

Edge Resource Pricing and Scheduling for Blockchain: A Stackelberg Game Approach

Sijie Huang, He Huang[✉], Senior Member, IEEE, Guojia Gao[✉], Member, IEEE, Yu-E Sun[✉], Member, IEEE, Yang Du[✉], Member, IEEE, and Jie Wu[✉], Fellow, IEEE

Abstract—Blockchain came to prominence as the distributed ledger underneath Bitcoin, which protects the transaction histories in a fully-connected, peer-to-peer network. The blockchain mining process requires high computing power to solve a Proof-of-Work (PoW) puzzle, which is hard to implement on users' mobile devices. So these miners may leverage the edge/cloud service providers (ESPs/CSP) to calculate the PoW puzzle. The existing edge-assisted blockchain networks assumed that all ESPs have a uniform propagation delay, which is unrealistic. In this article, we consider a more practical scene where ESPs locate in diverse positions of the blockchain network, which causes different propagation delays when supporting the computation of the PoW puzzle. Additionally, these ESPs connect to a remote CSP for resource scheduling when the computing tasks exceed their maximum capacity. The blockchain mining process generally involves complicated competition and games among CSP, ESPs, and miners. Each service provider focuses on how to determine his resource price so that he can maximize his utility. According to the set resource price, each miner concentrates on scheduling his resource requests for each ESP to maximize individual personal utility, which depends on ESPs' resource price and propagation delays. We first model such a resource pricing and scheduling problem as a three-stage multi-leader multi-follower Stackelberg game and aim at finding the Stackelberg equilibrium. Then, we analyze the subgame optimization problem in each stage and propose an iterative algorithm based on backward induction to achieve the Nash equilibrium of the Stackelberg game. Finally, extensive simulations are conducted to verify the significant performance of the proposed solution.

Index Terms—Blockchain, edge computing, game theory, resource pricing, resource scheduling, propagation delay

1 INTRODUCTION

NOWADAYS, electronic payment has become a daily transaction method that brings great convenience to people. Internet commerce relies almost entirely on trusted third-party financial institutions to process electronic payments. However, the existence of third-party financial institutions undoubtedly increases the additional cost of transactions and limits the actual transaction scale. In 2008, a new peer-to-peer electronic payment system, called "Bitcoin," was introduced, which can avoid the additional cost caused by the third-party financial institutions [2], [3], [4]. Bitcoin has been widely used in the past few years due to its decentralized particularity. As one popular digital cryptocurrency,

Bitcoin can be used across countries without worrying about being frozen by any financial institutions [5], and also can record and store all digital transactions in a decentralized append-only public ledger called "blockchain". Blockchain technology is applied in the Bitcoin field to record transactions and prevent tampering. Specifically, the data of digital transactions was packaged in the form of the linked blocks, in which each block is encrypted by using the Hash technique to ensure its security. With the background of blockchain technology, Bitcoin has attracted a lot of attention from digital transaction enthusiasts worldwide and has formed a substantial peer-to-peer transaction network. The core task of a blockchain network is to ensure that the trustless nodes in the network reach the agreement upon a single tamper-proof record of transactions, i.e., consensus mechanism [6]. The consensus mechanism allows thousands of nodes scattered around the world to agree on creating blocks. It also includes an incentive mechanism that promotes the effective operation of the blockchain system, which is the basis for building trust in the blockchain. Therefore, some blockchain networks, e.g., Bitcoin, incorporate an incentive-based block creation process known as "block mining" in their protocols. There are several consensus mechanisms like Proof of Work (PoW), Proof of Stake (PoS), Delegated Proof of Share (DPoS), and Byzantine Fault Tolerant (BFT) commonly used in blockchain [6], [7], [8]. Nakamoto proposed the PoW scheme for the Bitcoin blockchain network in [2]. PoW is a computation-intensive puzzle-solving race in which a node needs to achieve a hash querying rate as high as possible, which was used by the early Bitcoin network as a mining algorithm. Therefore, this

- Sijie Huang, He Huang, Guojia Gao, and Yang Du are with the School of Computer Science and Technology, Soochow University, Suzhou, Jiangsu 215006, China. E-mail: 20205227091@stu.suda.edu.cn, {huangh, gjgao, duyang}@suda.edu.cn.
- Yu-E Sun is with the School of Rail Transportation, Soochow University, Suzhou, Jiangsu 215006, China. E-mail: sunye12@suda.edu.cn.
- Jie Wu is with the Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122 USA. E-mail: jiewu@temple.edu.

Manuscript received 24 August 2021; revised 24 March 2022; accepted 18 May 2022. Date of publication 24 May 2022; date of current version 10 April 2023. The work of Sijie Huang, He Huang, Guojia Gao, Yu-E Sun, and Yang Du was supported by the National Natural Science Foundation of China (NSFC) under Grants U20A20182, 62102275, 61873177, 62072322, in part by the NSF of Jiangsu in China under Grant BK20210704, and in part by the NSF of the Jiangsu Higher Education Institutions of China under Grant 21KJB520025. The work of Jie Wu was supported by NSF Grants CPS 2128378, CNS 2107014, CNS 2150152, CNS 1824440, CNS 1828363, and CNS 1757533. (Corresponding authors: He Huang and Guojia Gao.) Digital Object Identifier no. 10.1109/TSC.2022.3177438

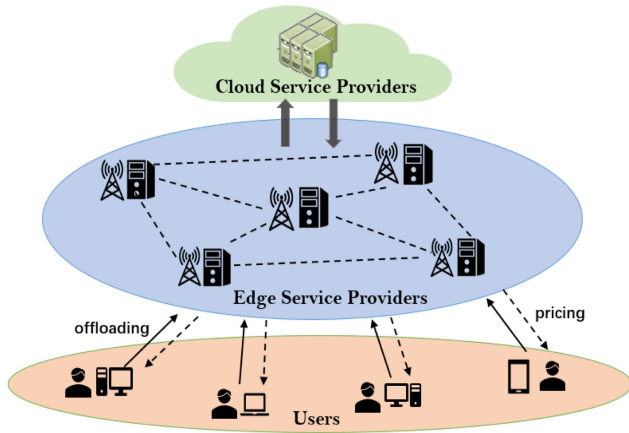


Fig. 1. Edge-assisted blockchain mining networks.

paper will study the block mining problem under the PoW mechanism based on the current work.

The block mining process in the Bitcoin system can be regarded as the following steps. In order to link a block to the blockchain, miners are first required to solve a computationally challenging puzzle. Then, each miner propagates his mined block to all blockchain network users to make this block be verified as soon as possible. This is because only when a block is confirmed by the majority of miners in this network can it be considered added to the end of the blockchain successfully. In other words, the consensus protocol of blockchain can be achieved. Only the miner who successfully links a block to the existing blockchain can gain a certain amount of Bitcoin as the mining incentive.

The PoW-based blockchain is a critical technology, which is considered as a technological innovation in the peer-to-peer network [9]. The security and reliability are thus ensured by this mechanism which requires numerous trials for a valid solution [10]. However, the PoW-based blockchain has a limitation: the blockchain mining process needs a mass of computation and storage resources, which is hard to be satisfied with a miner's terminal devices. Thanks to the development of edge/cloud computing techniques, the miners can take on lease some on-demand resources from the edge/cloud service providers (ESPs/CSP) to efficiently complete the mining process. For instance, [11] developed a cooperative computation offloading and resource allocation framework for blockchain-enabled mobile-edge computing systems; [8] studied a blockchain service market composed of the infrastructure provider, blockchain provider, cyber-insurer, and users. We call it the edge-assisted blockchain mining network, as shown in Fig. 1.

The mining process in edge-assisted blockchain networks is described as a speed game. First, miners send their mining requests to servers, i.e., renting some computation and storage resources to calculate the PoW puzzle. Then, suppose any server calculated the PoW puzzle. In that case, he needs to propagate the block to all of the other nodes in this edge-assisted blockchain network as soon as possible to make this block become the first one to realize the consensus principle. The miner who packages the block on this ESP and successfully takes the lead in reaching the consensus principle is the winner of the mining process. Agreeing on an identical blockchain by all nodes is also called "block

convergence". Ideally, nodes should hear about freshly mined blocks as quickly as possible. Here, a new block will be validated earlier by other nodes if it can be spread to the whole blockchain network faster [12]. The block convergence of blockchain may be disrupted by the increased network latency (i.e., propagation delay). In other words, even if two nodes solve the PoW problem simultaneously, the block packaged by one node may be discarded because the propagation time is longer than that of the other block.

Note that, since the mining process is a speed race, the transmission delay in offloading phase and the propagation delay in the block convergence phase are both crucial for a miner. The user who completes the above steps in a shorter period will have a higher probability of winning the mining game. Therefore, compared with the central cloud that is farther away and has higher latency, miners are more inclined to request edge servers located closer with shorter propagation latency, i.e., renting some computation and storage resources from ESPs. In the edge-assisted blockchain system, each ESP has its own maximum capacity that it can only provide limited computing services. Only when the application volume of miners exceeds its capacity will the ESP chooses to transfer some computing tasks to the central cloud to ensure the quality of service for miners.

However, the existing edge-assisted blockchain networks assumed that all ESPs have a uniform propagation delay, which is not practical in the real world. Due to the characteristics of the consensus protocol on the blockchain, the mining process is time-sensitive. The servers at the center of the blockchain network are more likely to be ahead during the block convergence phase than the other servers. Edge servers are affected by their geographic locations and even cause the propagation block to be abandoned due to the long convergence duration. Therefore, the geographical location of the edge server is an important consideration when studying edge-assisted blockchain networks. In this paper, we consider a price-based resource management mechanism with propagation delay in edge-assisted blockchain networks, in which ESPs have different propagation delays due to the different geographic locations. Further, we propose a three-stage multi-leader multi-follower Stackelberg game model between the computing service providers and miners, as shown in Fig. 2. In the first stage, the CSP first declares the unit price of its computing resources. In the second stage, ESPs set the unit price charged for providing services to miners. In the third stage, the miners decide the service demands to rent computation resources from ESPs according to the set prices. We discuss the Stackelberg equilibrium where both the profits of the ESPs and the utility of miners can be maximized simultaneously. In addition, we study the impact of propagation delays on block convergence.

Our contributions are summarized as follows:

- We consider a three-layer edge-assisted blockchain mining network model, i.e., miners, ESPs, and CSP. Each miner studies how to maximize his individual utility which depends on the resource price and the propagation delay of each ESP, while all ESPs focus on resource pricing and scheduling to maximize their utility. We are the first to consider the impact on block convergence of different propagation delays due to

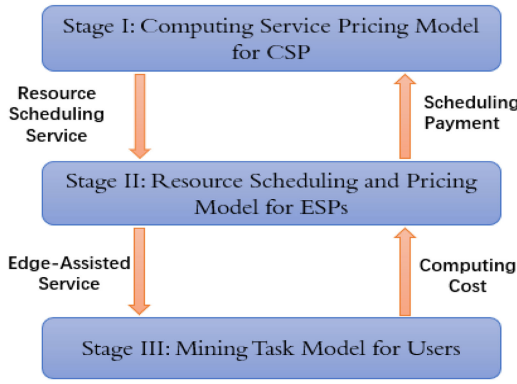


Fig. 2. Three-stage Stackelberg Game.

the distributed geographic locations. Here, ESPs' propagation delays will significantly affect the probability of blockchain mining success.

- To solve the competition and game among the CSP, ESPs, and miners, we propose a special three-stage multi-leader multi-follower Stackelberg game model. The CSP and ESPs are seen as resource renters and allocators in stages I and II, respectively; the miners apply for the computation assist from ESP as the followers in stage III. The proposed model takes the ESPs' propagation delays and the impact of CSP into consideration when calculating the miners' utility.
- We derive the explicit-form expressions of the most beneficial price strategies for each ESP and the optimal resource requests for each miner simultaneously. Furthermore, we analyze the existence and uniqueness of the Stackelberg equilibrium (SE), based on which we propose a corresponding algorithm to obtain the SE.
- Extensive simulations are conducted to verify the significant performance of the proposed solution.

The remainder of the paper is organized as follows. We first present the system model and formulate the optimization problem in Section 2. Next, we propose the solution and analyze the Nash Equilibrium point in Section 3. In Section 4, we conduct lots of simulations to verify the performance of the proposed algorithms. After reviewing the related work in Section 5, we conclude the paper in Section 6.

2 SYSTEM MODEL AND PROBLEM FORMULATION

We first present the edge-assisted blockchain model and introduce some corresponding notations. Next, we present the Stackelberg game on resource pricing and scheduling between miners and edge/cloud service providers, and further formalize the optimization problem.

In this paper, we consider the public blockchain mining networks based on the PoW consensus protocol. There are many ordinary network users (called miners) in the blockchain mining networks, trying to complete the transaction package (called block) to pursue some rewards. More specifically, the success of a miner appending the block to the end of the current blockchain contains two steps. 1) The miner needs to solve the PoW puzzle to ensure security and validity, called the mining procedure. 2) The miner must broadcast his results to other network users in the blockchain,

which is a broadcasting procedure to realize the consensus principle. During the mining procedure, the PoW puzzle that the miners try to solve highly depends on the computation resources of the miners' terminal devices. In other words, the miners with more computation resources will have a higher probability of solving this PoW puzzle. However, the PoW mining process needs to perform a large number of hash computing operations, and this computation-intensive mining task is too heavy for miners' terminal devices. With the help of the edge/cloud computing technique, the miners can take on lease some on-demand resources from ESPs. Hence, terminal devices in this blockchain network offload the computation-intensive PoW mining tasks to the ESPs. Note that the ESPs are geographically distributed at the network edge so that the network users can access the ESPs via the wireless local area networks. These ESPs connect to the remote CSP through a core network, as shown in Fig. 1. Each ESP has a limited computing resource capability, while the CSP is assumed to have unconstrained computation resources. For an ESP, when the total resources requested from the miners exceed his capacity, he will upload part of his requests to the CSP.

Fig. 1 shows a cooperative edge-assisted blockchain network, which consists of a CSP and several ESPs denoted as $\mathcal{N} = \{1, \dots, j, \dots, n\}$, and there are m miners, denoted as $\mathcal{M} = \{1, \dots, i, \dots, m\}$, allowed to communicate with ESPs simultaneously. To complete the PoW puzzle, the miners will purchase computing services from ESPs or CSP. When the ESPs or CSP calculates the PoW puzzle, he will try to broadcast its result to all miners as soon as possible. In such a way, the corresponding miner who rents the computation resources may become the first one to realize the consensual block ahead of other competitors. Only the first miner who reaches the consensual block principle can obtain the reward in the blockchain networks. Note that in addition to the time of calculating out the PoW puzzle, the propagation delay of spreading the results to other miners is also an important factor. In the system model, we assume that these n ESPs are sorted in descending order of the propagation delay. Also, the propagation delay of the CSP is greater than that of any ESP due to the remotest location.

We formulate the interaction process between miners, ESPs, and CSP as a three-stage Stackelberg game as Fig. 2. In the model, the miners who rent the computation resources from the ESPs can be seen as buyers, while the ESPs can be seen as the sellers of computation resources. However, when the total demands of an ESP exceed its maximum capacity, the ESP has to offload some requests to the CSP to obtain more calculation resources support. Under this situation, ESP becomes the buyer, and CSP acts as a seller by selling his computing resources. The three-stage Stackelberg game can be described as follows: The CSP first declares the unit price of its computing resources. Then, ESPs set the unit price charged for providing calculation services to miners. Miners determine their requests for each ESP's computing resources according to the set price and offload PoW mining tasks to ESPs. When the total requested resources of an ESP from the miners are over the maximum capacity, ESP will purchase the computing resource from CSP to complete the hash task of miners. Since all the miners, ESPs, and CSP are rational entities, they always focus on maximizing their individual

utility. Thus, there is an unavoidable game in the edge-assisted blockchain mining networks on the resource pricing and scheduling between the miners and service providers.

2.1 Stage I: CSP Side Utility

In stage I, the CSP acts as a leader, who determines the unit price of computing resources and provides the calculation service to ESPs. The utility function of the CSP can be expressed as the charged fee minus the cost of electricity, which is written as follows:

$$U^{csp} = (p^{csp} - c)Q, \quad (1)$$

where p^{csp} is the unit price of computing support resource paid by ESPs to CSP, and c is the unit electricity cost for providing computing services. Q represents the total request uploaded to the CSP by all ESPs due to their limited computing resource capacity, i.e., $Q = \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{N}} (1 - \tau_i^j) x_i^j$. Then, we formulate the subgame optimization problem at this stage, which maximizes the revenue of the CSP and seeks the optimal unit price of resource services. The subgame optimization problem P1 in stage I is:

$$\begin{aligned} \text{P1: maximize } & U^{csp} = (p^{csp} - c)Q \\ \text{subject to } & p^{csp} \geq 0, c \geq 0 \end{aligned} \quad (2)$$

2.2 Stage II: ESP Side Utility

In stage II, ESPs are the followers of stage I determining how many requests will be offloaded to the CSP. On the other hand, the ESPs also become the leader of Stage II and decide the unit price for providing hash computing services to miners. Actually, in this stage, the competition in ESPs forms as a non-cooperative subgame, where each ESP sets his unit price by considering miners' requests and other ESPs' prices. The profits of ESPs come from the payments of miners. When an ESP accepts a miner's request, the miner must pay the ESP for his computing resource services. The utility of each ESP is expressed as the received payment minus the corresponding cost. On the other hand, when a miner's request is uploaded to the CSP through one ESP, this ESP's utility equals the miner's payment minus the cost that this ESP provides for the CSP. The utility of j th ESP is defined as:

$$U_j^{esp} = (p_j - c) \cdot E_j + (p_j - p^{csp}) \cdot Q_j, \quad (3)$$

where c is the unit electricity cost for providing service on the ESPs and p^{csp} is the ESP's payment for renting unit computing resources from CSP. E_j denotes the total computation resources that the ESP j provides for the miners while Q_j means the total computation resources that ESP j rents from the CSP, that is, $E_j = \sum_{i \in \mathcal{M}} \tau_i^j x_i^j$ and $Q_j = \sum_{i \in \mathcal{M}} (1 - \tau_i^j) x_i^j$. According to this, we formulate the subgame optimization problem at Stage II. This problem maximizes the revenue of ESPs and aims at finding the optimal unit price of computing services charged by miners. We have the subgame optimization problem P2 as follows:

$$\begin{aligned} \text{P2: maximize } & U_j^{esp} = (p_j - c) \cdot E_j + (p_j - p^{csp}) \cdot Q_j \\ \text{subject to } & \sum_{i \in \mathcal{M}} \tau_i^j x_i^j \leq K_j \end{aligned} \quad (4)$$

where K_j is the maximum computation resource capacity of ESP j , and it is the common knowledge of all the parties in this game. Here, when the sum of requests from all miners on ESP j exceeds his capacity constraints, he has to upload part of the received requests to the CSP. However, the long propagation delay may decrease the winning probability for the applied miners, which will in turn reduce the miners' requests. Thus, it is quite challenging for each ESP to set a suitable unit price for the computing resources so that he can maximize his utility.

2.3 Stage III: Miner Side Utility

In stage III, miners are the followers of stage I and stage II. After knowing the price and capacity of the other parties, the miner needs to decide their computing demands offloaded to ESPs. For each miner i , we use $X_i = (x_i^1, x_i^2, \dots, x_i^n)$ to denote the request for these n ESPs, and use B_i to denote its budget. We suppose that each ESP in the edge-assisted blockchain mining networks has the same unit computing power as CSP. The probability of solving the PoW problem is related to computing power. Therefore, without loss of generality, we denote the computing power, i.e., the probability of miner i calculating the PoW problem on ESP j , as α_i^j , indicating the percentage of the resources he rents on ESP j to the total computation resources rented by ESPs. This computation offloading model in edge-assisted blockchain networks draws lessons from the existing blockchain mining pool leasing business model [13], that is:

$$\alpha_i^j = \frac{x_i^j}{\sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{N}} x_i^j}. \quad (5)$$

To facilitate writing, we denote $All = \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{N}} x_i^j$

As introduced above, the process of a miner winning the reward consists of two procedures, i.e., the mining procedure and the broadcasting procedure. The mining procedure is relative to the computing power of a miner applying for an ESP, while the propagation delay affects the winning probability of the broadcasting procedure. The long propagation delay may diminish the chances of winning if an ESP propagates a block slowly to other miners in the broadcasting procedure. In other words, the miner that first calculates the PoW puzzle and packages a block may fail to get the reward because someone else takes the lead in broadcasting the packaged block successfully and realizes the consensus protocol. This is because this block is likely to be discarded because of long propagation delay, which is called orphaning [14]. This paper considers that the propagation delay comes from two parts. The first one is block size: a large block will propagate slowly to other miners in the propagation step. Another factor that affects propagation delay is the geographic location of the ESP. The ESP located in the center of the blockchain network has a shorter propagation delay than that in the network edge to get the block convergence. Based on the existing works [13], [15], we also consider that the block mining time follows the Poisson distribution. The orphaning probability on the propagation delay caused by ESP j and the size of the block, denoted as $P_{\text{orphan}}(t_j, s_i)$, is approximated as:

$$P_{\text{orphan}}(t_j, s_i) = 1 - e^{-\lambda(t_j + s_i)}, \quad (6)$$

in which the parameter λ denotes the inter-arrival rate of the Poisson distribution. t_j means the propagation delay caused by ESP j due to the geographic location, and s_i is the number of transactions miner i chooses to be included in this block, i.e., the size of a block. Thus, the successful probability of mining game for miner i on ESP j is expressed as follows:

$$W_i^j = \alpha_i^j (1 - P_{\text{orphan}}(t_j, s_i)) = \frac{\tau_i^j x_i^j}{\text{All}} e^{-\lambda(t_j + s_i)}, \quad (7)$$

where W_i^j denotes the probability that ESP j is the first one who solves the PoW problem (i.e., packages a block) and broadcasts this block successfully, that is, making it be the first consensual block. Here, $\tau_i^j \in \{0, 1\}$ is the decision of the n -th ESP whether accepted the request of miner i according to its capacity. $\tau_i^j = 1$ means the request of miner i for ESP j is accepted by ESP j and the hash computing task will be carried out on this ESP. On the contrary, $\tau_i^j = 0$ denotes that this demand is uploaded to the CSP by ESP j . Note that an edge computing request may be sent to the remote CSP by the corresponding ESP due to his limited resource capacity. In such a case, the winning probability for the CSP is represented as follows:

$$W_i^{j'} = \frac{(1 - \tau_i^j) x_i^j}{\text{All}} e^{-\lambda(\bar{t} + s_i)}. \quad (8)$$

Here, due to the same unit computing power between ESPs and CSP, the probability of solving the PoW problem is relative to the overall computing power of all miners' demands. \bar{t} is the propagation delay of the CSP, which is bigger than t_j for $j \in \mathcal{N}$. Hence, the winning probability of miner i on all service providers can be summarized as:

$$W_i = \sum_{j \in \mathcal{N}} \left(\frac{\tau_i^j x_i^j}{\text{All}} e^{-\lambda(t_j + s_i)} + \frac{(1 - \tau_i^j) x_i^j}{\text{All}} e^{-\lambda(\bar{t} + s_i)} \right) \quad (9)$$

s.t. $\tau_i^j \in \{0, 1\}$

Miners' utility is defined as the expected reward minus the corresponding cost. The expected reward is computed by $R \cdot W_i$, in which R means the reward of successfully appending a block to the end of the existing blockchain, and W_i denotes the probability of the miner i winning the reward. On the other hand, the total cost of miner i is determined by the prices of ESPs, denoted as $\{p_1, \dots, p_j, \dots, p_n\}$, and this miner's service requests, i.e., $X_i = (x_i^1, x_i^2, \dots, x_i^n)$. Based on this, we formulate the subgame optimization problem at stage III, which maximizes the individual profits of the miner i as follows:

$$\begin{aligned} \text{P3: maximize } & U_i^{\text{miner}} = R \cdot W_i - \sum_{j \in \mathcal{N}} p_j \cdot x_i^j \\ \text{subject to } & \sum_{j \in \mathcal{N}} p_j \cdot x_i^j \leq B_i \end{aligned} \quad (10)$$

Here, the constraint B_i represents the budget of miner i . In addition, we list the commonly-used notations throughout this paper in Table 1.

3 STACKELBERG GAME EQUILIBRIUM ANALYSIS

We model the interactions between the CSP, ESPs, and miners as a multi-leader multi-follower three-stage Stackelberg

TABLE 1
Description of Commonly-Used Notations

Variable	Description
\mathcal{M}, \mathcal{N}	The sets of miners and ESPs, respectively.
i, j	The indexes for miners and ESPs.
x_i^j	The service demand of miner i for ESP j .
X_i	The service demand of miner i for all ESP.
K_j	The maximum capacity of ESP j .
B_i	The budget of miner i .
t_j / p_j	The propagation delay / unit price of ESP j .
τ_i^j	Miner i ' demand for ESP j is accepted or not.
c / h	The unit cost for providing service on ESP / the unit cost for uploading the request to CSP.
E_{all}	Total miners' request accepted by all ESPs.
C_{all}	Total request uploaded to CSP from all ESPs.
W_i^j	The probability of miner i winning on ESP j .
W_i	The probability of miner i
R	Reward of blockchain mining successfully.
U_i / V_j	The utility of miner i / ESP j

game with complete information. We aim at finding the Stackelberg equilibrium where the payoff of ESPs and miners can be maximized simultaneously. We first define the Stackelberg equilibrium point as follows.

Definition 1. Let \mathbf{x}^* , \mathbf{p}^* and $\mathbf{p}_{\text{csp}}^*$ denote the optimal service demand vector of all the miners, optimal unit price vector of ESPs and CSP computing service, respectively. Then, the point $(\mathbf{x}^*, \mathbf{p}^*, \mathbf{p}_{\text{csp}}^*)$ is the Stackelberg equilibrium if the following two conditions are satisfied:

$$U_j^{\text{esp}}(\mathbf{p}_j^*, \mathbf{p}_{\text{csp}}^*, \mathbf{x}^*) \geq U_j^{\text{esp}}(\mathbf{p}_j', \mathbf{p}_{\text{csp}}^*, \mathbf{x}^*) \quad (11)$$

$$U_i^{\text{miner}}(\mathbf{x}_i^*, \mathbf{x}_{-i}^*, \mathbf{p}^*) \geq U_i^{\text{miner}}(\mathbf{x}_i', \mathbf{x}_{-i}^*, \mathbf{p}^*) \quad (12)$$

Here, \mathbf{x}_{-i}^* is the best response service demand vector for all miners except miner i . Next, we will analyze the game equilibrium in the above model.

We use the backward induction method to obtain the Nash equilibrium of the Stackelberg game. We first solve the subgame problem P3 in stage III, then tackle the subgame problem P2 in stage II, and finally handle the stage I subgame problem P1.

3.1 Stage III: Miners' Requests Equilibrium

Based on the definition of Stackelberg game equilibrium, as the pricing strategies of all ESPs are given, each miner determines his service demands for each ESP as the best response. We first introduce the definition of the best response.

Definition 2. A request vector $\mathbf{x}_i^* \triangleq (x_i^{1*}, x_i^{2*}, \dots, x_i^{n*})$ is the optimal response service demand vector of the miner subgame if $U_i(\mathbf{x}_i^*, \mathbf{x}_{-i}^*, \mathbf{p}^*) \geq U_i(\mathbf{x}_i', \mathbf{x}_{-i}^*, \mathbf{p}^*)$.

In P3, the design variable X_n is the vector of computing requests offloaded to ESPs from miner n with the integer constraint. The strategy $\tau_i^j \in \{0, 1\}$ is a discrete variable. Therefore, the subgame optimization problem P3 is a mixed binary nonlinear integer programming problem. Generally, P3 can be solved by the traditional branch and bound algorithm to obtain the optimal solutions, but the algorithm has exponential complexity. To effectively tackle the subgame

problem $P3$, we need to design a low-complexity algorithm and make some simple relaxation operations.

We first continuously relax the binary variable τ_i^j to the following continuous constraint:

$$0 \leq \tau_i^j \leq 1 \quad i \in \mathcal{M}, j \in \mathcal{N} \quad (13)$$

Note that after continuous relaxation, τ_i^j can be regarded as a strategy factor. It represents the probability for j th ESP accepting the demand of i th miner, and further expresses the percentage of demands be accepted by ESP j from miner i , which is relative to the capacity of j th ESP. In this paper, we consider that all ESPs have the consistent capacity. Therefore, for $i \in \mathcal{M}, j \in \mathcal{N}$, τ_i^j to have the same value, we take the simple notation as τ .

Combining the above continuous relaxation of τ , Eq. (9) can be written as

$$W_i' = \sum_{j \in \mathcal{N}} \left(\frac{\tau x_i^j}{All} e^{-\lambda(t_j+s_i)} + \frac{(1-\tau)x_i^j}{All} e^{-\lambda(\bar{t}+s_i)} \right) \quad (14)$$

s.t. $0 \leq \tau \leq 1$

Then, the original subgame optimization problem $P3$ is equivalent to

$$\begin{aligned} P3' : \text{maximize} \quad & U_i^{miner} = R \cdot W_i' - \sum_{j \in \mathcal{N}} p_j \cdot x_i^j \\ \text{subject to} \quad & \sum_{j \in \mathcal{N}} p_j \cdot x_i^j \leq B_i \end{aligned} \quad (15)$$

Now, we analyze the existence of the Nash equilibrium in stage III subgame of the Stackelberg game. Before that, we first introduce an assumption as follows.

Assumption 1.

$$\Delta t < \frac{\ln m}{\lambda}, \text{ where } \Delta t = \max\{t_j - t_k\} \quad \forall i, k \in n \quad (16)$$

In fact, Assumption 1 is easy to be satisfied because λ is between 0 and 1 and much less than 1. Next, we prove that there exists a game equilibrium in our model, as described in Theorem 1.

Theorem 1. *Under Assumption 1, the existence and uniqueness of miner participation equilibrium, i.e., the Nash equilibrium of Stage II in this Stackelberg game, can be guaranteed.*

Proof. The strategy space of each miner is a non-empty, compact subset of the Euclidean space. From Eq. (15), U_i is apparently continuous with the variable x_i , which is the combination of requests for each ESP, i.e., x_i^j . We take the first order and second order derivatives of Eq. (15) with respect to x_i^j as follows:

$$\frac{\partial U_i^{miner}}{\partial x_i^j} = R \cdot \frac{\partial W_i'}{\partial x_i^j} - p_j, \quad (17)$$

$$\frac{\partial^2 U_i^{miner}}{\partial (x_i^j)^2} = R \cdot \frac{\partial^2 W_i'}{\partial (x_i^j)^2}. \quad (18)$$

Based on Eq. (14), we can take the second derivative of W_i as Eq. (19), Eq. (20) and Eq. (21), shown at the bottom of the page. It's obvious that $\sum_{k \in \mathcal{N}} x_i^k - All < 0$, then we need to prove,

$$\sum_{k \in \mathcal{N}} x_i^k e^{-\lambda(t_k - t_j)} - All < 0. \quad (22)$$

Eq. (22) can be transformed as

$$\sum_{k=1}^n \left(x_i^k e^{-\lambda(t_k - t_j)} - \sum_{i=1}^m x_i^k \right). \quad (23)$$

We take the expectation of Eq. (23) as follows:

$$\sum_{k=1}^n (x_i^k e^{-\lambda \Delta t} - m x_i^k). \quad (24)$$

Based on Assumption 1, we have the following result, i.e., $\sum_{k=1}^n (x_i^k e^{-\lambda \Delta t} - m x_i^k) < 0$, then Eq.(21) < 0 can be guaranteed. Thus, the miner participation sub-game is a concave game which always admits the Nash equilibrium.

By setting the first-order derivative of the miner's utility to 0, we have

$$\frac{\partial U_i^{miner}}{\partial x_i^j} = R \cdot \frac{\partial W_i'}{\partial x_i^j} - p_j = 0. \quad (25)$$

As for Eq. (26), shown at the top of the next page, we can obtain the best response of miner i as Eq. (27), shown at the top of the next page, where $All_{-i} = \sum_{l \neq i} \sum_{k \in \mathcal{N}} x_l^k$. The right part of Eq. (27) has nothing to do with x_i^j . In conclusion, if the amount of resources applied by other miners is large, the miner i needs to request more hash power to obtain a higher winning probability. \square

Since the model in this paper is relatively complex and involves multi-dimensional parameters, the solutions to

$$\frac{\partial W_i'}{\partial x_i^j} = -\frac{1}{All^2} \cdot \left[\sum_{k \in \mathcal{N}} \left(\tau x_i^k e^{-\lambda(t_k+s_i)} + (1-\tau)x_i^k e^{-\lambda(\bar{t}+s_i)} \right) \right] + \frac{1}{All} \left(\tau e^{-\lambda(t_j+s_i)} + (1-\tau)e^{-\lambda(\bar{t}+s_i)} \right) \quad (19)$$

$$= \frac{1}{All^2} \left[\tau \left(All e^{-\lambda(t_j+s_i)} - \sum_{k \in \mathcal{N}} x_i^k e^{-\lambda(t_k+s_i)} \right) + (1-\tau) e^{-\lambda(\bar{t}+s_i)} \left(All - \sum_{k \in \mathcal{N}} x_i^k \right) \right] \quad (20)$$

$$\frac{\partial^2 W_i'}{\partial (x_i^j)^2} = \frac{2}{All^3} \left[\tau e^{-\lambda(t_j+s_i)} \left(\sum_{k \in \mathcal{N}} x_i^k e^{-\lambda(t_k - t_j)} - All \right) + (1-\tau) e^{-\lambda(\bar{t}+s_i)} \left(\underbrace{\sum_{k \in \mathcal{N}} x_i^k - All}_{< 0} \right) \right]. \quad (21)$$

$$All^2 = \frac{R}{p_j} \left[\tau \sum_{k \neq j} x_i^k \left(e^{-\lambda(t_j+s_i)} - e^{-\lambda(t_k+s_i)} \right) + \tau All_{-i} e^{-\lambda(t_j+s_i)} + (1-\tau) All_{-i} e^{-\lambda(\bar{t}+s_i)} \right] \quad (26)$$

$$x_i^j = \sqrt{\frac{R}{p_j} \left[\tau \sum_{k \neq j} x_i^k \left(e^{-\lambda(t_j+s_i)} - e^{-\lambda(t_k+s_i)} \right) + \tau All_{-i} e^{-\lambda(t_j+s_i)} + (1-\tau) All_{-i} e^{-\lambda(\bar{t}+s_i)} \right]} - All_{-i} - \sum_{k \neq j} x_i^k. \quad (27)$$

the miners' Nash Equilibrium are infeasible to express in a symbolic manner. Fortunately, we are able to get the closed-form computation offloading solutions for the $P1$ in a special case. We consider a particular case where only one ESP is involved in this model. Its propagation delay is denoted t_0 , and this ESP determines its unit price of computing service as p_0 . We are interested in finding an NE where miners decide on a request to buy the computing service from ESP. Thus, the utility of miner is modified as

$$U_i^{\text{miner}} = R \cdot \left[\frac{\tau x_i}{\sum_{i \in \mathcal{M}} x_i} e^{-\lambda(t_0+s_i)} + \frac{(1-\tau)x_i}{\sum_{i \in \mathcal{M}} x_i} e^{-\lambda(\bar{t}+s_i)} \right] - x_i p_0 \quad (28)$$

Then, we let the first order derivatives of Eq.(28) equal 0 and we obtain

$$\frac{p_0}{R[\tau e^{-\lambda(t_0+s_i)} + (1-\tau)e^{-\lambda(\bar{t}+s_i)}]} = \frac{\sum_{k \neq i} x_k}{(\sum_{i \in \mathcal{M}} x_i)^2} \quad (29)$$

Next, we calculate the summation of this expression for all the miners as follows:

$$\sum_{i \in \mathcal{M}} \frac{p_0}{R[\tau e^{-\lambda(t_0+s_i)} + (1-\tau)e^{-\lambda(\bar{t}+s_i)}]} = \frac{M-1}{\sum_{i \in \mathcal{M}} x_i} \quad (30)$$

Combining Eq.(29) and Eq.(30), we have

$$\left(\frac{M-1}{\sum_{i \in \mathcal{M}} \beta} \right)^2 = \frac{\sum_{k \neq i} x_k}{\beta}, \quad (31)$$

where $\beta = \frac{p_0}{R[\tau e^{-\lambda(t_0+s_i)} + (1-\tau)e^{-\lambda(\bar{t}+s_i)}]}$. According to Eq. (30), we can get

$$\sum_{k \neq i} x_k = \frac{M-1}{\sum_{i \in \mathcal{M}} \frac{p_0}{R[\tau e^{-\lambda(t_0+s_i)} + (1-\tau)e^{-\lambda(\bar{t}+s_i)}]}} - x_i \quad (32)$$

Thus, we obtain the Nash equilibrium of the miner i as

$$x_i^* = \frac{M-1}{\sum_{i \in \mathcal{M}} \beta} - \beta \left(\frac{M-1}{\sum_{i \in \mathcal{M}} \beta} \right)^2 \quad (33)$$

There is a positive correlation between the best response and the total number of miners. As the number of miners increases, more competitors result in each user applying for more resources to increase the probability of winning.

3.2 Stage II: Optimal Pricing Mechanism

The ESPs are the followers of stage I choosing how many requests will be offloaded to the CSP. Next, ESPs act as the leaders of stage II determining the unit price of providing hash computing service to miners. Based on the Nash equilibrium of miners' demands in Stage I, we further analyze the benefits of ESPs defined in $P2$.

Theorem 2. U^{esp} is a concave function and Nash equilibrium of ESPs' subgame problem exists under the optimal strategy of miners.

Proof. To simplify the computation, we also consider the special case with only one ESP. We first keep the best response demands of miners as shown in Eq. (33) fixed and substitute the Eq. (33) into Eq. (4). Then, we have Eq. (34), shown at the bottom of this page. The first order and second order derivatives of U^{esp} with respect to p_0 are given as Eq. (36) and Eq. (37), shown at the bottom of this page. Due to the negative of Eq. (38), shown at the bottom of this page, the strict concavity of the objective function is ensured, which means that here exists a best response pricing strategy of ESP. Thus, the ESP can achieve the maximum profit with the unique optimal price, and it can achieve this goal by applying standard convex optimization algorithms, such as the interior-point algorithm and gradient projection algorithm. \square

$$U^{esp} = [\tau(p_0 - c) + (1-\tau)(p_0 - p^{csp})] \cdot \frac{M-1}{\sum_{i \in \mathcal{M}} \beta} \quad (34)$$

$$= [\tau(p_0 - c) + (1-\tau)(p_0 - p^{csp})] \cdot \frac{M-1}{p_0 \sum_{i \in \mathcal{M}} \frac{1}{R[\tau e^{-\lambda(t_0+s_i)} + (1-\tau)e^{-\lambda(\bar{t}+s_i)}]}} \quad (35)$$

$$\frac{\partial U^{esp}}{\partial p_0} = \frac{M-1}{\sum_{i \in \mathcal{M}} \beta} - [\tau(p_0 - c) + (1-\tau)(p_0 - p^{csp})] \frac{M-1}{\sum_{i \in \mathcal{M}} \frac{1}{R[\tau e^{-\lambda(t_0+s_i)} + (1-\tau)e^{-\lambda(\bar{t}+s_i)}]}} \frac{1}{p_0^2} \quad (36)$$

$$\frac{\partial^2 U^{esp}}{\partial (p_0)^2} = \frac{1}{p_0^2} \frac{2(M-1)}{\sum_{i \in \mathcal{M}} \frac{1}{R[\tau e^{-\lambda(t_0+s_i)} + (1-\tau)e^{-\lambda(\bar{t}+s_i)}]}} \left([\tau(p_0 - c) + (1-\tau)(p_0 - p^{csp})] \frac{1}{p_0} - 1 \right) \quad (37)$$

$$= \frac{1}{p_0^2} \frac{2(M-1)}{\sum_{i \in \mathcal{M}} \frac{1}{R[\tau e^{-\lambda(t_0+s_i)} + (1-\tau)e^{-\lambda(\bar{t}+s_i)}]}} \left(-\frac{\tau c + (1-\tau)p^{csp}}{p_0} \right) < 0 \quad (38)$$

3.3 Stage I: CSP's Participation Mechanism

CSP, the leader of stage I, decides the optimal unit price of the computing management service paid by ESPs to maximize his own revenue. In this stage, we construct the subgame optimization problem as $P1$.

Observing the structure of U^{csp} in problem $P1$, it can be shown that U^{csp} is a simple linear function of p^{csp} . Every participant in this game is selfish and rational, thus the revenue of CSP is non-negative and can be shown as:

$$p^{csp} \geq c \quad (39)$$

On the other hand, ESP's revenue in stage II shown in Eq. (34) is non-negative as well:

$$p^{csp} \leq p_0 + \frac{(p_0 - c)\tau}{(1 - \tau)} \quad (40)$$

Eq. (2) shows that the revenue function of CSP is monotonically increasing of p^{csp} . So we can get the maximum revenue of CSP when p^{csp} selects the maximum value, which is obtained by

$$p^{csp*} = p_0^* + \frac{(p_0^* - c)\tau}{(1 - \tau)}, \quad (41)$$

where p_0^* is the optimal price strategy of ESP in stage II and can be calculated by convex optimization algorithms.

Algorithm 1. Asynchronous Best Response

Input: Any feasible price p^{csp} $P = \{p_1, p_2, \dots, p_n\}$, miners' demands $X = \{X_1, X_2, \dots, X_m\}$, and the threshold ϵ

- 1: **for** iteration k **do**
- 2: storing last iteration $P^{(k-1)}$;
- 3: **Stage III: Miners Level Game**
- 4: **for** each miner i **do**
- 5: receiving the pricing strategy $P = \{p_1, p_2, \dots, p_n\}$ from all ESPs;
- 6: predicting the optimal requests of other miners;
- 7: calculating $x_i^{j(k)} = x_i^{j(k-1)} + \Delta \frac{\partial U_i(X_i^{(k-1)}, X_i^{(k-1)}, P)}{\partial x_i^j}$;
- 8: **if** Eq.(42) **then**
- 9: $x_i^{j(k-1)} \leftarrow x_i^{j(k)}$;
- 10: deciding his request $\mathbf{x}_i^{(k)} = \{x_i^1, x_i^2, \dots, x_i^n\}$;
- 11: sending $\mathbf{x}_i^{(k)}$ to ESPs;
- 12: **Stage II: ESPs Level Game**
- 13: **for** each ESP j **do**
- 14: Fix (X^*, p^{csp*}) .
- 15: increasing or decreasing the price with a step δ ;
- 16: predicting miners' optimal requests \mathbf{x}^* for each ESP;
- 17: **if** Eq.(43) \cap Eq.(44) **then**
- 18: $p_j^{(k)} \leftarrow p_j^{(k-1)} + \delta$;
- 19: **else if** Eq.(45) \cap Eq.(46) **then**
- 20: $p_j^{(k)} \leftarrow p_j^{(k-1)} - \delta$;
- 21: **Stage I: CSP Level Game**
- 22: Fix (P^*, X^*) .
- 23: obtain the optimal pricing strategy of $P1$ using Eq. (41).
- 24: **if** $\|P^{(k)} - P^{(k-1)}\| < \epsilon$ **then**
- 25: **return** $p^{csp*} P^{(k)}$ and \mathbf{x}^* ;
- 26: **else**
- 27: $k \leftarrow k + 1$;

3.4 Algorithm for Stackelberg Game and Analysis

Based on the analysis above, we adopt the backward induction method to achieve the Nash Equilibrium of the Stackelberg game. Backward induction is a method to solve the equilibrium of dynamic games. The dynamic game means that there is a sequence of actions of the player in the game, and the player in the latter actions can observe the previous actions [16]. The backward induction method first calculates the last stage of the dynamic game and then goes back to solve the equilibrium result. The rationale behind the backward induction method to solve the equilibrium of the Stackelberg game is explained below. The CSP acts first in the Stackelberg game and chooses the resource service price in stage I. The CSP will inevitably consider the behaviors of ESPs and miners in the next two stages. Each participant will be constrained by other participants in the course of action. Therefore, we construct a strategy adjustment with an iterative process. After each participant takes action, other participants need to judge whether they need to adjust their strategies to obtain higher profits. The iteration will finish until all the participants have no motivation to change their strategies.

In the three-stage Stackelberg game, there is at least one NE at each stage: the computing demand X^* by miners in stage III, the unit price of computing services P^* decided by ESPs in stage II, and the unit price of computing management services p^{csp*} determined by the CSP in stage I. We take advantage of a classic distributed algorithm (Algorithm 1) called Asynchronous Best Response to find the Nash equilibrium point in ESPs' subgame, where ESP is engaged in a gradient ascent process to maximize its utility. For simplicity of following descriptions, we define the following notations:

$$U_i^{(k)}(p^{csp*}, X_i^{(k)}, P) > U_i^{(k)}(p^{csp*}, X_i^{(k-1)}, P) \quad (42)$$

$$U_j^{esp}(p_j^{(k-1)} + \delta, p^{csp*}, \mathbf{x}^*) > U_j^{esp}(p_j^{(k-1)}, p^{csp*}, \mathbf{x}^*) \quad (43)$$

$$U_j^{esp}(p_j^{(k-1)} + \delta, p^{csp*}, \mathbf{x}^*) > U_j^{esp}(p_j^{(k-1)} - \delta, p^{csp*}, \mathbf{x}^*) \quad (44)$$

$$U_j^{esp}(p_j^{(k-1)} - \delta, p^{csp*}, \mathbf{x}^*) > U_j^{esp}(p_j^{(k-1)}, p^{csp*}, \mathbf{x}^*) \quad (45)$$

$$U_j^{esp}(p_j^{(k-1)} - \delta, p^{csp*}, \mathbf{x}^*) > U_j^{esp}(p_j^{(k-1)} + \delta, p^{csp*}, \mathbf{x}^*) \quad (46)$$

In Algorithm 1, the inputs include the random feasible price strategy of CSP, ESPs, the stochastic miners' demands, and the threshold ϵ . First, every miner tries to predict the optimal requests of the other miners and then adjust his demand according to the prices of ESPs using the gradient ascent method. If the miner's profit under the adjusted request strategy is higher than before, this miner will take the adjusted demands as a new request strategy in the next round. Then, every ESP tries to increase or decrease the price with a small step δ , and then predicts the miners' optimal requests under these price strategies. Similarly, if the adjusted price brings more benefits than the original price, each ESP will choose the price with maximum utility as the new price strategy in the next round; else, the price strategy will not change. Next, CSP changes its pricing strategy according to the price of ESPs and Eq. (41). These operations are conducted iteratively until the difference between

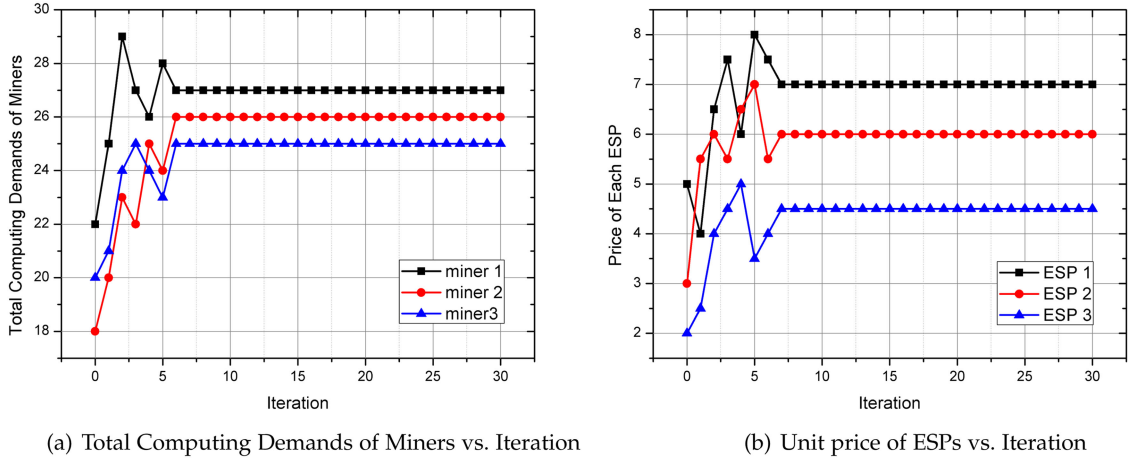


Fig. 3. Converge of the Stackelberg Game.

the previous and this round's Frobenius norms of the price strategy is less than a given threshold. The point now is the Nash equilibrium point that we are looking for. Finally, Algorithm 1 terminates and outputs the results.

Theorem 3. *Algorithm 1 achieves the approximate Nash Equilibrium, and it has a polynomial-time complexity.*

Proof. The computation overhead is dominated by stage III, i.e., miners level game from step 3 to step 11. Since stage III loops at most $|M * N|$ times, where M and N represent the number of miners and ESPs, respectively. Its computational complexity is denoted by $O(MN)$. Note that Algorithm 1 achieves the approximate Nash equilibrium of our proposed Stackelberg game. The approximate accuracy, measured by the Frobenius norms gap between the price strategy in two rounds, depends on the precision threshold ϵ . The convergence speed of the proposed algorithm also depends on the precision threshold ϵ . When ϵ is small, the number of iterations needed is large but the achieved results are more accurate. Conversely, when ϵ is large, the number of required iterations is small, but the achieved results are less accurate. Here, the number of loops for iteration is challenging to obtain explicit form through mathematical theoretical analysis. Still, we concluded that the algorithm could converge in polynomial time through amounts of simulations in Section 4. \square

4 PERFORMANCE EVALUATION

In this section, we verify the convergence of the proposed backward induction-based iterative algorithm of the Stackelberg game. We evaluate the performance of the proposed algorithm with extensive simulations. We conduct the simulations on a computer with Inter(R) Core(TM) i7-10700 CPU @2.90GHz 2.90GHz and 16.0GB RAM under a Windows platform. We first simulate the multi-leader multi-follower Stackelberg game between miners and ESPs, and further verify the practicality of our proposed utility function of miners. Then, numerical examples are provided to examine how miners figure out their optimal requests based on the prices of the ESPs, and how the ESPs optimize their unit price based on their available capability and the miners' budget. We assume that the parameter of propagation delay λ is fixed as $\frac{1}{600}$, as

introduced in the work [17]. In addition, when mentioning the prices set by the ESPs, no matter whether they are optimized or not, $p_j > c$, $p_j > p^{csp}$ and $\bar{t} \gg t_j$ always hold.

4.1 Convergence of the Iterative Algorithm

We first demonstrate the convergence of the Stackelberg game. For ease of illustration, we consider a simple example with three miners and three ESPs in this model. By referring to previous research, we set the propagation delay of each ESP following a normal distribution $\mathcal{N}(\mu_t, 0.1)$, where $\mu_t = 1ms$ is the average propagation delay of each ESP. In addition, we set the budget of each miner as 200.

When $N = 3$ and $M = 3$, Fig. 3 presents the convergence of the proposed backward induction-based iterative algorithm (i.e., Algorithm 1) for the Stackelberg game. We see that the total computing demands of each miner and the price of each ESP tend to be stable when the iteration number is larger than 7. This result also shows that the proposed backward induction-based iterative algorithm can achieve the Nash Equilibrium of the Stackelberg game in polynomial time. In addition, Fig. 4 illustrates the relation between the precision threshold ϵ versus the number of iteration rounds for convergence when the number of miners varies. That is, the iteration rounds are exponentially related to the threshold ϵ . Also, with the increase of the number of miners, the iteration rounds for convergence go up as well.

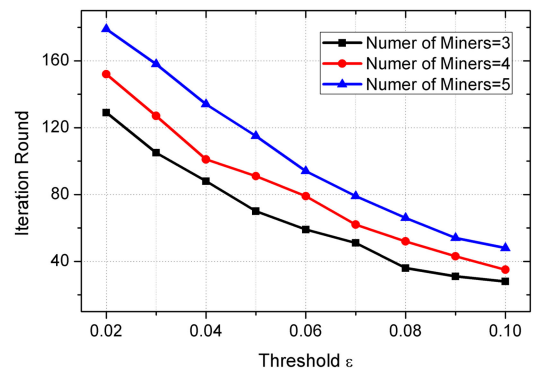
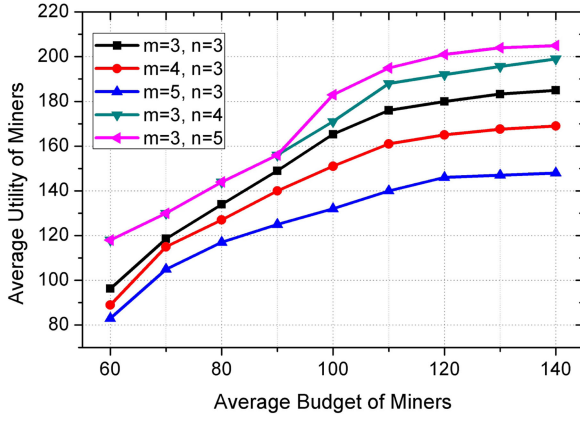
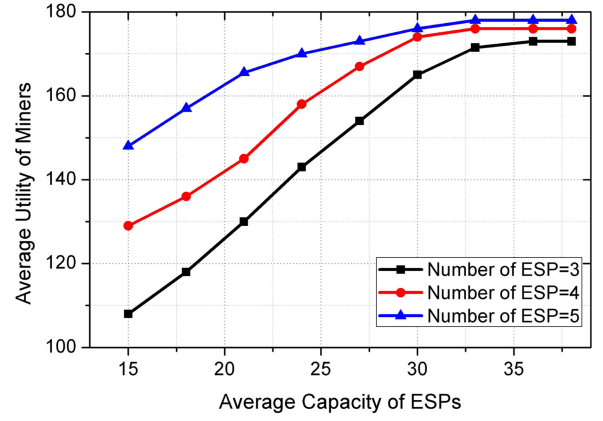


Fig. 4. Total Iteration Rounds vs. Threshold.

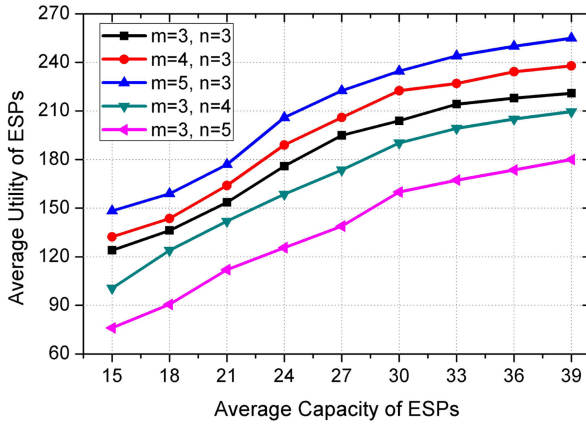


(a) Miners' Utility vs. Budget.

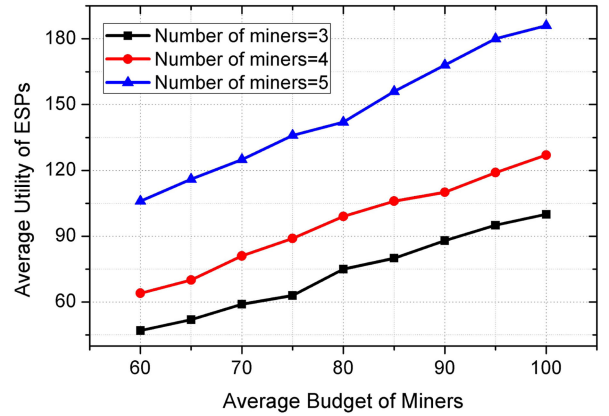


(b) Miners' Utility vs. ESP Capacity.

Fig. 5. The Effect of Budget and Capacity on Miner side.



(a) ESP Utility vs. Capacity.



(b) ESP Utility vs. Miners' Budget.

Fig. 6. The Effect of Budget and Capacity on ESP side.

4.2 The Effect of Different Parameters on Miner's Utility

We study miner utility as a function of miner budget and ESP capacity, as shown in Figs. 5a and 5b. Fig. 5a shows that when other miners' budgets are fixed at a certain value, increasing a miner's budget will bring a higher utility for miner. Obviously, increasing the miner's budget allows him to purchase more computing resources, which increases the probability of solving the PoW problem, thereby further enhancing his profits. On the contrary, as the number of ESPs is fixed, the utility of miners decreases as the number of miners increases. This is because more competition among more miners will result in a lower probability of each competitor, hence reducing the utility of each miner. Fig. 5a demonstrates that if the number of ESPs increases, each miner will get more profits. The reason is that more ESPs bring higher total capacity, which will allow more demands to be accepted by ESPs, resulting in a higher probability of winning the game due to the shorter propagation delay of ESPs than CSPs. Nevertheless, when the certain miner's budget is set as 90, there is no effect on the growth of the miner's benefit when changing the number of ESPs from 4 to 5. This is due to the fact that when the miner's budget is under a certain amount, simply increasing the number of ESPs does not help miners buy more computing services due to limited

budgets. As a result, the utility of one miner constantly holds until the budget grows up to a higher value.

Fig. 5b shows the impact of ESPs' capacity on miners' utility. The average of miners' utility increases as the average capacity of ESPs increases at first. When the average capacity of ESPs reaches a certain value, the changes in miners' utility tend to be flat. Miners specify their demand strategy based on their budget and the prediction of other competitors' anticipation. More capacity of ESPs facilitates the acceptance of miners' requests. Therefore, within a limited budget, increasing ESPs' capacity will bring a certain benefit growth. However, due to the limited budget of miners, they cannot pay for too many computing services, so the continuous growth of ESPs' capacity brings no benefit in their utility when their requests have reached the upper limit of their budgets.

4.3 The Effect of Different Parameters on ESP's Utility

We also address the comparison of how capacity affects the utility of ESPs in different numbers of miners or different numbers of ESPs cases. As illustrated in Fig. 6a, we observe that the profit of an ESP rises with the increase of its own capacity. The reason inside is that more capacity allows the ESP to accept more requests and reduces the number of computing tasks uploaded to the CSP. Here, the cost for

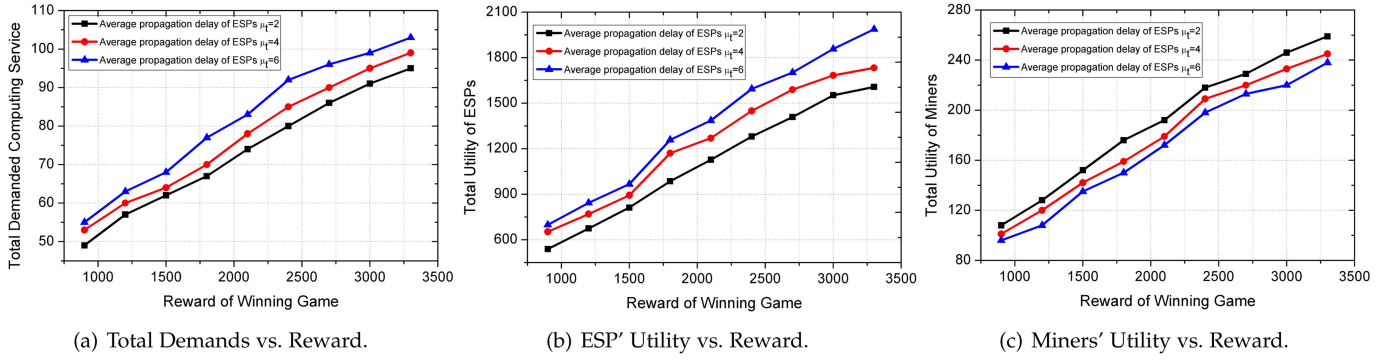


Fig. 7. The Effects of the Reward.

uploading requests to the CSP is higher than running the ESP tasks locally. Thus, more tasks implemented on ESP enhance the profit of ESP. We also see that when the number of ESPs is set as 3, the utility of ESP increases as the number of miners increases. On the contrary, when the number of miners is set as 3 and the number of ESPs grows up, the revenue of ESPs drops instead. This is because the greater number of miners will cause more computing service demands, and at the same time, the competition will be more intense. The miners are willing to request more computing resources to upgrade their winning probability in such settings.

However, with the fixed number of miners, the budget of miners is limited. We set the budget of each miner as $B_i = 100$ in this experiment, which means the total requests are limited under their budget constraint. Thus, more ESPs share the limited benefits, resulting in a decrease in the average profit of each ESP. Fig. 6b shows that the ESPs' utility rises when the average budget of miners B_i varies from 60 to 120. This is because the miners have more money to purchase the computing service, which further increases the utility of ESPs.

4.4 System Performance Evaluation

The system performance for miners and providers under the proposed algorithm is evaluated as illustrated in Figs. 7a, 7b, and 7c. We evaluate the utility of miners and ESPs as well as the computing service demanded by miners with the changes of the reward parameters. We can see that, with the increase of the reward, all values of total demanded computing service, ESPs' utility, and miners' utility increase. The reason lies in that the increasing reward motivates the miners to demand more computing services to win the mining game and hence increases the total demand for ESPs, which further improves

the profit of ESPs. Also, the increasing reward gives the miner higher profit and then improves the miners' utility.

In addition, we evaluate the effect of the average propagation delay of the system on participant profits. We can see that the average propagation delay of the system tends to be positively correlated with the computational service demand and utility of the ESPs in Figs. 7b and 7c. This can be explained in Eq. (9), that is, the probability that a miner wins the mining game W_i consists of two parts. One factor is the probability that the block he packs becomes the first consensus, which is negatively related to propagation delay. An increase in the average propagation delay results in a decrease in W_i . Therefore, miners must demand more computing power to compete with others. This, in turn, will increase ESP profits. However, the requirement for more computing services makes miners pay more, which reduces utility. This is why the utility of miners decreases as the average propagation delay of ESP increases.

Fig. 8 also studies the propagation delay for geographic location and number of miners two factors that affect the price and utility of ESPs. We can see that, ESP with higher propagation delay factor t will obtain a lower price and utility. This is because users are more inclined to buy ESPs with low latency to get a higher winning rate, so the platform has to lower the price to attract more users to purchase, thus obtaining less profit from providing computing services. In fact, by comparing Figs. 7b and 8, we can find that reducing the delay of a single ESP will bring an increase in its profit, but the effect will be counterproductive if all ESPs of the entire system reduce the delay. This phenomenon can be explained by the principle of the prisoner's dilemma in game theory. What's more, from the general trend, the increase in the number of miners leads

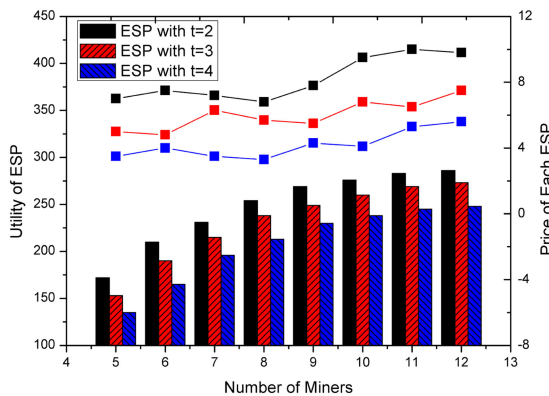


Fig. 8. The Effect of the Propagation Delay.

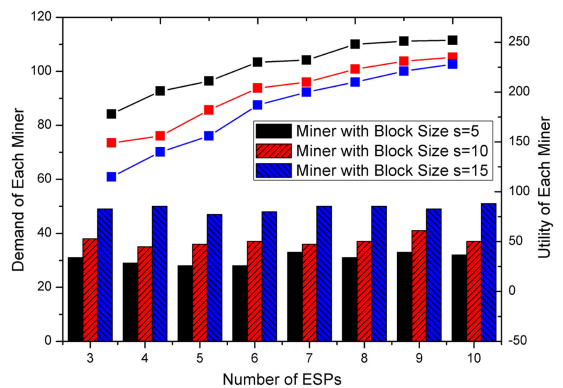


Fig. 9. The Effect of the Block Size.

to the improvement in ESPs' prices and benefits. The reason is that increase in the number of buyers leads to more intense competition and makes the platform a better sales situation, which we have discussed before.

In Fig. 9, we explore the relation between the number of ESPs and the miners' demand or utility. The increase in the number of ESPs doesn't bring obvious change to users' demand. The user's request is constrained by his budget, so he has no incentive to increase demand unconditionally. However, the increase in the number of ESPs will bring more fierce competition in the seller's market, thus leading to lower prices, which enhances the profit of miners. Furthermore, a miner who packages more transactions will spend more time in the propagation phase, so he tends to request more computing services to improve his winning probability. Thus, the higher demands will decrease the utility of the miner.

5 RELATED WORK

5.1 Widespread Applications of Blockchain

Blockchain came to prominence as the distributed ledger underneath Bitcoin. Recently, several studies on blockchain work [7], [18], [19] have applied the blockchain technology into the Internet of Things (IoT) field. [18] proposed secure support vector machine (SVM), which is a privacy-preserving SVM training scheme over blockchain-based encrypted IoT data and can enable data sharing while securing sensitive user information. A general architecture combining blockchain and IoT systems is presented by [19] to support the decentralized approach of data management in IoT systems. Furthermore, game theory [20] is usually combined with blockchain to be applied in the field of IoT to prevent malicious competition. [7] proposed a two-stage solution to ensure secure miner selection and prevent internal collusion among active miners, which further improve driving safety and enhance vehicular services. In addition to the successful applications in the IoT, blockchain has been treated as one of the most promising technologies to promote crowdsourcing by providing new nice features. The work [21] proposed a novel hybrid blockchain crowdsourcing platform to achieve the decentralization and privacy preservation. The reinforcement learning technology integrated into blockchain in crowdsourcing system was studied by [22], where an improved multi-blockchain structure and a blockchain-based hierarchical task management method were designed. Moreover, blockchain's achievements in privacy protection should not be underestimated either. The work [23], [24] used the blockchain technology in a decentralized way to identify and verify the users' information so that it can solve the security problem.

On the other hand, a large number of studies have been developed in mining schemes management for blockchain networks [25], [26]. In [17] the authors designed a noncooperative game among the miners. The miner's strategy is to choose the number of transactions to be included in a block, where solving the PoW puzzle for mining is modeled as a Poisson process. Then, [27] modeled the mining process as a sequential game where the miners compete for mining reward sequentially among them. Similarly, the authors in [28] formulated the stochastic game for

modeling the mining process, where miners decide on which blocks to extend and whether to propagate the mined block. In addition, [29] investigated the impact of network latency on blockchain forking behavior and possible violations of the aforementioned six confirmations convention for transaction approval.

5.2 Cloud/Edge Computing Based Blockchain

Cloud providers offer virtually unlimited computation and storage resources on demand, allowing for the elasticity and scalability of applications deployed. The mobile edge computing, which is emerging as an effective way to mitigate the problem of long latency and the current network architecture [30], [31], [32], has attracted more attention. With the development of cloud computing and edge computing, miners prefer offloading the PoW computations to local edge service due to the limited computing resource on their mobile terminals [15], [33], [34], [35], [36], [37]. The system model proposed in paper [38] allows unmanned aerial vehicles offloading to MEC through base stations to run some blockchain tasks. In [35], [39], the authors considered a blockchain-based mining game model with an ESP and a CSP in two situations, i.e., the ESP is connected (to the CSP) or standalone, and then analyzed the Stackelberg equilibrium in these models. However, in real scenarios, it is common for multiple ESPs to compete for pricing and sell resources which is not considered in this work. [14] studied the interactions among the cloud/edge providers and miners in blockchain using game theory approach, in which every miner has mobile device with limited computing power and can offload the PoW puzzle to the providers. The authors considered the propagation delay caused by the size of the block, i.e., the propagation delay is related to the number of transactions included in that block.

However, all previous studies did not consider the impact of ESPs' propagation delays due to their geographic locations, which will lead to the different probability of winning the mining game for miners. Therefore, this motivates us to take a step further to reconsider the mining strategies and resource management in a mobile environment. In fact, lots of real-world applications would benefit from our research. For example, our model can be applied in the task migration problem in mobile edge computing scenarios, the data synchronization problem in blockchain-based internet of vehicles systems, etc.

6 CONCLUSION & FUTURE WORK

In this paper, we investigate the resource pricing and scheduling problem in the edge-assisted blockchain mining networks by using the three-stage multi-leader multi-follower Stackelberg game theory. In particular, we first propose cooperative mobile-edge computing (MEC)-aided blockchain network. In the network, the user with devices can offload computation-intensive PoW mining tasks to edge computing service providers (ESPs) who connect with a remote cloud computing service provider (CSP). ESPs in this model are located in diverse positions of the blockchain network, which causes different propagation delays in mining work. If the computing demands from users exceed the maximum capacity of ESP, it will lease computing resources from CSP. Then, we study the joint computation offloading and resource service pricing

problem as a three-stage Stackelberg game. We analyze the subgame optimization problem in each stage and propose an iterative algorithm based on backward induction to achieve the Nash equilibrium of the Stackelberg game. Furthermore, we conduct extensive simulations to validate the convergence as well as evaluate the network performance. In future work, we will further explore the best response results without others' strategies and the practical algorithm implementation in the real world. We will also study other consensus mechanisms such as proof of stake, delegated proof of share, etc., in the next blockchain-based research. Note that the extended research for studying the task resource scheduling problem on the real blockchain system (e.g., Ethereum) is being carried out in our follow-up work.

REFERENCES

- [1] S. Huang, H. Huang, G. Gao, Y. Sun, Y. Du, and J. Wu, "Stackelberg game based resource pricing and scheduling in edge-assisted blockchain networks," in *Proc. 18th Int. Conf. Mobile Ad Hoc Smart Syst.*, 2021, pp. 289–296.
- [2] S. Nakamoto, "A peer-to-peer electronic cash system," 2008. [Online]. Available: <https://bitcoin.org/bitcoin.pdf>
- [3] P. Franco, *Understanding Bitcoin*. Hoboken, NJ, USA: Wiley, 2014.
- [4] R. Böhme, N. Christin, B. Edelman, and T. Moore, "Bitcoin: Economics, technology, and governance," *J. Econ. Perspectives*, vol. 29, no. 2, pp. 213–38, 2015.
- [5] Crypto-currency market capitalizations, 2017. [Online]. Available: <https://coinmarketcap.com>
- [6] W. Wang et al., "A survey on consensus mechanisms and mining strategy management in blockchain networks," *IEEE Access*, vol. 7, pp. 22328–22370, 2019.
- [7] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2906–2920, Mar. 2019.
- [8] S. Feng, W. Wang, Z. Xiong, D. Niyato, P. Wang, and S. S. Wang, "On cyber risk management of blockchain networks: A game theoretic approach," *IEEE Trans. Serv. Comput.*, vol. 14, no. 5, pp. 1492–1504, Sep./Oct. 2021.
- [9] X. Han, Y. Yuan, and F. Wang, "A fair blockchain based on proof of credit," *IEEE Trans. Computat. Social Syst.*, vol. 6, no. 5, pp. 922–931, Oct. 2019.
- [10] N. Bozic, G. Pujolle, and S. Secci, "A tutorial on blockchain and applications to secure network control-planes," in *Proc. 3rd Smart Cloud Netw. Syst.*, 2016, pp. 1–8.
- [11] J. Feng, F. Richard Yu, Q. Pei, X. Chu, J. Du, and L. Zhu, "Cooperative computation offloading and resource allocation for blockchain-enabled mobile-edge computing: A deep reinforcement learning approach," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6214–6228, Jul. 2020.
- [12] L. Wan, D. Evers, and H. Zhang, "Evaluating the impact of network latency on the safety of blockchain transactions," in *Proc. IEEE Int. Conf. Blockchain*, 2019, pp. 194–201.
- [13] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, "When mobile blockchain meets edge computing," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 33–39, Aug. 2018.
- [14] Z. Xiong, J. Kang, D. Niyato, P. Wang, and H. V. Poor, "Cloud/edge computing service management in blockchain networks: Multi-leader multi-follower game-based ADMM for pricing," *IEEE Trans. Serv. Comput.*, vol. 13, no. 2, pp. 356–367, Mar./Apr. 2020.
- [15] Z. Xiong, S. Feng, W. Wang, D. Niyato, P. Wang, and Z. Han, "Cloud/fog computing resource management and pricing for blockchain networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4585–4600, Jun. 2019.
- [16] Y. Zuo, S. Jin, S. Zhang, and Y. Zhang, "Blockchain storage and computation offloading for cooperative mobile-edge computing," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 9084–9098, Jun. 2021.
- [17] N. Houy, "The bitcoin mining game," *Ledger*, vol. 1, pp. 53–68, 2016.
- [18] M. Shen, X. Tang, L. Zhu, X. Du, and M. Guizani, "Privacy-preserving support vector machine training over blockchain-based encrypted IoT data in smart cities," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7702–7712, Oct. 2019.
- [19] Z. Xiong, Y. Zhang, N. C. Luong, D. Niyato, P. Wang, and N. Guizani, "The best of both worlds: A general architecture for data management in blockchain-enabled Internet-of-Things," *IEEE Netw.*, vol. 34, no. 1, pp. 166–173, Jan./Feb. 2020.
- [20] G. Gao, M. Xiao, J. Wu, L. Huang, and C. Hu, "Truthful incentive mechanism for nondeterministic crowdsensing with vehicles," *IEEE Trans. Mobile Comput.*, vol. 17, no. 12, pp. 2982–2997, Dec. 2018.
- [21] S. Zhu, H. Hu, Y. Li, and W. Li, "Hybrid blockchain design for privacy preserving crowdsourcing platform," in *IEEE Int. Conf. Blockchain*, 2019, pp. 26–33.
- [22] H. Lin, S. Garg, J. Hu, G. Kaddoum, M. Peng, and M. S. Hossain, "Blockchain and deep reinforcement learning empowered spatial crowdsourcing in software-defined internet of vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 6, pp. 3755–3764, Jun. 2021.
- [23] B. K. Mohanta, S. S. Panda, U. Satapathy, D. Jena, and D. Gountia, "Trustworthy management in decentralized IoT application using blockchain," in *Proc. 10th Int. Conf. Comput. Commun. Netw. Technol.*, 2019, pp. 1–5.
- [24] D. Fakhri and K. Mutijarsa, "Secure IoT communication using blockchain technology," in *Proc. Int. Symp. Electron. Smart Devices*, 2018, pp. 1–6.
- [25] P. De Giovanni, "Blockchain and smart contracts in supply chain management: A game theoretic model," *Int. J. Prod. Econ.*, vol. 228, 2020, Art. no. 107855.
- [26] E. Altman et al., "Blockchain competition between miners: A game theoretic perspective," *Front. Blockchain*, vol. 2, 2020, Art. no. 26.
- [27] J. Beccuti and C. Jaag, "The bitcoin mining game: On the optimality of honesty in proof-of-work consensus mechanism," *Swiss Econ. Work. Paper 0060*, pp. 1–20, 2017.
- [28] A. Kiayias, E. Koutsoupias, M. Kyropoulou, and Y. Tselekounis, "Blockchain mining games," in *Proc. ACM Conf. Econ. Comput.*, 2016, pp. 365–382.
- [29] L. Wan, D. Evers, and H. Zhang, "Evaluating the impact of network latency on the safety of blockchain transactions," in *Proc. IEEE Int. Conf. Blockchain*, 2019, pp. 194–201.
- [30] G. Gao, M. Xiao, J. Wu, H. Huang, S. Wang, and G. Chen, "Auction-based vm allocation for deadline-sensitive tasks in distributed edge cloud," *IEEE Trans. Serv. Comput.*, vol. 14, no. 6, pp. 1702–1716, Nov./Dec. 2021.
- [31] C.-F. Liu, M. Bennis, M. Debbah, and H. V. Poor, "Dynamic task offloading and resource allocation for ultra-reliable low-latency edge computing," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4132–4150, Jun. 2019.
- [32] H. Zhou, X. Chen, S. He, J. Chen, and J. Wu, "DRAIM: A novel delay-constraint and reverse auction-based incentive mechanism for wifi offloading," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 4, pp. 711–722, Apr. 2020.
- [33] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Privacy-preserved task offloading in mobile blockchain with deep reinforcement learning," *IEEE Trans. Netw. Service Manag.*, vol. 17, no. 4, pp. 2536–2549, Dec. 2020.
- [34] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Joint computation offloading and content caching for wireless blockchain networks," in *Proc. IEEE Conf. Comput. Commun. Workshops*, 2018, pp. 517–522.
- [35] S. Jiang, X. Li, and J. Wu, "Hierarchical edge-cloud computing for mobile blockchain mining game," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst.*, 2019, pp. 1327–1336.
- [36] S. Guo, Y. Dai, S. Guo, X. Qiu, and F. Qi, "Blockchain meets edge computing: Stackelberg game and double auction based task offloading for mobile blockchain," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5549–5561, May 2020.
- [37] M. Xiao, J. Wu, H. Huang, L. Huang, and C. Hu, "Deadline-sensitive user recruitment for probabilistically collaborative mobile crowdsensing," in *Proc. IEEE 36th Int. Conf. Distrib. Comput. Syst.*, 2016, pp. 721–722.
- [38] A. Asheralieva and D. Niyato, "Distributed dynamic resource management and pricing in the IoT systems with blockchain-as-a-service and UAV-enabled mobile edge computing," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 1974–1993, Mar. 2020.
- [39] S. Jiang, X. Li, and J. Wu, "Multi-leader multi-follower stackelberg game in mobile blockchain mining," *IEEE Trans. Mobile Comput.*, vol. 21, no. 6, pp. 2058–2071, Jun. 2022.



Sijie Huang received the BS degree from the School of Computer Science and Technology, Xidian University, Xi'an, China, in 2018. She is currently working toward the MS degree in electronic information with the School of Computer Science and Technology, Soochow University, Soochow, China. Her research interests include blockchain, mobile-edge computing, crowdsourcing and incentive mechanism.



Yang Du (Member, IEEE) received the BE degree from the Soochow University in 2015 and the PhD degree from the University of Science and Technology of China in 2020. He is currently a postdoctoral fellow with Soochow University. His research interests include network traffic measurement, mobile crowdsensing, and truth discovery.



He Huang (Senior Member, IEEE) received the PhD degree from the School of Computer Science and Technology, University of Science and Technology of China (USTC), China, in 2011. He is currently a professor with the School of Computer Science and Technology, Soochow University, China. From 2019 to 2020, he was a visiting research scholar with Florida University, Gainesville, USA. He has authored more than 100 papers in related international conference proceedings and journals. His current research inter-

ests include traffic measurement, computer networks, and algorithmic game theory. He is a member of the Association for Computing Machinery (ACM). He received the best paper awards from Bigcom 2016, IEEE MSN 2018, and Bigcom 2018. He has served as the technical program committee member of several conferences, including IEEE INFOCOM, IEEE MASS, IEEE ICC, and IEEE Globecom.



Guoju Gao (Member, IEEE) received the PhD degree in computer science and technology with the School of Computer Science and Technology, University of Science and Technology of China, Hefei, China. He is an Assistant Professor with the School of Computer Science and Technology at the Soochow University. He visited the Temple University, USA, from Jan. 2019 to Jan. 2020. His research interests include privacy preservation, game theory, and reinforcement learning.



Yu-E Sun (Member, IEEE) received the PhD degree from the Shenyang Institute of Computing Technology from Chinese Academy of Science. She is a professor with the School of Rail Transportation, Soochow University, China. From 2019 to 2020, she was a postdoc research scholar with Florida University, Gainesville, USA. She has authored more than 80 papers in related international conference proceedings and journals. Her current research interests span traffic measurement, privacy preserving, algorithm design and

analysis for wireless networks. She is a member of ACM. She received the best paper awards from Bigcom 2016, IEEE MSN 2018, and Bigcom 2018. She has served as the technical program committee co-chairs of ACM MSCC-MobiHoc 2016.



Jie Wu (Fellow, IEEE) is the director with the Center for Networked Computing and Laura H. Carnell professor with Temple University. He also serves as the director with International Affairs, College of Science and Technology. He served as chair with the Department of Computer and Information Sciences from the summer of 2009 to the summer of 2016 and associate vice provost for International Affairs from the fall of 2015 to the summer of 2017. Prior to joining Temple University, he was a program director with the National

Science Foundation and was a distinguished professor with Florida Atlantic University. His current research interests include mobile computing and wireless networks, routing protocols, cloud and green computing, network trust and security, and social network applications. He regularly publishes in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including *IEEE Transactions on Service Computing* and the *Journal of Parallel and Distributed Computing*. He was general co-chair for IEEE MASS 2006, IEEE IPDPS 2008, IEEE ICDCS 2013, ACM MobiHoc 2014, ICPP 2016, and IEEE CNS 2016, as well as program co-chair for IEEE INFOCOM 2011 and CCF CNCC 2013. He was an IEEE Computer Society distinguished visitor, ACM distinguished speaker, and chair for the IEEE Technical Committee on Distributed Processing (TCDP). He is a CCF distinguished speaker. He is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award.

► For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.