On Game-theoretic Computation Power Diversification in the Bitcoin Mining Network

Suhan Jiang and Jie Wu
Department of Computer and Information Sciences, Temple University
{suhan.jiang, jiewu}@temple.edu

***Abstract*—In the Bitcoin mining network, miners contribute computation power to solve crypto-puzzles in exchange for financial rewards. Due to the randomness and the competitiveness of mining, individual miners tend to join mining pools for low risks and steady incomes. Usually, a pool is managed by its central operator, who charges fees for providing risk-sharing services. This paper presents a hierarchical distributed computation paradigm where miners can distribute their power among multiple pools. By adding virtual pools, we separate miners' dual roles of being the operator as well as being the member when solo mining. We formulate a multi-leader multi-follower Stackelberg game to study the joint utility maximization of pool operators and miners, thereby addressing a computation power allocation problem. We investigate two practical pool operation modes, a uniform-share-difficulty mode and a nonuniform-share-difficulty mode. We derive analytical results for the Stackelberg equilibrium of the game under both modes, based on which optimal strategies are designed for all operators and miners. Numerical evaluations are presented to verify the proposed model.**

***Index Terms*—Bitcoin mining pool, reward variance, risk aversion, Stackelberg game.**

I. Introduction

Bitcoin [1] blockchain is maintained by a network of miners through a block-appending process called mining. Miners have to solve a computationally-difficult crypto-puzzle before adding a block to the blockchain. Block generators will receive monetary rewards as a mining incentive. The system periodically adjusts the difficulty of crypto-puzzles so that the probability of a miner generating a block depends on the ratio between its own computation power and the network-wide computation power. Plenty of computation power has been dedicated in mining, causing the probability extremely small for individuals, especially small miners, to find a block in a short time. Thus, most miners join mining pools, where they aggregate their computation power and cooperate in mining. If a member finds a block, the obtained block reward will be shared among all members. Pooled mining increases the chance of being awarded and effectively reduces the *variance* in the reward for individual miners. In reality, pooled mining has dominated the Bitcoin mining network, occupying more than 90% of the total computation power. There exist more than 20 Bitcoin mining pools, among which *F2pool*, *Antpool*, *Poolin*, *BTCpool*, and *Slushpool* [2–6] are the most popular.

This research was supported in part by NSF grants CNS 2128378, CNS 2107014, CNS 1824440, CNS 1828363, CNS 1757533, CNS 1629746, and CNS 1651947.

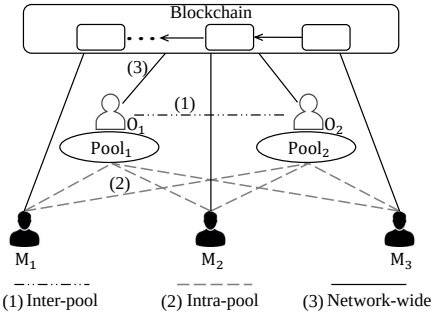


Fig. 1: Three competitions in the Bitcoin mining network: (1) inter-pool game where all pool operators compete with each other to attract miners, (2) intra-pool game where all pool members compete for pool rewards, and (3) network-wide game among all solo power and pooled power.

Usually, a pool is managed by a trusted operator who is responsible for identifying members' contributions and distributing rewards accordingly. The operator charges service fees by cutting a fixed percentage of a block reward. To verify each member's contribution, an operator will ask its members to solve sub-puzzles which are easier than the block puzzle. Each solution to a sub-puzzle, called a *share*, has a probability of being a valid solution to the block puzzle. Then, pool members show how much effort they have put into the pool by submitting shares. Based on the number of submitted shares, the operator can fairly distribute pool rewards. The pool operator should carefully decide its share difficulty, *i.e.*, the sub-puzzles' difficulty, as this value affects its own service cost as well as its member's benefits. Actually, miners can apply either solo mining or pooled mining, or even both. Pooled mining offers miners a steadier income but a lower long-term revenue due to the service fees. Solo mining charges no extra cost, but miners suffer from a high uncertainty of the reward. This *return-risk* tradeoff challenges each miner to determine a suitable ratio between solo mining and pooled mining to maximize its income at a steady rate. Meanwhile, rather than selecting a certain pool, a miner can join multiple pools. How a miner diversifies its computation power across different pools is also a non-trivial problem given the diversity of pools. Pools differ in their mechanisms, such as their service fees and reward distribution methods, incurring different expectations and variances of members' incomes.

In this paper, we present a hierarchical distributed computation paradigm consisting of multiple mining pools and a set of miners in the Bitcoin mining network. Miners can distribute their computation power to multiple pools as well as solo

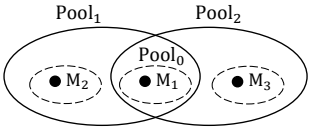


Fig. 2: A special configuration of Fig. 1, where all miners contribute partial power to solo mining and meanwhile, M_2 joins Pool_1, M_3 joins Pool_2, and M_1 joins both. Eclipses and circles represent pools and miners, respectively.

mining. As depicted in Fig. 1, three types of competitions exist in the proposed paradigm: (1) all operators form an *inter-pool* competition, each aiming to attract more computation power by designing a reasonable pool mechanism; (2) members in the same pool form an *intra-pool* competition, each aiming to maximize its utility by devoting a reasonable computation power to the pool; and (3) all individual mining power, including all solo-mining entities and pools, forms a *network-wide* mining competition. Obviously, the intricate relationship between pools and miners makes it hard to achieve the joint utility maximization for both sides.

We exploit game theory to analyze the complex interactions among pools and miners, who aim to maximize their own utilities. We propose a multi-leader multi-follower Stackelberg game to study the pool-mechanism-based power allocation problem. Such a clear two-level game model is simplified from the above-mentioned three-type-competition model by adding virtual players in the leader level. That is, for each miner, we add a virtual pool to recruit its solo-mining power. Each virtual pool is assumed to be exclusively open to its designated miner and charge no fees. Fig. 2 shows a possible miner allocation profile of Fig. 1. From the *global* view, there are five pools in total, two of which are real pools (solid eclipses) and the remaining three (dashed eclipses) are virtual pools. From a miner's *local* view, it faces three pools, among which one is private and another two are public. For example, M_1 can join Pool_0, Pool_1, and Pool_2. By adding virtual pools, we can separate a miner's dual roles of being an operator as well as being a member when he mines solo.

Our proposed game includes two subgames for the pools (as leaders) and the miners (as followers), respectively. In the leader subgame, each operator has a privilege to set its pool mechanism by anticipating miners' responses. In the follower subgame, miners decide their allocation strategies based on their own computation power and the observed pool mechanisms. We model miners as risk-averse players so that it is easy for them to join multiple pools. Thus, each miner is tagged with a risk-tolerance level, which affects its utility seriously. We analyze the equilibrium in the proposed Stackelberg game under two different pool operation modes, *i.e.*, operators adopting a uniform/non-uniform share difficulty among its members. By achieving the corresponding Stackelberg equilibrium, we design the optimal strategies for both operators and miners. The major contributions of this paper are as follows:

- We propose a Stackelberg game to study a pool-mechanism-based power allocation problem in a hierarchical mining network under two different pool operation modes.
- We characterize miners as risk-averse players and propose a variance-involved power function to reflect their utilities.
- We design optimal strategies for operators and miners by finding the Stackelberg equilibrium in the proposed game.
- We consider a special homogeneous-miner case and derive explicit-form expressions of the optimal strategies for both operators and miners in the uniform-share-difficulty mode.
- We perform numerical evaluations and conduct real-data experiments to confirm all the theoretical results.

II. System Mode

A. Preliminary

Bitcoin mining is a process where all miners are asked to solve a crypto-puzzle in order to create a block. Each block generator is rewarded with bitcoins in an amount we will denote R. The occurrence of finding a block can be well approximated by a random variable following a Poission process. The system adjusts the network difficulty, denoted D, so that the whole network finds a block every $T = 600s$ on average. D represents the difficulty of a crypto-puzzle (its current value is 1.6×10^{13}), which is related to the network-wide hash rate, *i.e.*, a measurement of computation power. Currently, its value is 1.3×10^{20} h/s, meaning that the number of hash the Bitcoin mining network can compute is 1.3×10^{20} per second for solving a crypto-puzzle. The target value is chosen so that every computed hash will lead to a valid block with a probability of $1/(2^{32}D)$.

B. A Hierarchical Bitcoin Mining Network

This paper focused on a hierarchical Bitcoin mining network. Corresponding notations are listed in Table I. We consider a set of M miners, and N public mining pools. Fig. 1 depicts an overview of this network. Operators are responsible for providing risk-sharing services to miners and make money by taking a cut from pool rewards. Operators decide their pool mechanisms individually, which will affect their own profits as well as members' benefits. Based on pools' mechanisms, each miner further determines how to allocate its power to different pools. Then, the whole network is involved in a series of repetitive block-appending competitions. Each of them aims for a maximum of utility (details are given in Section III), incurring a chain of non-trivial and intricate competitions due to the specificity of mining.

C. Strategy Space

1) Miners' Decisions: As is shown in Fig. 2, miner M_j of computation power h_j in total is facing $N+1$ pools: N mining pools and solo mining. Thus, M_j needs to determine a vector $\mathbf{m}_j = (\beta_j^0, \beta_j^1, \cdots, \beta_j^N)$ to show its power allocation decision, where $0 \leq \beta_j^i < 1$ and $\sum_{i=0}^{N} \beta_j^i = 1$ for for each $i \in [0, N]$. (Note that, the index 0 is used to represent solo mining.) The miner moves by choosing a $\mathbf{m_j}$ for one mining round. For simplicity, we assume that $\mathbf{m_j}$ remains constant for all miners in one mining round.

TABLE I: Summary of Notations.

Symbol	Description
D / R / T	blockchain mining difficulty / reward / interval
C	Constant of value $T/2^{32}$
N / M	number of pool operators / miners
O_i / M_j	the i-th operator, $i \in [0, N]$ / the j-th miner, $j \in [1, M]$
e	communication expense between any operator-miner pair
b_i	O_i's budget
d_i / f_i	Pool$_i$'s share difficulty / reward cutting rate
$\mathbf{o}_i = (d_i, f_i)$	O_i's strategy vector
$\mathbf{o}_{-i}$ / $\mathbf{o}$	all operators' except O_i's / all operators' strategy profile
α_j	M_j's risk tolerance level, $\alpha_j \in (0, 1)$
h_j /H_i /H	M_j's / O_i's /network-wide computation power
β_j^i	M_j's power allocation ratio in Pool$_i$, $\beta_j^i \in [0, 1]$
$\mathbf{m}_j = (\beta_j^i)$	M_j's power allocation vector
$\mathbf{m}_{-j}$ / $\mathbf{m}$	all miners except M_j's / all miners' allocation profile

2) Operators' Decisions: As we mentioned before, pools apply a share-based method to identify each member's computation power contribution. Thus, operator O_i assigns sub-puzzles with a share difficulty d_i ($d_i < D$) to its members. Each share has a probability d_i/D of being a valid solution to a new block. Shares do not have any real value other than acting as the main reference when distributing the reward. When a member in a pool finds a share that is also a valid block solution, the pool operator submits it to the blockchain and distributes the obtained reward to all pool members according to the number of shares they have submitted. Usually, O_i takes a fixed percentage cut f_i of the block reward as its service fees before distribution. Thus, operator O_i should make decisions on the share difficulty d_i, and how much its service fee should be, in the form of a reward cutting rate f_i. Each decision is vital and non-trivial. A big d_i makes it hard to truly reflect how much work a member has performed, especially for those small members, therefore deterring them from joining the pool. However, a small d_i means O_i has to communicate with each member frequently for share submissions, which inevitably brings extra communication costs. Similarly, a high f_i definitely hurts its members' incomes while a low f_i may not be able to cover its own cost of service.

There are multiple ways to design a fair reward distribution method in pooled mining, such as Pay-Per-Share and Pay-Per-Last-N-Shares. All of them aim to distribute the reward proportional to each member's computation power contribution. This paper does not focus on solving a fair reward distribution problem. For simplicity, the remaining reward $R(1 - f_i)$ will be distributed in proportion to the number of shares members submitted during that mining round. Thus, O_i's strategy vector can be expressed in the form of $\mathbf{o_i} = (d_i, f_i)$.

III. Problem Formulation

A. Operator-Miner Interaction: A Stackelberg Game

The interactions among operators and miners can be characterized as a leader-follower Stackelberg game by adding virtual players in the leader level. We add M more virtual pools, each exclusively accessible by a designated miner. Thus, a miner's power for solo mining can be considered as power contribution to its corresponding virtual pool. From the global view, there are $N + M$ pools in the mining network. From a miner's local view, it faces $N + 1$ pools, indexed from 0 to N, where Pool$_0$ is its corresponding virtual pool. In the proposed game, operators act as leaders and move first to reveal their own decisions by perceiving miners' reactions, and then followers, *i.e.*, miners, respond with their power allocations. It is a multi-leader multi-follower Stackelberg game, two levels of which can be described as follows. In the first level, the competition among pools forms a non-cooperative leader subgame, where each operator O_i optimizes its strategy vector $\mathbf{o_i} = (d_i, f_i)$ by predicting the miners' reactions as well as considering other operators' strategies. In the second level, each miner M_j, responds to the current pool mechanisms, by distributing its power to the target pools, considering its total computation power h_j and allocations of other miners'. Since decisions are made for individual utility maximization, a non-cooperative follower subgame is also formed.

B. Miner-side Problem

All miners are assumed to be risk-averse players, each aiming to optimally create a mining portfolio that maximizes its risk-adjusted rewards. Thus, we characterize a miner M_j by its total computation power h_j and its risk tolerance level α_j where $\alpha_j \in (0, 1)$. We denote u_j^i as M_j's expected utility in Pool$_i$, which can be expressed as Eq. (1):

$$u_j^i = Pr_i \cdot \left(p_j^i\right)^{\alpha_j}, \tag{1}$$

where Pr_i represents the probability of Pool$_i$ finding a block, and p_j^i represents the payoff M_j can obtain when Pool$_i$ successfully finds a block. Due to M_j's risk-averse nature, p_j^i is discounted by its risk tolerance level α_j. Besides, the expression of p_j^i is defined as below:

$$p_j^i = r_j^i - c_j^i - v_j^i, \tag{2}$$

where r_j^i, c_j^i, and v_j^i represent M_j's reward, cost, and variance in Pool$_i$. These three parameters are related to the amount of computation power M_j contributes to Pool$_i$. We show their accurate definitions in Section IV. Note that, our novelty lies in that we take the reward variance in the pool as a negative factor for miners' pooled mining payoff. This is reasonable and necessary since the main purpose for M_j to join mining pools is to reduce the high variance of its reward at the cost of losing partial profits. It is obvious that when choosing from two pools of identical rewards and costs, risk-averse miner M_j should select the pool offering a lower reward variance. Thus, U_j, which represents M_j's total expected utility, can be easily obtained as a sum over u_j^i, *i.e.*, $U_j = \sum_{i=0}^{N} u_j^i$. We define M_i's optimization problem below.

Problem 1 (OP$_{\text{MINER}}$)**.**

$$\textit{maximize} \qquad U_j = \sum\nolimits_{i=0}^{N} u_j^i, \tag{3a}$$

$$\textit{subject to} \qquad 0 \le \beta_j^i < 1, \quad \sum\nolimits_{i=0}^{N} \beta_j^i = 1, \tag{3b}$$

where (β_j^i) is a decision vector that represents M_j's power allocation ratios.

C. Operator-side Problem

Operator O_i's utility, denoted V_i, is defined as $V_i = \bar{r_i} - \bar{c_i}$, where $\bar{r_i}$ is its expected payoff and $\bar{c_i}$ is its cost. $\bar{r_i}$ is the product of Pr_i and O_i's service fees, and $\bar{c_i}$ is yielded due to the operator-member share submission communication cost. Thus, operator 0_i's optimization problem is defined in Eq.(4).

Problem 2 ($\text{OP}_{\text{OPERATOR}}$).

$$\textit{maximize} \qquad V_i = \bar{r_i} - \bar{c_i}, \tag{4a}$$

$$\textit{where} \qquad \bar{c_i} \leq b_i, \tag{4b}$$

where b_i represents O_i's budget constraint.

D. Operator-Miner Stackelberg Game

$\text{OP}_{\text{OPERATOR}}$ and OP_{MINER} together form the proposed Stackelberg game. To achieve equilibrium in this game, where neither the leaders (operators) nor the followers (miners) have incentives to deviate, we need to find its subgame perfect Nash equilibria (NE) in both the leader stage and the follower stage, by applying backward induction. Formally, the SE point(s) is defined as follows.

Definition 1. *Let $\mathbf{m}$ and $\mathbf{o}$ denote the optimal power allocation vector of all miners and the optimal strategy vector of operators, respectively. Let $(\mathbf{m}_j)_{j=1}^{M} = \mathbf{m}$ and $(\mathbf{o}_i)_{i=1}^{N} = \mathbf{o}$, then the point $(\mathbf{m}^*, \mathbf{o}^*)$ is the Stackelberg equilibrium if the following conditions hold:*

$$V_i(\mathbf{m}^*, \mathbf{o}^*) \geq V_i(\mathbf{m}, \mathbf{o}), \forall i, \tag{5a}$$

$$U_j(\mathbf{m}^*, \mathbf{o}^*) \geq U_j(\mathbf{m}, \mathbf{o}), \forall j. \tag{5b}$$

IV. Utility Analysis

A. Miner-side Utility

1) Probability: Pr_i represents the probability of Pool_i finding a block. Let H_i represent Pool_i's computation power and let H represent the network computation power, $Pr_i = H_i/H$ can be easily obtained.

2) A Variance-involved Payoff: p_j^i is the payoff M_j obtains from Pool_i when Pool_i successfully finds a block. r_j^i is related to M_j's power contribution to Pool_i as well as Pool_i's service fees. After O_i takes a fee of Rf_i, the remaining part will be shared among all members. As M_j's computation power to Pool_i is $h_j^i = h_j\beta_j^i$, M_j should receive a reward h_j^i/H_i times the remaining reward. Thus, $r_j^i = R(1-f_i)\cdot\left(h_j^i/H_i\right)$. Specifically, $r_j^0 = R$ given the virtual Pool_0 charges zero fee and is only open to M_j (*i.e.*, $H_0 = h_j^0$). c_j^i is the communication cost for share submission. Denote e as the communication cost used to submit a share. Then, c_j^i the product of e and the number of shares that it submits. During a mining round, M_j calculates a total of $h_j^i T$ hash values. Given Pool_i's share difficulty d_i, each of its computed hash has a probability $\xi = 1/(2^{32}d_i)$ of being a share specific to Pool_i. (Note that, a share for Pool_i won't be accepted by another Pool_j, as each pool's sub-puzzle is unique.) So M_j will find $\lambda = h_j^i T\xi$ shares on average. Obviously, the total cost is $c_j^i = eh_j^i T\xi$. Specifically, $c_j^i = 0$ since solo mining has no communication cost on share submission.

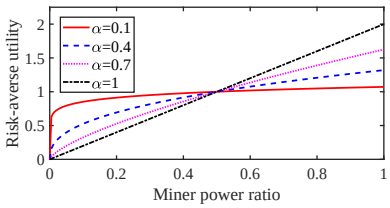


Fig. 3: Utility is affected by the risk tolerance level.

Similar to block mining, finding shares with a constant computation power h_j^i is a Poisson process with $h_j^i\xi$ as the rate parameter. We said that mining for time T results in λ shares on average. We can say further that the number of shares found follows the Poisson distribution with a parameter λ, so this quantity is also the variance of the number of shares found. The share difficulty d_i indicates that each share has a probability d_i/D to be a valid block puzzle solution. Thus, the variance of the reward is then $(Rd_i/D)^2\,\lambda$. We define v_j^0 as 0. This is because the purpose of pooled mining is to lower the reward variance, which doesn't apply to solo mining.

3) Risk Tolerance Level: α_j measures how much risk M_j is prepared to take in pursuit of its objective. The higher α_j is, the more risk M_j is willing to take. Note that, α=1 is a special case, indicating a risk-neutral miner. As is shown in Eq. (1), to capture the risk-averse nature of miners, we apply a power function to characterize their utilities. To show the risk tolerance level's effects on a power-function-based utility, Fig. 3 gives an example where the corresponding risk-neutral utility function is linear.

B. Operator-side Utility

As Pool_i's expected reward is RH_i/H in each mining round, O_i's payoff, namely its expected service fees, is $\bar{r_i} = f_i RH_i/H$ given its reward cutting rate f_i. In a mining round, all members can compute $H_i T$ hash, and each computed hash has a probability ξ of being a share. Thus, $H_i T\xi$ shares should be submitted to O_i on average, leading to a total cost of $\bar{c_i} = eH_i T\xi$. Note that, a virtual pool operator has a fixed strategy of $(D, 0)$, and thereby yielding a payoff of $\bar{r_i} = 0$ as well as a cost of $\bar{c_i} = 0$.

V. Optimal Strategy in Operator-Miner Game

In this section, we analyze the equilibrium in our proposed game, based on which we can design the optimal strategies for operators and miners. To find the leader-follower Stackelberg equilibrium, we have to figure out the Nash equilibrium in both the leader game and the follower game, where the concavity of $\text{OP}_{\text{OPERATOR}}$ and OP_{MINER} should be confirmed. Combining Lemma 1 and Theorem 1, we prove the Nash equilibrium in the follower game. Then we apply backward induction and prove the Nash equilibrium in the leader game based on Lemma 2 and Theorem 2. Corollary 1 and Corollary 2 show the explicit expressions of homogeneous-miner-homogeneous-operator optimal strategies. Furthermore, we investigate another pool operation mode of a non-uniform share difficulty. We also prove that the equilibrium is still achievable in this mode. All complex proofs are provided in Appendix.

A. Equilibrium Analysis

Lemma 1. *u_j^i is a strictly concave for $\forall i \in [1, N]$.*

Theorem 1. *A Nash equilibrium exists among all miners if all operators' strategies are fixed.*

Proof. Firstly, the strategy space for each miner, $\left\{\beta_j^i | \beta_j^i \in [0,1], \forall i \in [0,N]\right\} \cap \left\{\sum_{i=0}^{N} \beta_j^i = 1\right\}$, is a non-empty, convex, compact subset of the Euclidean space. Furthermore, we know U_j^i is apparently continuous in this space. To show the existence of Nash equilibrium among all miners, we need to prove U_j^i is strictly concave. Obviously, u_j^0 is a linear function, which is definitely concave. In Lemma 1, we have proved that u_j^i is a strictly concave for $\forall i \in [1, N]$. Given the fact that the sum of functions can be strictly concave as long as all addends are concave and at least one of them is strictly concave, we could conclude that U_j^i is strictly concave. □

Therefore, we can apply a classic best-response-dynamics algorithm [7] to obtain the Nash equilibrium of the multiple-player non-cooperative game among all miners. The equilibrium doesn't accomplish in one step in reality. Miners have to go through several iterations to update their strategies, then reach a steady point as it is impossible for each miner to know other miners' strategies before it makes decisions. Complete information is a assumption in the game theory but not realistic in practice. We give a simple example involving two pools and three miners with different risk tolerance levels and computation power, to show how miners' strategies evolve before remaining unchanged. As Table II shows, miners' strategies fix after several iterations. We also choose another starting point, *i.e.*, $\mathbf{m}_1 = (0, 0.5, 0.5)$, $\mathbf{m}_2 = (0.5, 0.3, 0.2)$, and $\mathbf{m}_3 = (0.15, 0.37, 0.48)$. In this case, miners' strategies converge to the same final values after 17 rounds.

Corollary 1. *Assume that all miners share an identical risk tolerance level, denoting α and all operators' budgets are unlimited, i.e., $b_i = +\infty, \forall i$. Each miner's optimal strategy can be explicitly as follows:*

$$h_j \beta_j^i = \begin{cases} h_j - \sum_{i=1}^{N} h_j \beta_j^i & i = 0, \\ \left[(M-1)\,\alpha + 1\right] / \left[(M\alpha + 1)\, M\right] \cdot (x/y) & else, \end{cases} \tag{6}$$

where $x = R(1 - f_i)$ and $y = eC/d_i - R^2 d_i C/D^2$. (Note that, x and y are just intermediate variables which are introduced for the simplicity of writing.)

The explicit expressions in such a special case allow us to intuitively observe that a pool's share difficulty and reward cutting rate vitally influence miners' decisions. In the following, we will discuss how each pool operator optimizes these two factors, based on the method of backward induction.

Lemma 2. *V_i is a strictly concave for $\forall i \in [1, N]$.*

Theorem 2. *Assuming e is less than 1 bitcoin (this is definitely holds in reality), each pool operator O_i can achieve utility maximization by optimally setting its share difficulty d_i and reward cutting rate f_i . That is, a Nash equilibrium exists among all operators.*

		Round					
Ratio	Init.	1	2	3	4	5	6-14
β_1^0	0.34	0.566	0.559	0.569	0.570	0.571	**0.572**
β_1^1	0.33	0.058	0.104	0.103	0.102	0.102	**0.102**
β_1^2	0.33	0.376	0.337	0.328	0.328	0.327	**0.326**
β_2^0	0.25	0.649	0.682	0.695	0.703	0.714	**0.732**
β_2^1	0.25	0.089	0.051	0.031	0.013	0.005	**0.000**
β_2^2	0.50	0.262	0.267	0.274	0.284	0.281	**0.268**
β_3^0	0.50	0.739	0.741	0.742	0.753	0.762	**0.781**
β_3^1	0.25	0.204	0.232	0.237	0.230	0.227	**0.219**
β_3^2	0.25	0.057	0.027	0.021	0.017	0.011	**0.000**

TABLE II: Miner strategy iterations in a setting of $(\alpha_1, \alpha_2, \alpha_3) = (0.1, 0.4, 0.9)$, $h_1 : h_2 : h_3 = 3 : 7 : 10$, $\mathbf{o}_1 = (40, 0.01)$, $\mathbf{o}_2 = (8, 0.09)$, and $(R, D, e) = \left(12.5, 800, 2^{-8}\right)$.

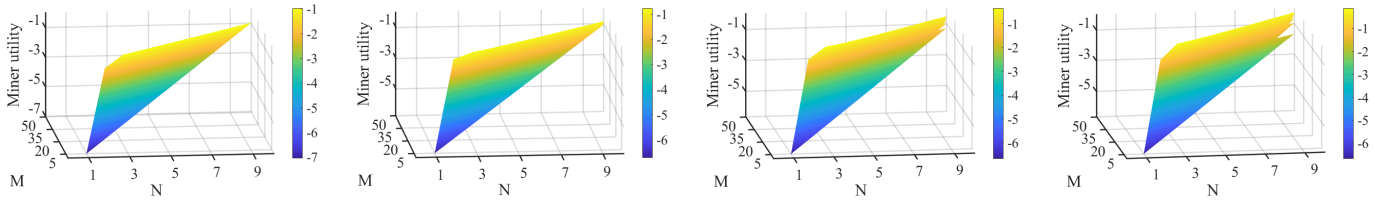
Proof. Based on the Nash equilibrium achieved among all miners, each pool operator O_i can optimize its strategies to achieve profit (defined in Problem 2) maximization. Obviously, O_i's strategy space, $\{0 < f_i < 1\} \cap \{d_i \geq 1\}$, is a non-empty, convex, compact subset of the Euclidean space. Further, we know V_i is apparently continuous over the domain of $\{0 < f_i < 1\} \cap \{d_i \geq 1\}$. Since we have proved that V_i is a strictly concave function in its domain in Lemma 2, we can confirm the existence of operators' Nash equilibrium. □

Corollary 2. *Assume that all miners share an identical risk tolerance level α and and all operators' budgets are unlimited, i.e., $b_i = +\infty, \forall i$.. Each operator's optimal strategy can be explicitly as follows: $f_i^* = (xd_i + y) / (2xd_i)$ and $d_i^* = d_i$ so that f_i^* and d_i^* satisfy $2yzd + xyf_i = xzf_i d_i^2$, where $x = R/H$, $y = eC$, and $z = R^2 C/D^2$. (Note that, x, y and z are just intermediate variables which are introduced for the simplicity of writing.)*

B. Non-Uniform-Share-Difficulty Pool Operation Mode

Previously, we focus on a pool operation mode where an operator sets a uniform share difficulty for its members. In this case, members devoting different computation power will have different numbers of submitted shares on average, and each share has the same worth. Now, we move on to another practical mode where a mining pool supports a non-uniform share difficulty for its members. In this mode, all members will submit shares in similar paces, indicating that they have an identical communication cost in expectation. However, the share difficulty is variable among members to match their own computation power, and each member's share is priced proportional to its share difficulty.

1) A Pool Member's Reward, Cost and Variance: In the non-uniform-share-difficulty mode, the reward cutting rate f_i is still in operator O_i's decision vector. Instead of deciding the share difficulty d_i, O_i determines share collecting speed s_i, so that each of its members will submit $s_i T$ shares in expectation during a mining round. The corresponding communication cost is $c_j^i = e s_i T$. When Pool_i with a total computation power of



(a) $\alpha = 0.01$. (b) $\alpha = 0.1$. (c) $\alpha = 0.4$. (d) $\alpha = 0.8$.

Fig. 4: Homogeneous miners' utility under different risk tolerance levels.

Power ratio	SN	SA	MR	MNO	MAO
0.05	0.5482	0.5477	0.5578	0.5890	*0.5719*
0.10	1.0982	1.0964	1.1773	1.1780	*1.1757*
0.15	1.6446	1.6446	1.7334	1.7670	*1.8007*
0.20	2.1954	2.1929	2.3451	2.3560	*2.4257*
0.25	2.7411	2.7501	2.8068	2.9449	*3.0507*

TABLE III: Miner's average income under different investment methods.

Power ratio	SN	SA	MR	MNO	MAO
0.05	560	562	147	123	*99*
0.10	378	391	108	115	*97*
0.15	282	282	110	107	*94*
0.20	180	185	111	105	*92*
0.25	128	123	102	101	*90*

TABLE IV: Miner's variance under different investment methods.

H_i finds a block,, miner M_j with computation power $h_j\beta_j^i$ will receive a reward of $p_j^i = R(1-f_i)\cdot(h_j\beta_j^i/H_i)$. M_j's share difficulty can be expressed as $d_j^i = h_j\beta_j^i/(2^{32}s_i)$, indicating its share is worth a value of Rd_j^i/D. Thus, its reward variance can be calculated as $v_j^i = s_iT\cdot\left(Rd_j^i/D\right)^2 = R^2d_j^ih_j\beta_j^iC/D^2$. Obviously, a big s_i leads to a low variance for members.

2) Utility Reformulation and Equilibrium Analysis: M_j's utility U_i can be easily reformulated by applying these updated p_j^i, c_j^i, and v_j^i in the non-uniform-share-difficulty mode. Now, we rewrite the utility function for each pool operator in this mode. O_i's expected payoff $\bar{r_i}$ remains unchanged, while its communication cost should change into $\bar{c_i} = Ns_ieT$. Thus, V_i in this mode can be easily updated accordingly.

Theorem 3. *When operators' strategies are fixed, there exists a Nash equilibrium among all miners.*

Theorem 4. *Assuming e is less than 1 bitcoin, each pool operator O_i can achieve utility maximization by optimally setting its share collecting speed s_i and reward cutting rate f_i . That is, a Nash equilibrium exists among all operators.*

Theorem 3 and Theorem 4 are obtained in the non-uniform-share-difficulty mode. Due to the page limitation, we cannot prove them in details, while their proofs are similar to the proofs of Theorem 1 and Theorem 2. Therefore, the Stackelberg equilibrium also can be achieved in the non-uniform-share-difficulty mode.

VI. Evaluation

Our evaluation includes two main parts. First, we examine how miners (Subsection VII.A) and operators (Subsection VII.B) decide their optimal strategies. Second, we analyze how Bitcoin market price influences the achieved equilibrium (Subsection VII.C).

A. Miner-side Equilibrium Analysis

1) Miner Payoff under Different Investment Methods: Since we highlight our novel miner utility function, to show its advantage, we will compare it with some existing works that use different utility functions to guide miners on how to select mining pools. All methods to be compared are explained as follows. (1) SN: each miner is risk-neutral and is allowed to either mine solo or join in a single mining pool for utility maximization; (2) SA: it is slightly different from SN in that all miners are risk-averse; (3) MR: each miner randomly allocates its computation power to multiple pools and solo mining; (4) MNO: each miner is risk-neutral and optimizes its utility by power diversification among multiple pools and solo mining; and (5) MAO: it is our proposed method, which differs from MNO on the risk-averse-miner assumption. We assume there are 3 pool operators with fixed strategies and 20 miners of different computation powers. All miners share the same risk attitudes, *i.e.*, $\alpha = 1$ in the risk-neutral setting and $\alpha = 0.5$ in the risk-averse setting. We specify power ratios for 5 miners as 0.05, 0.1, 0.15, 0.2, and 0.25 and randomly assign the remaining power to another 15 miners. We model the mining process during which 1000 blocks are generated. We treat every 20 blocks as a period and record the income for those 5 miners. Then, we calculate their average income over 50 periods. The results generated from different methods are given in Table III. As we only run 1000 mining rounds, our results show that for each size of miners, diversifying power always works better than concentrating it. Although it should not be such a case in the long run, this observation reflects the randomness of mining itself and that multiple-pool investment gives miners more income sources. We also record how often they get paid for those 5 miners in Table IV, which reflects the reward variance of each miner. Obviously, miners applying our utility function can obtain incomes more frequently. We can conclude that our proposed method brings miners steady and relatively high incomes.

2) Factors Affects Miner's Utilities: Now, we investigate some factors that influence each miner's utility. We consider two personal reasons, *i.e.*, a miner's computation power as well as its risk tolerance level, and one external reason, *i.e.*, the number of pools for miners to join in. We assume that there are 20 miners with different computation powers and an identical risk tolerance level. From Fig. 4, we can conclude: (1) a miner with more computation power definitely has higher

TABLE V: Miners' strategy profiles under different risk tolerance levels

	M_1			M_2			M_3		
$\alpha_1 : \alpha_2 : \alpha_3$	β_1^0	β_1^1	β_1^2	β_2^0	β_2^1	β_2^2	β_3^0	β_3^1	β_3^2
.01 : 0.3 : 0.7	0	0.50	0.50	0.38	0.31	0.31	0.70	0.15	0.15
0.1 : 0.4 : 0.9	0	0.50	0.50	0.50	0.25	0.25	0.90	0.05	0.05
0.3 : 0.5 : 0.8	0	0.50	0.50	0.52	0.24	0.24	0.80	0.10	0.10

(a) Miner power ratio $h_1 : h_2 : h_3 = 3 : 7 : 10$.

	M_1			M_2			M_3		
$\alpha_1 : \alpha_2 : \alpha_3$	β_1^0	β_1^1	β_1^2	β_2^0	β_2^1	β_2^2	β_3^0	β_3^1	β_3^2
.01 : 0.3 : 0.7	0.02	0.49	0.49	0.18	0.41	0.41	0.5	0.25	0.25
0.1 : 0.4 : 0.9	0.28	0.36	0.36	0.40	0.30	0.30	0.86	0.07	0.07
0.3 : 0.5 : 0.8	0.32	0.34	0.34	0.42	0.29	0.29	0.70	0.15	0.15

(b) Miner power ratio $h_1 : h_2 : h_3 = 1 : 1 : 1$.

utility compared with a miner of less computation power; (2) more mining pools gives more income sources for miners and incurs a higher utility as well; and (3) a miner's risk tolerance level also affects its utility. In Table V, we show optimal strategy profiles for miners with heterogeneous risk tolerance levels under different power ratio settings.

B. Operator-side Equilibrium Analysis

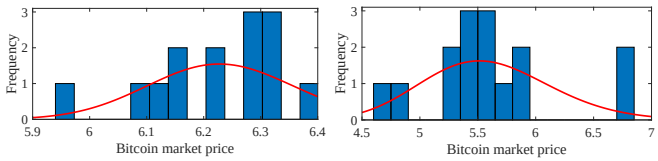
This part investigates two different ways that an operator can determine its share difficulty. The first is the one we apply in our previous analysis, where the operator sets an identical share difficulty for its members. In this case, members devoting different computation power will have different average numbers of submitted shares and each share has the same worth. Another way is a variable share difficulty where each member shares the same average numbers of submitted shares, indicating that each member has the same communication cost in expectation. In this case, each share is priced proportional to its difficulty. we calculate the different difficulty settings under the optimal fee rate for each operator. We find that O_1's optimal difficulty level should be 510 and O_1's optimal value is 480. This result is quite reasonable as O_2's optimal fee rate is higher than that of O_1's. To attract more computation power, O_2 has to lower its difficulty level. Otherwise, miners cannot gain from its pool.

C. Time-varying Bitcoin Market Price

Bitcoin market price is time-varying and can be modeled as a log-normal distribution. We show how the oscillation of such a distribution affects the equilibrium achieved by operators and miners below. We assume there exist 3 pools in total and 100 homogeneous miners. We compare three settings: (1) Bitcoin market price is fixed as 6.25, (2) Bitcoin market price follows a lognormal distribution of which the mean is 6.25 and the variance is 0.01 in Fig. 5(a), and (3) Bitcoin market price follows a lognormal distribution of which the mean is 6.25 and the valiance is 1 in Fig. 5(b). From Fig. 6, we can see the miners contribute more power to pooled mining as the market price becomes more unstable.

VII. Related Work

1) Bitcoin Mining Pools: There exist several mining pools [2–6] in the Bitcoin network. Members in the same pool collaborate in mining and share obtained rewards fairly. Many feasible reward mechanisms have been proposed [8–11], among which proportional, PPS, and PPLNS are commonly adopted by mining pools in practice. In a proportional system, miners make profits until the pool finds a block, each of whom will be rewarded in proportion to the number of shares it has submitted. The PPS approach offers an instant, guaranteed payout to a miner for its contribution to the probability that the pool finds a block. The PPLNS method is similar to Proportional, but the miner's reward is calculated on a basis of N last shares. We apply a generalized method where rewards are distributed based on the number of submitted shares in the mining round when the pool finds a block.



(a) $\mu = 6.25$ and $\sigma^2 = 0.01$. (b) $\mu = 6.25$ and $\sigma^2 = 1$.

Fig. 5: Variable Bitcoin market price.

2) Cryptocurrency Investment: The discussion on cryptocurrency investment is hot and there are many related works from both economics [12–15] and computer science [16–19]. Previous papers focus on the analysis of miners' optimal investment portfolios. In [20], authors model how miners form mining pools as a cooperative game with coalitional structures, allowing a miner to join a single pool or mine solo. [21] guides a miner on how to invest across different blockchains with the same PoW algorithm. It leverages modern portfolio theory to predict a miner's allocation over time using price data and inferred risk tolerance. [22] presents an analytical tool that allows risk-averse miners to optimally create a mining portfolio that maximizes their risk-adjusted rewards. This model allows miners to invest on multiple cryptocurrencies. Our paper is a case study on Bitcoin. Besides miners, we also consider the investment from the pool operator side. We combine economic theory by considering miners' risk tolerance level and their reward variance, and game theory by modeling the interactions among pool operators and miners as a Stackelberg game.

3) Stackelberg Game in Cryptocurrency Mining: Stackelberg Game is a widely-used model in the field of cryptocurrency mining. [23, 24] use this model to characterize the interaction between cloud/edge computing provider and a set of mobile miners seeking for computation offloading, and solve an optimal pricing-based computing resource management problem. In [25], authors model a pool operator as the leader and a fixed set of pool members as followers, and figure out the optimal fee the pool operator should charge and the optimal power a member should devote to. [26] considers the Proof-of-Stake mining and applies Stackelberg game to maximize profits for both a stake pool operator and stakeholders in a blockchain-based mobile roaming management system. Our paper focuses on Bitcoin, *i.e.*, Proof-of-Work mining, and is more difficult than the above-mentioned works as our game is multi-leader multi-follower while their models only include a single leader.

Our game is different from traditional Stackelberg games, as leaders are part of the follower game.

VIII. Conclusion

In this paper, we have proposed a Stackelberg game between the pool operators for optimizing their pool mechanisms and the miners for optimal computation power allocation strategies. We adopt classic economic theories and characterize miners as risk-averse players, where a power function is applied to model miners' utilities. We consider two practical pool operation modes, a uniform-share-difficulty mode and a uniform-communication-frequency mode. We analyze the existence and uniqueness of Stackelberg equilibrium (SE) for the proposed game, and derive explicit solutions for miners and operators. We study the impacts of Bitcoin's time-varying market price, which incurs more power devoted to mining pools. Numerical experiments have been conducted to further confirm our analysis.

Appendix

A. Proof of Lemma 1

In this part, we will show that u_j^i is a strictly concave function for $i \in \{1, 2, \cdots, N\}$. For the simplicity of writing, we will ignore the subscript/superscript of i in the below. Thus, H is the shorthand of H_i, and we directly use $\sum h_j$ to represent the total computation power in the network. Similarly, h_j is short for h_j^i, u_j for u_j^i, f for f_i, and d for d_i. We further define H_{-j}, where $H = h_j + H_{-j}$. We define $\phi_j = u_j \sum h_j$. Since $\sum h_j$ is a positive constant, u_j and ϕ_j share the same concavity. Let $r = R(1-f)$ and $c = eC/d - R^2 dC/D^2$. Eq. (7) shows ϕ_j's second derivative.

$$\frac{\partial^2 \phi_j}{\partial h_j^2} = \alpha_j \left(\frac{r h_j}{H} - c h_j\right)^{\alpha_j} \frac{-\psi_j}{h_j^2 H \left(cH - r\right)^2} \tag{7}$$

where $\psi_j = (1-\alpha_j) r^2 H_{-j}^2 + 2crH^2 \left(h_j - H_{-j} + \alpha_j H_{-j}\right) - c^2 H^3 \left(h_j - H_{-j} + \alpha_j H\right).$

Since we assume $r > cH$, the following inequation holds.

$$\begin{aligned}\psi_j &> (1-\alpha_j) c^2 H^2 H_{-j}^2 + 2c^2 H^3 \left(h_j - H_{-j} + \alpha_j H_{-j}\right) \\ &\quad - c^2 H^3 \left(h_j - H_{-j} + \alpha_j H\right) \\ &= (1-\alpha_j) c^2 H^2 H_{-j}^2 + c^2 H^3 (1-\alpha_j) \left(h_j - H_{-j}\right) \\ &= (1-\alpha_j) c^2 h_j^2 H^2 \end{aligned} \tag{8}$$

Obviously, $(1-\alpha_j) c^2 h_j^2 H^2 \geq 0$ always holds for $\forall h_j$. Therefore, ψ_j is bigger than 0, indicating that $\partial^2 \phi_j / \partial h_j^2 < 0$. We can conclude that ϕ_j as well as u_j is strictly concave.

B. Proof of Corollary 1

If we assume all miners have an identical risk tolerance level, denoting α, we can derive explicit expressions of each miner's computation power allocation in the equilibrium. Since we have shown that u_j is strictly concave, its maximal point can be obtained by finding the solution to $\partial u_j / \partial h_j = 0$, which is the same as the solution to $\partial \phi_j / \partial h_j = 0$.

$$\frac{\partial \phi_j}{\partial h_j} = \left(\frac{r h_j}{H} - c h_j\right)^{\alpha} \frac{\alpha c H^2 + c h_j H - r\left(h_j + \alpha H_{-j}\right)}{h_j \left(cH - r\right)} \tag{9}$$

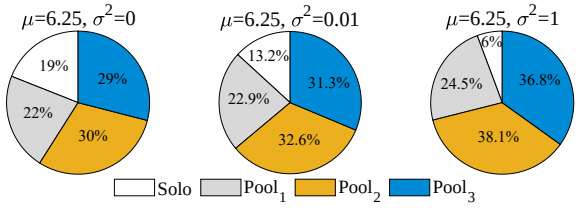


Fig. 6: Homogeneous miners' power allocation evolution.

Given the first derivative of ϕ_j in Eq. (9), we could say that $\alpha c H^2 + c h_j H - r\left(h_j + \alpha H_{-j}\right) = 0$, which can be simplified as below $h_j = \alpha r H - \alpha c H^2 / \left(cH + \alpha r - r\right)$. Accumulating both sides over M miners yields the following results:

$$\sum_{j=1}^{M} h_j = \sum_{j=1}^{M} \frac{\alpha r H - \alpha c H^2}{cH + \alpha r - r}, \tag{10}$$

$$H = M \frac{\alpha r H - \alpha c H^2}{cH + \alpha r - r}. \tag{11}$$

We further simplify Eq. (11) and obtain the explicit expression of H shown in Eq. (12):

$$H = \frac{(M-1)\alpha + 1}{M\alpha + 1} \frac{r}{c}. \tag{12}$$

Thus, the explicit expression of h_j can be easily obtained by replacing H using Eq. (12) into Eq. (9).

C. Proof of Lemma 2

For the simplicity of writing, we ignore the subscript/superscript of i in the following part. As V, short for V_i, is a two-variable function, the definiteness of its hessian function determines its concavity. Denote $\mathcal{H}$ for the Hessian matrix of V, with respect to f_i and d_i, as below:

$$\mathcal{H} := \begin{bmatrix} V_{ff} & V_{fd} \\ V_{df} & V_{dd} \end{bmatrix}, \tag{13}$$

where

$$V_{ff} = \frac{\partial^2 V}{\partial f^2}, \quad V_{fd} = \frac{\partial^2 V}{\partial f \partial d}, \quad V_{df} = \frac{\partial^2 V}{\partial d \partial f}, \quad V_{dd} = \frac{\partial^2 V}{\partial d^2}.$$

We provide the explicit-form expressions of the Jacobian elements first as below:

$$\frac{\partial V}{\partial f} = \frac{R}{\sum h_j}\left(H + f\frac{\partial H}{\partial f}\right) - \frac{eC}{d}\frac{\partial H}{\partial f}, \tag{14}$$

$$\frac{\partial V}{\partial d} = \frac{R}{\sum h_j} f \frac{\partial H}{\partial d} + \frac{eCH}{d^2} - \frac{eC}{d}\frac{\partial H}{\partial d}. \tag{15}$$

Then, we derive Hessian elements as follows:

$$V_{ff} = \frac{2R}{\sum h_j}\frac{\partial H}{\partial f} + \left(\frac{Rf}{\sum h_j} - \frac{eC}{d}\right)\frac{\partial^2 H}{\partial f^2}, \tag{16}$$

$$V_{fd} = \frac{R}{\sum h_j}\frac{\partial H}{\partial d} + \frac{eC}{d^2}\frac{\partial H}{\partial f} + \left(\frac{Rf}{\sum h_j} - \frac{eC}{d}\right)\frac{\partial^2 H}{\partial f \partial d}, \tag{17}$$

$$V_{df} = \frac{R}{\sum h_j}\frac{\partial H}{\partial d} + \left(\frac{Rf}{\sum h_j} - \frac{eC}{d}\right)\frac{\partial^2 H}{\partial d \partial f}, \tag{18}$$

$$V_{dd} = \frac{eC}{d^2}\frac{\partial H}{\partial d} - \frac{2eC}{d^3} + \left(\frac{Rf}{\sum h_j} - \frac{eC}{d}\right)\frac{\partial^2 H}{\partial d^2}. \tag{19}$$

As H is an affine function over h_j, each miner's risk tolerance level just affects the scalars of this linear combination. In the following, we will show $\mathcal{H}$ is positive definite in the special case where all miners share an identical risk tolerance level where H

can be explicitly expressed. The proof is enough to reflect the concavity of V in all general cases. Let $\theta = R/\sum h_j$, $\sigma = eC$, and $\mu = R^2C/D^2$. Given the expression of H in Eq. (12), $\mathcal{H}$ can be expressed as follows:

$$\mathcal{H} := R\frac{(M-1)\,\alpha+1}{M\alpha+1}\begin{bmatrix} v_{ff} & v_{fd} \\ v_{df} & v_{dd}\end{bmatrix}, \tag{20}$$

where

$$v_{ff} = \frac{-2\theta d}{\mu d^2+\sigma}, \tag{21}$$

$$v_{fd} = \frac{2\theta\mu d^2(2f-1)-2\sigma\mu d}{(\mu d^2+\sigma)^2} - \frac{\theta(2f-1)}{\mu d^2+\sigma}, \tag{22}$$

$$v_{df} = \frac{2\theta\mu d^2(2f-1)-2\sigma\mu d}{(\mu d^2+\sigma)^2} - \frac{a(2f-1)}{\mu d^2+\sigma}, \tag{23}$$

$$v_{dd} = \frac{8\mu^2 d^2(f-1)(\sigma-\theta f d)}{(\mu d^2+\sigma)^3} - \frac{2\mu(f-1)(\sigma-3\theta df)}{(\mu d^2+\sigma)^2}. \tag{24}$$

We remove the scalar $R\left[(M-1)\,\alpha+1\right]/\left(M\alpha+1\right)$, as it won't affect the sign of $det(\mathcal{H})$. Eq. (25) shows the simplified $det(\mathcal{H})$.

$$\begin{aligned} det(\mathcal{H}) =& \frac{4\theta\sigma\mu d(\theta d+\sigma f)-4\sigma\mu^2 d^2(\sigma-2\theta d)}{(\mu d^2+\sigma)^4} \\ &+ \frac{4\theta^2\sigma f(1-f)+\theta^2(\mu d^2+\sigma)}{(\mu d^2+\sigma)^3}. \end{aligned} \tag{25}$$

Obviously, the right part of $det(\mathcal{H})$ is positive. Now we show the of the left addend is also positive if e is less than 1 bitcoin.

$$\begin{aligned} &4\theta\sigma\mu d(\theta d+\sigma f)-4\sigma\mu^2 d^2(\sigma-2\theta d) \\ =&4\sigma\mu d\left[2\theta\mu d^2+\theta\sigma f+d\left(\theta^2-\sigma\mu\right)\right] \\ =&4\sigma\mu d\left(2\theta\mu d^2+\theta\sigma f\right)+4\sigma\mu d^2\left[\left(\frac{R}{\sum h_j}\right)^2-e\left(\frac{RT}{D2^{32}}\right)^2\right] \\ =&4\sigma\mu d\left(2\theta\mu d^2+\theta\sigma f\right)+4\sigma\mu d^2\left(\frac{R}{\sum h_j}\right)^2(1-e)>0 \end{aligned} \tag{26}$$

Now, we can conclude that the sign of $det(\mathcal{H})$ is positive, and thus, V is strictly concave over f and d.

D. Proof of Corollary 2

The optimal f and d can be obtained by solving $\frac{\partial V}{\partial f}=0$ and $\frac{\partial V}{\partial d}=0$ shown in the below:

$$\begin{cases} \dfrac{\partial V}{\partial f} = \dfrac{\sigma+\theta d-2\theta df}{\mu d^2+\sigma}, & \text{(27a)} \\ \dfrac{\partial V}{\partial d} = \dfrac{2\sigma\mu d+\theta\sigma f-\theta\mu f d^2}{\left(\mu d^2+\sigma\right)^2}. & \text{(27b)} \end{cases}$$

REFERENCES

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Manubot, Tech. Rep., 2019.

[2] "F2pool." [Online]. Available: https://www.f2pool.com/

[3] "Antpool." [Online]. Available: https://v3.antpool.com/home

[4] "Poolin." [Online]. Available: https://www.poolin.com/

[5] "Btc.com." [Online]. Available: https://pool.btc.com/

[6] "Slushpool." [Online]. Available: https://slushpool.com/

[7] "Potential games." [Online]. Available: https://homepages.cwi.nl/~apt/stra13/potential.pdf

[8] M. Rosenfeld, "Analysis of bitcoin pooled mining reward systems," *arXiv preprint arXiv:1112.4980*, 2011.

[9] S. Zhu, W. Li, H. Li, C. Hu, and Z. Cai, "A survey: Reward distribution mechanisms and withholding attacks in bitcoin pool mining," *Mathematical Foundations of Computing*, vol. 1, no. 4, p. 393, 2018.

[10] S. Kim, "Group bargaining based bitcoin mining scheme using incentive payment process," *Transactions on Emerging Telecommunications Technologies*, vol. 27, no. 11, pp. 1486–1495, 2016.

[11] B. Fisch, R. Pass, and A. Shelat, "Socially optimal mining pools," in *International Conference on Web and Internet Economics*. Springer, 2017, pp. 205–218.

[12] D. L. K. Chuen, L. Guo, and Y. Wang, "Cryptocurrency: A new investment opportunity?" *The Journal of Alternative Investments*, vol. 20, no. 3, pp. 16–40, 2017.

[13] Y. Andrianto and Y. Diputra, "The effect of cryptocurrency on investment portfolio effectiveness," *Journal of finance and accounting*, vol. 5, no. 6, pp. 229–238, 2017.

[14] A. Brauneis and R. Mestel, "Cryptocurrency-portfolios in a mean-variance framework," *Finance Research Letters*, vol. 28, pp. 259–264, 2019.

[15] N. K. Yilmaz and H. B. Hazar, "Predicting future cryptocurrency investment trends by conjoint analysis," *Journal of Economics Finance and Accounting*, vol. 5, no. 4, pp. 321–330, 2018.

[16] T. A. Borges and R. F. Neves, "Ensemble of machine learning algorithms for cryptocurrency investment with different data resampling methods," *Applied Soft Computing*, p. 106187, 2020.

[17] J. Bae and H. Lim, "Random mining group selection to prevent 51% attacks on bitcoin," in *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*. IEEE, 2018, pp. 81–82.

[18] X. Liu, W. Wang, D. Niyato, N. Zhao, and P. Wang, "Evolutionary game for mining pool selection in blockchain networks," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 760–763, 2018.

[19] R. Qin, Y. Yuan, and F.-Y. Wang, "Research on the selection strategies of blockchain mining pools," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 3, pp. 748–757, 2018.

[20] Y. Lewenberg, Y. Bachrach, Y. Sompolinsky, A. Zohar, and J. S. Rosenschein, "Bitcoin mining pools: A cooperative game theoretic analysis," in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*. Citeseer, 2015, pp. 919–927.

[21] G. Bissias, B. N. Levine, and D. Thibodeau, "Using economic risk to model miner hash rate allocation in cryptocurrencies," in *Data Privacy Management, Cryptocurrencies and Blockchain Technology*. Springer, 2018, pp. 155–172.

[22] P. Chatzigiannis, F. Baldimtsi, I. Griva, and J. Li, "Diversification across mining pools: Optimal mining strategies under pow," *arXiv preprint arXiv:1905.04624*, 2019.

[23] Z. Xiong, S. Feng, D. Niyato, P. Wang, and Z. Han, "Optimal pricing-based edge computing resource management in mobile blockchain," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–6.

[24] S. Jiang, X. Li, and J. Wu, "Hierarchical edge-cloud computing for mobile blockchain mining game," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 1327–1336.

[25] G. Xue, J. Xu, H. Wu, W. Lu, and L. Xu, "Incentive mechanism for bitcoin mining pool based on stackelberg game," in *International Conference on Science of Cyber Security*. Springer, 2019, pp. 190–198.

[26] C. T. Nguyen, D. N. Nguyen, D. T. Hoang, H.-A. Pham, N. H. Tuong, and E. Dutkiewicz, "Blockchain and stackelberg game model for roaming fraud prevention and profit maximization," in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2020, pp. 1–6.