Computers & Operations Research 134 (2021) 105365

Contents lists available at ScienceDirect

Computers and Operations Research

journal homepage: www.elsevier.com/locate/cor

A discrete-event simulation model for the Bitcoin blockchain network with strategic miners and mining pool managers

Kejun Li [a], Yunan Liu [a], Hong Wan [a], Yining Huang [b]

[a] Department of Industrial and Systems Engineering, North Carolina State University, Raleigh, NC 27606, USA
[b] Operations Research Graduate Program, North Carolina State University, Raleigh, NC 27607, USA

ARTICLE INFO

Keywords:
Blockchain
Discrete-event simulation
Bitcoin mining policy
Mining competition

ABSTRACT

As the first and most famous cryptocurrency-based blockchain technology, Bitcoin has attracted tremendous attention from both academic and industrial communities in the past decade. A Bitcoin network is comprised of two interactive parties: individual miners and mining pool managers, each of which strives to maximize its own utility. In particular, individual miners choose which mining pool to join and decide on how much mining power to commit under limited constraints on the mining budget and mining power capacity; managers of mining pools determine how to allocate the mining reward and how to adjust the membership fee. In this work we investigate the miners' and mining pool managers' decisions in repeated Bitcoin mining competitions by building a Monte-Carlo discrete-event simulation model. Our simulation model (i) captures the behavior of these two parties and how their decisions affect each other, and (ii) characterizes the system-level dynamics of the blockchain in terms of the mining difficulty level and total mining power. In addition, we study the sensitivity of system performance metrics with respect to various control parameters. Our analysis may provide useful guidelines to mining activity participants in the Bitcoin network.

1. Introduction

Blockchain is a digital append-only database that maintains a dynamic list of records. At its core, it is a consensus-based distributed system across a peer-to-peer network. The ingenuity of blockchain technology lies in its decentralized nature which enables the development of secured environment against tampering and revision. Since its first application to the Bitcoin cryptocurrency proposed by Nakamoto et al. (2008), blockchain technology has grown rapidly in the past decade and has been applied to supply chain (Sharma et al., 2020), finance (Chang et al., 2020), healthcare (Griggs et al., 2018), and energy (Li et al., 2017). The backbone of the Bitcoin system is a winner-take-all game, in which each participant tries to be the first to solve a highly complicated computational problem. Due to the high payoff[1] of the game, many players (a.k.a., miners) are incentivized to participate in the Bitcoin mining competition, despite the fact that miners have to pay for the costs of purchasing, operating, and maintaining mining machines. Because an individual participant's winning probability is extremely small as the total mining power increases[2], mining pools with integrated mining power arises. Mining pool provide a platform for individual players to cooperate with each other to reduce the mining risk as well as share the mining reward. To regulate the operations of a mining pool, the pool manager is responsible for the adjustment of membership fees and reward sharing policy.

Motivated by the growing interest in Bitcoin blockchain technology, we built a Monte-Carlo simulation model to study the system dynamics of the Bitcoin block system. Our work investigates the behavior of individual miners and pool managers, and how they interact with each other. From the perspective of individual miners, we study how to seek appropriate mining policy so as to maximize profit with limited monetary budgets and hashing power capacities; they decide which pool to join and when to turn on (off) their mining machines. From the perspective of pool managers, we examine how they adopt different reward share policies and adjust pool membership fees. Before reviewing relevant literature and presenting our contributions, we first provide a brief review of the background of the Bitcoin blockchain.

E-mail address: kli15@ncsu.edu (K. Li).

[1] The winner is rewarded by 12.5 Bitcoins. There was a bull run of the Bitcoin price in late 2017, which almost reached $20,000. Even though the Bitcoin price is highly volatile, it is still around $8,000 in May 2020 (Blockchain.com contributors, 2020).

[2] A miner's winning probability is proportional to the mining capacity she possesses.

https://doi.org/10.1016/j.cor.2021.105365
Received 1 August 2020; Received in revised form 3 April 2021; Accepted 30 April 2021
Available online 19 May 2021

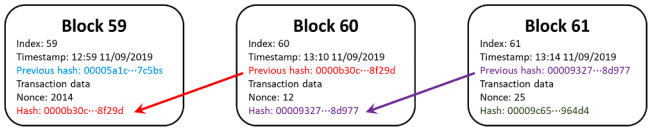

Fig. 1. A demonstration of blockchain: the Bitcoin blockchain.

1.1. Background of the Bitcoin blockchain

Blockchain is a digital, append-only, timestamped ledger with the following key features: decentralization, immutability, security, and transparency. As shown in Fig. 1, each block contains the index, timestamp, previous hash, hash, and other information. Hash refers to the output of a cryptographic function with other data as input,

$$f(\text{index}, \text{timestamp}, \text{previoushash}, \text{etc.}) = \text{hash},$$

which is a mathematical algorithm that maps data of arbitrary size to a bit string of a fixed size, and is designed noninvertible.[3] The previous hash of current block points to the hash of the previous one, constituting the single-chain structure.

Blockchain technology has been widely applied in different industries, such as cryptocurrencies, banking and payments, supply chain management, voting, and online music. Various mechanisms and consensuses are designed depending on the diverse needs of implementations, which generates different types of blockchains. Based on different levels of accessibility and authorization, blockchains could be mainly classified into four types: public permissionless, public permissioned, private permissionless, and private permissioned (Peters and Panayi, 2016; Wan, 2020). Specifically, the public and permissionless blockchain realizes the full decentralization and gets rid of trust, which is used by a majority of cryptocurrencies.

Bitcoin is the most well-known cryptocurrency and has been highly discussed since 2017, thanks to its controversiality, e.g., the high valuation and huge energy consumption. In the Bitcoin network, the decentralized community of individual participants displaces the trusted third party in traditional centralized systems. Contrary to conventional distributed systems employing Byzantine fault tolerance consensus (Bracha and Toueg, 1985; Correia et al., 2011), the novelty of the Bitcoin system comes from that it achieves a consensus of the ordering and confirmation of transactions among untrusted distributed participants through a *mining* game, which is referred to as the Nakamoto consensus. Specifically, participants of the mining game, also called *miners*, compete with each other to first solve a extremely complex computation puzzle: search for an appropriate *nonce* (one of the inputs) of the SHA-256 function[4] by brute force, in order to make the output hash value below *target* (a certain threshold given by the system). See Fig. 1. The process is called *proof of work* (PoW). The winner will obtain the confirmation right of the next block of transactions, and more importantly, earn the corresponding payoff. The incentive mechanism of the system is designed as follows: winning miners of each mining game are rewarded with a certain amount of newly minted Bitcoin (12.5 Bitcoins) and transaction fees from general users as compensation.

Continuous participation in the mining activity incurs a nontrivial amount of costs, consisted of hardware procurement, maintenance cost, and utility cost. To solve the cryptographic puzzle is excessively energy-consuming, so the electricity bill constitutes the majority of all costs. Nevertheless, incentivized by the massive reward of the mining competition, there have been nearly 10,000 active Bitcoin network nodes (participants) on a daily basis in recent years (Yeow, 2020; Coin Dance contributors, 2020). These miners conduct the daily operations, such as minting new coins and recording transactions, to support the system. It is well accepted that solo-mining will no longer be able to sustain any profitable mining activities in the current days. In order to win mining competitions more steadily, profit-driven individual miners have conglomerated to form mining pools. Miners in the same mining pool collaborate with each other to compete in the mining contest, and more importantly, share the mining reward. Forming mining pools do not increase the mean number of competitions that a miner wins, but it does reduce the variance (Rosenfeld, 2011). The mining pool collects hashing power and provides the infrastructure for this collaboration. In return, the mining pool charges a membership fee from each applicant and determines the reward allocation policy among miners. This creates interplays between individual miners and mining pool managers, which we provide more details in later sections.

1.2. Literature review

The Bitcoin system and its underneath technology, i.e., blockchain, have caught much attention of researchers and scientists in the latest years. As powerful analytical tools, game-theoretic approaches are widely applied to characterize interactions among different players in the Bitcoin/blockchain system. The current related research mainly focuses on the security (Liao and Katz, 2017; Wu et al., 2019) and mining activities (Lewenberg et al., 2015; Tsabary and Eyal, 2018) of the system. Meanwhile, the collaboration of blockchain and machine learning could be efficient and effective, including supervised learning (Yin and Vatrapu, 2017; Dey et al., 2020), unsupervised learning (Akcora et al., 2018; Abay et al., 2019) and reinforcement learning (Liu et al., 2018; Nguyen et al., 2020). Additionally, the operations research (OR) society is also getting interested in this emerging technology. Cretarola and Figà-Talamanca (2019), Koutmos (2019) and Atsalakis et al. (2019) investigated the cryptocurrency price. Kawase and Kasahara (2017) and Huberman et al. (2019) employed queuing theory to study transactions in the Bitcoin system. Finally, considering the core of our work is the simulation-based sequential mining decision making to optimize players' utilities, in this subsection, we mainly review recent blockchain research from the point of simulation's view. Two main simulation methodologies are employed: agent-based and discrete-event models.

Agent-based simulation. Kaligotla and Macal (2018) provided a generalized framework of modeling blockchain simulation by illustrating the essential agents and functioning of the system. Cocco and Marchesi (2016) reproduced the economy of the mining process with heterogeneous agents by presenting a complex artificial cryptocurrency market model with the Bitcoin transactions and price series. Later, Cocco et al. (2019) studied the trading of the currency pair BTC/USD by applying a *genetic algorithm* in this artificial market, where there are two types of agents called *Chartists* and *Random Traders*. Rosa et al. (2019) developed a security attack testing platform with extended scalability by taking advantage of *parallel and distributed simulation* techniques. Intimated by the design of algorithmic trading and reinforcement learning systems, Chitra et al. (2019) built a multi-agent simulator to model censorship properties in parallelized PoW chains. They showed how endogenous design choices affect practical protocol performance and how simulations can interact with exogenous data. Brousmichc et al. (2018) built an agent-based framework to simulate the local energy marketplace by integrating realistic consumption/production behaviors and interacting with a private blockchain network. Bottone et al. (2018) developed an extendable agent-based simulation model for a block-free and fee-less distributed ledger, in which they exploited NetLogo to provide a 3D visualization of the *Tangle* (Popov, 2016). With the integration of *inverse reinforcement learning*, Lee et al. (2018) proposed a novel agent-based model to predict the movement of Bitcoin price. Using agent-based modeling and simulation technique, Wei et al. (2020) compare different consensus protocols and trade network topologies quantitatively.

Discrete-event simulation. To investigate the large-scale

[3] Even a tiny alteration of the input will generate a totally different hash output.

[4] From the *secure hash algorithm* (SHA) family, its output is 256-bit.

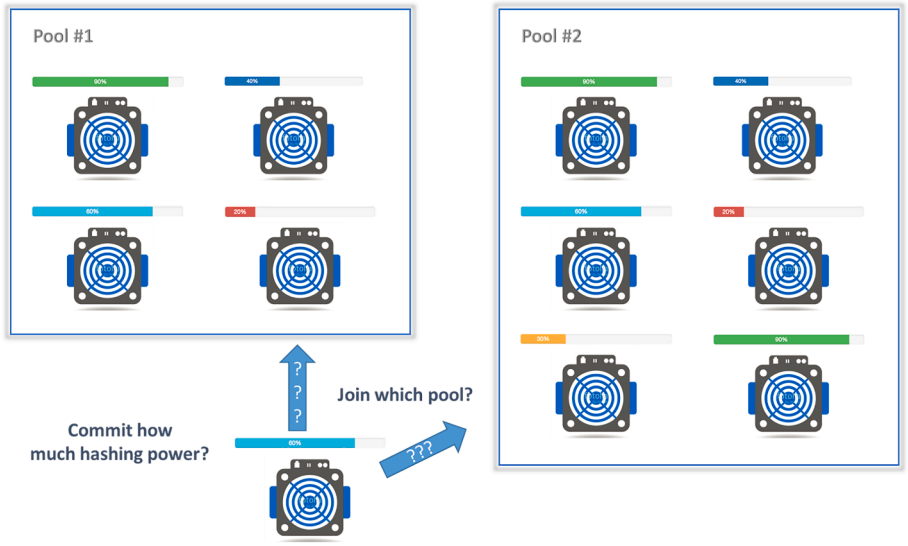

Fig. 2. An illustration of the two-layer miner and mining pool decision model.

blockchain networks, Wang et al. (2018) collected and defined a number of performance metrics to quantify the *quality of blockchain*. A queuing theory-based model is built by Memon et al. (2019) to characterize the realistic behaviors of both a memory and a mining pool for any blockchain system. Utilizing a generalized representation of consensus protocols, Foytik et al. (2020) presented a blockchain simulator to provide insights into the performance of the consensus protocols under different networking conditions. Alharby and van Moorsel (2019) proposed a discrete-event simulation with transactions and emphasized on the generation of block via PoW. Aoki et al. (2019) involved the events of block creation, block propagation, and message communication. Miller and Jansen (2015) enabled the scalable execution of thousands of Bitcoin nodes on a single machine in their work and included the *denial-of-service* attack to demonstrate the proposed simulator. Göbel et al. (2016) used an event-driven model to study the *selfish-mining* attack under a network with communication delay among miners. Gervais et al. (2016) studied optimal countermeasures for *double-spending* and *selfish mining* attacks based on Markov decision processes. They constructed a Bitcoin blockchain simulator to analyze the security and performance of different configurations.

1.3. Contributions and organizations

Our contributions. In this paper, we develop a practical discrete-event simulation model to study the dynamics of the Bitcoin blockchain. One realistic feature of our model is the inclusion of the individual budget constraint during the Bitcoin mining competition. In addition, we embed more details of the process of block production through share submissions within mining pools. Meanwhile, different share-based pool reward policies are also considered. To investigate the miner behavior, two crucial decisions are involved, i.e., mining and pool selection (Fig. 2). Aiming to provide a platform with comprehensive functionalities and configurations of the Bitcoin system, the proposed model also includes the membership fee adjustment and the adaptive difficulty mechanism. Furthermore, an alternative mining policy for the miner with insufficient mining budget is proposed and tested, and the emergence and behavior of the monopolist of the Bitcoin mining market are studied when eliminating the limitation of pool capacity.

Some interesting results and conclusions are generated from the simulation.

- For individual miners, our result shows that the Bitcoins gained by them are proportional to their mining capacities and budget volumes restrict mining behaviors. By applying different pool reward allocation policies, it validates that *pay per share* (PPS) policy could bring the most steady incomes, *pay per last N shares* (PPLNS) policy the second, and *proportional system* (PROP) policy the least.
- For mining pools, we find that an initial oligopolistically distributed mining market does not eventually develop into a monopoly in the course of time. On the other hand, it develops into an oligopoly with identically initialized pools. We also observe that medium-size mining pools can attract more individual miners than both small-size and large-size ones; this may provide some guidelines to the manager/operator of the emerging Bitcoin mining pool.
- From the point of view of the system, we validate that the dynamic adjustment of the mining difficulty is effective in terms of maintaining a stable block generation rate. Moreover, we reveal an interesting relationship between the overall hashing power implementation and difficulty level; such a relationship may help an individual miner to predict the dynamics of the total hashing rate of a new campaign by observing the change of difficulty level, in order to optimize her mining strategy. Furthermore, we find that more individual miners introduce higher stochasticity to the system.
- By extending the simulation model, we provide conditions for the alternative policy to be effective and observe the *herd behavior* introduced by the policy. Additionally, the factors affecting the monopolist of the mining market are investigated.

Organization of the paper. The remainder of this paper is organized as follows. Model settings, assumptions and detailed algorithms are presented in Section 2. Section 3 gives input data and parameters, presents experiment results, and explores extensions of the simulation. Finally, we give concluding remarks and discuss future research opportunities in Section 4.

2. Simulation model

2.1. Multi-layer model scope

Our simulation model aims to explore what influences the collective and heterogeneous behaviors of miners and mining pools.

Layer 1: individual miner. The first and most fundamental layer of decision is individual miners. Because they participate in mining to gain profits, miners will make their mining decisions based on their expected profit. That is, miners continue to stay in the contest if they can sustain a positive gain, and leave if otherwise. For each block appended to the

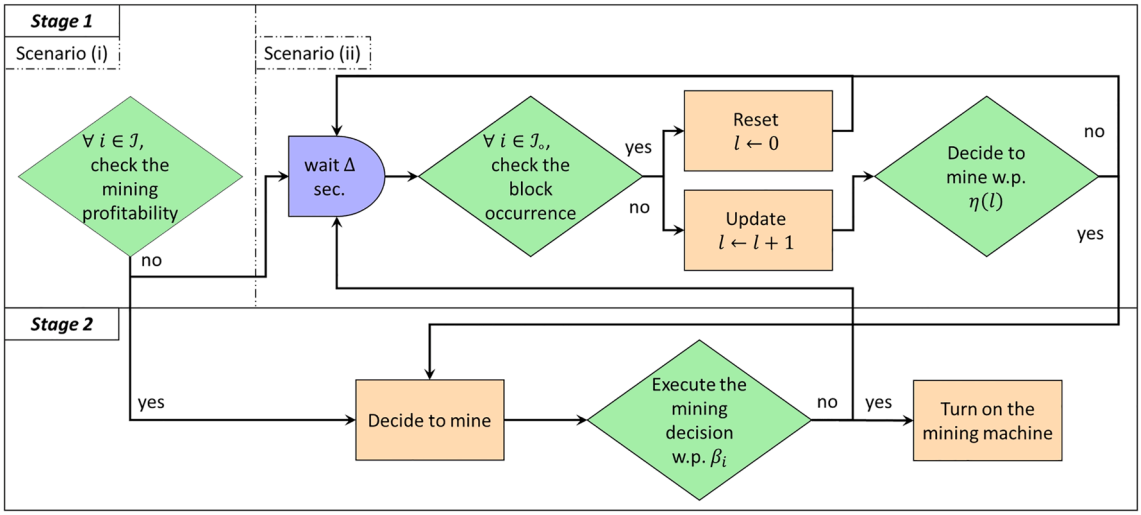

Fig. 3. The two-stage individual mining policy.

main chain, the reward of the winner is composed of two parts: transaction fees (roughly $1 \sim 2$ Bitcoins) and the Coinbase reward (currently 12.5 Bitcoins). The costs of mining include hardware costs and utility costs. The fixed cost of mining hardware ranges from \$50 to $\$10,500$ in the market. According to Morgan Stanley 2017 data, the total energy consumption of the Bitcoin network is equivalent to the total electricity supporting 2 million U.S. homes.

In addition to the mining decision (i.e., *to mine or not to mine*), miners will also decide which mining pool to join. Although the mining was first visioned to be performed by personal computers, over the years, individual and group miners have conglomerated to form mining pools, because mining pools provide a more steady income stream. On the other hand, mining pools charge membership fees, which is around 1% $\sim$ 3% of the total reward for the main-stream ones. Furthermore, depending on the reward-sharing policy, joining a mining pool affects the reward and cost structure for individual miners.

Layer 2: mining pool. Mining pool managers form the second layer in the Bitcoin blockchain network. We consider the case where the mining pool only serves as a centralized collaboration platform for miners, which is the direct opposite of the original Bitcoin design as a decentralized network. The primary objective in managing a mining pool is to make a profit, which consequently requires the manager to balance the incoming of new miners and the departure of old ones. New miners bring along hashing capacities, which increase the winning probabilities. However, should there exist any mining pool whose total hashing power is large enough to dominate the mining competition, Bitcoin participants will inevitably question the credibility of the system, which may result in the abandonment of network supporters and eventually the collapse of the entire network. This would deprive the purpose and economic opportunities of a mining pool. Given Bitcoin system states and available hashing power, pool managers aim to set the proper membership fees and reward policies.

Metrics of the Bitcoin system. We monitor the system-level dynamics, such as difficulty level and average block generation times, while individual miners and mining pool managers are making decisions on different levels. At the beginning of each 2-week period, the hashing difficulty level will be automatically adjusted; the goal of this is to keep the block generation rate steady.

2.2. Model assumptions

In our simulation model, players (i.e., individual miners and pool managers) participate in a campaign-repeated tournament. We set the maximum of campaigns to be w and the counter of campaigns to be W. Within each campaign, there are exactly $n = 2,016$ valid blocks generated. We denote by X^N the time to first generate the N^{th} valid block in a single campaign. By Satoshi's design, mining difficulty[5] is updated at the end of the W^{th} campaign by

$$D^{W+1} = D^W \frac{600n}{\sum_{N=1}^{n} X^N} \tag{1}$$

with D^W representing the difficulty of the W^{th} campaign. The scaling is to maintain a nearly constant block generating rate (1 per 600 s on average), so the time length of each campaign is roughly 2 weeks (Narayanan et al., 2016). The residual time of the current campaign is estimated by $\widehat{T}(N) = 600(n - N)$. Let $\mathscr{I}$ and $\mathscr{J}$ denote the sets of miners and pools. Furthermore, we denote by $\mathscr{I}_\circ$ the set of "idle" miners, $\mathscr{I}_p$ the set of passive miners, and $\mathscr{I}_a = \mathscr{I} \setminus \mathscr{I}_p$ the set of active miners, which will be explained later.

Assumptions on miner behavior. From the perspective of miner $i \in \mathscr{I}$, she is characterized by a type vector $\theta_i = (b_i, c_i, \gamma_i, p_i)$:

- b_i: the mining budget (\$) within a campaign (correspondingly, B_i is her residual budget (\$) in the campaign);
- c_i: the mining cost (\$/hash);
- γ_i: the individual valuation parameter of Bitcoin;
- p_i: the maximal mining power (hash/s).

An individual miner's first decision is to mine or not to mine. Let q_i be binary variable, we write $h_i = q_i p_i$ as the mining power (hash/s) an individual miner invests in mining. The "idle" miner set is formally defined as $\mathscr{I}_\circ = \{i \in \mathscr{I} | q_i = 0\}$. We denote by $\widehat{T}_i = B_i/(c_i h_i)$ the estimated residual time until exhausting the budget under the mining policy h_i, $\widehat{T} = \min_i\{\widehat{T}_i\}$, and $I_\circ = \text{argmin}_i\{\widehat{T}_i\}$. When a miner is making a decision, she may face the following two scenarios.

(i) When a block is mined and broadcast to the whole network, each miner will decide to turn on her mining machine if it is profitable to participate in the mining of the next block in expectation.
(ii) In addition to Scenario (i), each "idle" miner $i \in \mathscr{I}_\circ$ will periodically check if a new block is released (this occurs once every Δ seconds). After l consecutive attempts with "negative" outcomes, the

[5] A measure of how difficult it is to find a hash value below a given target.

minor will, with probability $\eta(l)$, decide whether to run her mining machine. We assume $\eta(l)$ is increasing in l.

This monitoring mechanism mentioned in Scenario (ii) above can (a) maintain an acceptable fraction of open mining machines at all times, and (b) help an "idle" miner to closely track the trend of the overall hashing rate (regardless of other factors, such as Bitcoin market price, it can be a promising opportunity to participate in mining when the overall hashing rate is low). If a miner decides to participate in mining the next block, she will execute the mining decision with probability

$$\beta_i = \min\left\{\frac{B_i}{\widehat{T}\left(N\right)c_i p_i}, 1\right\}.$$

A miner turns on the machine if her residual budget can cover the estimated mining expenditure (i.e., $\widehat{T}(N)c_i p_i$) until the end of the campaign. Because all budgets will be refilled at the beginning of a new campaign, miners can be more aggressive in executing mining decisions towards the end of the current campaign. We summarize the whole process of individual mining to a *two-stage policy* in Fig. 3. For a miner i, the time to generate the N^{th} block (i.e., X_i^N) after updating difficulty is exponentially distributed (Narayanan et al., 2016) with rate

$$\mu_i = \frac{h_i}{D^W/D^*},$$

where D^* is the minimal difficulty. Since we also include the pool hopping decision in the model, we use $P_i \in \mathscr{J}$ for miner i's pool index. Moreover, we assume that miner i's valuation of a Bitcoin V_i follows a continuous distribution $F_V(\cdot;\Gamma,\gamma_i)$, parameterized by the exogenous market valuation Γ and her own valuation factor γ_i. In particular, $V_i = \widetilde{V}_i \mathbf{1}_{\{\widetilde{V}_i \geqslant 0\}}$ and $\widetilde{V}_i \sim \mathscr{N}(\Gamma, \gamma_i^2)$, where Γ and γ_i are estimated by the historical data of Bitcoin market price (see Table 3).

Following Salimitari et al. (2017), we apply *prospect theory* (Kahneman and Tversky, 1979; Liu et al., 2014) to model the loss and risk aversion nature of miners during the pool hopping process. Suppose a miner with expected profit x joins a pool with mining power share y, her utility is given by

$$U(x, y; \lambda_i, \phi_i, \omega_i, \rho_i) = V(x; \lambda_i, \phi_i) \cdot W(x, y; \omega_i, \rho_i). \tag{2}$$

The value function $V(x; \lambda, \phi)$ characterizes the *reflection effect*[6]. In particular, it has the form of

$$V\left(x; \lambda_i, \phi_i\right) := \begin{cases} x^{\phi_i} & \text{if } x \geqslant 0 \\ -\lambda_i(-x)^{\phi_i} & \text{otherwise} \end{cases}$$

with parameters $\lambda_i > 1$ and $0 < \phi_i < 1$. Moreover, to include the effect of the mining power distribution among pools, the weight function $W(x, y; \omega_i, \rho_i)$ is used to account for the *certainty effect*[7]. In particular,

$$W\left(x, y; \omega_i, \rho_i\right) := \begin{cases} y^{\rho_i}[y^{\rho_i} + (1-y)^{\rho_i}]^{-1/\rho_i} & \text{if } x \geqslant 0 \\ y^{\omega_i}[y^{\omega_i} + (1-y)^{\omega_i}]^{-1/\omega_i} & \text{otherwise} \end{cases}$$

with parameters $0.5 \leqslant \omega_i < \rho_i < 1$.

Assumptions on pool policies. We denote by F_j^W the membership fee of the W^{th} campaign for the j^{th} mining pool from set $\mathscr{J}$ (i.e., a proportion of the total reward set by the pool manager). We will update the membership fee based on *shares*[8]. The submitted shares can be used to statistically measure the computational power a miner/pool controls (Liu and Liu, 2019). We assume that $F_j^1 = F_j^2 \sim \mathscr{U}(a, b)$, and

$$F_j^{W+1} = F_j^W + \alpha_j \mathbf{1}_{\{S_j^W / S_j^{W-1} > 1+\epsilon\}} - \alpha_j \mathbf{1}_{\{S_j^W / S_j^{W-1} < 1-\epsilon\}}, \quad W \geqslant 2, \tag{3}$$

where S_j^W counts the total shares produced by the j^{th} pool within the W^{th} campaign, the constant $\epsilon \in (0,1)$ controls the sensitivity in the change of share numbers, and α_j is the step size for the membership fee adjustment. If the production of shares in a specific pool changes significantly, which leads to a notable fluctuation of its mining power, the manager will take action to adjust the membership fee. A variety of mining pool reward policies has been proposed (Rosenfeld, 2011; Cong et al., 2019; Bitcoin Wiki contributors, 2020). In this simulation, we apply the following three share-based schemes most commonly adopted in practise (Qin et al., 2018; Qin et al., 2019).

- PROP: at the end of every *round*[9], the pool manager will distribute the block reward among miners, in direct proportion to the number of shares they submitted during this round.
- PPLNS: instead of using the total number of shares in a round, the pool manager focuses on the last "N" shares, regardless of round boundaries.
- PPS: unlike the above two polices, a miner will get instant compensation according to the expected value of a submitted share's contribution.

To implement these two policies, we introduce the following additional parameters: the difficulty discount factor of the j^{th} pool δ_j; the current difficulty to generate a valid share for the miners in the j^{th} pool $D_j^W = D^W/\delta_j$; the number of shares the i^{th} miner needs to get a valid block S_i^b, which is a geometric random variable with parameter $1/\delta_{P_i}$; the miner's inter-share times $\{X_{ik}^s\}_{k=1,\cdots,S_i^b}$, which are independent exponential random variables with rate

$$\nu_i = \frac{h_i}{D_{P_i}^W / D^*} = \delta_{P_i}\mu_i.$$

System level metrics. On the system level, we record the following metrics:

- X^N: the time first generating the N^{th} valid block in a campaign, $X^N = \min_i\{X_i^N\}$;
- I^N: the index of the miner first finds the N^{th} valid block in a campaign, $I^N = \text{argmin}_i\{X_i^N\}$;
- J^N: the index of the mining pool that the miner I^N comes from, $J^N = P_{I^N}$;
- bd_j^W: the proportion of blocks of the j^{th} pool in the W^{th} campaign;
- md_j^W: the proportion of miners of the j^{th} pool in the W^{th} campaign;
- pd_j^W: the proportion of mining power capacity of the j^{th} pool in the W^{th} campaign;
- $hd_j(\tau)$: the proportion of mining power committed in the j^{th} pool at system time τ.

Other assumptions. Furthermore, we list other assumptions as follows.

- The individual valuation of Bitcoin V_i is independent.
- Considering a significant portion of hashing power belongs to the pool manager herself in practice, we introduce the passive miner set $\mathscr{J}_p$ (Cong et al., 2019): passive miners stick to the same mining pool

[6] People are risk-averse over gains; people are risk-seeking over losses.

[7] In practise, people tend to over-react to lower probabilities (resp. smaller pools) and under-react to higher probabilities (resp. larger pools).

[8] A share is a partial solution to the original puzzle of generating a valid block, which is corresponding to a lower difficulty.

[9] The time elapsed between two valid blocks successfully mined by the same pool.

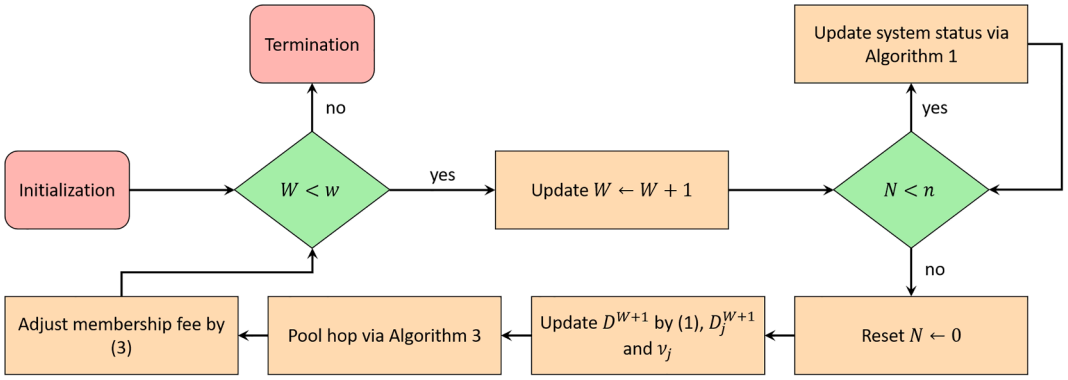

Fig. 4. Main routine of simulation.

instead of hopping periodically. We assume that a miner is passive with probability π_p.

- Pool hopping with probability: at the end of each campaign, a miner prioritizes pools and joins the k^{th} one with probability hp_k.
- Miners pool-hop at the end of each campaign, and the hopping time window is negligible.
- To avoid the collapse of the trustworthiness of the Bitcoin system caused by the over-centralization of the hashing power, each mining pool sets an upper threshold *UB* of pool capacity.
- We ignore transaction fees included in each block, which are negligible compared with the Coinbase reward.
- We ignore the fixed cost of purchasing mining machines and the depreciation of hardware.

2.3. Simulation algorithms

We present detailed simulation algorithms in this subsection. The framework of the simulation is illustrated in Fig. 4. We describe the following *event list*:

- t_A: the time point of the next block generation;
- t_B: the time point of the next block check;
- t_C: the time point of the next miner to run out her budget.

The system state is updated by the *next-event time advance* approach in Algorithm 1.

Algorithm 1 Subroutine: update system status

switch $t \leftarrow \min\{t_A, t_B, t_C\}$
  case $t_A = t$ (The next event is block generation.)
    Reset $l \leftarrow 0$, update $N \leftarrow N+1$.
    if $N = n$ **then** reset $B_i \leftarrow b_i, i \in \mathscr{I}$; else update $B_i \leftarrow B_i - c_i h_i(t_A - \tau), i \in \mathscr{I}$. **end if**
    Update $\beta_i \leftarrow \min\left\{\frac{B_i}{\widehat{T}(N)c_i p_i}, 1\right\}, i \in \mathscr{I}$; $\tau \leftarrow t_A$.
    Distribute block reward within pool J^N.
    Update $\mathbb{P}_i$ by (4), $\mathbb{E}[R_i] \leftarrow 12.5(1-F_{P_i})V_i\mathbb{P}_i - 600c_i p_i, i \in \mathscr{I}$.
    if $\mathbb{E}[R_i] > 0$ **then** update $q_i \leftarrow 1$ w.p. β_i; $q_i \leftarrow 0$ w.p. $(1-\beta_i)$.**end if**
    Update $\nu_i \leftarrow \frac{q_i p_i}{D^c_{P_i}/D^m}, i \in \mathscr{I}$.
    Update $\widehat{T}_i \leftarrow \frac{B_i}{c_i h_i}, i \in \mathscr{I}$; $\widehat{T} \leftarrow \min_i\left\{\widehat{T}_i\right\}, I_* \leftarrow \operatorname{argmin}_i\left\{\widehat{T}_i\right\}$.
    Update $X_i^{N+1}, i \in \mathscr{I}$. (See details in Algorithm 2.)
    Update $X^{N+1} \leftarrow \min_{i:X_i^{N+1} \leqslant \widehat{T}_i}\{X_i^{N+1}\}, I^{N+1} \leftarrow \operatorname{argmin}_{i:X_i^{N+1} \leqslant \widehat{T}_i}\{X_i^{N+1}\}, J^{N+1} \leftarrow P_{I^{N+1}}$.
    Update $t_A \leftarrow \tau + X^{N+1}, t_B \leftarrow \tau + \Delta$.
  end case
  case $t_B = t$ (The next event is block check.)
    Update $l \leftarrow l+1, \mathscr{I}_* \leftarrow \{i \in \mathscr{I} | q_i = 0\}$.
    Update $B_i \leftarrow B_i - c_i h_i(t_B - \tau), \beta_i \leftarrow \min\left\{\frac{B_i}{\widehat{T}(N)c_i p_i}, 1\right\}, i \in \mathscr{I}$; $\tau \leftarrow t_B$.

(*continued on next column*)

(*continued*)

    Update $q_i \leftarrow 1$ w.p. $\beta_i\eta(l)$; $q_i \leftarrow 0$ w.p. $(1-\beta_i\eta(l)), i \in \mathscr{I}_*$. Update $\nu_i \leftarrow \frac{q_i p_i}{D^c_{P_i}/D^m}$, $i \in \mathscr{I}_*$.
    Update $\widehat{T}_i \leftarrow \frac{B_i}{c_i h_i}, i \in \mathscr{I}$; $\widehat{T} \leftarrow \min_i\left\{\widehat{T}_i\right\}, I_* \leftarrow \operatorname{argmin}_i\left\{\widehat{T}_i\right\}$.
    Update $X_i^{N+1}, i \in \mathscr{I}_*$. (See details in Algorithm 2.)
    Update $X^{N+1} \leftarrow \min_{i:X_i^{N+1} \leqslant \widehat{T}_i}\{X_i^{N+1}\}, I^{N+1} \leftarrow \operatorname{argmin}_{i:X_i^{N+1} \leqslant \widehat{T}_i}\{X_i^{N+1}\}, J^{N+1} \leftarrow P_{I^{N+1}}$.
    Update $t_A \leftarrow \tau + X^{N+1}, t_B \leftarrow \tau + \Delta$.
  end case
  case $t_C = t$ (The next event is one miner runs out her budget.)
    Update $B_i \leftarrow B_i - c_i h_i(t_C - \tau), \beta_i \leftarrow \min\left\{\frac{B_i}{\widehat{T}(N)c_i p_i}, 1\right\}, i \in \mathscr{I}$; $\tau \leftarrow t_C$.
    Update $q_{I_*} \leftarrow 0, \nu_{I_*} \leftarrow 0$.
    Update $\widehat{T}_i \leftarrow \frac{B_i}{c_i h_i}, i \in \mathscr{I}$; $\widehat{T} \leftarrow \min_i\left\{\widehat{T}_i\right\}, I_* \leftarrow \operatorname{argmin}_i\left\{\widehat{T}_i\right\}$.
  end case
  Update $t_C \leftarrow \tau + \widehat{T}$.
end switch

Suppose miner i turns on her machine, the individual winning probability is estimated by

$$\mathbb{P}_i = \frac{\widetilde{\mu}_i}{\widetilde{\mu}_i + \widehat{\mu}}, \quad \text{where} \quad \widetilde{\mu}_i = \frac{p_i}{D^W/D^e} \quad \text{and} \quad \widehat{\mu} = \frac{N}{\sum_{k=1}^{N} X^k} \tag{4}$$

are the individual exponential rate when turning on her machine and the estimated overall mining rate, respectively. Then we can use $\mathbb{P}_i$ to update the expected profit of the miner in Step 7 of Algorithm 1.

We next show how miner i produces a valid block through submitting shares via Algorithm 2.

Algorithm 2 Subroutine: generate individual inter-block time via inter-share time

Generate $S_i^b \sim \text{Geo}(1/\delta_{P_i})$.
Generate $\{X_{ik}^s\}_{k=1,\cdots,S_i^b} \sim \exp(\nu_i)$.
Update the first inter-share time $X_{i1}^s \leftarrow X_{i1}^s + l\Delta$.
Update $X_i^{N+1} \leftarrow \sum_{k=1}^{S_i^b} X_{ik}^s$.

We finally discuss the pool hopping decision in Algorithm 3. Suppose the i^{th} miner joins the j^{th} pool and a new valid block is found by the specific pool, the average Coinbase reward share of the miner is estimated by

$$\mathbb{P}_{ij} = S_i^W \Big/ S_j^W \mathbf{1}_{\{j=P_i\}} + \frac{S_i^W \Big/ S_{P_i}^W bd_{P_i}^W}{bd_j^W + S_i^W \Big/ S_{P_i}^W bd_{P_i}^W} \mathbf{1}_{\{j \neq P_i\}}, \tag{5}$$

where S_i^W is the counter of shares generated by the i^{th} miner in the W^{th} campaign (the definition is similar to S_j^W). The proportion of shares yielded by the miner in the current pool is $S_i^W/S_{P_i}^W$, approximating the

Table 1
Typical ASICs in mining market.

	SHA-256 Mining Equipment	Hashing power (Th/s)	Efficiency (J/Gh)	Cost ($/Th)
1	MicroBT Whatsminer 10S	55	0.064	8.84E −07
2	Bitfily Snow Panther B1+	25	0.086	1.19E −06
3	Bitmain Antminer T9	13	0.126	1.75E −06
4	Canaan AvalonMiner 741	7	0.158	2.19E −06
5	Bitmain Antminer S7	5	0.273	3.80E −06

Table 2
An overview of top 9 Bitcoin mining pools and others.

	Mining pool	# of blocks	Percent
1	AntPool	27,026	17.2%
2	F2Pool	19,282	12.3%
3	BTC.com	17,488	11.2%
4	ViaBTC	12,100	7.7%
5	SlushPool	12,002	7.7%
6	BTC.TOP	11,256	7.2%
7	BTCC	10,586	6.8%
8	BitFury	8,754	5.6%
9	BW.COM	7,315	4.7%
10	Others	30,886	19.7%

ratio of hashing power she owns within the pool. The proportion of blocks produced by the j^{th} pool during the W^{th} campaign is bd_j^W, approximating the ratio of hashing power occupied by the pool. Then we can use $\mathbb{P}_{ij}$ to update the expected profit of miner i to join pool j in Step 1 of Algorithm 3.

Algorithm 3 Subroutine: pool hopping

1: Update $\mathbb{P}_{ij}$ by (5), $\mathbb{E}[R_{ij}] \leftarrow 12.5(1-F_{P_i})V_i\mathbb{P}_{ij} - 600c_ip_i, i \in \mathscr{I}_a, j \in \mathscr{J}$.
2: Update $u_{ij} \leftarrow U\left(\mathbb{E}[R_{ij}], bd_j^W; \lambda_i, \phi_i, \omega_i, \rho_i\right)$ by (2), $i \in \mathscr{I}_a, j \in \mathscr{J}$.
3: Order $\{u_{ij}\}_{j\in\mathscr{J}}$ to get the hopping priority $\{j_1^i, j_2^i, \cdots, j_k^i, \cdots, j_K^i\}$ such that $u_{ij_1^i} \geqslant u_{ij_2^i} \geqslant \cdots \geqslant u_{ij_k^i} \geqslant \cdots \geqslant u_{ij_K^i}$, where $K = |\mathscr{J}|, i \in \mathscr{I}_a$.
4: Update $P_i \leftarrow j_k^i$ w.p. $hp_k, i \in \mathscr{I}_a$.

3. Numerical experiments

3.1. Input data and parameters

Mining machines. Rauchs (2020) collects the information of more than 80 different SHA-256 mining equipments, among which, we list 5 popular *application-specific integrated circuits* (ASICs)[10] with distinct hashing power in Table 1. The mining cost of each machine is calculated based on the electricity price of 0.05 $/kWh. For details, see Rauchs (2020) and references therein. To initialize the mining capacity in the simulation, we assume a miner purchases her mining machine from Table 1 with equal probabilities (see Table 3).

Mining power distribution. From February 2016 to January 2019, there are in total 156,695 valid blocks generated. Wang et al. (2019) summarizes the distribution of valid blocks generated from top Bitcoin mining pools, which is the estimator of the mining power distribution over the network. We consider the top 9 pools and group all other minor pools and solo miners into the 10^{th} category, see details in Table 2. This shows that oligopoly indeed exists in the Bitcoin mining system: several players control a large proportion of the hashing power (Cong et al., 2019), but none of them can dominate the entire mining market solely (i.e., exceed the 50% threshold). Our model assumes that the i^{th} miner's pool index P_i is initialized by a discrete probability distribution estimated by the proportions in Table 2.

Input parameters are listed in Table 3.

Table 3
Summary of input parameter design.

	Parameter	Description
System	$r = 40$	Number of simulation replications
	$w = 15$	Number of campaigns in each replication
	$n = 2,016$	Number of blocks in each campaign
	$\Delta = 600$	Period of block check
	$\eta(l) = 0.35^{6-l}\mathbf{1}_{\{1\leqslant l\leqslant 6\}} + \mathbf{1}_{\{l>6\}}$	Probability of deciding to mine after l "negative" checks in a row
	$\Gamma = \$8,807.71$	Market valuation parameter of Bitcoin
Miner	$\lvert\mathscr{I}\rvert = 300$	Number of miners
	$\pi_p = 20\%$	Initialization probability to generate the passive set $\mathscr{I}_p$
	$\gamma_i = \gamma = \$1,490.16$	Homogeneous individual valuation parameter of Bitcoin
	(p_i, c_i) generated by $\mathscr{U}\{1, 5\}$	Maximal mining power and the corresponding mining cost
	$\xi_i = 200\%, 100\%, 50\%$ equally likely	Scaling factor of budget
	$b_i = 2,016{\cdot}600{\cdot}p_i{\cdot}c_i{\cdot}\xi_i$	Mining budget of a single campaign
	λ_i generated by $\mathscr{U}(1, 2)$	Loss aversion parameter of value function in prospect theory
	ϕ_i generated by $\mathscr{U}(0, 1)$	Risk aversion parameter of value function in prospect theory
	ω_i generated by $\mathscr{U}(0.5, 1)$	Parameter of weight function for loss in prospect theory
	ρ_i generated by $\mathscr{U}(\omega_i, 1)$	Parameter of weight function for gain in prospect theory
Pool	$\lvert\mathscr{J}\rvert = 10$	Number of mining pools
	$a = 0.01$	Lower boundary of the uniform distribution to initialize the membership fee
	$b = 0.03$	Upper boundary of the uniform distribution to initialize the membership fee
	$\alpha_j = \alpha = 0.002$	Step size of the membership fee adjustment
	$\epsilon = 0.002$	Sensitivity parameter of the membership fee adjustment
	$\delta_j = \delta = 1,000$	Homogeneous difficulty discount factor
	$s\delta_j = 2\delta_j$	Window size parameter "N" of PPLNS policy
	$hp_k =$ $\begin{cases} 0.8 & \text{if } k = 1 \\ 0.8(1 - \sum_{g=1}^{k-1} hp_g) & \text{if } 2\leqslant k\leqslant K-1 \\ 1 - \sum_{g=1}^{k-1} hp_g & \text{if } k = K \end{cases}$	Hopping probability to the k^{th} most profitable pool
	$UB = 0.40$	Upper threshold of pool capacity

[10] ASICs were the next step of development after CPUs, GPUs and FPGAs in Bitcoin mining hardware.

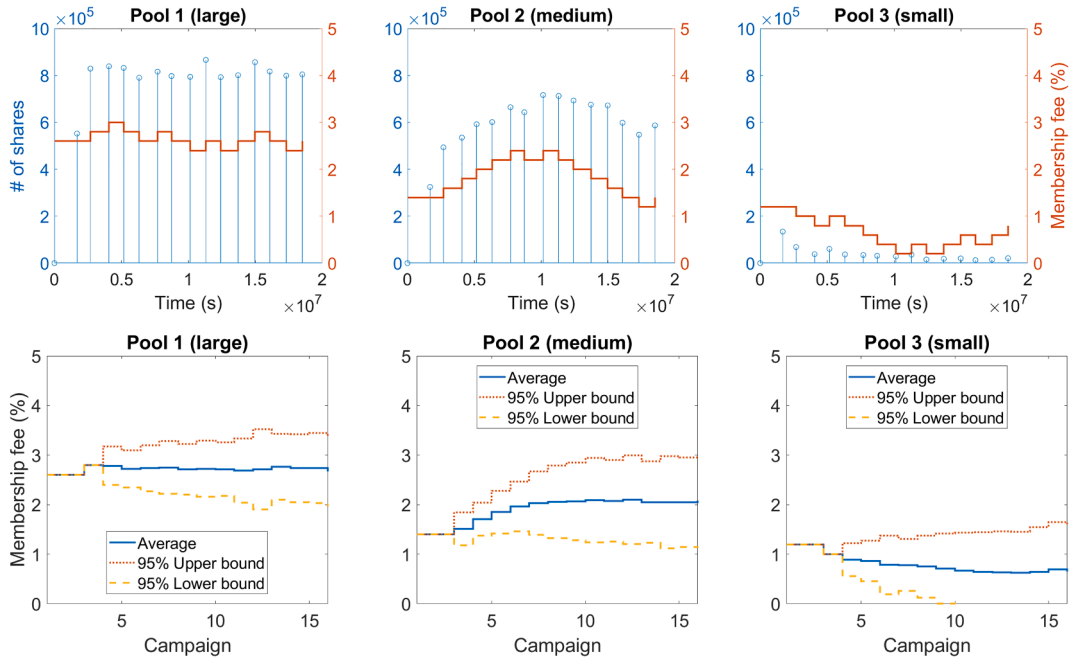

Fig. 5. Simulation of 3 types of pools: large, medium and small.

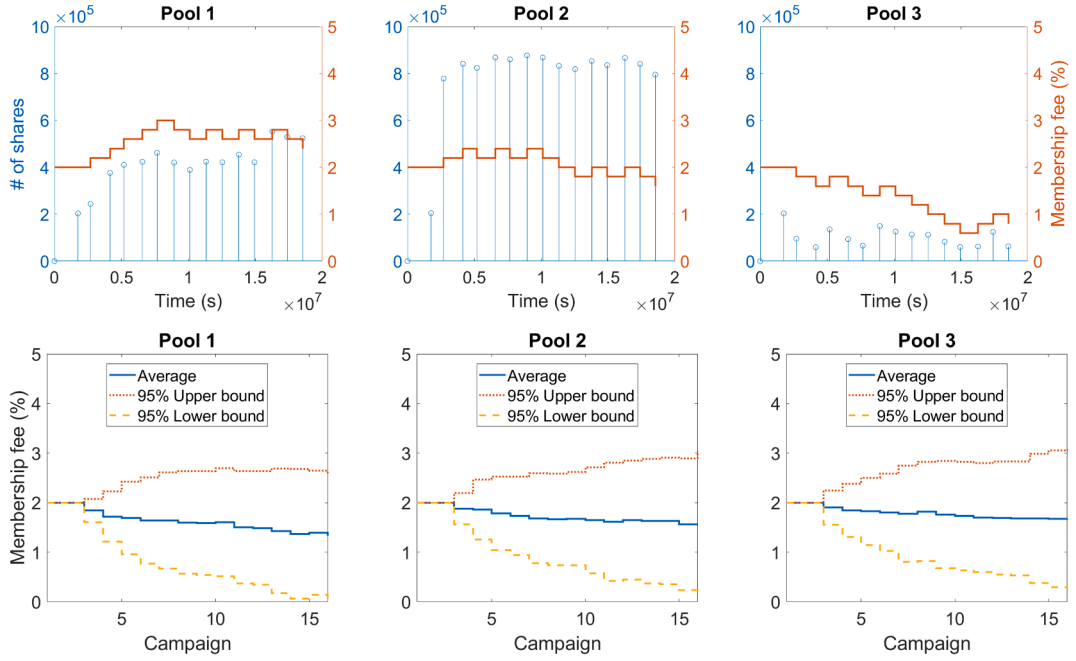

Fig. 6. Simulation of identically initialized pools (illustrate the first three pools).

3.2. Experiment results

We now presents our simulation results and discuss how they can be used to generate useful insights. The preliminary results are reported in the proceedings of the 2020 Winter Simulation Conference. Figs. 5 and 6 depict the dynamics of shares and membership fees from three representative pools[11], having large, medium and small mining capacities. Fig. 7 illustrates the dynamics of mining difficulty and the histogram of inter-block times. Fig. 8 shows the dynamics of the total hashing rate of the whole system.

3.2.1. Individual miners

We calculate the mean squared error (MSE) between the average proportions of Bitcoins gained by individual miners over replications and their corresponding shares of maximal mining power, and the average variance of Bitcoins earned of all miners in Table 4. The extremely small value of MSE indicates that the Bitcoins gained by miners are proportional to their mining capacities, even though they turn on machines probabilistically. The average variance under PPS policy has the smallest value. Considering a miner is immediately rewarded once upon the submission of a valid share in the PPS system,

[11] Full results of all mining pools are presented in A.

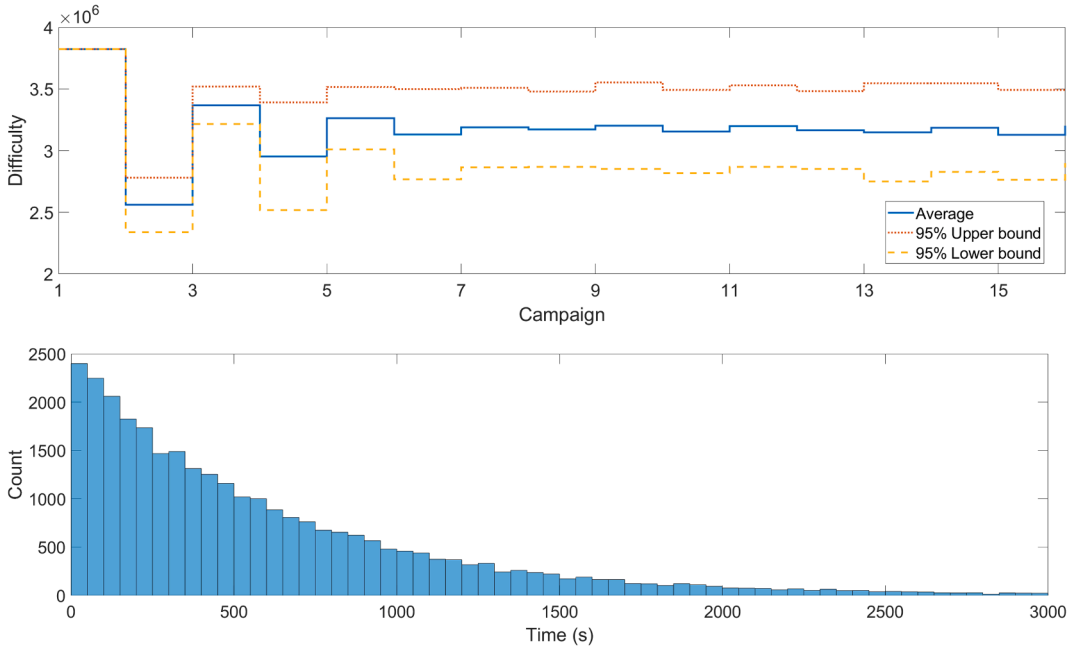

Fig. 7. Bitcoin mining difficulty level and inter-block time.

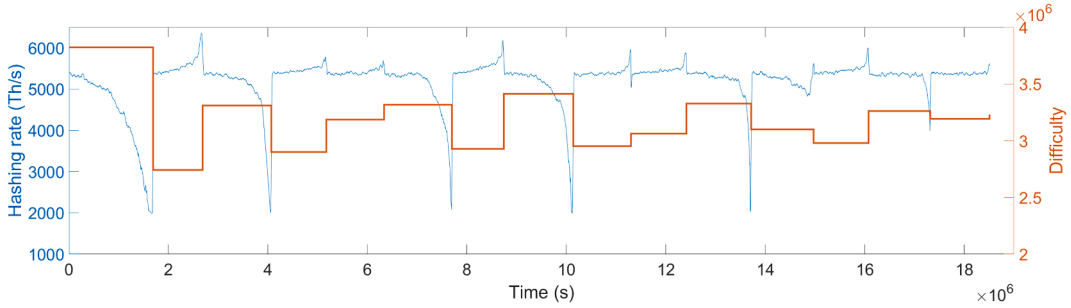

Fig. 8. The dynamics of overall hashing rate (100-point moving average, a single replication).

Table 4
Descriptive statistics of individual mining rewards.

	MSE	Average variance
PPS	1.44E−06	306.29
PPLNS	1.44E−06	881.14
PROP	1.46E−06	1039.17

Table 5
Proportions of time when running mining machines for 15 typical individual miners.

			ξ_i		
			50%	100%	200%
p_i	55	Mean	48.39%	92.41%	100.00%
		Std.	0.36%	1.52%	0.00%
	25	Mean	48.40%	92.41%	100.00%
		Std.	0.36%	1.52%	0.00%
	13	Mean	48.39%	92.41%	100.00%
		Std.	0.37%	1.52%	0.00%
	7	Mean	48.42%	92.41%	100.00%
		Std.	0.36%	1.52%	0.00%
	5	Mean	48.41%	92.41%	100.00%
		Std.	0.36%	1.52%	0.00%

the pool manager needs to absorb the risk associated with finding full solutions. In other words, PPS offers zero variance in the reward per share, but there is still some variance in the number of shares found by the miner in unit time (Rosenfeld, 2011). The average variance with PPLNS policy is less than the one with PROP, which validates the argument by Rosenfeld (2011). Additionally, we pick 15 typical miners with distinct hashing capacities (p_i's) and budget scaling factors (ξ_i's), and compute their time proportions when running mining machines. From Table 5, we can see, regardless of mining capacities, that miners' mining behaviors are restricted by their budget sizes in the simulation.

3.2.2. *Mining pools*

Given that pool managers apply the simple policy as in (3) for membership fee adjustment, it is not surprising to see that the membership fees and the number of shares are coping with each other (Fig. 5a). Furthermore, in the real Bitcoin mining competition, the mining power implementation is unobservable among players. Even in the same pool, the hashing rate is not transparent to the pool manager or individual miners. The number of valid shares submitted can be an effective estimator to measure the true hashing rate within the pool (Liu and Liu, 2019). So we conclude that the mining power dynamics in a pool could be inferred by the membership fee change. Because mining power is not equally distributed over the pools (Table 2), pools 1 and 2 own larger proportions of mining power, who yield the majority production of shares. Nevertheless, as time evolves, we do not observe any

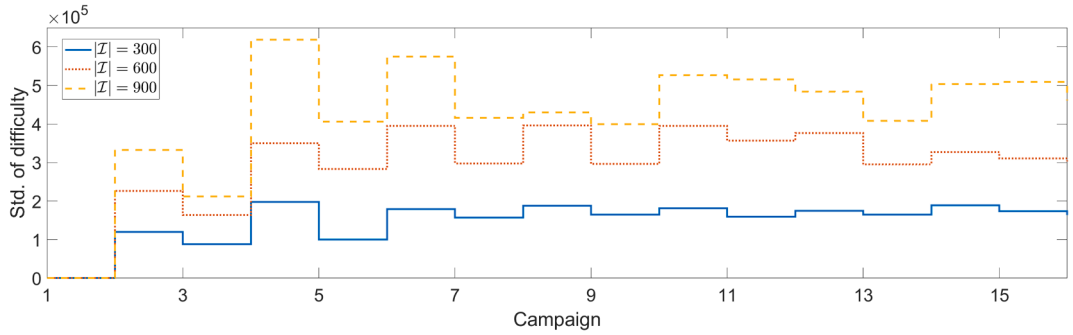

Fig. 9. Standard deviations of D^W over 40 replications with different numbers of total miners.

monopolistic structure. Contributing factors of this result are the finiteness of the pool capacity and the existence of passive miners.

From Fig. 5b, we can observe three different membership fee dynamics: (i) a steady curve in large-size pools (e.g., pool 1), (ii) an increasing trend in medium-size pools (e.g., pool 2), and (iii) a decreasing trend in small-size pools (e.g., pool 3). As mentioned before, the increase and decrease of the mining power occupied by each pool can be related to the corresponding changes of the membership fee. We give some explanations: the expansion of the large size pool is limited by the capacity threshold as well as the under-reaction of miners (modeled by prospect theory) during the pool selection process; on the other hand, the medium-size pool has a higher potential to attract individual miners. This observation may be able to provide guidelines for some consortium interested in setting up a new mining pool: sufficient initial mining power is essential to attract individual miners; however, the pool size cannot be too large, which prevents the system's mining power from being over-centralized and the blockchain network from collapsing due to miners' loss of trust.

Besides initializing pools based on the proportions of Table 2, we simulate the case that all pools have identically initial states: same miner compositions and same membership fees. In a single replication, although all pools are the same at the beginning, Fig. 6a reveals that several oligopolists (i.e., pools 1 and 2) are formed as time goes, since they generate most of the shares. By observing the dynamics of average membership fees over replications in Fig. 6b, they are all more unvarying than those with different initialized pools in Fig. 5b.

3.2.3. System metrics

Fig. 7a shows the convergence of the Bitcoin mining difficulty level as campaigns evolve. It validates the effectiveness of the adaptive difficulty mechanism designed by Satoshi. The block mining rate depends on the difficulty level and mining power committed by miners, which are updated every 2,016 blocks and fluctuated according to individual mining decisions in real time, respectively. As a result, new valid blocks occur according to a *nonhomogeneous* Poisson process (Bowden et al., 2018). Nevertheless, the histogram reported in Fig. 7b is similar to an exponential distribution. Hence, the new block arrival rate function $\mu(t)$ is steady thanks to the bi-weekly difficulty adjustment mechanism. On the other hand, we recognize that the estimated mean of inter-block time is 616.61 with standard deviation 643.49, which is slightly greater than 600, the idealistic value designed by Satoshi. This may be attributed to the delay in updating the difficulty, see Kraft (2016), Meshkov et al. (2017), Garay et al. (2017).

From Fig. 8, we observe an interesting relationship between the mining power allocation rule and the difficulty level dynamics: (i) if the current difficulty decreases significantly from the previous one, the overall hashing rate exhibits an upward spike when approaching the end of the current campaign; (ii) conversely, there is a downward spike towards the end of the current campaign. The reason for the first scenario is that the current difficulty level is too high for miners with respect to their mining capacities and budgets. Even though the mining activity is profitable, some miners have already exhausted their budgets. On the other hand, the current difficulty level in the second scenario is relatively low so that miners will be more risk-seeking (i.e., increasing β_i's) to execute mining decisions, in order to spend all residual budgets by the end of the campaign. This finding may help miners to predict the future allocation of the overall mining power by taking advantage of the dynamics of the mining difficulty, and to select the optimal timing to begin mining. For example, if a miner has a lower budget, she choose to apply an *alternative policy*: she could mine more actively at the beginning of a new campaign in Scenario (i); she should not turn on her machine idle until the end of a new campaign in Scenario (ii). We will discuss the alternative policy with more details in SubSection 3.3.

Table 6
Typical lower-budget miners with $\xi_i = 50\%$.

p_i (Th/s)	i			
	$(\mathscr{I}_1)$	$(\mathscr{I}_2)$	$(\mathscr{I}_3)$	$(\mathscr{I}_4)$
5	1	4	10	20
7	19	21	25	31
13	84	90	98	164
25	33	38	51	58
55	3	6	35	45

To supplement the case that the number of miners $|\mathscr{I}| = 300$, we also conduct experiments with $|\mathscr{I}| = 600$ and 900, and obtain similar results to those as illustrated in Figs. 5, 7 and 8. The standard deviation of Bitcoin mining difficulty level increases as the number of miners increases (see Fig. 9). To see this, note that an increased number of miners leads to higher system stochasticity, which in turn increases the vacillation of overall hashing rate and difficulty level.

3.3. Extensions

In this subsection, we discuss the alternative mining policy for the miner with insufficient mining budget and investigate the emergence and behavior of the monopolist when eliminating the limitation of pool capacity.

3.3.1. Alternative mining policy

Except for the "default" individual mining policy used in the simulation model (Fig. 3), we also introduce an *alternative policy* for some low-budget miners with $\xi_i = 50\%$ (Table 6): they start to turn on machines from the beginning until depleting budgets, if a campaign's difficulty level is lower than the previous; they only mine the blocks of the second half of the campaign, otherwise. To test the alternative policy, we need some new notations: the set of miners applying the alternative mining policy $\mathscr{I}_{ap}$; the proportion of mining power for miners applying the alternative policy P_{ap}; the relative increase of the Bitcoins gained by miner i from scenario S0 ζ_i; the relative increase of the Bitcoins gained by the miners applying the alternative policy from scenario S0 ζ_s; the

Table 7
Mining scenarios.

Scenario	$\mathscr{I}_{ap}$	P_{ap}
S0	∅	0.00%
S1_1	{1}	0.08%
S1_2	{19}	0.11%
S1_3	{84}	0.20%
S1_4	{30}	0.39%
S1_5	{3}	0.86%
S2	$\mathscr{I}_1$	1.65%
S3	$\mathscr{I}_1 \cup \mathscr{I}_2$	3.30%
S4	$\mathscr{I}_1 \cup \mathscr{I}_2 \cup \mathscr{I}_3$	4.95%
S5	$\mathscr{I}_1 \cup \mathscr{I}_2 \cup \mathscr{I}_3 \cup \mathscr{I}_4$	6.59%

average relative increase of the Bitcoins gained by each miner applying the alternative policy from scenario S0 $\overline{\zeta}$. Meanwhile, we define mining scenarios in Table 7.

The related results under different mining scenarios are shown in Table 8. We can see that the alternative policy indeed helps most of "testing" miners improve their payoff. Moreover, for a specific miner with an insufficient budget, her relative increase of Bitcoins has a trend of raising first and then dropping (e.g., the 6th and 7th rows of Table 8). The changes of ζ_s and $\overline{\zeta}$ also reflect similar trends. In other words, if more miners adopt the alternative policy, the marginal mining payoff will shrink. Recall that the alternative policy is designed according to the relationship between the mining power allocation rule and the difficulty level dynamics (Fig. 8). Combined with Fig. 10, the shrinkage of the marginal payoff is due to that the increasing value of P_{ap} changes the relationship. In addition, compared with Fig. 8, Figs. 10a and 10b present significantly different time series of overall hashing rate, even though the P_{ap}'s (4.95% and 6.59%, respectively) are comparatively small, which may be explained by the *herd mentality* of miners. In conclusion, when the overall hashing rate keeps the pattern in Fig. 8, the alternative mining policy might increase the advantage of a "poor" miner with sufficient hashing power (on the contrary, the miner with smaller hashing power will probably have less reward, see the 3rd row of Table 8).

3.3.2. *Monopolist of the Bitcoin mining system*

In our simulation model, the distribution of the overall mining power may potentially be influenced by

- the pool membership fee (F_j) adjustment mechanism,
- the upper threshold (UB) of pool capacity,
- the certainty effect of prospect theory (under-react to larger pools and over-react to smaller pools),
- and passive miners.

Table 8
Relative increases of Bitcoins gained by typical lower-budget miners from scenario S0.

Scenario	S1_1	S1_2	S1_3	S1_4	S1_5	S2	S3	S4	S5
P_{ap}	0.08%	0.11%	0.20%	0.39%	0.86%	1.65%	3.30%	4.95%	6.59%
ζ_1	−0.53%	–	–	–	–	−0.24%	0.05%	−1.05%	−1.10%
ζ_{19}	–	0.42%	–	–	–	0.85%	0.11%	0.55%	−0.03%
ζ_{84}	–	–	0.78%	–	–	0.59%	0.29%	0.50%	0.30%
ζ_{33}	–	–	–	0.97%	–	0.71%	1.06%	0.08%	0.08%
ζ_3	–	–	–	–	1.17%	1.13%	1.19%	0.09%	0.27%
ζ_s	−0.53%	0.42%	0.78%	0.97%	1.17%	0.88%	0.92%	0.60%	0.44%
$\overline{\zeta}$	−0.53%	0.42%	0.78%	0.97%	1.17%	0.61%	0.76%	0.50%	0.22%

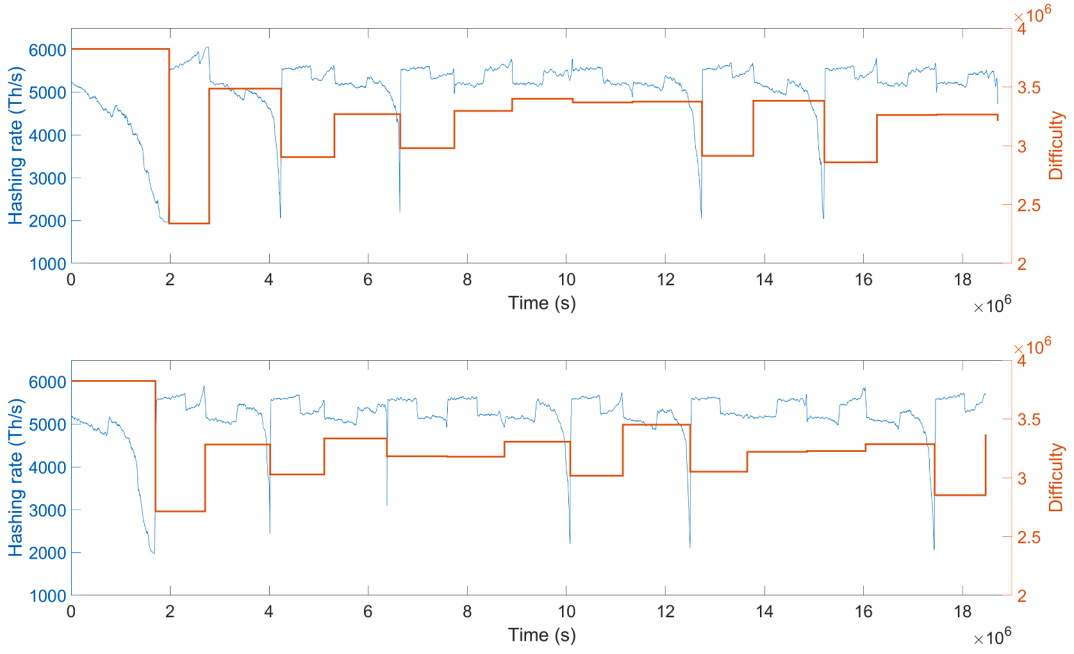

Fig. 10. The dynamics of overall hashing rate in different mining scenarios (100-point moving average, a single replication).

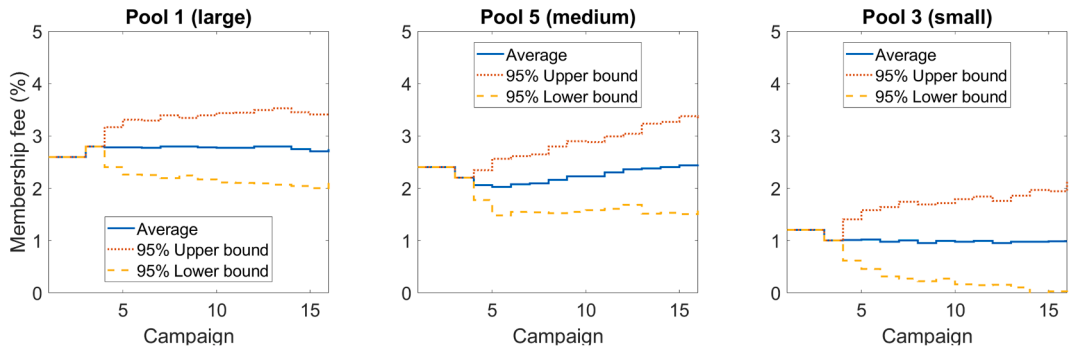

Fig. 11. The dynamics of membership fees without *UB* (average and 95% C.I. of 40 replications).

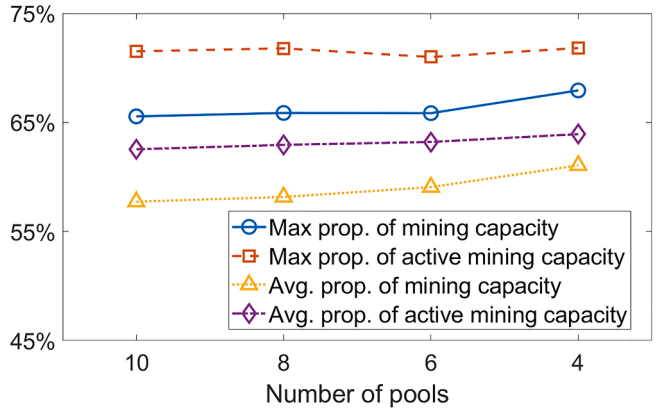

Fig. 12. The monopolist's proportions of (active) mining capacity with different pool numbers when removing *UB*.

The first two factors above are regarding mining pools. Recall Eq. (3), the dynamics of F_j could indicate the change of the corresponding pool's mining capacity. Nevertheless, F_j makes less impact on the overall mining power of the network than *UB*, because it fluctuates within the range of 0% $\sim$ 4% (see Fig. 5b), which is close to zero. In other words, the increasing membership with a fixed step size will not discourage miners from joining the pool effectively.

Suppose pool managers do not care about the credit collapse of the Bitcoin system and remove *UB*, pool 1 will quickly arise as a monopolist which occupies the majority (more than 50%) of the system's overall mining capacity. Afterward, its mining capacity becomes steady, so does its membership fee (Fig. 11). In our experiment, the monopolist possesses around 60% $\sim$ 70% of the total mining capacity. In other words, it attracts most, but not all, of the active miners. Because miners over-react to smaller pools and under-react to larger pools (the certainty effect of prospect theory).

In order to analyze the monopoly of the Bitcoin mining system, extra experiments are performed without the pool capacity's upper limitation. First, we find that the size of the monopolist's pool increases as the number of mining pools decreases (Fig. 12). Similarly, by reducing the proportion of passive miners, the monopolist will attract more active miners (Fig. 13). In conclusion, keeping the number of competitors as well as the ratio of passive miners could impose restrictions on monopoly, and the latter has more impact than the former. Additionally, we have an interesting observation: independent of the number of pools and the proportion of passive miners, the average (maximal) ratio of active mining capacity controlled by the monopolistic pool stays invariant, which is around 63% (72%).

4. Conclusion and future work

Summary. We develop a discrete-event Monte-Carlo simulation model to study the behavior of individual miners and mining pool managers, with the objective of testing different mining and pool managing policies. Compared with previous works, our model is more flexible and practical, because it involves realistic features including hashing rate, mining cost, monetary budget, Bitcoin market price, mining pool reward policies, and membership fees. Our simulation results may provide useful insights for individual miners and pool managers in the realistic mining system. First, for an individual miner participating in pool mining, the Bitcoin income stability of three popular remuneration schemes is as follows: PPS > PPLNS > PROP. Next, to approach the ideal block generation rate designed by Satoshi, the current adaptive difficulty recalculation algorithm is demonstrated to be efficient. Another interesting finding here is that medium pools may have a greater growth potential than small and large ones. Moreover, the stochasticity of the system will be higher if we involve more individual miners in our simulation. Furthermore, the relationship between the mining power allocation rule and the real-time difficulty level may help individual miners to make more profitable mining decisions by choosing the appropriate timing to mine. Based on this, we introduce an alternative mining policy and validate its effectiveness under certain conditions. Finally, we provide the factors influencing the Bitcoin mining

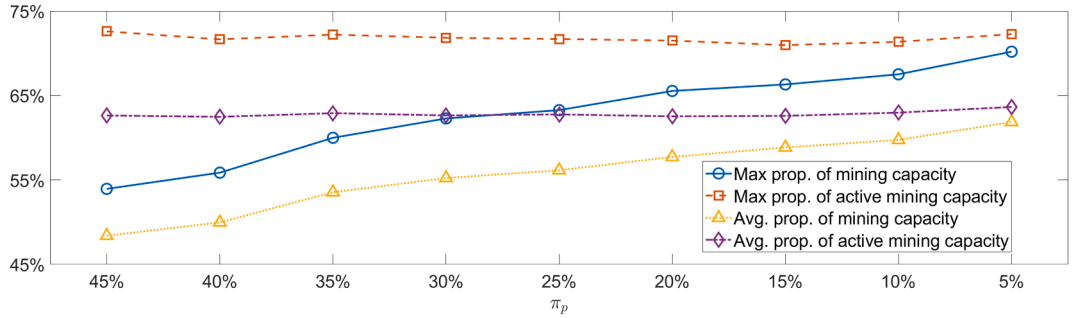

Fig. 13. The monopolist's proportions of (active) mining capacity with different π_p's when removing *UB*.

monopoly in the simulation model.

Future works. Since security issues, e.g., selfish-mining attack (Eyal and Sirer, 2014), block withholding attack (Wu et al., 2019), and fork after withholding attack (Kwon et al., 2017), are critical in the Bitcoin network, we next plan to extend our simulation model to include various attacks and coping strategies. Furthermore, as an increasing number of Bitcoin/blockchain research from the OR community (Pun et al., 2018; Roşu and Saleh, 2020; Gan et al., 2021), we also consider taking more advantage of OR methodologies in the future. Meanwhile, we may involve Black Swan events, such as the skyrocket or plummet of Bitcoin price and the outbreak of epidemic diseases.

CRediT authorship contribution statement

Kejun Li: Writing - original draft, Formal analysis, Data curation, Visualization. **Yunan Liu:** Writing - review & editing, Methodology, Supervision. **Hong Wan:** Writing - review & editing, Conceptualization, Supervision. **Yining Huang:** Resources.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The authors are grateful for Ling Zhang's suggestions in the early stage of the model building.

Appendix A. Additional numerical experiment results

The dynamics of shares and membership fees of all mining pools are illustrated in Fig. A.14. Moreover, Fig. A.15 is regarding the case of all pools with the same initial states; Fig. A.16 shows the results without the upper limitation of pool capacity *UB*.

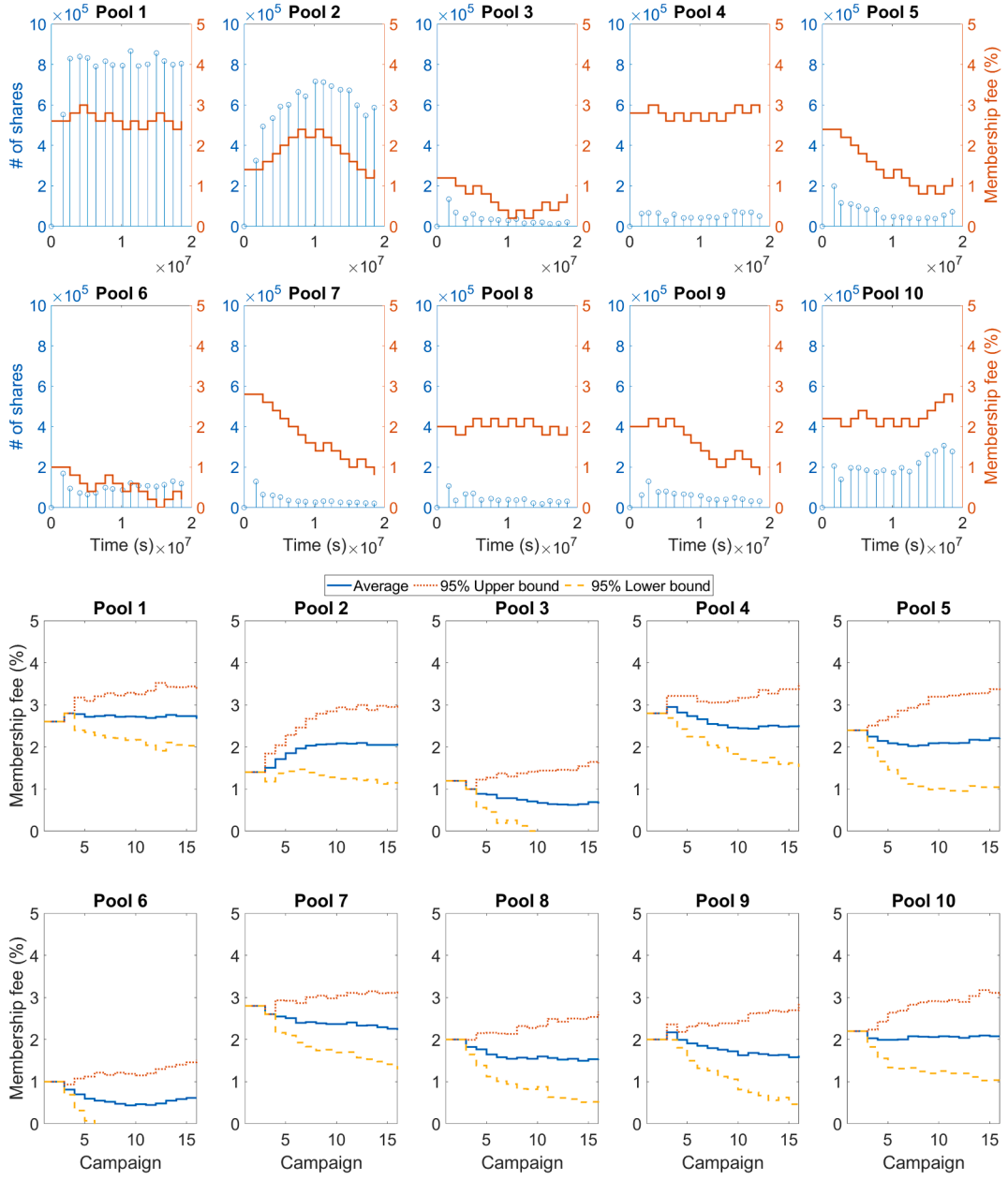

Fig. A.14. Simulation of 10 pools.

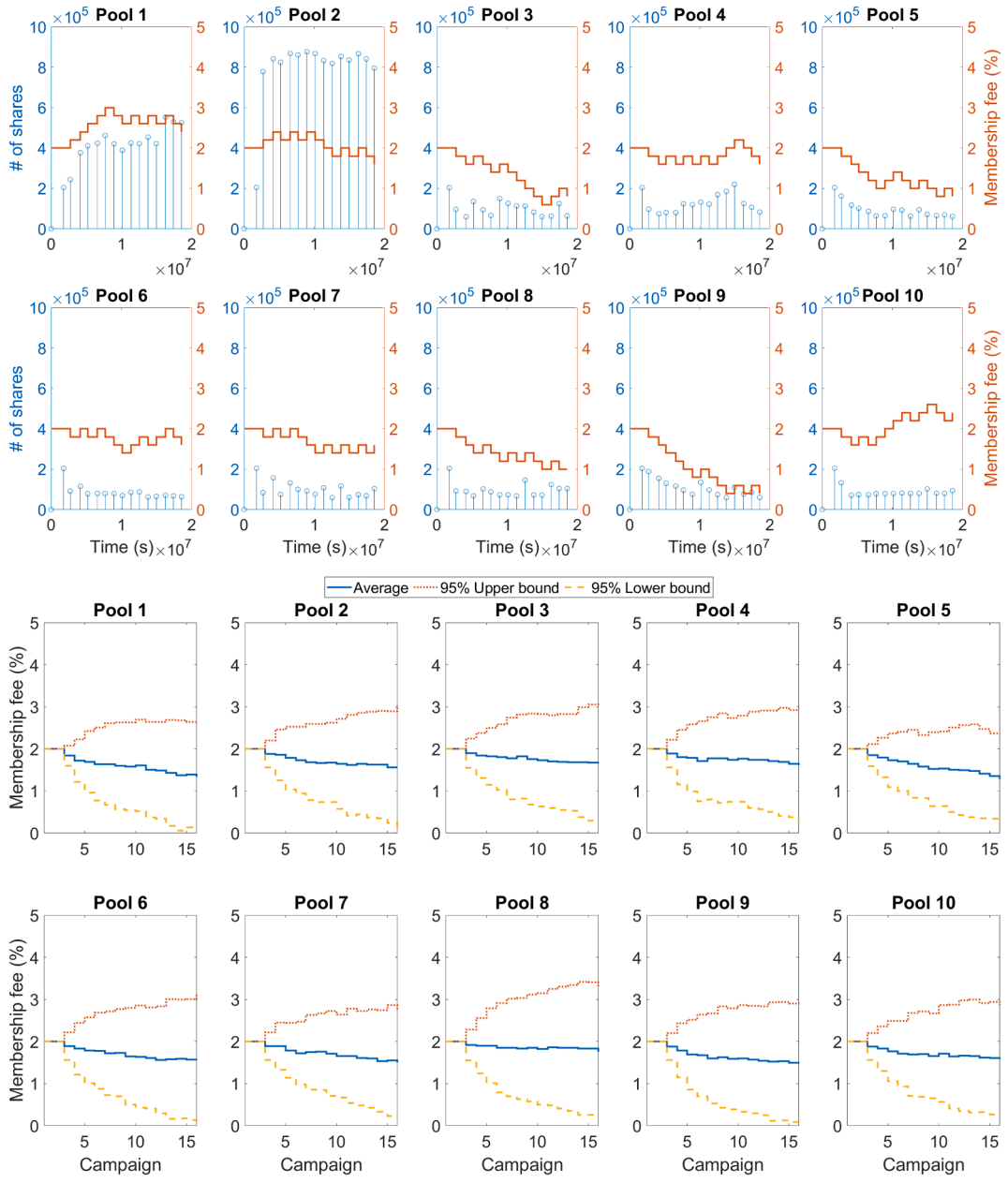

Fig. A.15. Simulation of 10 pools (identically initialized pools).

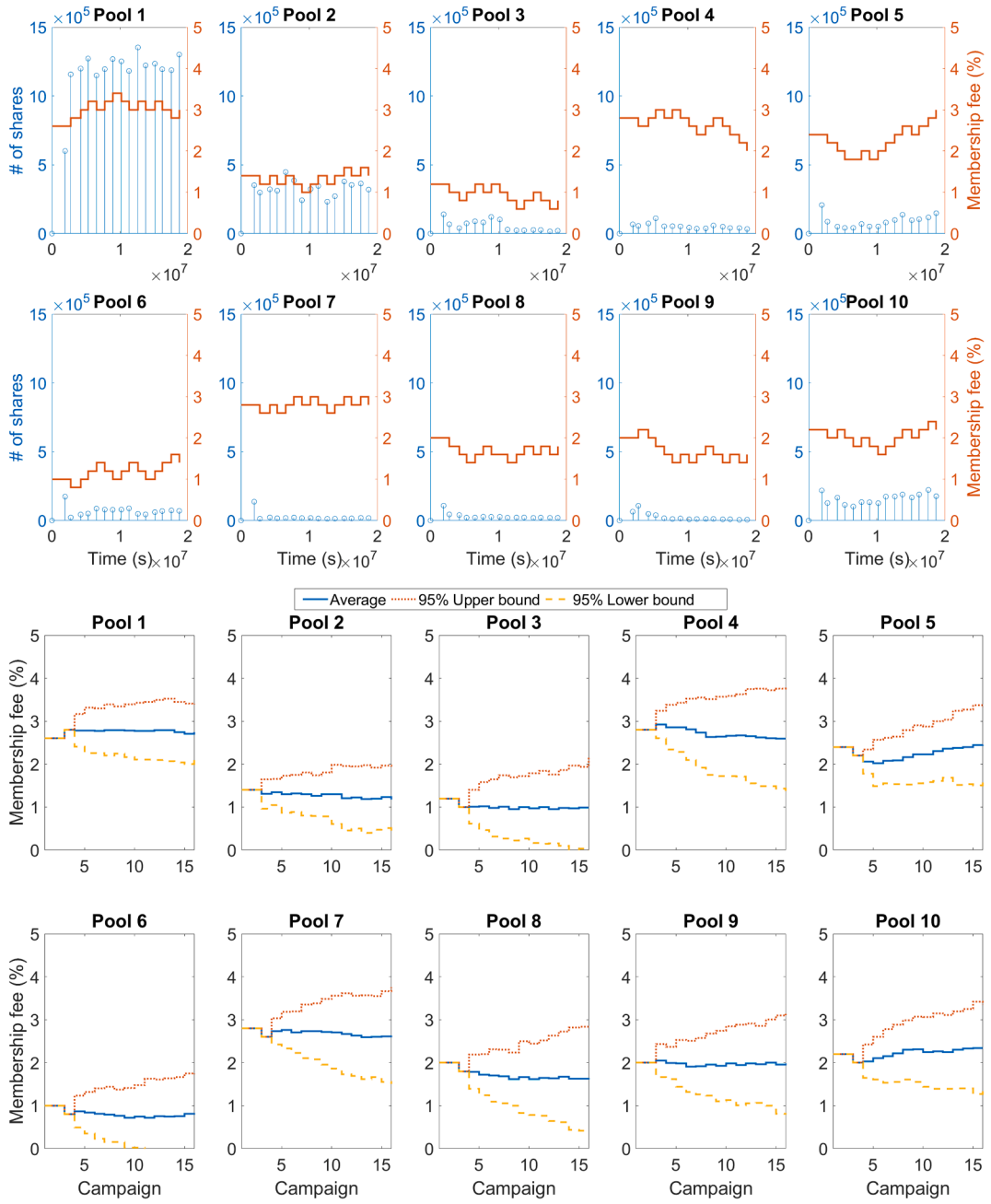

Fig. A.16. Simulation of 10 pools (without *UB*).

References

Abay, N.C., Akcora, C.G., Gel, Y.R., Islambekov, U.D., Kantarcioglu, M., Tian, Y., Thuraisingham, B., 2019. Chainnet: Learning on blockchain graphs with topological features. arXiv preprint arXiv:1908.06971.

Akcora, C.G., Dey, A.K., Gel, Y.R., Kantarcioglu, M., 2018. Forecasting bitcoin price with graph chainlets. Pacific-Asia conference on knowledge discovery and data mining, Springer 765–776.

Alharby, M., van Moorsel, A., 2019. Blocksim: A simulation framework for blockchain systems. ACM SIGMETRICS Performance Evaluation Review 46, 135–138.

Aoki, Y., Otsuki, K., Kaneko, T., Banno, R., Shudo, K., 2019. Simblock: A blockchain network simulator. In: IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). Institute of Electrical and Electronics Engineers Inc, Piscataway, New Jersey, pp. 325–329.

Atsalakis, G.S., Atsalaki, I.G., Pasiouras, F., Zopounidis, C., 2019. Bitcoin price forecasting with neuro-fuzzy techniques. Eur. J. Oper. Res. 276, 770–780.

Bitcoin Wiki contributors, 2020. Comparison of mining pools - Bitcoin Wiki. url:https://en.bitcoin.it/wiki/Comparison_of_mining_pools, accessed 30th May.

Blockchain.com contributors, 2020. BTC to USD: Bitcoin to US Dollar Market Price - Blockchain. URL: https://www.blockchain.com/charts/market-price, accessed 30th May.

Bottone, M., Raimondi, F., Primiero, G., 2018. Multi-agent based simulations of block-free distributed ledgers. In: 2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA). Institute of Electrical and Electronics Engineers Inc, Piscataway, New Jersey, pp. 585–590.

Bowden, R., Keeler, H.P., Krzesinski, A.E., Taylor, P.G., 2018. Block arrivals in the bitcoin blockchain. arXiv preprint arXiv:1801.07447.

Bracha, G., Toueg, S., 1985. Asynchronous consensus and broadcast protocols. J. ACM (JACM) 32, 824–840.

Brousmichc, K.L., Anoaica, A., Dib, O., Abdellatif, T., Deleuze, G., 2018. Blockchain energy market place evaluation: An agent-based approach. In: 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). Institute of Electrical and Electronics Engineers Inc, Piscataway, New Jersey, pp. 321–327.

Chang, S.E., Luo, H.L., Chen, Y., 2020. Blockchain-enabled trade finance innovation: A potential paradigm shift on using letter of credit. Sustainability 12, 188.

Chitra, T., Quaintance, M., Haber, S., Martino, W., 2019. Agent-based simulations of blockchain protocols illustrated via kadena's chainweb. In: 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW). Institute of Electrical and Electronics Engineers Inc, Piscataway, New Jersey, pp. 386–395.

Cocco, L., Marchesi, M., 2016. Modeling and simulation of the economics of mining in the bitcoin market. PloS one 11.

Cocco, L., Tonelli, R., Marchesi, M., 2019. An agent-based artificial market model for studying the bitcoin trading. IEEE Access 7, 42908–42920.
Coin Dance contributors, 2020. Coin Dance — Bitcoin Nodes Summary. url: https://coin.dance/nodes, accessed 30th May.
Cong, L.W., He, Z., Li, J., 2019. Decentralized mining in centralized pools. Rev. Financ. Stud.
Correia, M., Veronese, G.S., Neves, N.F., Verissimo, P., 2011. Byzantine consensus in asynchronous message-passing systems: a survey. IJCCBS 2, 141–161.
Cretarola, A., Figà-Talamanca, G., 2019. Detecting bubbles in bitcoin price dynamics via market exuberance. Ann. Oper. Res. 1–21.
Dey, A.K., Akcora, C.G., Gel, Y.R., Kantarcioglu, M., 2020. On the role of local blockchain network features in cryptocurrency price formation. Can. J. Stat. 48, 561–581.
Eyal, I., Sirer, E.G., 2014. Majority is not enough: Bitcoin mining is vulnerable. International conference on financial cryptography and data security, Springer 436–454.
Foytik, P., Shetty, S., Gochhayat, S.P., Herath, E., Tosh, D., Njilla, L., 2020. A blockchain simulator for evaluating consensus algorithms in diverse networking environments. In: Proceedings of the 2020 Spring Simulation Conference. Institute of Electrical and Electronics Engineers Inc, Piscataway, New Jersey, pp. 1–12.
Gan, J., Tsoukalas, G., Netessine, S., 2021. Initial coin offerings, speculation, and asset tokenization. Manage. Sci. 67, 914–931.
Garay, J., Kiayias, A., Leonardos, N., 2017. The bitcoin backbone protocol with chains of variable difficulty. Annual International Cryptology Conference, Springer 291–323.
Gervais, A., Karame, G.O., Wüst, K., Glykantzis, V., Ritzdorf, H., Capkun, S., 2016. On the security and performance of proof of work blockchains. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. Association for Computing Machinery, New York, NY, USA, pp. 3–16.
Göbel, J., Keeler, H.P., Krzesinski, A.E., Taylor, P.G., 2016. Bitcoin blockchain dynamics: The selfish-mine strategy in the presence of propagation delay. Performance Eval. 104, 23–41.
Griggs, K.N., Ossipova, O., Kohlios, C.P., Baccarini, A.N., Howson, E.A., Hayajneh, T., 2018. Healthcare blockchain system using smart contracts for secure automated remote patient monitoring. J. Med. Syst. 42, 1–7.
Huberman, G., Leshno, J.D., Moallemi, C., 2019. An economist's perspective on the bitcoin payment system. AEA Papers and Proceedings 93–96.
Kahneman, D., Tversky, A., 1979. Prospect theory: An analysis of decision under risk. Econometrica 47, 363–391.
Kaligotla, C., Macal, C.M., 2018. A generalized agent based framework for modeling a blockchain system, in: Rabe, M., Juan, A.A., Mustafee, N., A. Skoogh, S.J., Johansson, B. (Eds.), Proceedings of the 2018 Winter Simulation Conference, Institute of Electrical and Electronics Engineers, Inc., Piscataway, New Jersey. pp. 1001–1012.
Kawase, Y., Kasahara, S., 2017. Transaction-confirmation time for bitcoin: A queueing analytical approach to blockchain mechanism. International Conference on Queueing Theory and Network Applications, Springer 75–88.
Koutmos, D., 2019. Market risk and bitcoin returns. Ann. Oper. Res. 1–25.
Kraft, D., 2016. Difficulty control for blockchain-based consensus systems. Peer-to-Peer Networking Appl. 9, 397–413.
Kwon, Y., Kim, D., Son, Y., Vasserman, E., Kim, Y., 2017. Be selfish and avoid dilemmas: Fork after withholding (faw) attacks on bitcoin, in. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 195–209.
Lee, K., Ulkuatam, S., Beling, P., Scherer, W., 2018. Generating synthetic bitcoin transactions and predicting market price movement via inverse reinforcement learning and agent-based modeling. J. Artif. Soc. Soc. Simul. 21.
Lewenberg, Y., Bachrach, Y., Sompolinsky, Y., Zohar, A., Rosenschein, J.S., 2015. Bitcoin mining pools: A cooperative game theoretic analysis, in. In: Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, pp. 919–927.
Li, Z., Kang, J., Yu, R., Ye, D., Deng, Q., Zhang, Y., 2017. Consortium blockchain for secure energy trading in industrial internet of things. IEEE Trans. Ind. Inf. 14, 3690–3700.
Liao, K., Katz, J., 2017. Incentivizing blockchain forks via whale transactions. International Conference on Financial Cryptography and Data Security, Springer 264–279.
Liu, C.H., Lin, Q., Wen, S., 2018. Blockchain-enabled data collection and sharing for industrial iot with deep reinforcement learning. IEEE Trans. Industr. Inf. 15, 3516–3526.
Liu, J., Liu, Z., 2019. A survey on security verification of blockchain smart contracts. IEEE Access 7, 77894–77904.
Liu, Y., Fan, Z.P., Zhang, Y., 2014. Risk decision analysis in emergency response: A method based on cumulative prospect theory. Comput. Oper. Res. 42, 75–82.
Memon, R.A., Li, J.P., Ahmed, J., 2019. Simulation model for blockchain systems using queuing theory. Electronics 8, 234.
Meshkov, D., Chepurnoy, A., Jansen, M., 2017. Short paper: revisiting difficulty control for blockchain systems, in: Data Privacy Management, Cryptocurrencies and Blockchain Technology. Springer, pp. 429–436.
Miller, A., Jansen, R., 2015. Shadow-bitcoin: Scalable simulation via direct execution of multi-threaded applications, in: 8th Workshop on Cyber Security Experimentation and Test ({CSET} 15), USENIX Association, Washington, D.C.
Nakamoto, S., et al., 2008. Bitcoin: A peer-to-peer electronic cash system.
Narayanan, A., Bonneau, J., Felten, E., Miller, A., Goldfeder, S., 2016. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction. Princeton University Press, Princeton, New Jersey.
Nguyen, D.C., Pathirana, P.N., Ding, M., Seneviratne, A., 2020. Privacy-preserved task offloading in mobile blockchain with deep reinforcement learning. IEEE Trans. Netw. Serv. Manage. 17, 2536–2549.
Peters, G.W., Panayi, E., 2016. Understanding modern banking ledgers through blockchain technologies: Future of transaction processing and smart contracts on the internet of money. Banking beyond banks and money. Springer 239–278.
Popov, S., 2016. The tangle. url:http://www.descryptions.com/Iota.pdf, accessed 30th May.
Pun, H., Swaminathan, J.M., Hou, P., 2018. Blockchain adoption for combating deceptive counterfeits. Kenan Institute of Private Enterprise Research Paper.
Qin, R., Yuan, Y., Wang, F.Y., 2019. A novel hybrid share reporting strategy for blockchain miners in ppln pools. Decis. Support Syst. 118, 91–101.
Qin, R., Yuan, Y., Wang, S., Wang, F.Y., 2018. Economic issues in bitcoin mining and blockchain research. In: 2018 IEEE Intelligent Vehicles Symposium (IV), IEEE, pp. 268–273.
Rauchs, M., 2020. Cambridge Bitcoin Electricity Consumption Index (CBECI). url: https://www.cbeci.org/, accessed 30th May.
Rosa, E., D'Angelo, G., Ferretti, S., 2019. Agent-based simulation of blockchains. In: Tan, G. (Ed.), Methods and Applications for Modeling and Simulation of Complex Systems. Springer Singapore, Singapore, pp. 115–126.
Rosenfeld, M., 2011. Analysis of bitcoin pooled mining reward systems. arXiv preprint arXiv:1112.4980.
Roşu, I., Saleh, F., 2020. Evolution of shares in a proof-of-stake cryptocurrency. Manage. Sci.
Salimitari, M., Chatterjee, M., Yuksel, M., Pasiliao, E., 2017. Profit maximization for bitcoin pool mining: A prospect theoretic approach. In: 2017 IEEE 3rd International Conference on Collaboration and Internet Computing (CIC). Institute of Electrical and Electronics Engineers Inc, Piscataway, New Jersey, pp. 267–274.
Sharma, R., Kamble, S.S., Gunasekaran, A., Kumar, V., Kumar, A., 2020. A systematic literature review on machine learning applications for sustainable agriculture supply chain performance. Comput. Oper. Res. 119, 104926.
Tsabary, I., Eyal, I., 2018. The gap game. In: Proceedings of the 2018 ACM SIGSAC conference on Computer and Communications Security, pp. 713–728.
Wan, H., 2020. Blockchain beyond cryptocurrency: An overview. In: Frontiers of Engineering: Reports on Leading-Edge Engineering from the 2019 Symposium. National Academies Press.
Wang, B., Chen, S., Yao, L., Liu, B., Xu, X., Zhu, L., 2018. A simulation approach for studying behavior and quality of blockchain networks. In: International Conference on Blockchain. Springer International Publishing, Cham, pp. 18–31.
Wang, C., Chu, X., Yang, Q., 2019. Measurement and analysis of the bitcoin networks: A view from mining pools. arXiv preprint arXiv:1902.07549.
Wei, X., Li, A., Zhou, H., 2020. Impacts of consensus protocols and trade network topologies on blockchain system performance. J. Artif. Soc. Soc. Simul. 23, 1–2.
Wu, D., Liu, X.d., Yan, X.b., Peng, R., Li, G., 2019. Equilibrium analysis of bitcoin block withholding attack: A generalized model. Reliab. Eng. Syst. Saf. 185, 318–328.
Yeow, A., 2020. Global Bitcoin Nodes Distribution - Bitnodes. URL: https://bitnodes.io/, accessed 30th May.
Yin, H.S., Vatrapu, R., 2017. A first estimation of the proportion of cybercriminal entities in the bitcoin ecosystem using supervised machine learning. In: 2017 IEEE International Conference on Big Data (Big Data), IEEE, pp. 3690–3699.