Sensitivity-Based Optimization for Blockchain Selfish Mining*

Jing-Yu Ma[1][0000−0002−0396−1232] and Quan-Lin Li[2]

[1] Bussiness School, Xuzhou University of Technology, Xuzhou 221018, China
mjy0501@126.com
[2] School of Economics and Management, Beijing University of Technology, Beijing 100124, China
liquanlin@tsinghua.edu.cn

Abstract. In this paper, we provide a novel dynamic decision method of blockchain selfish mining by applying the sensitivity-based optimization theory. Our aim is to find the optimal dynamic blockchain-pegged policy of the dishonest mining pool. To study the selfish mining attacks, two mining pools is designed by means of different competitive criterions, where the honest mining pool follows a two-block leading competitive criterion, while the dishonest mining pool follows a modification of two-block leading competitive criterion through using a blockchain-pegged policy. To find the optimal blockchain-pegged policy, we set up a policy-based continuous-time Markov process and analyze some key factors. Based on this, we discuss monotonicity and optimality of the long-run average profit with respect to the blockchain-pegged reward and prove the structure of the optimal blockchain-pegged policy. We hope the methodology and results derived in this paper can shed light on the dynamic decision research on the selfish mining attacks of blockchain selfish mining.

Keywords: Blockchain · Selfish mining · Blockchain-pegged policy · Sensitivity-based optimization · Markov decision process.

1 Introduction

Blockchain is used to securely record a public shared ledger of Bitcoin payment transactions among Internet users in an open P2P network. Though the security of blockchain is always regarded as the top priority, it is still threatened by some *selfish mining attacks*. In the PoW blockchain, the probability that an individual miner can successfully mine a block becomes lower and lower, as the number of joined miners increases. This greatly increases the mining risk of each individual miner. In this situation, some miners willingly form a mining pool. Blockchain selfish mining leads to colluding miners in dishonest mining pools,

* Supported by the National Natural Science Foundation of China under grant No. 71932002, and by the Beijing Social Science Foundation Research Base Project under grant No. 19JDGLA004.

one of which can obtain a revenue larger than their fair share. The existence of the selfish mining not only means that it is unfair to solve PoW puzzles but also is a severe flaw in integrity of blockchain.

The existence of such selfish mining attacks was first proposed by Eyal and Sirer [4], they set up a Markov chain to express the dynamic of the selfish mining attacks efficiently. After then, some researchers extended and generalized such a similar method to discuss other attack strategies of blockchain. The newest work is Li et al. [13], which provided a new theoretical framework of pyramid Markov processes to solve some open and fundamental problems of blockchain selfish mining under a rigorously mathematical setting. Göbel et al. [7], Javier and Fralix [9] used two-dimensional Markov chain to study the selfish mining. Furthermore, some key research includes stubborn mining by Nayak et al. [16]; Ethereum by Niu and Feng [17]; multiple mining pools by Jain [8]; multi-stage blockchain by Chang et al. [3]; no block reward by Carlsten et al. [2]; power adjusting by Gao et al. [5].

In the study of blockchain selfish mining, it is a key to develop effective optimal methods and dynamic control techniques. However, little work has been done on applying Markov decision processes (MDPs) to set up optimal dynamic control policies for blockchain selfish mining. In general, such a study is more interesting, difficult and challenging. Based on Eyal and Sirer [4], Sapirshtein et al. [19] extended the underlying model for selfish mining attacks, and provided an algorithm to find ϵ-optimal policies for attackers within the model through MDPs. Furthermore, Wüst [20] provided a quantitative framework based on MDPs to analyse the security of different PoW blockchain instances with various parameters against selfish mining. Gervais et al. [6] extended the MDP of Sapirshtein et al. [19] to determine optimal adversarial strategies for selfish mining. Recently, Zur et al. [24] presented a novel technique called ARR (Average Reward Ratio) MDP to tighten the bound on the threshold for selfish mining in Ethereum.

The purpose of this paper is to apply the MDPs to set up an optimal parameterized policy (i.e., blockchain-pegged policy) for blockchain selfish mining. To do this, we first apply the sensitivity-based optimization theory in the study of blockchain selfish mining, which is an effective tool proposed for performance optimization of Markov systems by Cao [1]. Li [11] and Li and Cao [10] further extended and generalized such a method to a more general framework of perturbed Markov processes. A key idea in the sensitivity-based optimization theory is the performance difference equation that can quantify the performance difference of a Markov system under any two different policies. The performance difference equation gives a straightforward perspective to study the relation of the system performance between two different policies, which provides more sensitivity information. Thus, the sensitivity-based optimization theory has been applied to performance optimization in many practical areas. For example, the energy-efficient data centers by Xia et al. [21] and Ma et al. [14, 15]; the inventory rationing by Li et al. [12]; the multi-hop wireless networks by Xia and Shihada [22] and the finance by Xia [23].

The main contributions of this paper are twofold. The first one is to apply the sensitivity-based optimization theory to study the blockchain selfish mining for the first time, in which we design a modification of two-block leading competitive criterion for the dishonest mining pool. Different from previous works in the literature for applying an ordinary MDP to against the selfish mining attacks, we propose and develop an easier and more convenient dynamic decision method for the dishonest mining pool: the sensitivity-based optimization theory. Crucially, this sensitivity-based optimization theory may open a new avenue to the optimal blockchain-pegged policy of more general blockchain systems. The second contribution of this paper is to characterize the optimal blockchain-pegged policy of the dishonest mining pool. We analyze the monotonicity and optimality of the long-run average profit with respect to the blockchain-pegged policies under some restrained rewards. We obtain the structure of optimal blockchain-pegged policy is related to the blockchain reward. Therefore, the results of this paper give new insights on understanding not only competitive criterion design of blockchain selfish mining, but also applying the sensitivity-based optimization theory to dynamic decision for the dishonest mining pool. We hope that the methodology and results given in this paper can shed light on the study of more general blockchain systems.

The remainder of this paper is organized as follows. In Section 2, we describe a problem of blockchain selfish mining with two different mining pools. In Section 3, we establish a policy-based continuous-time Markov process and introduce some key factors. In Section 4, we discuss the monotonicity and optimality of the long-run average profit with respect to the blockchain-pegged policy by the sensitivity-based optimization theory. Finally, we give some concluding remarks in Section 5.

2 Problem Description

In this section, we give a problem description of blockchain selfish mining with two different mining pools. Also, we provide system structure, operational mode and mathematical notations.

Mining pools: There are two different mining pools: honest and dishonest mining pools.

(a) The honest mining pool follows the Bitcoin protocol. If he mines a block, he will broadcast to whole community immediately. To avoid the 51% attacks, we assume the honest mining pool are majority in the blockchain system.

(b) The dishonest mining pool has the selfish mining attacks. When the dishonest mining pool mines a block, he can earn more unfair revenue. Such revenue will attract some rational honest miners to jump into the dishonest mining pool. We denote the efficiency-increased ratio of the dishonest mining pool and the net jumping's mining rate by τ and γ, respectively.

Selfish mining processes: We assume that the blocks mined by the honest and dishonest mining pools have formed two block branches forked a tree root, and the growths of the two block branches are two Poisson processes with

block-generating rates α_1 and α_2, respectively. In the honest mining pool, the block-generating rate α_1 is equal to the net mining rate, but the situation in the dishonest mining pool is a bit different. The block-generating rate for the dishonest mining pool is $\alpha_2 = \widetilde{\alpha}_2 (1 + \tau)$, where $\widetilde{\alpha}_2$ is regarded as the net mining rate when all the dishonest miners become honest. Following the protocol can not earn more rewards, the honest miners like to jump to the dishonest mining pool with the net jumping rate γ, the real mining rates of the honest and dishonest mining pools are given by $\lambda_1 = \alpha_1 - \gamma$ and $\lambda_2 = (\widetilde{\alpha}_2 + \gamma)(1 + \tau)$, respectively.

Note that mining costs of both mining pools contains two parts: (a) Power consumption cost. Let c_P be the power consumption price per unit of net mining rate and per unit of time. It is easy to see that the power consumption costs per unit of time with respect to the honest and dishonest mining pools are given by $c_P (\alpha_1 - \gamma)$ and $c_P (\widetilde{\alpha}_2 + \gamma)$, respectively. (b) Administrative cost. Let c_A be the administrative price per unit of real mining rate and per unit of time. Then the administrative costs per unit of time with respect to the honest and dishonest mining pools are given by $c_A (\alpha_1 - \gamma)$ and $c_A (\widetilde{\alpha}_2 + \gamma)(1 + \tau)$, respectively.

Competitive criterions: In the blockchain selfish mining, the honest and dishonest mining pools compete fiercely in finding the nonces to generate the blocks, and they publish the blocks to make two block branches forked at a common tree root. For the two block branches, the longer block branch in the forked structure is called a *main chain*, which or the part of which will be pegged on the blockchain. Under the selfish mining attacks, such two mining pools follow the different competitive criterions.

(a) A two-block leading competitive criterion for the honest mining pool. The honest chain of blocks is taken as the main chain pegged on the blockchain, as soon as the honest chain of blocks is two blocks ahead of the dishonest chain of blocks.

(b) A modification of two-block leading competitive criterion for the dishonest mining pool. Once the dishonest chain of blocks is two blocks ahead of the honest chain of blocks, the dishonest chain of blocks can be taken as the main chain. To get more reward, the dishonest mining pool may prefer to keep its mined blocks secret, and continue to mine more blocks rather than broadcast all the mined information.

Since the dishonest miners are minority, their mining power is limited, the dishonest mining pool will not be extend infinitely. We assume that once the dishonest main chain contains m blocks, its part n blocks ($n \leq m$) must be pegged on the blockchain immediately. In addition, the limitation of the dishonest main chain leads to that the honest main chain containing at most $n - 2$ blocks due to the two-block leading competitive criterion.

Blockchain-pegged processes: If the main chain is formed, then the mining processes are terminated immediately. The honest main chain or the part of the dishonest main chain is pegged on the blockchain, and the blockchain-pegged times are i.i.d. and exponential with mean $1/\mu$. The mining pool of the main chain can obtain an appropriate amount of reward (or compensation) from two different parts: A block reward r_B by the blockchain system and an average

total transaction fee r_F in the block. At the same time, all the blocks of the other non-main chain become orphan and immediately return to the transaction pool without any new fee. Note that no new blocks are generated during the blockchain-pegged process of the main chain.

We assume that all the random variables defined above are independent of each other. Fig. 1 provides an intuitive understanding for the two cases.

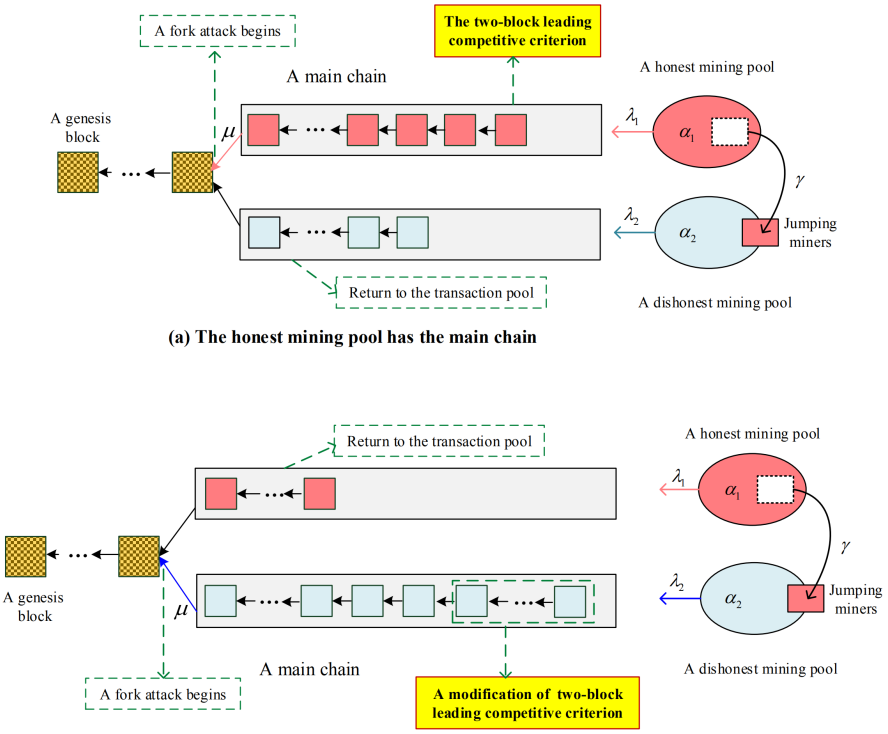

Fig. 1. A blockchain selfish mining with two different mining pools.

3 Optimization Model Formulation

In this section, we establish an optimization problem to find an optimal blockchain-pegged policy for the dishonest mining pool. To do this, we set up a policy-based continuous-time Markov process and introduce some key factors.

3.1 The states and policies

To study the blockchain-pegged policy of the blockchain selfish mining with two different mining pools, we first define both 'states' and 'policies' to express such a stochastic dynamic.

Let $N_1(t)$ and $N_2(t)$ be the numbers of blocks mined by the honest and dishonest mining pools at time t, respectively. Then $(N_1(t), N_2(t))$ is regarded as the state of this system at time t. Obviously, all the cases of State $(N_1(t), N_2(t))$ form a state space as follows:

$$\boldsymbol{\Omega} = \bigcup_{k=0}^{m+2} \boldsymbol{\Omega}_k,$$

where

$$\begin{aligned}
\boldsymbol{\Omega}_0 &= \{(0,0), (0,1), \ldots, (0,m)\}, \\
\boldsymbol{\Omega}_1 &= \{(1,0), (1,1), \ldots, (1,m)\}, \\
\boldsymbol{\Omega}_k &= \{(k,k-2), (k,k-1), \ldots, (k,m)\}, k = 2,3,\ldots,m+2.
\end{aligned}$$

Actually, the blockchain-pegged policy of the dishonest mining pool can be represented by blockchain-pegged probability p. The dishonest mining pool pegs the main chain on the blockchain according to the probability p at the state (n_1, n_2) for $(n_1, n_2) \in \boldsymbol{\Omega}$. From the problem description in Section 2, it is easy to see that

$$p = \begin{cases} a \in [0,1], & n_1 = 0,1,\ldots,m-3,\ n_2 = n_1+2, n_1+3, \ldots, m-1, \\ 1, & n_1 = 0,1,\ldots,m-2,\ n_2 = m, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

It is obviously that the Markov process is controlled by the blockchain-pegged policy (the probability p). Let all the possible probabilities p given in (1) compose a policy space as follows:

$$\mathcal{P} = \{p : p \in [0,1], \text{ for } (n_1, n_2) \in \boldsymbol{\Omega}\}.$$

It is readily seen that State $(0,0)$ is a key state, which plays a key role in setting up the Markov process of two block branches forked at the tree root. In fact, State $(0,0)$ describes the tree root as the starting point of the fork attacks, e.g., see Fig. 2. If the Markov process enters State $(0,0)$, then the fork attack ends immediately, and the main chain is pegged on the blockchain.

Now, from Fig. 2, we provide an interpretation for the blockchain-pegged probability p as follows:

(1) In Part A-1, i.e., $n_1 = 0,1,\ldots,m-3$ and $n_2 = n_1+2, n_1+3, \ldots, m-1$, the dishonest mining pool follows the modification of two-block leading competitive criterion and forms the dishonest main chain, then the probability $p \in [0,1]$.

(2) In Part A-2, i.e., $n_1 = 0,1,\ldots,m-2$ and $n_2 = m$, for the limitation of dishonest mining power, the dishonest main chain must be pegged on the

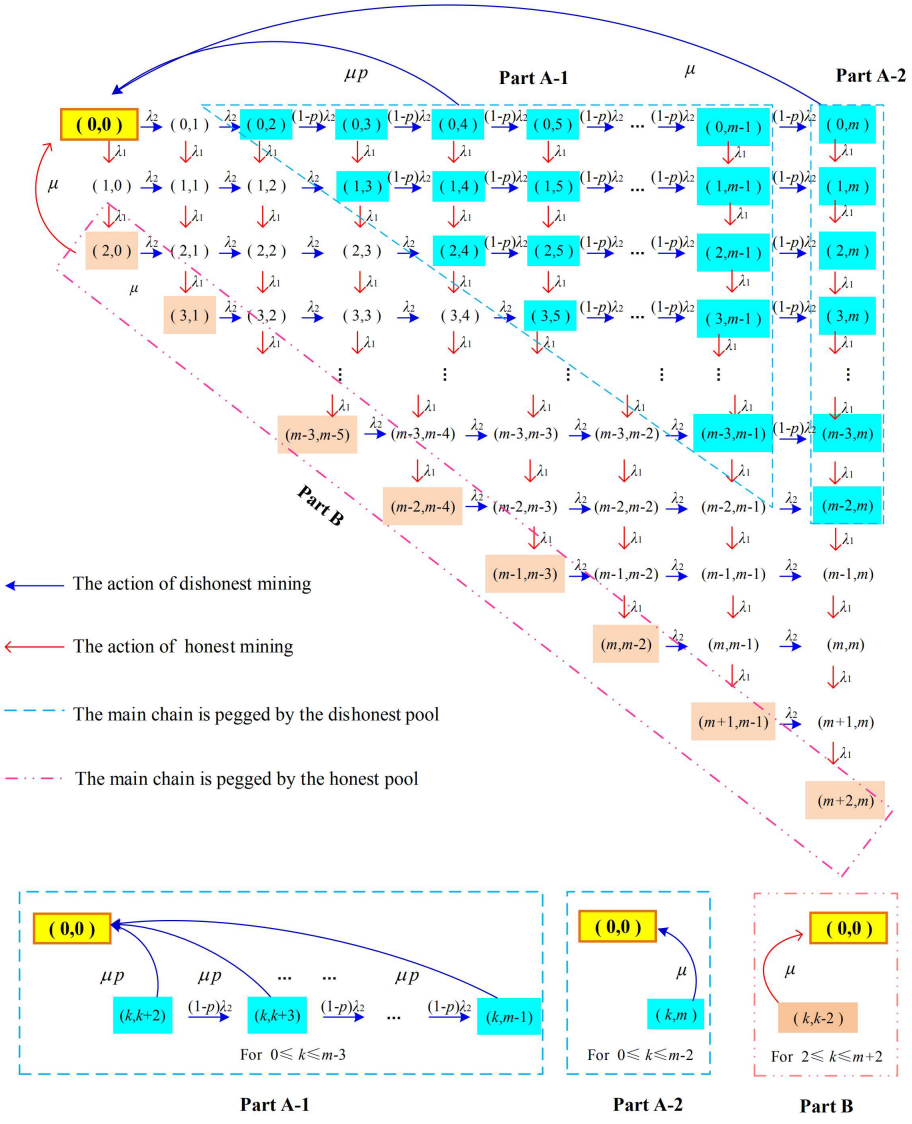

Fig. 2. The state transition relation of the Markov process.

blockchain, or there is a risk of getting no reward. It is easy to see that the probability p is taken as 1.

(3) In the rest of Fig. 2, it is the competitive process of honest and dishonest mining pools. Therefore, $p = 0$ for the dishonest main chain hasn't formed. In addition, the states in Part B mean that the honest main chain is formed.

Due to the modification of two-block leading competitive criterion, the limitation m of the dishonest mining pool must be more than 2, so that there exist the blockchain-pegged policy for the dishonest mining pool. If $m \geq 5$, the infinitesimal generator has a general expression (note that the special cases of $m = 3$ and $m = 4$ are omitted here). In what follows, we assume $m \geq 5$ for convenience of calculation, but the analysis method is similar.

Let $\mathbf{X}^{(p)}(t) = (N_1(t), N_2(t))^{(p)}$ be the system state at time t under any given policy $p \in \mathcal{P}$. Then $\left\{\mathbf{X}^{(p)}(t) : t \geq 0\right\}$ is a policy-based continuous-time Markov process on the state space $\boldsymbol{\Omega}$ whose state transition relation is depicted in Fig. 2. Obviously, such a Markov process is a special form of the pyramid Markov process given in Li et al. [13]. Based on this, the infinitesimal generator of the Markov process $\left\{\mathbf{X}^{(p)}(t) : t \geq 0\right\}$ is given by

$$\mathbf{Q}^{(p)} = \begin{pmatrix} Q_{0,0} & B_0 & & & & & \\ Q_{1,0} & Q_{1,1} & B_1 & & & & \\ Q_{2,0} & & Q_{2,2} & B_2 & & & \\ \vdots & & & \ddots & \ddots & & \\ Q_{m+1,0} & & & & Q_{m+1,m+1} & B_{m+1} & \\ Q_{m+2,0} & & & & & Q_{m+2,m+2} & \end{pmatrix}. \tag{2}$$

Here, we omit the details of the submatrices in the infinitesimal generator $\mathbf{Q}^{(p)}$.

3.2 The stationary probability vector

Based on some special properties of the infinitesimal generator, we provide the stationary probability vector for the policy-based continuous-time Markov process $\left\{\mathbf{X}^{(p)}(t) : t \geq 0\right\}$.

For $n_1 = 0, 1, \ldots, m-3$, $n_2 = n_1 + 2, n_1 + 3, \ldots, m-1$ and $0 \leq p < 1$, it is clear from the finite states that the policy-based continuous-time Markov process $\mathbf{Q}^{(p)}$ must be irreducible, aperiodic and positive recurrent.

We write the stationary probability vector of the Markov process $\left\{\mathbf{X}^{(p)}(t) : t \geq 0\right\}$ as follows:

$$\boldsymbol{\pi}^{(p)} = \left(\boldsymbol{\pi}_0^{(p)}, \boldsymbol{\pi}_1^{(p)}, \ldots, \boldsymbol{\pi}_{m+2}^{(p)}\right), \tag{3}$$

where

$$\begin{aligned} \boldsymbol{\pi}_0^{(p)} &= \left(\pi^{(p)}(0,0), \pi^{(p)}(0,1), \ldots, \pi^{(p)}(0,m)\right), \\ \boldsymbol{\pi}_1^{(p)} &= \left(\pi^{(p)}(1,0), \pi^{(p)}(1,1), \ldots, \pi^{(p)}(1,m)\right), \\ \boldsymbol{\pi}_k^{(p)} &= \left(\pi^{(p)}(k,k-2), \pi^{(p)}(k,k-1), \ldots, \pi^{(p)}(k,m)\right), \ 2 \leq k \leq m+2. \end{aligned}$$

Let

$$\begin{aligned}&\mathbf{D}_0 = 1,\\ &\mathbf{D}_k = B_{k-1}\left(-Q_{k,k}\right)^{-1},\ k = 1, 2, \ldots, m+2.\end{aligned} \tag{4}$$

Then the following theorem provides an explicit expression for the stationary probability vector $\boldsymbol{\pi}^{(p)}$ by means of the system of linear equations: $\boldsymbol{\pi}^{(p)}\boldsymbol{Q}^{(p)} = \mathbf{0}$ and $\boldsymbol{\pi}^{(p)}\boldsymbol{e} = 1$.

Theorem 1. *The stationary probability vector $\boldsymbol{\pi}^{(p)}$ of the Markov process $\boldsymbol{Q}^{(p)}$ is given by*

$$\boldsymbol{\pi}_k^{(p)} = \boldsymbol{\pi}_0^{(p)} \prod_{l=1}^{k} \mathbf{D}_l, \tag{5}$$

where $\boldsymbol{\pi}_0^{(p)}$ is determined by the system of linear equations

$$\begin{aligned}\boldsymbol{\pi}_0^{(p)} \left(\sum_{k=0}^{m+2} \prod_{l=0}^{k} \mathbf{D}_l Q_{k,0} \right) &= \mathbf{0},\\ \boldsymbol{\pi}_0^{(p)} \left(\sum_{k=0}^{m+2} \prod_{l=0}^{k} \mathbf{D}_l \boldsymbol{e} \right) &= 1.\end{aligned}$$

3.3 The reward function

A reward function of the dishonest mining pool with respect to both states and policies is defined as a profit rate (i.e., the total revenues minus the total costs per unit of time).

Let $R = r_B + r_F$ and $C = (\widetilde{\alpha} + \gamma)\left[c_P + c_A\left(1 + \tau\right)\right]$. Then R and C denote the blockchain-pegged reward and the mining cost for the dishonest mining pool, respectively. According to Fig. 2, the reward function at State $\left(N_1\left(t\right), N_2\left(t\right)\right)^{(p)}$ under the blockchain-pegged policy p is defined as follows:

$$f^{(p)}\left(n_1, n_2\right) = \begin{cases} n_2 R\mu p - C, & \text{if } 0 \le n_1 \le m-3 \text{ and } n_1 + 2 \le n_2 \le m-1,\\ mR\mu - C, & \text{if } 0 \le n_1 \le m-2 \text{ and } n_2 = m,\\ -C, & \text{otherwise.}\end{cases}$$

We futher define a column vector $\boldsymbol{f}^{(p)}$ composed of the elements $f^{(p)}\left(n_1, n_2\right)$ as

$$\boldsymbol{f}^{(p)} = \left(\left(\boldsymbol{f}_0^{(p)}\right)^T, \left(\boldsymbol{f}_1^{(p)}\right)^T, \ldots, \left(\boldsymbol{f}_{m+2}^{(p)}\right)^T \right)^T, \tag{6}$$

where

$$\begin{aligned}\boldsymbol{f}_0^{(p)} &= \left(f^{(p)}\left(0,0\right), f^{(p)}\left(0,1\right), \ldots, f^{(p)}\left(0,m\right) \right)^T,\\ \boldsymbol{f}_1^{(p)} &= \left(f^{(p)}\left(1,0\right), f^{(p)}\left(1,1\right), \ldots, f^{(p)}\left(1,m\right) \right)^T,\\ \boldsymbol{f}_k^{(p)} &= \left(f^{(p)}\left(k,k-2\right), f^{(p)}\left(k,k-1\right), \ldots, f^{(p)}\left(k,m\right) \right)^T,\ k = 2, 3, \ldots, m+2.\end{aligned}$$

In the remainder of this section, the long-run average profit of the dishonest mining pool under a blockchain-pegged policy p is defined as

$$\eta^p = \lim_{T\to+\infty} E\left\{\frac{1}{T}\int_0^T f^{(p)}\left(\left(N_1(t), N_2(t)\right)^{(p)}\right)\mathrm{d}t\right\} = \boldsymbol{\pi}^{(p)}\boldsymbol{f}^{(p)}, \tag{7}$$

where $\boldsymbol{\pi}^{(p)}$ and $\boldsymbol{f}^{(p)}$ are given by (5) and (6), respectively.

3.4 The performance potential

The sensitivity-based optimization theory has a fundamental quantity called performance potential by Cao [1], which is defined as

$$g^{(p)}(n_1, n_2) = E\left\{\int_0^{+\infty}\left[f^{(p)}\left(\mathbf{X}^{(p)}(t)\right) - \eta^p\right]\mathrm{d}t \,\middle|\, \mathbf{X}^{(p)}(0) = (n_1, n_2)\right\}, \tag{8}$$

where η^p is defined in (7). For any blockchain-pegged policy $p \in \mathcal{P}$, $g^{(p)}(n_1, n_2)$ quantifies the contribution of the initial State (n_1, n_2) to the long-run average profit of the dishonest mining pool. Here, $g^{(p)}(n_1, n_2)$ is also called the relative value function or the bias in the traditional MDP theory, see, e.g., Puterman [18]. We further define a column vector $\boldsymbol{g}^{(p)}$ as

$$\boldsymbol{g}^{(p)} = \left(\left(\boldsymbol{g}_0^{(p)}\right)^T, \left(\boldsymbol{g}_1^{(p)}\right)^T, \ldots, \left(\boldsymbol{g}_{m+2}^{(p)}\right)^T\right)^T, \tag{9}$$

where

$$\begin{aligned}
\boldsymbol{g}_0^{(p)} &= \left(g^{(p)}(0,0), g^{(p)}(0,1), \ldots, g^{(p)}(0,m)\right)^T,\\
\boldsymbol{g}_1^{(p)} &= \left(g^{(p)}(1,0), g^{(p)}(1,1), \ldots, g^{(p)}(1,m)\right)^T,\\
\boldsymbol{g}_k^{(p)} &= \left(g^{(p)}(k,k-2), g^{(p)}(k,k-1), \ldots, g^{(p)}(k,m)\right)^T,\ k = 2,3,\ldots,m+2.
\end{aligned}$$

A similar computation to that in Ma et al. [14, 15] is omitted here, we can provide an expression for the vector $\boldsymbol{g}^{(p)}$

$$\boldsymbol{g}^{(p)} = R\boldsymbol{a} + \boldsymbol{b}, \tag{10}$$

where $\boldsymbol{a}$ and $\boldsymbol{b}$ can be given by $\mathbf{Q}^{(p)}$, $\boldsymbol{\pi}^{(p)}$ and $\boldsymbol{f}^{(p)}$. It is seen that all the entries $g^{(p)}(n_1, n_2)$ in $\boldsymbol{g}^{(p)}$ are the linear functions of R. Therefore, our objective is to find the optimal blockchain-pegged policy p^* such that the long-run average profit of the dishonest mining pool η^p is maximize, that is,

$$p^* = \arg\max_{p\in\mathcal{P}}\left\{\eta^p\right\}. \tag{11}$$

However, it is very challenging to analyze some interesting structure properties of the optimal blockchain-pegged policy p^*. In the remainder of this paper, we will apply the sensitivity-based optimization theory to study such an optimal problem.

4 Monotonicity and Optimality

In this section, we use the the sensitivity-based optimization theory to discuss monotonicity and optimality of the long-run average profit of the dishonest mining pool with respect to the blockchain-pegged policy. Based on this, we obtain the optimal blockchain-pegged policy of the dishonest mining pool.

In an MDP, system policies will affect the element values of infinitesimal generator and reward function. That is, if the policy p changes, then the infinitesimal generator $\mathbf{Q}^{(p)}$ and the reward function $\boldsymbol{f}^{(p)}$ will have their corresponding changes. To express such a change mathematically, we take two different policies $p, p' \in \mathcal{P}$, both of which correspond to their infinitesimal generators $\mathbf{Q}^{(p)}$ and $\mathbf{Q}^{(p')}$, and to their reward functions $\boldsymbol{f}^{(p)}$ and $\boldsymbol{f}^{(p')}$.

The following lemma provides the performance difference equation for the difference $\eta^{p'} - \eta^{p}$ of the long-run average performances for any two blockchain-pegged policies $p, p' \in \mathcal{P}$. Here, we only restate it without proof, while readers may refer to Cao [1] and Ma et al. [14] for more details.

Lemma 1. *For any two blockchain-pegged policies $p, p' \in \mathcal{P}$, we have*

$$\eta^{p'} - \eta^{p} = \boldsymbol{\pi}^{(p')} \left[\left(\mathbf{Q}^{(p')} - \mathbf{Q}^{(p)} \right) \boldsymbol{g}^{(p)} + \left(\boldsymbol{f}^{(p')} - \boldsymbol{f}^{(p)} \right) \right]. \tag{12}$$

Therefore, to find the optimal blockchain-pegged policy p^*, we consider such two blockchain-pegged policies $p, p' \in \mathcal{P}$. Suppose the blockchain-pegged policy is changed from p to p', which corresponding the states (n_1, n_2) for $n_1 = 0, 1, \ldots, m-3$ and $n_2 = n_1 + 2, n_1 + 3, \ldots, m-1$, i.e., Part A-1 of Fig. 2.

Using Lemma 2, we examine the sensitivity of blockchain-pegged policy on the long-run average profit of the dishonest mining pool. Substituting (2) and (6) into (12), we have

$$\begin{aligned} & \eta^{p'} - \eta^{p} \\ &= \boldsymbol{\pi}^{(p')} \left[\left(\mathbf{Q}^{(p')} - \mathbf{Q}^{(p)} \right) \boldsymbol{g}^{(p)} + \left(\boldsymbol{f}^{(p')} - \boldsymbol{f}^{(p)} \right) \right] \\ &= (p' - p) \sum_{n_1=0}^{m-3} \sum_{n_2=n_1+2}^{m-1} \pi^{(p')}(n_1, n_2) \left[\mu - (\mu - \lambda_2) g^{(p)}(n_1, n_2) - \lambda_2 g^{(p)}(n_1, n_2+1) + n_2 R \mu \right]. \end{aligned} \tag{13}$$

With the difference (13), we can easily obtain the following equation

$$\frac{\triangle \eta^{p}}{\triangle p} = \sum_{n_1=0}^{m-3} \sum_{n_2=n_1+2}^{m-1} \pi^{(p')}(n_1, n_2) \left[\mu - (\mu - \lambda_2) g^{(p)}(n_1, n_2) - \lambda_2 g^{(p)}(n_1, n_2+1) + n_2 R \mu \right], \tag{14}$$

where $\triangle \eta^{p} = \eta^{p'} - \eta^{p}$ and $\triangle p = p' - p$. As $p' \to p$,

$$\left. \frac{\mathrm{d}\eta^{p}}{\mathrm{d}p} \right|_{\triangle p \to 0} = \lim_{\triangle p \to 0} \frac{\eta^{p'} - \eta^{p}}{\triangle p},$$

we derive the following derivative equation

$$\frac{\mathrm{d}\eta^p}{\mathrm{d}p}=\sum_{n_1=0}^{m-3}\sum_{n_2=n_1+2}^{m-1}\pi^{(p)}\left(n_1,n_2\right)\left[\mu-(\mu-\lambda_2)\,g^{(p)}\left(n_1,n_2\right)-\lambda_2 g^{(p)}\left(n_1,n_2+1\right)+n_2R\mu\right]. \tag{15}$$

According to (10), $g^{(p)}\left(n_1,n_2\right)$ and $g^{(p)}\left(n_1,n_2+1\right)$ are both linear functions w.r.t. R. Thus, we denote $g^{(p)}\left(n_1,n_2\right)$ and $g^{(p)}\left(n_1,n_2+1\right)$ as $a_{n_1,n_2}R+b_{n_1,n_2}$ and $a_{n_1,n_2+1}R+b_{n_1,n_2+1}$, respectively. Substituting into (15), we have

$$\frac{\mathrm{d}\eta^p}{\mathrm{d}p}=\overline{a}R+\overline{b}, \tag{16}$$

where

$$\begin{aligned}\overline{a}&=\sum_{n_1=0}^{m-3}\sum_{n_2=n_1+2}^{m-1}\pi^{(p)}\left(n_1,n_2\right)\left[\left(\lambda_2-\mu\right)a_{n_1,n_2}-\lambda_2 a_{n_1,n_2+1}+n_2\mu\right],\\ \overline{b}&=\sum_{n_1=0}^{m-3}\sum_{n_2=n_1+2}^{m-1}\pi^{(p)}\left(n_1,n_2\right)\left[\left(\lambda_2-\mu\right)b_{n_1,n_2}+\lambda_2 a_{n_1,n_2+1}b_{n_1,n_2+1}+\mu\right].\end{aligned}$$

It is clear that $\frac{\mathrm{d}\eta^p}{\mathrm{d}p}$ is also a linear function w.r.t. R, and depends only on the current policy.

Remark 1 *It is seen from (16) that we only need to know the sign of $\frac{d\eta^p}{dp}$, instead of its precise value. The estimation accuracy of a sign is usually better than that of a value. Therefore, this feature can help us find the optimal blockchain-pegged policy effectively. Moreover, we see that we do not have to know some prior system information. Thus, the complete system information is not required in our approach and this is an advantage during the practical application.*

Remark 2 *The key idea of the sensitivity-based optimization theory is to utilize the performance sensitivity information, such as the performance difference, to conduct the optimization of stochastic systems. Therefore, even if the competition criteria become more complicated, it does not affect the applicability of our method.*

The following theorems discuss monotonicity and optimality of the long-run average profit η^p of the dishonest mining pool with respect to the blockchain-pegged policy p.

Theorem 2. *If $R>-\overline{b}/\overline{a}$, then the long run average profit η^p is strictly monotone increasing with respect to each decision element $p\in[0,1]$, and the optimal blockchain-pegged policy $p^*=1$.*

This theorem follows directly (16). It is seen that the optimal blockchain-pegged policy $p^*=1$ just corresponding to any State (n_1,n_1+2) in Part A-1 of Fig. 2, and the state transition has changed. In this case, the dishonest chain

of blocks is only two blocks ahead of the honest chain of blocks, the dishonest mining pool should peg on the blockchain, also follows the two-block leading competitive criterion.

Therefore, when the blockchain-pegged reward is higher with $R > -\overline{b}/\overline{a}$, it is seen that the dishonest miners become honest, all miners will follow the PoW protocol and broadcast to the whole community. In this case, the selfish mining attacks should be invalid.

Theorem 3. *If $0 \leq R < -\overline{b}/\overline{a}$, then the long run average profit η^p is strictly monotone decreasing with respect to each decision element $p \in [0,1]$, and the optimal blockchain-pegged policy $p^* = 0$.*

Simlar to Theorem 2, this theorem also follows directly (16). It is seen that the optimal blockchain-pegged policy $p^* = 0$ corresponding to any State (n_1, n_2) in Part A-1 of Figure 2.

In the blockchain selfish mining, if the dishonest mining pool makes decision not to peg on the blockchain, i.e., $p^* = 0$, the main chain is detained to continue mining more blocks so that it is not broadcasted in the blockchain network, until the number of blocks reaches m for the limited mining bound. In this case, the dishonest mining pool prefer to obtain more mining profit through winning on mining more blocks, rather than peg on the blockchain prematurely.

Therefore, when the blockchain-pegged reward is lower with $0 \leq R < -\overline{b}/\overline{a}$, it is seen that the dishonest mining pool follows the m-block leading competitive criterion under the selfish mining attacks.

Theorem 4. *If $R = -\overline{b}/\overline{a}$, then the change of blockchain-pegged policy p no longer improve the long-run average profit η^p.*

With Theorem 4, the dishonest miners don't care about when the main chain is pegged on the blockchain, thus the blockchain-pegged policy can be chosen randomly in set $[0,1]$.

5 Concluding Remarks

In this paper, we propose a novel dynamic decision method by applying the sensitivity-based optimization theory to study the optimal blockchain-pegged policy of blockchain selfish mining with two different mining pools.

We describe a more general blockchain selfish mining with a modification of two-block leading competitive criterion, which is related to the blockchain-pegged policies. To find the optimal blockchain-pegged policy of the dishonest mining pool, we analyze the monotonicity and optimality of the long-run average profit with respect to the blockchain-pegged policy under some restrained blockchain-pegged rewards. We prove the structure of optimal blockchain-pegged policy with respect to the blockchain-pegged rewards. Different from those previous works in the literature on applying the traditional MDP theory to the blockchain selfish mining, the sensitivity-based optimization theory used in this

paper is easier and more convenient in the optimal policy study of blockchain selfish mining.

Along such a research line of applying the sensitivity-based optimization theory, there are a number of interesting directions for potential future research, for example:

• Extending to the blockchain selfish mining with multiple mining pools, for example, a different competitive criterion, no space limitation of the dishonest pool and so on;

• analyzing non-Poisson inputs such as Markovian arrival processes (MAPs) and/or non-exponential service times, e.g. the PH distributions;

• discussing the long-run average performance is influenced by some concave or convex reward (or cost) functions; and

• studying individual or social optimization for the blockchain selfish mining from a perspective of combining game theory with the sensitivity-based optimization.

References

1. Cao, X. R.: Stochastic learning and optimization—A sensitivity-based approach. Springer, New York (2007)
2. Carlsten, M., Kalodner, H. A., Weinberg, S. M., et al.: On the instability of bitcoin without the block reward. In: ACM SIGSAC Conference on Computer and Communications Security, pp. 154–167. Association for Computing Machinery, New York (2016)
3. Chang, D., Hasan, M., Jain, P.: Spy based analysis of selfish mining attack on multi-stage blockchain. IACR Cryptol, ePrint: 2019/1327, pp. 1–34 (2019)
4. Eyal, I., Sirer, E. G.: Majority is not enough: Bitcoin mining is vulnerable. In: International Conference on Financial Cryptography and Data Security, pp. 436–454. Springer, Berlin (2014)
5. Gao, S., Li, Z., Peng, Z. et al.: Power adjusting and bribery racing: Novel mining attacks in the bitcoin system. In: ACM SIGSAC Conference on Computer and Communications Security, pp. 833–850. Association for Computing Machinery, New York (2019)
6. Gervais, A., Karame, G. O., Wüst, K., et al.: On the security and performance of Proof of Work blockchains. In: ACM SIGSAC Conference on Computer and Communications Security, pp. 3–16. Association for Computing Machinery, New York (2016)
7. Göbel, J., Keeler, H. P., Krzesinski, A. E., et al.: Bitcoin blockchain dynamics: The selfish-mine strategy in the presence of propagation delay. Performance Evaluation, **104**, 23–41 (2016)
8. Jain, P.: Revenue generation strategy through selfish mining focusing multiple mining pools. Bachelor Thesis, Computer Science & Applied Mathematics, Indraprastha Institute of Information Technology, India (2019)
9. Javier, K., Fralix, B.: A further study of some Markovian Bitcoin models from Göbel et al.. Stochastic Models, **36**(2), 223–250 (2020)
10. Li, Q. L., Cao, J.: Two types of RG-factorizations of quasi-birth-and-death processes and their applications to stochastic integral functionals. Stochastic Models, **20**(3), 299–340 (2004)

11. Li, Q. L.: Constructive computation in stochastic models with applications: the RG factorizations. Springer, Heidelberg (2010)
12. Li, Q. L., Li, Y. M., Ma, J. Y., et al.: A complete algebraic transformational solution for the optimal dynamic policy in inventory rationing across two demand classes. arXiv: 1908.09295v1 (2019)
13. Li, Q. L., Chang, Y. X., Wu, X., et al.: A new theoretical framework of pyramid Markov processes for blockchain selfish mining. Journal of Systems Science and Systems Engineering, 1–45 (2021)
14. Ma, J. Y., Xia, L., Li, Q. L.: Optimal energy-efficient policies for data centers through Sensitivity-based optimization. Discrete Event Dynamic Systems, **29**(4), 567–606 (2019)
15. Ma, J. Y., Li, Q. L., Xia, L.: Optimal asynchronous dynamic policies in energy-efficient data centers. arXiv: 1901.03371 (2019)
16. Nayak, K., Kumar, S., Miller A., et al.: Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. In: IEEE European Symposium on Security and Privacy, pp. 305–320. IEEE, Saarbruecken (2016)
17. Niu, J., Feng, C.: Selfish mining in Ethereum. arXiv: 1901.04620 (2019)
18. Puterman, M. L.: Markov decision processes: discrete stochastic dynamic programming. John Wiley & Sons, Hoboken (1994)
19. Sapirshtein, A., Sompolinsky, Y., Zohar, A.: Optimal selfish mining strate gies in Bitcoin. In: The 20th International Conference on Financial Cryptography and Data Security, pp. 515–532. Springer, Berlin (2016)
20. Wüst, K.: Security of blockchain technologies. Master Thesis, Department of Computer Science, ETH Zürich, Switzerland (2016)
21. Xia, L., Zhang, Z. G., Li, Q. L.: A c/μ-rule for job assignment in heterogeneous group-server queues. Production and Operations Management, 1–18 (2021)
22. Xia, L., Shihada, B.: A Jackson network model and threshold policy for joint optimization of energy and delay in multi-hop wireless networks. European Journal of Operational Research, **242**(3), 778–787 (2015)
23. Xia, L.: Risk-sensitive Markov decision processes with combined metrics of mean and variance. Production and Operations Management, **29**(12): 2808–2827 (2020)
24. Zur, R. B., Eyal, I., Tamar, A.: Efficient MDP analysis for selfish-mining in blockchains. In: The 2nd ACM Conference on Advances in Financial Technologies, pp. 113–131. Association for Computing Machinery, New York (2020)