Incentive Assignment in PoW and PoS Hybrid Blockchain in Pervasive Edge Environments

Yaodong Huang, Yiming Zeng, Fan Ye, Yuanyuan Yang
Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794, USA
{yaodong.huang, yiming.zeng, fan.ye, yuanyuan.yang}@stonybook.edu

***Abstract*—Edge computing is becoming pervasive in our daily lives with emerging smart devices and the development of communication technology. Resource-rich smart devices and high-density supportive networks make data transactions prevalent over edge environments. To ensure such transactions are unmodifiable and undeniable, blockchain technology is introduced into edge environments. In this paper, we propose a hybrid blockchain system in edge environments to enhance the security for transactions and determine the incentive for miners. We propose a Proof of Work (PoW) and Proof of Stake (PoS) hybrid consensus blockchain system utilizing the heterogeneity of devices to adapt to the characteristic of edge environments. We raise the incentive assignment problem that gives the corresponding PoW miner when a new block generates. We further formulate it into a two-stage Stackelberg game. We propose an algorithm and prove that it can obtain the global optimal results for the incentive that the miner will receive for a new block. Numerical simulation results show that our proposed algorithm can give reasonable incentive to miners under different system parameters in edge blockchain systems.**

***Keywords*—Pervasive edge computing, Hybrid blockchain, Proof of Work**

I. Introduction

The arriving 5G networks aim at providing low-latency, high-throughput and energy-saving computing to a massive number of devices. Thanks to the backbone technology, edge computing is becoming increasingly crucial to enhance the quality of service for thriving smart edge devices. Such devices like phones, IoT sensors or even vehicles offer an immense amount of data, which can be shared and transferred among different clients. With the abundance of devices and data, edge computing can process data locally without the involvement of cloud or other centralized services. New business models have emerged to provide paid information services to users for income. An example is “We media”, where data producers sell contents like video clips or texts to interested customers to make money.

Consider a situation where data producers have for-profit contents and some users want to access such contents and pay for them. The subscriptions allow paid users to access corresponding contents quickly and securely while denying unpaid users from obtaining them. Most current solutions require a trusted third-party to manage such contents and subscriptions. For instance, Gumroad [1] provides services for data producers to sell digital contents directly to consumers. Although considerable amounts of text, audio, and video contents are sold on these platforms, there are still adverse events [2], mostly related to security, trust, and privacy concerns. In peer edge environments, micro-access control and micro-payment transactions provide fast identity verification and data accessing without trusted third parties. For example, vehicles can sell pictures and road information directly to peers without using a cloud-based backend platform. With such micro-access control and micro-payment, peer devices can directly manage subscription payments and data delivery on edge, helping both producers and consumers in a distributed manner.

Recently, the blockchain technology becomes widely used in distributed systems. It makes its debut in cryptocurrencies like Bitcoin, where the transactions are sent between users in a peer-to-peer network [3]. The blockchain consists of multiple blocks. Each block serves as a ledger and stores information of the previous block to form a chain. The blockchain contains many security features in a distributed system. First, the complete history of blocks and transactions are stored throughout the network. It can provide quick restoration and verification as well as prevent “a central point of failure”. Second, the blockchain cannot be easily manipulated unless more than a quarter of the total computational power is controlled by malicious parties [4]. Changing a block is hard since each block has a hash that is designed to be hard to obtain and changing a block also affects a chain of blocks. The blockchain technology improves the efficiency, security, and privacy of transactions in a distributed manner without the help of centralized trusted third parties.

Despite the advantages of blockchain technology in such distributed systems, edge environments have limitations over resources, especially storage and energy. Maintaining the security of blocks in blockchain systems often requires a tremendous amount of energy and storage space. Such resource requirement is beyond the capabilities of most edge devices (e.g., phones, IoT sensors), and will make them less inclined to participate the blockchain. Meanwhile, the edge devices are also heterogeneous, some of the devices (e.g, edge servers, vehicles) will have a relatively larger amount of resources to conduct computing intensive work, which may help enhance the security of blockchain running over edges. How to combine heterogeneous devices to design an effective blockchain system that all devices can participate, assign fair incentive and improve security remains a challenging problem.

In this paper, we propose a consensus-hybrid Proof of Work (PoW) and Proof of Stake (PoS) blockchain coordinating between resource-limited and resourceful edge devices to improve security when running the private blockchain

978-1-7281-6887-6/20/$31.00 ©2020 IEEE

system over edge devices. We focus on the fair assignment of incentives to users using different forms of consensus for new blocks in the edge blockchain system. We propose the incentive assignment problem to determine how much incentive is given to PoW miners for mining a new block. A Stackelberg game is formulated to describe the incentive assignment problem and we propose an iterative algorithm to solve it. We also prove that the algorithm can converge and achieve global optimal results. Extensive simulations show our proposed algorithm can offer a reasonable number of coins as the incentive to PoW miners under different settings of hybrid blockchain system parameters.

We make the following contributions in this paper.

- We propose a PoS-PoW hybrid blockchain system in edge environments to enhance the security for the transactions conducted. The blockchain system considers the heterogeneity of edge devices to encourage the participation of both resource-limited and resource-rich devices. The ratio between PoS and PoW blocks can be adjusted to fit the capabilities of users in the network.
- We propose a novel incentive assignment mechanism to determine the incentive for a new block for miners in the edge blockchain system. We raise the problem to give corresponding PoW miners fair incentive and formulate it into a two-stage Stackelberg game. We propose an iterative algorithm to solve the problem and offer theoretical analysis to prove it can converge to the global optimal result.
- We implement the incentive assignment algorithm and conduct extensive numerical evaluations. The results show that our proposed mechanism can give a reasonable number of coins to the miner that mines the new block under different system parameters settings.

The rest of the paper is organized as follows. Section II discusses some related work on blockchain and edge computing. Section III discusses the system overview of the hybrid blockchain. Section IV presents the design of PoW and PoS blocks and mining processes. In Section V we formulate the PoW mining incentive assignment problem and offer the solution. We conduct numerical simulations in Section VI. Finally, we conclude the paper and discuss future work in Section VII.

II. Related Work

The blockchain technology is proposed in 2008 by Satoshi Nakamoto [3] and has been widely used in cryptocurrencies ever since. It consists of a series of blocks linked using cryptography. The blockchain can serve as a distributed ledger for storing data among devices [5] and can prevent unauthorized changes of its contents due to cryptography features. If a malicious user wants to tamper with a piece of data, it has to counterfeit a whole branch of a chain from the block that it intends to modify, which is nearly impossible unless it controls half of the computational power of the network [6]. Even just posing some threat to the system needs more than a quarter of total computational power [7]. These features can make the blockchain system a safe ledger perfectly for cryptocurrencies, e.g., Bitcoin [3], Litecoin [8], and Ethereum [9].

On the contrary of cloud computing which moves the computing to the centralized cloud, edge computing moves the computing work to distributed nodes on the edge of the network. The computing mostly or entirely happens on nodes near to or inside the edge devices [10]. With the increasing powerful edge smart devices and fast-growing networking technology like 5G and Wi-Fi 6, data sharing among edge devices and clouds creates many novel applications [11]–[13]. Recently, blockchain technology has been introduced for the data transaction secureness for edge environments. Many edge applications such as IoT [14], [15], vehicle network [16], and network function virtualization [17] have applied blockchain to enhance security, privacy, scalability, and robustness. Although blockchain can bring such advantages, limitations of edge environments make it impractical to directly deploy blockchain. Huang et al. [18] discuss the storage allocation and Proof of Stake implementation on peer edge environments with limited resources. Wu et al. [19] discuss the task offloading of mining on mobile edge networks.

The traditional concept of PoW mining is for miners competing with each other solving a mathematics puzzle. Whoever solves the puzzle first will be granted the privilege to write the next block. This concept is called Proof of Work (PoW) [3], [20]. Another emerging concept is called Proof of Stake (PoS) [21], [22], which in contrast, achieves the consensus from the publicly owned data of the users such as wealth or age. Since PoS does not rely on exhaustively solving mathematical puzzles, it saves a lot of energy for mining a new block. Recently, the hybrid blockchain protocol has drawn much attention as it may get advantages from both PoW and PoS. Liu et al. [23] propose a fork-free PoW protocol and combine with PoS to make a hybrid blockchain, Santos et al. [24] discuss measurement of the complexity of different consensus protocols including PoW-PoS hybrid protocol.

To tackle the complicated collaboration in edge networks, the game theory is a promising technique that has been widely adopted in various networks. In [25], the authors consider a D2D communication framework in which the operator of the base station offers incentives to owners of devices to motivate content communication. In [26], a wireless sensor network consisting of many private sensor networks is considered.

III. System Overview

In this section, we introduce our hybrid blockchain model. We first introduce some background information, and later our proposed system and incentive assignment designs.

A. Background

1) Proof of Work: It is a well-known consensus mechanism and presented in Bitcoin [3], which grants the privilege to users who solve a computationally intensive math problem. The user needs to hash certain information and a random number so the hash value meets some preset patterns. The user is called miner and the process is called mining. For instance, Bitcoin miners need to hash the timestamp, hash value of Merkle (a type of tree for transaction data) root, the previous block hash value, current target (indicate difficulty) and a nonce to get a hash value. The hash value must be smaller than a given number (target). The miner can change the hash value by picking a different nonce. The process that finding the nonce thus certain hash value is called mining. The smaller the target is, the harder the mining process will be. Currently, in February 2020, the target is 19 consecutive 0's in the front of the hash value (in hexadecimal form)[1].

The security of the PoW mechanism is that, when changing any part of the value, the hash value of the block will change. When a hash of a block changes, the hash value of all subsequent blocks will also change. Thus, if a user wants to change certain information in a block, to make sure each block is legitimate, it has to create a whole new branch calculating all subsequent blocks. Since finding the nonce is a time and energy consuming process, unless it controls more than a quarter to half of the total computational power in the network [6], [7], it cannot get other users to accept this branch.

2) Proof of Stake: This consensus aims at reducing the amount of power consumption of PoW. The total amount of energy consumed per year for Bitcoin mining is 73 TWh (7.3×10^{10}kWh)[2]. PoS, on the other hand, is an energy-saving method to reach consensus to generate new blocks. Users who create PoS blocks are called minters, and the process is called minting. Unlike PoW mining where miners have to solve a math problem, the specific minter of the next block is randomly chosen based on the history related factors (e.g, wealth, age, storage). These factors are often assigned as tokens (e.g., coins times ages). Recently, PoS gains much attention as a low energy cost alternative over block consensus. Many cryptocurrencies appear based on this concept, e.g., Nxt [21] and Peercoin [22], and Ethereum has such plans to move to PoS in the future [27].

Although PoS has advantages over PoW on energy consumption, it has certain drawbacks that prevent it from being widely used. First, due to the low complexity of computation work, working on different chains has a less computational burden. This may create more branches and some users can work on multiple branches to make more profit. This is also vulnerable to data corruption since forging a long chain with faulty information is not a time and energy consuming since information inside each block is not as protected as PoW (no pattern required over hash value). Second, it also vulnerable to the 51% attack with an entity obtains 51% of the tokens. This can also cause a large wealth gap when the entity obtains a much larger amount of cryptocurrency, meaning richer minters will become richer and poorer will stay poorer.

B. Hybrid Blockchain Design

The hybrid blockchain chain is made up of two different kinds of blocks, PoS and PoW blocks. The blocks are created by users using different consensus. The blockchain is briefly described in Fig. 1.

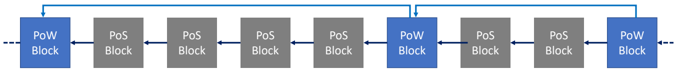


Fig. 1. The brief description of hybrdi blockchain. PoS blocks are the majority in the blockchain. PoW blocks are inserted in between PoS blocks.

As we can see from Fig. 1, there are more PoS blocks than PoW blocks. PoS blocks are generated (minted) by the users (minters) with limited resources, which are the majority of users in edge environments. The PoS blocks process and store most of the transactions. These transactions are quickly processed, and a part of the transaction fees are given to the block minters.

Purely PoS based blockchain has certain limitations over security as we mentioned above. Hybrid blockchain, by inserting a certain amount of PoW blocks into the blockchain, can improve the security of the blockchain system due to the designed computational hardness in PoW. PoW blocks are generated (mined) by users (miners) with more resources. The mining process is basically the same as the traditional PoW. The difference in our scenarios is that, instead of the longest chain, they choose the chain contains all legal transactions and has the most PoW blocks to append the new block. Since the PoS chain may create many branches, PoW blocks can freeze a certain branch (by appending the new block on a certain branch). Other users will continue to work on the frozen branch. Freezing a certain branch will reduce the number and length of branches. Meanwhile, transactions need to be validated by PoW miners. A part of the transaction fees is also given to miners. We discuss the detail mining process in Section IV-D.

PoW blocks are inserted into the chain frequently. Low fraction of PoW blocks will increase the processing time for transactions. High fraction of PoW blocks will take too much computational power that exceeds the capacities of users. Thus, a balanced frequency of PoW blocks is needed, and an expected ratio between PoS and PoW block, denoted as r, is crucial for the stableness of the blockchain.

C. Incentive Assignment

Incentive, such as coins, is given to minters and miners to incentivize the participation. Compared to the minting process, the mining process costs much more resource and increases the security. To get compensation for their work and encourage more users to participate in mining PoW blocks, fair incentive must be given to the miners of PoW blocks. Thus, we design a PoW mining incentive assignment mechanism for PoW-PoS hybrid blockchain. The incentive assignment mechanism determines how many coins a miner gets when it mines a new PoW block. The mechanism considers the participation for the

[1] Data obtained from recent Bitcoin blocks on https://blockchain.info in February 2020.

[2] Data obtained from https://digiconomist.net/bitcoin-energy-consumption in February 2020.

PoW miners as well as the ratio between PoW and PoS blocks. We formulate the problem as a two-stage Stackelberg Game and propose an iterative algorithm to solve it. The detailed formulation and solutions are presented in Section V.

IV. Blockchain Design

In the section, we discuss the block structure of PoS and PoW blocks, and introduce the mining and minting processes.

A. *Proof of Stake Blocks*

The structure of PoS blocks is illustrated in Fig. 2.

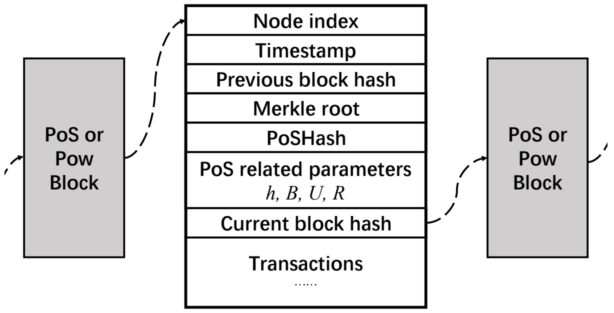


Fig. 2. The structure of a PoS block

The purpose of PoS in the blockchain is to reduce power consumption and encourage users (as minters) with limited resources to participate and process all transactions in the edge environments. Minters do not need compete each other by solving a computational complex problem. The PoS blocks record the information for the minting process that other users can validate. A block consists of a header and contents. The header records basic information of the block. Contents record all transactions happen between after the previous block generates. In the header, first, the timestamp, index and previous block hash are similar as those in normal blockchain systems. Second, transactions are encoded in a binary tree of hash values called Merkle trees. The root of the tree, called Merkle root is recorded in the header of the PoS block. Third, PoSHash and related parameters are used to validate whether the block is legal. PoSHash is used to generate and validate corresponding parameters. Expected time amendment B can adjust the expected time between two consecutive PoS blocks. Target value R and hit value h are for other users to verify that the block comes from the minter that has the privilege. Settings of these PoS related parameters are introduced in Section IV-D. Finally, the minter will hash such information in the header and store the hash value into the current block hash entry. The hash value does not need to satisfy certain patterns.

B. *Proof of Work Blocks*

The structure of PoW blocks is illustrated in Fig. 3.

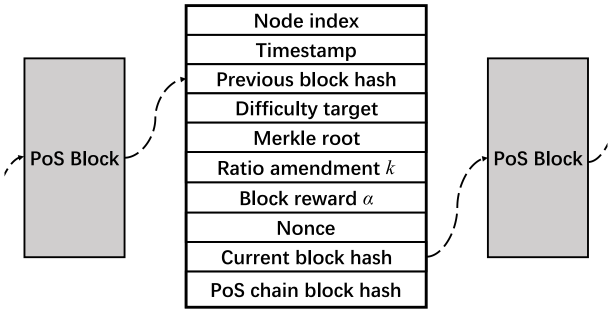


Fig. 3. The structure of a PoW block

PoW block is introduced to enhance the security of the blockchain. Since the hash value must satisfy certain patterns, miners must enumerate a nonce number to make the hash value of the header satisfy such pattern. Thus, counterfeiting a block costs large time and power consumptions. To record such information, the design of the PoW blocks has some major differences from the PoS block. First, it has the difficulty target part recoding the pattern of the hash value, which indicates how hard for mining the block will be. Second, nonce n must be recorded for other users to validate whether the hash value is correct. Third, ratio amendment k is presented which adjusts the number of PoS block between two PoW blocks. It is used to adjust the ratio between PoS and PoW blocks and make sure the PoW block generation time is proper. Fourth, the Merkle root design is different from that of PoS blocks, which is not for the transaction in the block but rather the Merkle root in PoS blocks after the previous PoW block. The PoW block, in our design, does not contain any transaction. All transactions are processed in PoS blocks.

The purpose of Merkle root is to validate whether the transaction in a block is changed. A slight change in a transaction will make Merkle root very different, thus will also affect the hash value of a block. In our design, transactions are processed in PoS blocks, and each block has a Merkle root entry. Miners later will mine PoW blocks and want to get part of the transaction fees from these transactions. It will validate Merkle roots in these PoS blocks. The miner will build a new binary from PoS Merkle root and the new tree root will stored in the PoW block Merkle tree entry. It can ensure that the transactions are hard to counterfeit, and for the benefit of PoW miners, it will choose honest transactions since it can get the most transaction fees.

Note that no transactions in the PoW block can increase transaction safety. Since PoW blocks do not contain transactions, a counterfeit transaction must be encoded in the PoS block of a malicious minter first. It then needs to be accepted by other PoS minters and later accepted by a miner to be encoded in the Merkle tree. It is too hard to coordinate all parties unless it has both more than half of the wealth (tokens) and more than half of the computational power.

C. *Minting Process*

PoS blocks are generated by the minting process. Minters compare the contribution to the system (i.e., mint more blocks) and randomly select one minter to be granted the privilege to mint the next block. In edge environments, the contribution of users is crucial for blocks generation and transaction processing. Our design goal is to make sure that a minter contributed more will have more advantage to mint the next block while still preserve the probability for new minters to participate.

Inspired by [18], [21], we describe our minting process here. Each node i will have a hit value h_i, which is directly calculated from PoSHash and its account number. The target value varies from minter to minter since each minter will have different account numbers. Basically,

$$POSHash(d+1,i) = \text{Hash}[POSHash(d) + Account_i],$$

$$h_i = POSHash(d+1,i) \mod M,$$

where M is the largest possible hit value. $POSHash(d)$ is the POSHash in the previous block, and $POSHash(d+1,i)$ will be the next PoSHash. Each minter will also have a target value R_i based on the stake of the minter. R_i is defined as

$$R_i = U_i t B,$$

where U_i is the token of minter i, t is the time passes from the previous block, and B is the value to control the PoS block generation rate. Minter i will be granted the privilege to mint a new block when it is the first minter that satisfies

$$h_i \leqslant R_i.$$

Since R_i will grow as time passes, eventually R_i will be larger than h_i. Minter with larger U_i will have higher possibility to be granted privilege, since R_i will grow faster. According to [18], when the expected time for generating a new PoS block is t_0, expectation time amendment B is equals $\frac{M}{(N+1)t_0 U}$, where U is the average token of all minters.

D. Mining Process

PoW blocks are generated by the mining process. Miners compete with each other to first obtain the correct nonce to make the hash value satisfy certain patterns. The nonce finding process is similar to the traditional blockchain where miners enumerate different numbers of the nonce to get a hash value to satisfy a certain pattern. Despite the nonce finding process, our proposed mining process focuses more on the hybrid blockchain system, which needs to consider the PoS blocks and transactions. Since miners consume more computational power and help validate transactions, they are expected to get more incentive. Meanwhile, a certain number of PoS blocks between PoW blocks is crucial for a stable hybrid blockchain. Thus, the system needs to adjust the hardness and the ratio amendment k.

A miner will get an incentive as α coins when it mines a new PoW block. The determination of expected coin number α is a game between all miners. Ratio amendment k is a crucial factor in the game and the coordination with the PoS process. Ratio amendment k is the ratio between the PoW block generation period T and expected incentive α for PoW blocks, i.e., $k = \frac{T}{\alpha}$. A longer period between two PoW blocks indicates more PoS blocks in between, i.e., more blocks and transactions need to be validated, and increases computational power cost for miners. If the cost for validation exceeds the revenue a miner can get, no user will participate in mining PoW blocks. Miners will require more coins to compensate if the PoW block generation period is longer. Ratio amendment k makes sure block generation period and expected incentive are reasonable. We discuss the game and ratio amendment in Section V.

Note that changing the hardness target can also change the generation period T, i.e., increasing hardness to make miners use a longer time to get the right nonce. However, increasing hardness will increase power consumption, which is not abundant in edge environments. Adjusting hardness is reasonable when the total computational power in the system changes, which will affect the efficiency of PoW. We will discuss the hardness target changing in our future work.

V. Incentive Assignment Fomulation and Solution

In this section, we propose a two-stage dynamic Stackelberg game to model the incentive assignment among PoW block miners. We then discuss the equilibrium of the model and show the algorithm to the solutions. The notations used in this section are shown in Table I.

TABLE I
NOTATIONS USED IN THE GAME FORMULATION

$i \in \mathcal{I}$	Miner ID and the miner set
x_i	The amount of computational power miner i devoted
s_i	The total amount of computational power of miner i
j_i	The total amount of computational power devoted other than miner i
α	The number of coins as incentive
β_i	The unit revenue than miner i can get for purpose other than mining
k	The ratio amendment of PoW process
Z	The hardness factor of PoW process
T	The expected time between two PoW blocks
r	The expected ratio between number of PoS and PoW blocks
L	The largest possible number for h_i
h_i	A hit of node i, $h_i \sim \mathcal{U}(0, M)$
N	The number of PoS minters in the network
B	The expectation time amendment, the value to adjust the time between two PoS blocks
U	The average token number of all minters

A. PoW Miners Incentive Assignment Formulation

1) Miners: As we mentioned before, PoW mining needs a lot of computational power. A simple computational power measurement is how many times of hashing a machine can do in a period. For PoW block miners, they devote a part of their computational power to the mining process. The remaining power can be used for other purposes. Each miner wants to gain as much profit as possible. For a miner i, denote the revenue (coin) of mining a PoW block as α, the profit is defined as follows.

$$M_i(\alpha, x_i) = \alpha \mathbf{P}(x_i) + \beta_i(s_i - x_i) - c \tag{1}$$

In (1), x_i is the amount of computational power that miner i is willing to devote for the Proof of Work mining process. s_i is the total computational power for miner i. β_i is the non-mining revenue factor. $\beta_i(s_i - x_i)$ means the revenue that it can gain using the reminder of the computational power. c represents a constant cost in the mining process. $\mathbf{P}(x_i)$ is the probability for miner i to mine the block, In the mining process,

$$\mathbf{P}(x_i) = \frac{x_i}{x_i + j_i}.$$

Here, $j_i = \sum_i \mathbf{x_{-i}}$, $\mathbf{x_{-i}} = \{x_1, x_2, ..., x_{i-1}, x_{i+1}, ..., x_n\}$, i.e., $\mathbf{x_{-i}} = \mathbf{x} \backslash x_i$.

Note that for the fairness over mining, α is the same for all miners, meaning that no matter which miner mines the block, it will get α coins; β_i can be different for each miner i, since each miner may use the remaining computational power for different purposes.

The problem formulation for miner i to get the most profit is as follows.

$$\max \quad M_i(\alpha, x_i) \tag{2}$$

$$\text{s.t.} \quad 0 \leqslant x_i \leqslant s_i. \tag{3}$$

The objective function (2) is the profit for miner i as we address above. Constraint (3) makes sure miner i cannot use more computational power than its capacity.

2) Virtual System: Since the blockchain system is distributed, there is no central authority that controls the system settings. Every miner will compete over others to get the incentive. It is actually a game among miners. We define a virtual system that serves as a leader in the Stackelberg game. It should be regarded as a protocol that every miner agrees with. The equilibrium obtained from the game is a consensus through different miners. If a miner gives unreasonable large number of coins to itself, it will affect the income of others, and the block will be rejected.

The goal of the virtual system is to minimize the coins for each block, i.e., everyone prevents other miners to get too many coins for a block. The problem formulation is as follows.

$$\min \quad \alpha \tag{4}$$

$$\text{s.t.} \quad \sum_i x_i \geqslant \frac{Z}{T}, \tag{5}$$

$$T \geqslant r\frac{M}{(N+1)BU}. \tag{6}$$

Objective function (4) is the number of coins for a block α. Constraint (5) makes sure that the total computational power of all miners contributed can satisfy the mining settings of specific hardness and expected period. Z stands for hardness. $T = \alpha k$ indicates the block generation period as we discuss in Section IV-D. Constraint (6) makes sure that the expected PoW period is larger than r times of the expected PoS generation blocks, which we discuss in Section III-B.

B. Game Model and Equilibrium Analysis

We now discuss the two-stage Stackelberg game. The game is defined as follows.

- *Followers*: PoW block miners.
- *Leader*: the virtual system.
- *Strategies*: the virtual system determines the number of coins α and miners determine the amount of computational power x_i to devote.
- *Payoff*: minimize the coin α and maximize the total profit for miners $M_i(x_i)$.

We now analyze the equilibrium derivation of the problem.

Definition 1. Stackelberg Equilibrium*: The outcome* $\{x^*, \alpha^*\}$ *is the Nash equilibrium of the game* G_B*, if the following conditions are concurrently satisfied for every miner* $i \in \mathcal{I}$ *and the virtual system:*

$$M(\alpha, x_i^*) \geqslant M(\alpha, x_i), \forall i \in \mathcal{I},$$
$$\alpha^* \leqslant \alpha.$$

This problem is challenging because the number of coins and the amount of computational power each miner devotes are coupled together. The processes of the interaction between the virtual system and miners are dynamic. To analyze the problem, we separate the process of the game into two different stages. In Stage I, the virtual system presents the expected coin α to miners. In Stage II, miners receive the expected coin α and adjust the computational power to participate. The results are returned to the virtual system. The virtual system and miners adjust the value until an equilibrium is reached. This game jointly solves the two problems.

1) Stage II: Individual miner problem (IMP)

We first address the case in Stage II. The objective for resellers is to maximize their total profit. After getting the expected number of coins α of the leader, the determination of miners is decided as the response for participation. We analyze the existence and uniqueness of the Nash equilibrium in the IMP.

Definition 2. *A computation resource assignment vector* $\mathbf{x}^* = (x_1^*, \cdots, x_n^*)$ *is the Nash equilibrium of the IMP, if, for each miner* $i \in \mathcal{I}$, $M_i(\alpha, x_i, j_i^*) \leqslant M_i(\alpha, x_i^*, j_i^*)$, *where* $j_i^* = \sum_{l \neq i | l \in \mathcal{I}} x_l^*$.

We prove the existence of the equilibrium of IMP in Theorem 1.

Theorem 1. *A Nash equilibrium exists and is unique in the game IMP.*

Proof. We investigate followers (miners) first. The strategy space of the miner i is defined as $[0, s_i]$ from the constraint (3), which is non-empty, convex and compact. The utility function (2) of miner i is continuous in $[0, s_i]$.

To prove the concavity of the utility function, we calculate the first and second-order derivation of (2) which are written as follows,

$$\frac{\partial M_i(x_i)}{\partial x_i} = \frac{\alpha j_i}{(x_i + j_i)^2} - \beta_i,$$

and

$$\frac{\partial^2 M_i(x_i)}{\partial^2 x_i} = -\frac{2\alpha j_i}{(x_i + j_i)^3} < 0.$$

The second order partial derivation is less than 0 which indicates that the utility function $M_i(x_i)$ is strictly concave about x_i. Accordingly, the Nash equilibrium exists in this non-cooperative IMP [28].

$M_i(x_i)$ is continuous and concave about x_i in the space $[0, s_i]$. It has only one optimal solution for this subproblem. Thus, the equilibrium is unique. □

The optimal strategy of the miner is decided by solving the optimization problem (2)-(3) for x_i, given that α is obtained from the virtual system and using it as the input. To get the optimal solution, for each node i, let

$$\frac{\partial M_i}{\partial x_i} = 0.$$

Since $x_i \geqslant 0$, we cat get

$$x_i^* = \sqrt{\frac{\alpha j_i}{\beta_i}} - j_i. \tag{7}$$

After obtaining the devoted computational power of miner i, it is regarded as the input to the virtual system problem (4)-(6) to minimize the number of coin α in Stage I.

2) Stage I: Virtual system problem

Now we discuss Stage I. In this stage, the virtual system minimizes the incentive coin number α. This is regarded as miners try to limit the incentive with others while satisfying the computational hardness target. The virtual system considers the anticipated strategy from each miner and later determines the incentive coin number. Thus, for each miner i, we introduce the optimal storage strategy (7), and constraint (5) can be written as

$$\alpha^{\frac{3}{2}} \sqrt{\frac{j_i}{\beta_i}} \geqslant \frac{Z}{k}. \tag{8}$$

To solve the problem, each miner will calculate an α based on its own. We denote the alpha from each miner i as α_i. The goal is to minimize each α_i and make sure all α_i value is close to each other. Thus, for a specific miner i, the problem (4) to (6) is formulated as

$$\begin{aligned} \min \quad & U(\alpha_i) = \sum_{i \in \mathcal{I}} \alpha_i, \\ \text{s.t.} \quad & (6), \\ & \alpha_i^{\frac{3}{2}} \sqrt{\frac{j_i}{\beta_i}} \geqslant \frac{Z}{k}, \forall i \in \mathcal{I}. \end{aligned} \tag{9}$$

The problem formulation above is a quasiconvex optimization [29] of the number of coins α for a PoW block. Equations (6) and (8) have square-root, which makes the problem quasiconvex. It is challenging to update the individual x_i from all the miners to the virtual system synchronously to minimize 5. Note that in Stage II, miner i needs to acquire the amount of aggregated computing resources to derive the optimal solution.

C. Iterative Algorithm

Motivated by this observation, we propose a sequentially updating algorithm based on the Gauss-Seidel iteration shown in Algorithm 1. The iteration rounds of Algorithm 1 are indexed by τ. Each iteration is divided into $|\mathcal{I}| + 1$ phases. In phase 0 of iteration 1, all miners concurrently calculate $x_i(1)$ based on their own resources condition. In phase n of iteration τ, miner n determines its computing resource from (7), and $j_n(\tau)$ is updated as follows,

$$j_n(\tau) = \sum_{l<n} x_l(\tau + 1) + \sum_{l>n} x_l(\tau).$$

Then miner i calculates the $x_i(\tau)$ from (7) and transmits it to the virtual system. In phase $n + 1$ of iteration τ, the virtual system transmits the update j_{n+1} to the miner $n + 1$.

Algorithm 1 Incentive Allocation Iterative Algorithm

```
Input: A feasible solution set $\mathbf{x}$ of (3)
Output: $\mathbf{x}$
 1: while $DIF > Thres$ or $round < 100$ do
 2:     for all $i \in \mathcal{I}$ do
 3:         Get $j_i = \sum \mathbf{x_{-i}}$
 4:         Solve (9) to obtain $\alpha_i$
 5:         Get $x_i$ from (7)
 6:         Update $x_i$ in $\mathbf{x}$
 7:     end for
 8:     Calculate $DIF$
 9:     $round+ = 1$
10: end while
```

The basic idea is to solve the problem (9) to get an incentive miner i proposes. Then, obtain x_i from (7) and using this x_i as a known quantity to solve the problem of other miners. In Algorithm 1, we first need a feasible solution of (3). Then, for each miner i, we get j_i as the summation of computational power devoted from all other miners (lines 2-3). Next, we solve the quasiconvex optimization problem to get α for this specific miner (line 4). We then obtain x_i from (7) and update $\mathbf{x}$ (line 5). After solving x_i, the next miner will use this x_i in its j_i and solve the problem (line 6). After getting all x_i, the algorithm will iterate until certain criteria are reached (line 8 and line 1). Since the problem is to get an α that every miner agrees, the algorithm will terminate when the variation of α is smaller than a certain threshold (DIF).

Theorem 2. *Algorithm 1 can converge to the global optimum for the edge block system.*

Proof. First, we prove the convergence and global optimality of the Algorithm 1. It is a distributed modified Gauss-Seidel Algorithm. According to Proposition 2.1 of CH.3 in [30] which gives if

- $U_{(}\alpha_i)$ is continuously differentiable and the constraints in convex and compact.
- Given a fixed α_i, miner i determines a unique and optimal response x_i to the virtual system.
- $U(\alpha_i)$ is a unique minimizer when given the feasible x_i.

It can be obviously checked that $U_{(}\alpha_i)$ can satisfy the first requirement because $U_{(}\alpha_i)$ is a linear combination of α_i.

From Theorem 1, $M_i(x_i)$ is concave about x_i and has a unique solution in the give compact space. The second requirement is satisfied.

$U(\alpha_i)$ is a linear function, it has a unique solution given the fixed space which satisfies the third requirement. □

Next, we prove the equilibrium of game G_B.

Theorem 3. *A Nash equilibrium exists and is unique in the game* G_B.

Proof. When given the j_i, the miner i has one unique optimal solution. When x_i is returned to the virtual system as the response of incentives α_i, the unique optimal incentives are determined by minimizing $U(\alpha)$ which has been mentioned above. This is a one to one correspondence between x_i and α_i, the computing resource x_i can be written as a function of α_i. For each $i \in \mathcal{I}$, the following Nash equilibrium exists,

$$U(\alpha_i, \alpha_{-i}*) > U(\alpha_i^*, \alpha_{-i}^*),$$

where $\alpha_{-i} = \sum_{l \neq i | l \in \mathcal{I}} \alpha_l$. This equilibrium can be conducted sequentially for each $i \in \mathcal{I}$. For the linear property of $U(\alpha_i)$, the equilibrium is unique. □

VI. Numerical Simulation Results

In this section, we conduct several simulations to evaluate our proposed hybrid blockchain incentive assignment mechanism. We focus on evaluating the incentive coin for PoS miners and computational power a miner devoted to PoW process under different settings of PoW mining incentives in the hybrid blockchain, and different capabilities of miners.

A. Simulation Settings

In edge environments, most users will have fewer resources. Most users will tent to mint PoS blocks to get reasonable revenue. A small number of users will conduct PoW mining to maximize their profit. In the simulation, we set 20 miners mining PoW blocks. Having 20 miners is reasonable for a private blockchain on the edge. The maximum computational power for a miner is set to 1% of the expected power to generate a PoW block per time unit [31]. We set the expected ratio between PoS and PoW blocks τ to 20, and the related PoS factor $\frac{M}{(N+1)BU}$ is set to 1 for easy computation. We test the distribution of computational power devoted to mining and the coins for a new PoW block under different settings of non-mining revenue factor β_i and ratio amendment k. The Algorithm 1 will stop its iteration when the variance of α of different miners becomes less than 3%. We implement the Stackelberg game and algorithm, and solve the quasiconvex optimization problem (9) using CVXPY [32].

B. Performance under Different Incentive Assignment Settings

We first evaluate the incentive coin number α and the computational power denotation of miners for a new block under different incentive assignment settings k and β_i. We set β_i for each miner the same for a fair comparison and denoted as β in the following. We conduct Algorithm 1 under k and β_i equal 1 to 5 respectively, and the computational capacity of miners is set to 10 units. The algorithm iteration will on average terminate in 4.88 rounds, with the maximum 10 rounds and the minimum 2 rounds.

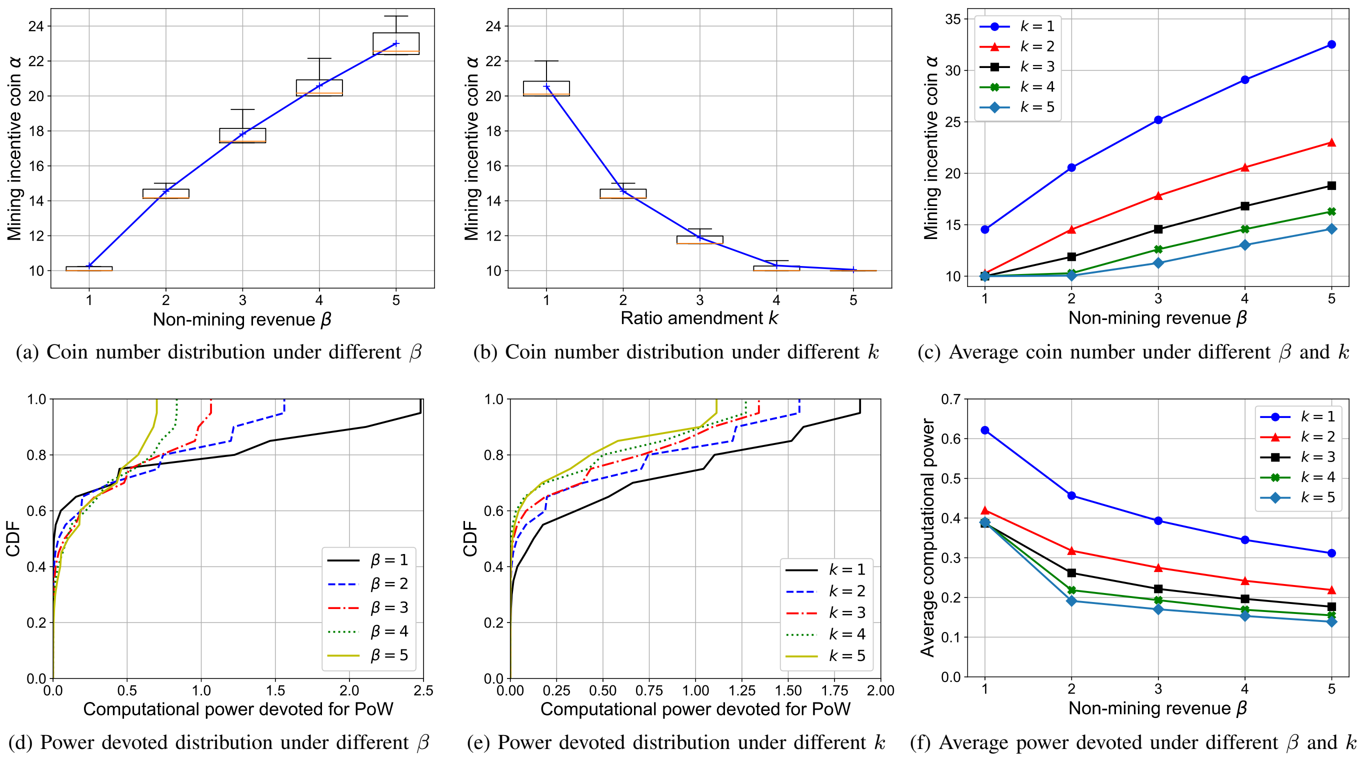
Fig. 4(a) shows the incentive coin number α a miner can get under different non-mining revenue β when $k = 2$. The larger the β is, the higher the α will be. Since larger β will make miners use less computational power on PoW mining, the incentive will increase to lure more miners to participate in mining. Meanwhile, under each parameter setting, the mean α (shown on the blue line) is larger than the median (the orange bar). This indicates that most miners tend to have a lower α in which case other miners mine the next block will not have too many coins. In real cases, the mean value can be chosen as the values for the number of coins for a new block.

Fig. 4(b) shows the incentive coin number α a miner can get under different ratio amendment k when $\beta = 2$. The α decreases close to inverse proportion as the k value increases. Since $k = \frac{T}{\alpha}$, longer time between to PoW blocks when k increases. Under the same computational hardness requirement, miners can devote less computational power to satisfy the hardness. Thus, the overall α will decrease, and the inverse proportion matches the time settings. Meanwhile, a longer time between two PoW blocks also means more PoS in between, which means miners may need more resources to validate, which in turn can give miners more transaction fees to compensate costs.

Fig. 4(c) denotes the overall change of incentive coin numbers under different k and β. The changing of α matches the observation as we describe above in a range under different k and β. Note that with higher k settings and lower β settings, the α will not decrease too much as it must satisfy the minimum requirement of the computational hardness target. The minimum α under such settings is about 9.99 coins per block.

Fig. 4(d) shows the cumulative distribution of the computational power devoted by miners to the PoW mining process under different non-mining revenue factor β when $k = 2$. When β is larger, miners will devote less to the mining since less revenue can get from mining. When $\beta = 5$, all miners devoted 0.75 computational power unit or less, while 20% miners devoted more than 1 computational power unit when $\beta = 1$. Meanwhile, smaller β makes the computational power denotation more disperse. There are 60% miners devoted 0.05 or less when $\beta = 1$, and will increase to 0.2 for the same ratio of miners when $\beta = 5$. This is because lower β will have lower α, which may reduce the motivation for some miners when other miners want to devote a lot in mining.

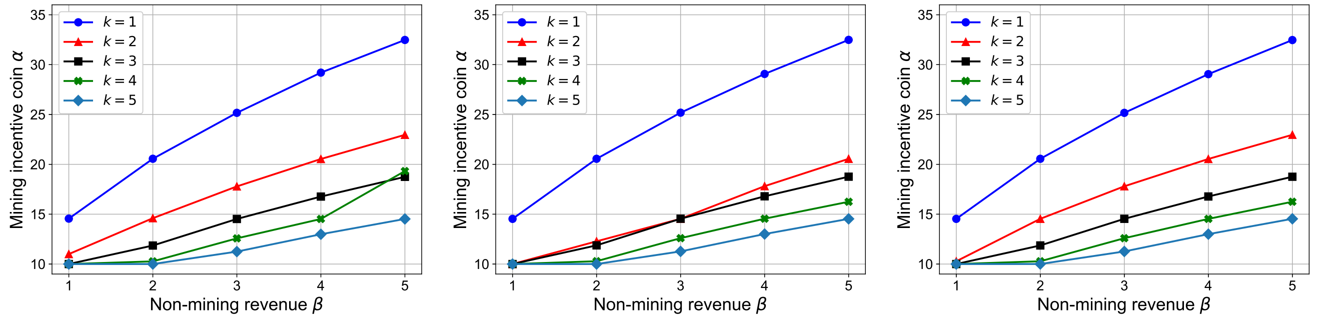
Fig. 4(e) shows the cumulative distribution of the computational power devoted by miners to the PoW mining process under different ratio amendment k when $\beta = 2$. When k is larger, miner will devote less to the mining since α is decreasing. When $k = 5$, all miners devoted 1.2 computational power unit or less, while 20% miners devoted more than 1.2 computational power unit when $k = 1$. Miners will have a longer time for mining blocks thus can use less computational power on mining.



(a) Coin number distribution under different β (b) Coin number distribution under different k (c) Average coin number under different β and k

(d) Power devoted distribution under different β (e) Power devoted distribution under different k (f) Average power devoted under different β and k

Fig. 4. The number of coins for mining a new block and computational power devoted to PoW process under different parameter settings. The number of coins increases as non-mining revenue β grows, and decreses as ratio amendment k grows. The computational power devoted decrease as β or k grwos.



(a) Zipf distribution of computational capacity (b) Normal distribution of computational capacity (c) Cluster distribution of computational capacity

Fig. 5. The number of coins for mining a new block under different distributions of computational capacities of miners. The general trend shows that the number of coins increases as non-mining revenue β grows, and decreases as ratio amendment k grows under all diffrent distributions.

Fig. 4(f) denotes the average computational power devoted under different k and β. The trend matches the observation as we describe above. The higher k or β is, the smaller the average computational power will be devoted from miners. Note that when $\beta = 1$, increasing k will not decrease the average computational power devoted. This is because, under such circumstances, α is relatively low, increasing k can not satisfy the PoS ratio requirement, thus minimum PoW block generation time is determined, and certain computational power is needed for mining.

C. Performance under Different Computational Capacities of Miners

We now evaluate the incentive coin number α under different computational capacity distributions of miners. In real edge environments, devices are of different makes and models, which do not have the same capacity on computational power. To address this issue, we test the incentive assignment mechanism under different computational capacity distributions. We test two popular distribution as Zipf and normal distribution. We also propose a situation that there are several different clusters of miners that have the same capacity inside each group. We called this cluster distribution. We then conduct Algorithm 1 under k and β_i equal 1 to 5 respectively. The average computational capacity of miners is set to 10 units. For cluster distribution, we set 4 clusters of miners which have the computational capacity 4, 8, 12, and 16 units respectively.

Fig. 5 denotes the overall change of incentive coin numbers under different k and β for (a) Zipf distribution, (b) normal distribution, and (c) cluster distribution. The changing of α

matches the results for the uniform distribution as we denoted in previous simulations results. The lower k or the higher β is, the larger α will be. This shows that our proposed mining incentive assignment mechanism can work under networks with high device heterogeneity. It is worth noting that the minimum α requirement also holds here as the same parameter settings, which further indicates the device heterogeneity does not have a high impact on the overall performance of our proposed mechanism.

VII. Conclusion and Future Work

In this paper, we have proposed a PoS-PoW hybrid blockchain system considering the limitations of the edge environments. The system utilizes the heterogeneity of devices making some resource-rich users conduct Proof of Work to enhance the security for transactions in edge environments. We have raised the incentive assignment problem for Proof of Work miners to get fair incentives when mining the new block. We have formulated the problem to a two-stage Stackelberg game and have proposed a Gauss-Seidel based iterative algorithm. We have proven that the proposed algorithm can converge and obtain global optimum. Simulation results show that our proposed incentive assignment mechanism let miners for new PoS block get reasonable incentive under different system parameters in a small-scale, private edge blockchain.

Over time, users will join or leave the blockchain network. This will affect the total computational power for Proof of Work, which may affect the security of the blockchain and the participation of users in the Proof of Work process. The computational hardness needs to be adjusted when the system running over time. In the future, we will discuss the computational hardness target adjustment based on dynamic user behaviors.

Acknowledgements

This work is supported in part by US National Science Foundation under grant numbers 1513719 and 1730291.

References

[1] "Gumroad," https://gumroad.com/, [Online; accessed 10-Feb-2019].

[2] "Is gumroad a scam?" https://www.quora.com/Is-Gumroad-a-scam, [Online; accessed 10-Feb-2019].

[3] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[4] I. Eyal and E. G. Sirer, "Majority is not enough: Bitcoin mining is vulnerable," in *International conference on financial cryptography and data security*. Springer, 2014, pp. 436–454.

[5] M. Iansiti and K. R. Lakhani, "The truth about blockchain," *Harvard Business Review*, vol. 95, no. 1, pp. 118–127, 2017.

[6] C. Ye, G. Li, H. Cai, Y. Gu, and A. Fukuda, "Analysis of security in blockchain: Case study in 51%-attack detecting," in *2018 5th International Conference on Dependable Systems and Their Applications (DSA)*. IEEE, 2018, pp. 15–24.

[7] N. T. Courtois and L. Bahack, "On subversive miner strategies and block withholding attack in bitcoin digital currency," *arXiv preprint arXiv:1402.1718*, 2014.

[8] K. Fanning and D. P. Centers, "Blockchain and its coming impact on financial services," *Journal of Corporate Accounting & Finance*, vol. 27, no. 5, pp. 53–57, 2016.

[9] V. Buterin *et al.*, "A next-generation smart contract and decentralized application platform," *white paper*, 2014.

[10] M. Research, "Edge computing," https://www.microsoft.com/en-us/research/project/edge-computing/, Oct. 2008.

[11] Y. Huang, X. Song, F. Ye, Y. Yang, and X. Li, "Fair caching algorithms for peer data sharing in pervasive edge computing environments," in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2017, pp. 605–614.

[12] A. Samanta, L. Jiao, M. Mühlhäuser, and L. Wang, "Incentivizing microservices for online resource sharing in edge clouds," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 420–430.

[13] Y. Zhong, K. Xu, X.-Y. Li, H. Su, and Q. Xiao, "Estra: Incentivizing storage trading for edge caching in mobile content delivery," in *2015 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2015, pp. 1–6.

[14] J. Huang, L. Kong, G. Chen, M.-Y. Wu, X. Liu, and P. Zeng, "Towards secure industrial iot: Blockchain system with credit-based consensus mechanism," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3680–3689, 2019.

[15] A. S. Sani, D. Yuan, W. Bao, P. L. Yeoh, Z. Y. Dong, B. Vucetic, and E. Bertino, "Xyreum: A high-performance and scalable blockchain for iiot security and privacy," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 1920–1930.

[16] H. Liu, Y. Zhang, and T. Yang, "Blockchain-enabled security in electric vehicles cloud and edge computing," *IEEE Network*, vol. 32, no. 3, pp. 78–83, 2018.

[17] D. B. Rawat, "Fusion of software defined networking, edge computing, and blockchain technology for wireless network virtualization," *IEEE Communications Magazine*, vol. 57, no. 10, pp. 50–55, 2019.

[18] Y. Huang, J. Zhang, J. Duan, B. Xiao, F. Ye, and Y. Yang, "Resource allocation and consensus on edge blockchain in pervasive edge computing environments," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2019, pp. 1476–1486.

[19] L. Wu, L. Li, X. Li, Y. Yu, L. Zhang, M. Pan, and Z. Han, "Resource allocation in blockchain system based on mobile edge computing networks," in *2019 11th International Conference on Wireless Communications and Signal Processing (WCSP)*. IEEE, 2019, pp. 1–6.

[20] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten, "Sok: Research perspectives and challenges for bitcoin and cryptocurrencies," in *Security and Privacy (SP), 2015 IEEE Symposium on*. IEEE, 2015, pp. 104–121.

[21] Nxt community, "Nxt whitepaper," http://nxtwiki.org/wiki/Whitepaper:Nxt, 2014, [Online; accessed 10-Feb-2019].

[22] S. King and S. Nadal, "PPcoin: Peer-to-peer crypto-currency with proof-of-stake," https://peercoin.net/whitepaper, 2012, [Online; accessed 10-Feb-2019].

[23] Z. Liu, S. Tang, S. S. Chow, Z. Liu, and Y. Long, "Fork-free hybrid consensus with flexible proof-of-activity," *Future Generation Computer Systems*, vol. 96, pp. 515–524, 2019.

[24] R. P. d. Santos, "Pow, pos, & hybrid protocols: A matter of complexity?" *arXiv preprint arXiv:1805.08674*, 2018.

[25] Z. Chen, Y. Liu, B. Zhou, and M. Tao, "Caching incentive design in wireless d2d networks: A stackelberg game approach," in *2016 IEEE International Conference on Communications (ICC)*. IEEE, 2016, pp. 1–6.

[26] Y. Zeng, P. Zhou, J. Liu, and Y. Yang, "A stackelberg game framework for mobile data gathering in leasing residential sensor networks," in *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*. IEEE, 2018, pp. 1–6.

[27] P. Fairley, "Ethereum will cut back its absurd energy use," *IEEE Spectrum*, vol. 56, no. 1, pp. 29–32, 2018.

[28] Z. Han, D. Niyato, W. Saad, T. Başar, and A. Hjørungnes, *Game theory in wireless and communication networks: theory, models, and applications*. Cambridge university press, 2012.

[29] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[30] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods*. Prentice hall Englewood Cliffs, NJ, 1989, vol. 23.

[31] T. T. A. Dinh, J. Wang, G. Chen, R. Liu, B. C. Ooi, and K.-L. Tan, "Blockbench: A framework for analyzing private blockchains," in *Proceedings of the 2017 ACM International Conference on Management of Data*, 2017, pp. 1085–1100.

[32] S. Diamond and S. Boyd, "Cvxpy: A python-embedded modeling language for convex optimization," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2909–2913, 2016.