proposition.

One important point to note is that, in traditional Blockchain systems, every miner gains access to large amounts of computation resources via joining mining pools. Therefore, in this paper, we will verify two important observations regarding blockchain systems from a logit equilibrium standpoint:

(a) How does $\pi_i(c_i)$ change with the computational cost $c_i$?,
(b) How does the arrival of a new miner affect $\pi_i(c_i)$?

We will study each of these questions in the following two subsections respectively. Lastly, we have also summarized all the symbols used in this paper in Table III for the sake of readability.

| Symbol | Description |
|---|---|
| $\mathcal{M}$ | Set of miners: $\{1, \cdots, N\}$ |
| $A$ | Reward provided to the successful miner |
| $a_{i,k}$ | Binary outcome of $k^{th}$ attempt made by $i^{th}$ miner in guessing the nonce. |
| $p_i$ | Probability that the $i^{th}$ miner is successful in a single guess |
| $K$ | Number of random guesses after which one of miners has successfully guessed the nonce |
| $c_i$ | A non-negative cost bid per attempt at the $i^{th}$ miner which represents computational power and/or energy price |
| $q_i(c_i, \boldsymbol{c}_{-i})$ | Conditional probability that $i$th miner solves the puzzle before other miners, given the bid profile $\boldsymbol{c} = (c_i, \boldsymbol{c}_{-i})$ |
| $Q_i(c_i)$ | Average probability with which the $i$th miner solves the puzzle before other miners |
| $U_i(c_i)$ | Expected utility at $i^{th}$ miner for choosing cost-bid $c_i$ |
| $f(\boldsymbol{c})$ | *Blockchain allocation function*, i.e. the probability of solving the cryptopuzzle in one attempt per unit cost-bid. |
| $\pi_i(c_i)$ | Probability that $i^{th}$ miner solves the cryptopuzzle successfully before any other agent |
| $\phi(\boldsymbol{c}_{-i})$ | Joint belief at the $i^{th}$ miner about all the other agents' cost-bids |

TABLE I
SUMMARY OF SYMBOLS AND THEIR DESCRIPTION

## IV. Impact of Computational Cost on Allocation Function

For the sake of easy notation, let us denote

$$\begin{aligned} q_i(c_i, \boldsymbol{c}_{-i}) &= P\left(\sum_{k=1}^{K} a_{i,k} = 1, \sum_{k=1}^{K} a_{-i,k} = 0 \;\middle|\; c_i, \boldsymbol{c}_{-i}\right) \\ &= p_i(1-p_i)^{K-1} \cdot \prod_{j \neq i} \left[(1-p_j)^K\right]. \end{aligned} \tag{12}$$

In other words, we have $Q_i(c_i) = \mathbb{E}_{\pi(c_{-i})}\left[q_i(c_i, \boldsymbol{c}_{-i})\right]$. Then, the impact of probability $q_i$ on the mining auction is characterized by the following proposition.

**Proposition 1.** *A mining auction discourages its miners to adopt higher computational capabilities if*

$$\frac{\partial q_i(c_i, \boldsymbol{c}_{-i})}{\partial c_i} \leq \frac{K}{A}$$

*holds true for all $i \in \mathcal{M}$.*

*Proof:* Note that, in order to demotivate miners to accumulate higher computational capabilities, we desire $\pi_i(c_i)$ to be a decreasing function of $c_i$. This can happen only when

$$\frac{\partial \pi_i(c_i)}{\partial c_i} = \frac{\pi_i(c_i)}{\mu}\left(A \cdot \frac{\partial Q_i(c_i)}{\partial c_i} - K\right) \leq 0. \tag{13}$$

In other words, an idealistic mining auction satisfies the condition

$$\frac{\partial Q_i(c_i)}{\partial c_i} = \int_{\mathbb{R}^{N-1}} \frac{\partial q_i(c_i, \boldsymbol{c}_{-i})}{\partial c_i}\, \pi_{-i}(\boldsymbol{c}_{-i})\; d\boldsymbol{c}_{-i} \;\leq\; \frac{K}{A}, \tag{14}$$

whenever agents employ quantal responses as opposed to fixed best responses. Note that, if

$$\frac{\partial q_i(c_i, \boldsymbol{c}_{-i})}{\partial c_i} \leq \frac{K}{A}$$

holds true, the inequality in Equation (14) holds true as well. ■

In the remainder of this paper, our goal is to identify a mining auction (i.e. an appropriate allocation function $f(\boldsymbol{c})$) that satisfies the condition presented in Proposition 1. In this journey, we rely on some minor results, which are first stated as lemmas.

**Lemma 1.** *If $f(c_i, \boldsymbol{c}_{-i})$ is an increasing function of $c_i$, $p_i$ is increasing in $c_i$ for a fixed profile $\boldsymbol{c}_{-i}$. Furthermore, if $f(c_i, \boldsymbol{c}_{-i})$ is $\left(\frac{1}{\sum\limits_{j\in\mathcal{M}} c_j}\right)$-Lipschitz in $c_i$, then $p_i$ is $\left(\frac{1}{\sum\limits_{j\in\mathcal{M}} c_j}\right)$-Lipschitz in $c_i$ for all $i \in \mathcal{M}$.*

*Proof:* We compute the partial derivative of $p_i$ with respect to $c_i$ as shown below.

$$\frac{\partial p_i}{\partial c_i} = \frac{\partial f}{\partial c_i} \cdot \frac{c_i}{\sum\limits_{j\in\mathcal{M}} c_j} + f \cdot \frac{\sum\limits_{j\neq i} c_j}{\left(\sum\limits_{j\in\mathcal{M}} c_j\right)^2}. \tag{15}$$

Note that the right side is always non-negative, as long as $\frac{\partial f}{\partial c_i}$ is non-negative.

Now, if $f$ is $\left(\frac{1}{\sum\limits_{j\in\mathcal{M}} c_j}\right)$-Lipschitz in $c_i$ for all $i \in \mathcal{M}$, we have

$$\frac{\partial p_i}{\partial c_i} \quad \leq \quad \frac{c_i}{\left(\sum\limits_{j\in\mathcal{M}} c_j\right)^2} + f \cdot \frac{\sum\limits_{j\neq i} c_j}{\left(\sum\limits_{j\in\mathcal{M}} c_j\right)^2} \quad \leq \quad \frac{1}{\sum\limits_{j\in\mathcal{M}} c_j}. \tag{16}$$

■

**Lemma 2.** *If $f(c_i, \boldsymbol{c}_{-i})$ is increasing in $c_i$ for all $i \in \mathcal{M}$, then we have*

$$\frac{\partial p_j}{\partial c_i} \quad \geq \quad \frac{-c_i}{\left(\sum\limits_{j\in\mathcal{M}} c_j\right)^2}. \tag{17}$$

*Proof:* We compute the partial derivative of $p_j$ with respect to $c_i$ for any $j \neq i$, as shown below.

$$\frac{\partial p_j}{\partial c_i} = \frac{\partial f}{\partial c_i} \cdot \frac{c_i}{\sum\limits_{j\in\mathcal{M}} c_j} - f \cdot \frac{c_i}{\left(\sum\limits_{j\in\mathcal{M}} c_j\right)^2}. \tag{18}$$

If $f(c_i, \boldsymbol{c}_{-i})$ is increasing in $c_i$ and since $f \leq 1$, then we have Equation (17). ■

Next, we state the main result in this paper in the following theorem.

**Theorem 1.** *There does not exist a $\left(\frac{1}{\sum\limits_{j\in\mathcal{M}} c_j}\right)$-Lipschitz allocation function $f(c_i, \boldsymbol{c}_{-i})$ that increases unilaterally with $c_i$ for all $i \in \mathcal{M}$, which discourages miners to bid higher costs at logit equilibrium.*

*Proof:* In the following, we compute the partial derivative of $g_i = \log q_i$ with respect to $c_i$:

$$\frac{1}{g_i} \cdot \frac{\partial g_i}{\partial c_i} = \left[\frac{1}{p_i} - \frac{K-1}{1-p_i}\right] \frac{\partial p_i}{\partial c_i} - K \cdot \sum_{j\neq i} \left(\frac{1}{1-p_j}\right) \frac{\partial p_j}{\partial c_i} \tag{19}$$

$$\begin{aligned} &\leq \frac{1-Kp_i}{p_i(1-p_i)} \cdot \frac{1}{\sum\limits_{m\in\mathcal{M}} c_m} \\ &\quad + \ K \cdot \frac{c_i}{\left(\sum\limits_{j\in\mathcal{M}} c_j\right)^2} \cdot \sum_{j\neq i} \frac{1}{1-p_j} \end{aligned} \tag{20}$$

Since $g_i(\boldsymbol{c}) \leq 1$ and $0 \leq p_i \leq f$, we have

$$\begin{aligned} \frac{\partial g_i}{\partial c_i} \quad &\leq \quad \frac{1}{c_i f(1-f)} \ + \ K \cdot \frac{c_i}{\left(\sum\limits_{j\in\mathcal{M}} c_j\right)^2} \cdot \frac{N-1}{1-f} \\ &= \quad \frac{c_{tot.}^2 + Kc_i f(N-1)}{c_i f(1-f) c_{tot.}^2} \end{aligned} \tag{21}$$

where $c_{tot.} = \sum\limits_{j\in\mathcal{M}} c_j$.

From Proposition 2, the allocation function $f$ demotivates miners to adopt higher computational capabilities if

$$\frac{c_{tot.}^2 + Kc_i f(N-1)}{c_i f(1-f) c_{tot.}^2} \quad \leq \quad \frac{K}{A}. \tag{22}$$

In other words, we expect $f$ to satisfy

$$f^2 + \left[\frac{A(N-1)}{c_{tot.}^2} - 1\right] f + \frac{A}{Kc_i} \quad \leq \quad 0, \tag{23}$$

for all $i \in \mathcal{M}$.

In other words, if we denote $c_{min} = \min\limits_{i\in\mathcal{M}} c_i$, then it is sufficient if $f$ satisfies

$$f^2 + \left[\frac{A(N-1)}{c_{tot.}^2} - 1\right] f + \frac{A}{Kc_{min}} \quad \leq \quad 0. \tag{24}$$

Note that the above equation can be equivalently written as

$$\begin{aligned}&\left\{f+\frac{1}{2}\left[\frac{A(N-1)}{c_{tot.}^2}-1\right]\right\}^2+\frac{A}{Kc_{min}} \\ &\quad \le \frac{1}{4}\left[\frac{A(N-1)}{c_{tot.}^2}-1\right]^2.\end{aligned} \tag{25}$$

This inequality cannot be achieved since the left side of the above inequality is always larger than the right side. ■

In other words, the theorem says that it is impossible to design a lenient allocation function for blockchain systems that discourages miners to adopt higher computational capabilities. That is, from the perspective of logit equilibrium, blockchain systems need to take severe actions (in terms of controlling the cryptopuzzle difficulty) against its miners to discourage them towards lower costs.

## V. Impact of a New Miner on Allocation Function

In order to study how allocation function changes with an additional miner, we study this problem in two stages. In the first stage, we assume that there are $N$ miners (denoted as the set $\mathcal{M}^{(N)}$) with fixed computational costs $\boldsymbol{c}_N = \{c_1, \ldots, c_N\}$. In the second stage, we assume that a new miner with computational cost of $c_{N+1}$ joins the mining auction to form a set $\mathcal{M}^{(N+1)}$ with computational costs $\boldsymbol{c}_{N+1} = \{c_1, \ldots, c_N, c_{N+1}\}$. Similar to our notation for the set of miners $\mathcal{M}^{(N)}$ and $\mathcal{M}^{(N+1)}$ in the two stages, we henceforth introduce superscripts $(N)$ and $(N+1)$ for all relevant symbols in the first and second stages respectively.

**Lemma 3.** *If $f(\boldsymbol{c}_N)$ decreases with increasing $N$, then $p_i^{(N)}$ also decreases as $N$ increases.*

*Proof:* Rewriting Equation (9) using the new notation, we have

$$p_i^{(N)} = f_N(\boldsymbol{c}_N)\cdot\frac{c_i}{\displaystyle\sum_{j\in\mathcal{M}^{(N)}} c_j}. \tag{26}$$

Using the above equation, we identify a recursive relation between $p_i^{(N)}$ and $p_i^{(N+1)}$ as shown below:

$$\begin{aligned} p_i^{(N+1)} &= f_{N+1}(\boldsymbol{c}_{N+1})\cdot\frac{c_i}{\displaystyle\sum_{j\in\mathcal{M}^{(N+1)}} c_j} \\ &= p_i^{(N)}\cdot\frac{f_{N+1}(\boldsymbol{c}_{N+1})}{f_N(\boldsymbol{c}_N)}\cdot\frac{\displaystyle\sum_{j\in\mathcal{M}_N} c_j}{\displaystyle\sum_{j\in\mathcal{M}^{(N+1)}} c_j}.\end{aligned} \tag{27}$$

Let $\eta$ denote the ratio of $p_i^{(N+1)}$ to $p_i^{(N)}$, i.e. we have

$$\eta = \frac{f_{N+1}(\boldsymbol{c}_{N+1})}{f_N(\boldsymbol{c}_N)}\cdot\frac{\displaystyle\sum_{j\in\mathcal{M}_N} c_j}{\displaystyle\sum_{j\in\mathcal{M}^{(N+1)}} c_j}.$$

Assuming that $f_N(\boldsymbol{c}_N)$ decreases with increasing $N$, we have $f_{N+1}(\boldsymbol{c}_{N+1}) \le f_N(\boldsymbol{c}_N)$. Therefore, $\eta$ is always less than one, which leads to $p_i^{(N+1)} \le p_i^{(N)}$. ■

**Lemma 4.** *If $f(\boldsymbol{c}_N)$ decreases with increasing $N$, then*

$$\begin{aligned}& Q_i^{(N+1)}(c_i) \\ &\quad \le \eta^{NK}\int_{\mathbb{R}^{N-1}} q_i^{(N)}(c_i,\boldsymbol{c}_{-i})\cdot\phi^{(N+1)}(\boldsymbol{c}_{-i})\ d\boldsymbol{c}_{-i}.\end{aligned} \tag{28}$$

*Proof:* Using the new notation in Equation (6), the probability that the $i^{th}$ player solves the puzzle before the remaining $N-1$ players is given by

$$\begin{aligned} Q_i^{(N)}(c_i) &= \int_{\mathbb{R}^{N-1}} q_i^{(N)}(c_i,\boldsymbol{c}_{-i})\cdot\phi^{(N)}(\boldsymbol{c}_{-i})\ d\boldsymbol{c}_{-i} \\ &= \int_{\mathbb{R}^{N-1}} p_i^{(N)}\left(1-p_i^{(N)}\right)^{K-1} \\ &\qquad \cdot\prod_{j\neq i}\left[\left(1-p_j^{(N)}\right)^K\right]\cdot\phi^{(N)}(\boldsymbol{c}_{-i})\ d\boldsymbol{c}_{-i}.\end{aligned} \tag{29}$$

Note that $p_i^{(N+1)} = \eta\cdot p_i^{(N)}$ as shown in the proof of Lemma 3. Therefore, we have the inequality $1-p_i^{(N+1)} \le \eta\cdot\left(1-p_i^{(N)}\right)$, which when substituted in the expression for $Q_i^{(N+1)}(c_i)$, we get

$$\begin{aligned} Q_i^{(N+1)}(c_i) &= \int_{\mathbb{R}^N} q_i^{(N+1)}(c_i,\boldsymbol{c}_{-i})\cdot\phi^{(N+1)}(\boldsymbol{c}_{-i})\ d\boldsymbol{c}_{-i} \\ &\le \eta^{NK}\int_{\mathbb{R}^N} p_i^{(N)}\left(1-p_i^{(N)}\right)^{K-1} \\ &\qquad \cdot\prod_{j\neq i}\left[\left(1-p_j^{(N)}\right)^K\right]\cdot\phi^{(N+1)}(\boldsymbol{c}_{-i})\ d\boldsymbol{c}_{-i} \\ &= \eta^{NK}\int_{\mathbb{R}^{N-1}} q_i^{(N)}(c_i,\boldsymbol{c}_{-i}) \\ &\qquad\qquad \cdot\phi^{(N+1)}(\boldsymbol{c}_{-i})\ d\boldsymbol{c}_{-i},\end{aligned} \tag{30}$$

where the integral limits in the last equation reduced from $\mathbb{R}^N$ to $\mathbb{R}^{N-1}$, since $q_i^{(N)}(c_i,\boldsymbol{c}_{-i})$ is independent of $c_{N+1}$. ■

Then, the *ex ante* and *ex post* probabilities of solving the puzzle (denoted as $\pi_i^{(N)}(c_i)$ and $\pi_i^{(N+1)}(c_i)$ respectively) are respectively given by

$$\begin{aligned} \pi_i^{(N)}(c_i) &= \delta_i^{(N)}\exp\left(\frac{A\cdot Q_i^{(N)}(c_i)-K\cdot c_i}{\mu}\right),\ \text{and} \\ \pi_i^{(N+1)}(c_i) &= \delta_i^{(N+1)}\exp\left(\frac{A\cdot Q_i^{(N+1)}(c_i)-K\cdot c_i}{\mu}\right),\end{aligned} \tag{31}$$

where $\delta_i^{(N)} = \pi_i^{(N)}(0)$ and $\delta_i^{(N+1)} = \pi_i^{(N+1)}(0)$. In this paper, we will investigate how the probability of solving a puzzle changes with the number of miners as shown in the following theorem.

**Theorem 2.** *The necessary condition for the existence of logit equilibrium is that the allocation function $f(\boldsymbol{c})$ should decrease with increasing $N$.*

*Proof:* Let us consider the log-ratio between *ex post* and *ex ante* probabilities of solving the cryptopuzzle, as defined below:

$$\begin{aligned} R_{N,i}(c_i) &= \log \frac{\pi_i^{(N+1)}(c_i)}{\pi_i^{(N)}(c_i)} \\ &= \log \frac{\delta_i^{(N+1)}}{\delta_i^{(N)}} + \frac{A}{\mu} \left[ Q_i^{(N+1)}(c_i) - Q_i^{(N)}(c_i) \right]. \end{aligned} \tag{32}$$

Upon substituting the bound for $Q_i^{(N+1)}(c_i)$ as stated in Lemma 4, we obtain

$$\begin{aligned} & R_{N,i}(c_i) \\ & \leq \log \frac{\delta_i^{(N+1)}}{\delta_i^{(N)}} + \frac{A}{\mu} \int_{\mathbb{R}^{N-1}} \left[ \eta^{NK} \phi^{(N+1)}(\boldsymbol{c}_{-i}) - \phi^{(N)}(\boldsymbol{c}_{-i}) \right] \\ & \qquad \cdot q_i^{(N)}(c_i, \boldsymbol{c}_{-i}) \; d\boldsymbol{c}_{-i} \\ & = \log \frac{\delta_i^{(N+1)}}{\delta_i^{(N)}} + \frac{A}{\mu} \int_{\mathbb{R}^{N-1}} \left[ \eta^{NK} \sum_{j \neq i} R_{N,j}(c_j) - 1 \right] \\ & \qquad \cdot q_i^{(N)}(c_i, \boldsymbol{c}_{-i}) \cdot \phi^{(N)}(\boldsymbol{c}_{-i}) \; d\boldsymbol{c}_{-i} \end{aligned} \tag{33}$$

for all $i = 1, \ldots, N$.

Note that the probability of solving the puzzle $\pi_i^{(N)}(c_i)$ will decrease with increasing $N$ since the additional miner can have a non-negative probability of solving the puzzle before others, especially when their computational costs remain fixed. As a result, the term $\log \frac{\delta_i^{(N+1)}}{\delta_i^{(N)}}$ is negative. Furthermore, if $R_{N,i}(c_i) \leq 0$ for all $i = 1, \ldots, N$, then the above inequality holds true at all the $N$ original miners, thus resulting in a lower probability of success for everyone. ■

## VI. Discussion

The implications of the above results on the security of permissionless blockchain systems which use proof-of-work (PoW) consensus algorithms in their mining protocols (e.g. Bitcoin and Ethereum) are discussed in this section. The main reason for choosing PoW consensus algorithms in blockchain systems is to promote decentralization of hashing power so that no one miner (or a small group of miner, as in mining pools) can have the power to control the total hash rate. However, miners have every incentive to gain greater control of the hash rate in order to improve their chances of getting the reward. This can be achieved in one of the two following methods: (i) invest in more powerful hardware (e.g. ASICs) to increase their own computational capabilities, and (ii) team up with other miners to share resources and reduce competition.

The first method can be interpreted as increasing $c_i$ at the $i^{th}$ miner. Theorem 1 states that the blockchain system should increase its cryptopuzzle difficulty (allocation function) at least with a rate of $\left( \dfrac{1}{\displaystyle\sum_{j \in \mathcal{M}} c_j} \right)$ in order to discourage such aggregation of resources by one miner. Although the hardware costs have been reducing drastically over time, such a strategy is practically difficult to adopt at the miners due to significant energy costs.

On the other hand, the second method is to eliminate competition and reduce the total number of effective competitors in the mining competition. This is quite easy and is currently being observed in the form of mining pools. Theorem 2 states that a logit equilibrium exists only when the cryptopuzzle difficulty (allocation function) $f$ decreases with increasing $N$. This is quite contradicting to the original goal because current blockchain systems are forced to increase $f$ if there are fewer mining pools. However, this is not the case with blockchain systems which are only designed based on global difficulty metrics (e.g. total hash rate), and not how these hash rates are controlled by the miners (or mining pools).

This fundamental predicament in controlling decentralization of blockchain mining using global difficulty metrics (e.g. total hash rate) can be deterred via designing novel PoW consensus protocols which rely on hashing power at individual miners and/or mining pools. One approach is to design incentive-compatible online mechanisms where miners reveal their hashing rates truthfully to the blockchain system periodically so that cryptopuzzle difficulty (allocation function) can be adjusted in a manner that steers miners towards decentralization of computational power.

## VII. Summary and Future Work

In summary, we modeled mining in blockchain systems as an all-pay auction. Since many studies have shown that miners exhibit overbidding behavior, we investigated the problem of designing a mining auction mechanism which induces a logit equilibrium amongst miners. We found that the miners cannot be discouraged to bid higher costs at logit equilibrium, if the rate of change of allocation probability $f$, i.e. the probability with which the cryptopuzzle can be solved in one attempt per unit normalized-cost, with respect to each miner's cost is bounded by the inverse of the sum of costs at all the miners. In other words, it is necessary to punish the miners severely if they choose higher computational costs, in order to motivate a distributed trust system. Furthermore, we also showed that it is necessary to have allocation functions that decrease with the number of players. In other words, if new miners join the game, then they tend to pool together from the perspective of logit equilibrium. In other words, it is difficult to realize the promises upon which a secure blockchain system is built using proof-of-work consensus protocols. Consequently, if a mining protocol is controlled only using cryptopuzzle difficulty, it is practically hard to mitigate security threats such as forking attacks on blockchain systems.

In the future, we will validate our theoretical findings on real datasets, and develop improved Proof-of-Work Consensus protocols in blockchain systems via identifying allocation functions whose rate of change is not upper-bounded by the inverse of the sum of costs at all the miners. We will also investigate QRE formulations where there is also incomplete information on competitor strengths to see how the interaction between these two mechanism design strategies play out. On the other hand, we will also investigate incentive-compatible online mechanisms where miners are motivated to reveal their hashing power truthfully so that cryptopuzzle difficulty can be designed to achieve decentralization.

## References

[1] A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. Princeton, NJ, USA: Princeton University Press, 2016.

[2] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," *Communications of the ACM*, vol. 61, no. 7, pp. 95–102, Jul. 2018.

[3] S. Chow and M. E. Peck, "The bitcoin mines of China," *IEEE Spectrum*, vol. 54, no. 10, pp. 46–53, Oct. 2017.

[4] N. Arnosti and S. M. Weinberg, "Bitcoin: A natural oligopoly," Nov. 2018, arXiv:1811.08572 [cs.CR].

[5] M. R. Baye, D. Kovenock, and C. G. de Vries, "The all-pay auction with complete information," *Economic Theory*, vol. 8, no. 2, pp. 291–305, Jun. 1996.

[6] A. L. Hillman and J. G. Riley, "Politically contestable rents and transfers," *Economics & Politics*, vol. 1, no. 1, pp. 17–39, Mar. 1989.

[7] E. Dechenaux, D. Kovenock, and R. M. Sheremeta, "A survey of experimental research on contests, all-pay auctions and tournaments," *Experimental Economics*, vol. 18, no. 4, pp. 609–669, Dec. 2015.

[8] G. V. Ranade and L. R. Varshney, "The role of information patterns in designing crowdsourcing contests," in *Creating and Capturing Value through Crowdsourcing*, C. L. Tucci, A. Afuah, and G. Viscusi, Eds. Oxford, UK: Oxford University Press, 2018, pp. 154–177.

[9] L. R. Varshney, J. B. Rhim, K. R. Varshney, and V. K. Goyal, "Categorical decision making by people, committees, and crowds," in *Proceedings of the 2011 Information Theory and Applications Workshop*, Feb. 2011.

[10] K. J. Boudreau, N. Lacetera, and K. R. Lakhani, "Incentives and problem uncertainty in innovation contests: An empirical analysis," *Management Science*, vol. 57, no. 5, pp. 843–863, May 2011.

[11] C. Noussair and J. Silver, "Behavior in all-pay auctions with incomplete information," *Games and Economic Behavior*, vol. 55, no. 1, pp. 189–206, Apr. 2006.

[12] S. P. Anderson, J. K. Goeree, and C. A. Holt, "Rent seeking with bounded rationality: An analysis of the all-pay auction," *Journal of Political Economy*, vol. 106, no. 4, pp. 828–853, Aug. 1998.

[13] S. Azouvi and A. Hicks, "SoK: Tools for game theoretic models of security for cryptocurrencies," May 2019, arXiv:1905.08595 [cs.CR].

[14] E. Budish, "The economic limits of bitcoin and the blockchain," Jun. 2018.

[15] G. Huberman, J. D. Leshno, and C. Moallemi, "An economist's perspective on the bitcoin payment system," *AEA Papers and Proceedings*, vol. 109, pp. 93–96, May 2019.

[16] N. Leonardos, S. Leonardos, and G. Piliouras, "Oceanic games: Centralization risks and incentives in blockchain mining," Apr. 2019, arXiv:1904.02368 [cs.GT].

[17] Z. Liu, N. C. Luong, W. Wang, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "A survey on applications of game theory in blockchain," Feb. 2019, arXiv:1902.10865 [cs.GT].

[18] J. Sun, P. Tang, and Y. Zeng, "Games of miners," in *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2020)*, May 2020, pp. 1323–1331.

[19] N. Dimitri, "Bitcoin mining as a contest," *Ledger*, vol. 2, pp. 31–37, 2017.

[20] J. Chiu and T. V. Koeppl, "Incentive compatibility on the blockchain," Bank of Canada, Staff Working Paper 2018-34, Jul. 2018.

[21] E. S. Pagnotta, "Bitcoin as decentralized money: Prices, mining, and network security," Dec. 2018.

[22] K. E. Train, *Discrete Choice Methods with Simulation*, 2nd ed. Cambridge University Press, 2009.

[23] R. D. McKelvey and T. R. Palfrey, "Quantal response equilibria for normal form games," *Games and Economic Behavior*, vol. 10, no. 1, pp. 6–38, Jul. 1995.

[24] V. Bagaria, S. Kannan, D. Tse, G. Fanti, and P. Viswanath, "Deconstructing the blockchain to approach physical limits," Oct. 2018, arXiv:1810.08092 [cs.CR].

[25] M. Vilim, H. Duwe, and R. Kumar, "Approximate bitcoin mining," in *Proceedings of the 53rd Design Automation Conference (DAC '16)*, Jun. 2016, pp. 97:1–97:6.

[26] "Difficulty: A Bitcoin Wiki Page," https://en.bitcoin.it/wiki/Difficulty, 2019, [Online; Accessed 12-Mar-2021; Last Updated 17-Dec-2019].

[27] O. Delgado-Mohatar, M. Felis-Rota, and C. Fernández-Herraiz, "The bitcoin mining breakdown: Is mining still profitable?" *Economics Letters*, vol. 184, p. 108492, 2019.