Distributed Resource Management for Blockchain in Fog-Enabled IoT Networks

Lichao Yang, Ming Li, *Associate Member, IEEE*, Heli Zhang, *Member, IEEE*, Hong Ji, *Senior Member, IEEE*, Mingyan Xiao, *Graduate Student Member, IEEE*, and Xi Li, *Member, IEEE*

***Abstract*—Blockchain, an emerging decentralized but trusted system, has been applied in many applications, such as the Internet of Things (IoT), supply chains, and smart grid. However, due to the large amount of computing and storage resources blockchain typically demands, its wide deployment is faced with the sustainability issue. To resolve this issue, a viable solution is to empower the IoT system with fog computing that can offload the computation-demanding tasks. Due to varieties of mining tasks and heterogeneous resource capabilities at fog nodes (FNs), it is not an easy task to schedule mining tasks and manages resource allocation among FNs of conflicting interests and independent IoT devices in a distributed manner. In this article, under the framework of matching theory, we design a distributed matching mechanism to maximize the social welfare of resource-restricted FNs while guaranteeing various mining requirements of FNs. Besides, we also provide formal proof regarding the convergence and computational complexity of a distributed matching algorithm (DMA). Finally, we verify that DMA not only improves the social welfare of FNs but also reduces the mining latency compared with the existing algorithms through extensive simulations.**

***Index Terms*—Blockchain, fog computing, Internet of Things (IoT), matching.**

I. Introduction

A. Background

With the explosive growth of decentralized cryptocurrencies, its key enabler, blockchain, is attracting more and more attention. Blockchain is a peer-to-peer (P2P) distributed ledger technology to build trust between multiple entities without relying on any third parties. Due to this nice property, blockchain has been applied to many applications, such as the Internet of Things (IoT) [1], [2], supply chain [3], [4], and smartgrid [5]. In IoT networks, IoT devices produce a large chunk of data in a distributed manner [6]. Due to the lack of a centralized trusted third party, how to ensure the authenticity and integrity of these data becomes critical for many IoT systems. Thanks to the consensus protocol, blockchain thus serves as an ideal candidate to address these issues. The consensus protocol is deemed as the basis of blockchain technique [7]. The protocol consists of multiple stages. First, miners use their computation power to solve a Proof-of-Work (PoW) puzzle of mining tasks. Once it is done, the miner broadcasts the mined block so that other miners can verify the block and reach a consensus. By then, the block is successfully loaded onto the existing blockchain. The winning miner obtains a reward.

Manuscript received May 1, 2020; revised July 23, 2020 and August 18, 2020; accepted September 19, 2020. Date of publication October 1, 2020; date of current version February 4, 2021. This work was supported in part by National Natural Science Foundation of China under Grant 61671088; in part by the National Natural Science Foundation of China under Grant 61771070; and in part by the BUPT Excellent Ph.D. Students Foundation under Grant CX2018203. *(Corresponding author: Heli Zhang.)*

Lichao Yang, Heli Zhang, Hong Ji, and Xi Li are with the Key Laboratory of Universal Wireless Communications, Ministry of Education, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: lichaoyang@bupt.edu.cn; zhangheli@bupt.edu.cn; jihong@bupt.edu.cn; lixi@bupt.edu.cn).

Ming Li and Mingyan Xiao are with the Department of Computer Science and Engineering, University of Texas at Arlington, Arlington, TX 76019 USA (e-mail: ming.li@uta.edu; mingyan.xiao@mavs.uta.edu).

Digital Object Identifier 10.1109/JIOT.2020.3028071

B. Motivation

The implementation of blockchain is faced with a critical challenge. It is resource consuming for IoT devices to execute mining and consensus protocols. Following some prior works [8]–[10], we propose to incorporate edge computing into IoT networks to support blockchain. Here, the system is composed of a large amount of IoT devices, fog nodes (FNs) with edge computing capabilities, and one primary FN (PFN) with storage capability [11], [12]. FNs are equipped with different edge computing capabilities that connect to IoT devices via wireless links [13]–[15]. Additionally, we assume that FNs can join and leave the network anytime they like. The reason is while FNs can join and leave the network at the time unit of a minute. In another word, there is no frequent update of FN deployment. To accommodate the system dynamics, we can rerun the proposed algorithm whenever an FN joins or leaves.

Blockchain applications on IoT devices first release mining tasks to PFN, which is in charge of collecting tasks from IoT devices and forwarding them to FNs. FNs equipped with different edge computing serve as miners. FNs collect and verify tasks and integrate them into blocks by solving the PoW puzzle. The newly generated block is then immediately broadcast to the rest FNs to achieve consensus.

One critical issue of running blockchain on this fog-enabled IoT network is resource management and scheduling among FNs. There have been some prior works to tackle these issues, using, for example, game theory and auction theory. However, game-theoretic approaches need to know the actions of the other players, which is not always practical in a blockchain system. Auction theory addresses how to elicit miner's private truthful cost so as to maximize the platform's utility. However, this issue is not the focus of this work. Recently, matching theory has attracted great attention for its capacity of achieving *stability* in a two-sided market without the need for a centralized controller.

In this work, we study the problem of mining task allocation among FNs in an edge-computing assisted blockchain system. Thus, no centralized controller exists. We first formulate a social welfare maximization problem. Then, a matching theory-based distributed mechanism is proposed to transform the original optimization problem into a two-sided matching game with one-sided preference. Then, a distributed matching algorithm (DMA) is designed to find a suboptimal solution. Analysis and simulations manifest that the proposed DMA is a flexible and efficient scheme compared with the existing solutions, such as alternating direction method of multipliers (ADMMs)-based algorithm [16] and gradient iterative algorithm (GIA) [17].

C. Our Contribution

Our main contributions of this article are summarized as follows.

1) *We Formulate a Social Welfare Maximization Problem:* We explore the implementation of a PoW-based blockchain in fog-enabled IoT networks, allowing to offload mining tasks of solving the PoW puzzle to FNs. The optimization problem takes into account various constraints on resource availability and mining requirements at FNs.
2) *We Design a Distributed Matching Mechanism:* By analyzing the behaviors of all players in a blockchain system, we then transform the social welfare maximization problem as a two-sided matching game with one-sided preference.
3) *We Derive a Stable Matching Result With a Suboptimal Solution:* Based on the stability property of matching theory, a DMA is designed to achieve a suboptimal solution. The stability and computational efficiency of DMA are theoretically analyzed. Simulations show the effectiveness of the proposed algorithm compared with two other existing algorithms.

D. Organization and Notation

The remainder of this article is organized as follows. We begin with related work in Section II. In Section III, we briefly introduce the blockchain system model in the fog-enabled IoT network. We then formulate a social welfare maximization problem in Section Iv. With the help of matching theory, we reformulate the original problem into a two-sided matching game with one-sided preference and design a distributed matching mechanism integrated with blockchain technology in Section V. In Section VI, we get a stable matching result and do the computational complexity analysis of DMA based on edge computing. Simulations are conducted in Section VII compared with other existing algorithms in a blockchain system. Finally, we conclude this article in Section VIII.

II. Related Work

Recently, there have been more and more studies on resource management in edge computing-enabled IoT networks [18]–[20]. Especially, we review the merits and demerits of the existing offloading strategies for IoT applications in fog-enabled IoT networks. Wang *et al.* [18] investigated a partial computation offloading strategy between smart mobile devices and edge clouds by jointly optimizing the overall energy consumption of the edge cloud servers. Bibani *et al.* [19] designed a hybrid platform as a service for IoT applications in fog and edge cloud environments to guarantee low latency. Qiu *et al.* [20] proposed agent mining and edge cloud mining approaches to solve complicated puzzles and propagate blocks. Then, a joint optimization problem of access selection of users, computing resources allocation, and networking resources allocation is formulated in the blockchain-enabled IoT. It can be observed cloud, edge, and fog computing are wide to solve wireless resource allocation. Inspired by the existing works, to solve the limitations of computing resources in IoT devices and reduce the transmission delay in the blockchain network, we also adopt fog-enabled network architecture integrated with blockchain technology in this article.

Then, although there has been a recent surge in works that propose to use auction theory and game theory to optimize resource management integrated with blockchain technology, the matching theory is seldom adopted to solve wireless resource allocation in fog-enabled IoT networks. For example, Wu and Ansari [21] proposed the concept of FN clusters, then the blockchain was customized for FN clusters to reduce the required computing power consumption and storage spaces. Jiao *et al.* [22] focused on the trading between the fog service providers and miners, and proposed an auction-based market model to allocate the computing resource with the aim of maximizing social welfare in the blockchain system. Xiong *et al.* [17] studied the interaction between the fog providers and miners in a Pow-based blockchain network using a game-theoretic approach to realize the decentralized resource management. Sun *et al.* [23] jointly considered incentives and cross-server resource allocation in blockchain-driven MEC by virtual of the double auction theory.

Nevertheless, these works mainly focused on the mining strategies, ignoring the effect of varieties of transactions and the conflicting interests among heterogeneous FNs. As a branch of the game theory, because of its distribution nature, individuality, stability, and self-organization of the matching theory, it has been widely used to deal with resource management in various areas, such as D2D communication [24], [25], spectrum allocation [26], [27], and edge computing [28], [29]. Yuan *et al.* [24] aimed to maximize the throughput of D2D pairs while suppressing the interference to cellular links and proposed a novel local-global channel state information distributed channel-power allocation scheme according to the many-to-one matching theory. Wang *et al.* [25] employed social networking and D2D communication in the IoT to achieve content sharing among smart devices via multihop cooperative D2D communications, aiming at maximizing the overall success rate by matching theory. Sanguanpuak *et al.* [26] studied the nonorthogonal spectrum assignment with the goal of maximizing the expected weighted sum rate of the cooperators, by adopting the many-to-one stable matching framework. Chen *et al.* [27] traded the service providers with spare spectrum with who were in urgent demand of additional spectrum to present a spectrum under both maximum quota and minimum requirements. Jia *et al.* [28] investigated the computing resource allocation problem as a double two-sided matching problem in fog computing networks based on cost efficiency. Liu *et al.* [29] studied the task offloading problem from a many-to-one matching perspective and aimed to optimize the total network delay by processing the massive workload tasks in the proximity of vehicles.

In conclusion, different from the existing works, it is significant to consider maximizing the social welfare of FNs under the consideration of varieties of mining tasks and the resource-restricted FNs with heterogeneous edge computing capabilities in a blockchain system. Then, powered by matching theory it is the first adopted to design a distributed matching mechanism integrated with blockchain technology, in the wireless resource allocation problem and one suboptimal solution is achieved. Finally, we conduct an extensive analysis to evaluate the stability and computational complexity of DMA theoretically.

III. System Model

A. Network Model

We consider a fog-enabled IoT network with K IoT devices, one PFN with storage capability at the edge of radio access networks that can store various transactions. Additionally, considering that the mining demands of the wide varieties of IoT devices are in diversities, we deploy a set of M FNs with different edge computing capabilities, denoted as $\mathcal{M} = \{1, \ldots, M\}$ at the edge of the radio access network. Next, since each IoT device has a mobile blockchain application to record and verify transactions or data on its device, however, due to the computing resource limitation on IoT devices, mining tasks (transactions) are first offloaded to the nearby PFN (mempool). Then, the mining tasks wait for arrangement within the rounded FNs (miners) and mempool through a matching mechanism to mine a new block. After a new block is mined, it is instantaneously propagated across the network for verification to complete the consensus process. If the consensus is achieved by the blockchain system, the mined block will be added to the public blockchain and the miner who first successfully mines a block will earn the reward.

In this section, we give one network topology to identify a PFN within a field of mining. Especially, we consider a network topology: different FNs first form a circle. When multiple circles are covered with each other, we will choose one FN covered by all circles as a PFN, then we deploy adequate storage capability to store and schedule mining tasks. The other remaining FNs are then served as miners with limited computation capacity. The detailed deployment is illustrated in Fig. 1.

B. Mining Reward Computed by Fog Computing

First, assume the arrival rate of the kth IoT device priority to submit transactions to PFN follows a Poisson process with rate γ_k. Thus, the total number of transactions is $\Gamma = \sum_k \gamma_k$. Upon those transactions being confirmed, they will be removed from the PFN updating again. In addition, denote w_m by the computation power of each FN m. Then, $W = \{w_1, \ldots, w_m\}$ is the computation power set of all FNs. As a result, the probability of mining a block for FN m in one consensus round can be expressed as

$$P_m^{\text{mine}} = \frac{w_m}{\sum_{m \in M} w_m} \tag{1}$$

such that $\sum_{m \in M} P_m^{\text{mine}} = 1$, and $P_m^{\text{mine}} > 0$.

In the mining tournament, let $S = \{S_1, S_2, \ldots, S_m\}$ denote the block size of each FN m mining, and s_n be the data size of transaction n, where $n \in N$. The relationship between the block size of FN m and transaction n satisfies $S_m = \sum_n x_{nm} s_n$, where $x_{nm} = 1$ means that transaction n is allocated to FN m, and 0 otherwise.

In the mining process, assume that the occurrence of obtaining a hash value by FNs can be modeled as a random variable following a Poisson process with the mean rate of λ [22]. Once the FN successfully mines a block, it will broadcast the mined block to the rest of FNs in the whole network to reach a consensus. Upon the block is added to the blockchain, the FN who first mines the block will receive the mining reward (token reward). The reward consists of two parts: 1) a fixed reward and 2) a variable reward. Denote by R the fixed reward for mining a new block, and $\sum_n x_{nm} \rho_n$ as the variable reward, i.e., transaction fee. Here, ρ_n is the transaction fee of transaction n. Thus, FN m's token reward R_m can be shown as

$$R_m = \left(R + \sum_n x_{nm} \rho_n \right) P_m^{\text{win}} \tag{2}$$

where P_m^{win} is the probability that FN m successfully mines a new block, we will introduce with more details as follows.

The process of successfully mining a block is composed of two phases, i.e., the mining phase and consensus phase. In the mining phase, notice that the probability that FN m mines a block is P_m^{mine}. In the consensus phase, when a valid block is mined, it will be propagated across the whole network to execute a consensus process. However, there may exist an orphan block during the consensus phase. Specifically, other FNs also discover a new block at the same time, only the block which is first verified by the network will obtain the token reward, and other candidate blocks will be discarded by blockchain, called as an orphan block. Suppose the orphaning probability of FN m is P_m^{orphan}, thus, the probability of successfully mining p_m^{win} can be derived as

$$P_m^{\text{win}} = P_m^{\text{mine}} \left(1 - P_m^{\text{orphan}} \right). \tag{3}$$

Here, we regard the total delay in a consensus phase is composed of propagation delay and the verification delay. For the propagation delay, it can be modeled as $\alpha_p = (\sum_n x_{nm} / \gamma_0 a)$, where γ_0 is a scale-related network parameter and a is the average channel capacity of each link. Also, FN's verification

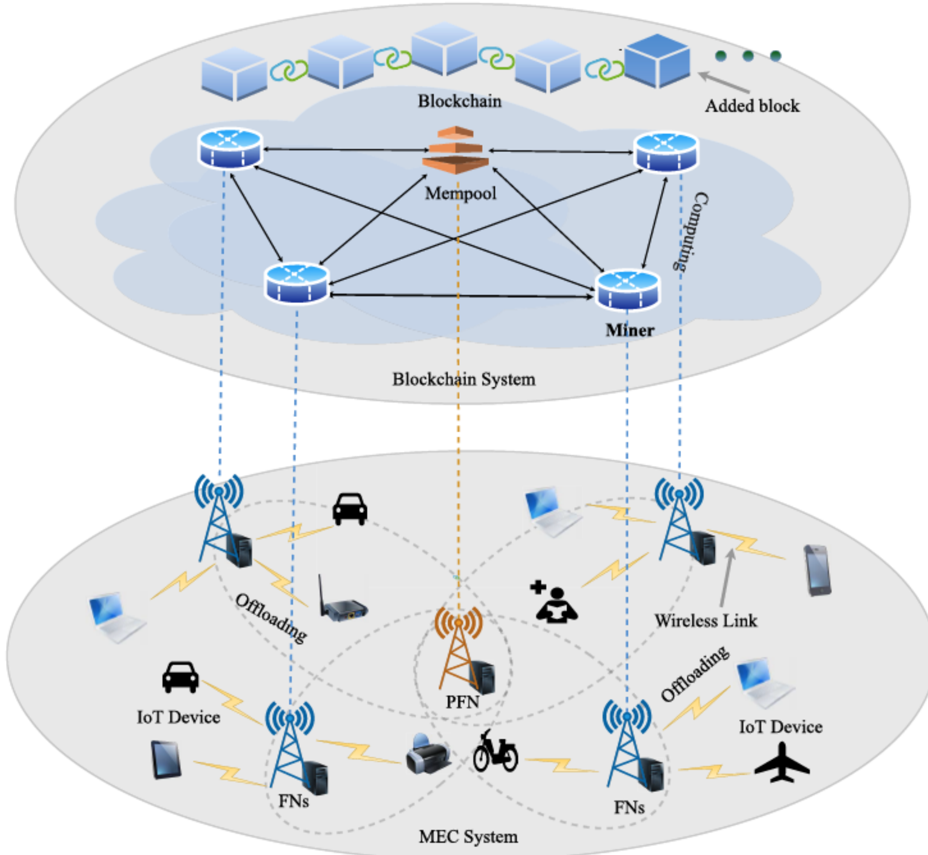

Fig. 1. System model.

delay α_v is assumed linear to the block size, i.e., $\alpha_v = fS_m$, where f is a constant parameter determined by network scale and the average verification speed of nodes [22], [30]. Then, the propagation delay in the consensus phase for block m is $\alpha_m = \alpha_p + \alpha_v$. Additionally, since the generation of new blocks follows a Poisson process with a parameter λ reflecting the complexity of mining a block, FN m's orphaning probability is expressed as

$$P_m^{\text{orphan}} = 1 - e^{-\lambda \alpha_m}. \tag{4}$$

Above all, the expected reward R_m of FN m can be expressed R_m with more details as follows:

$$R_m = \left(R + \sum_n x_{nm} \rho_n \frac{w_m}{\sum_{m \in M} w_m} e^{-\lambda \alpha_m} \right). \tag{5}$$

C. Service Cost in Mining Phase

1) Mining Transmission Model: When a mining block is transmitted from PFN to one FN, we need to compute the transmission rate. Especially, assume that a wireless backbone is connected between PFN and FNs. Then, assume each FN m consumes a fixed transmission power P_m, and the mining block is transmitted over an orthogonal channel. Then, the achievable transmission rate γ_m is given by Shannon capacity

$$\gamma_m = B_m \log_2 \left(1 + \frac{P_m H_m}{\delta^2} \right) \tag{6}$$

where H_m is the channel gain between PFN and FN m and δ^2 represents the noise power.

The transmission delay of block m should not exceed the maximization tolerant delay ϱ_m of FN m, which is calculated by

$$\frac{S_m}{\gamma_m} \le \varrho_m. \tag{7}$$

Finally, we give the transmission cost TC_m in the mining phase, which is defined as follows:

$$\text{TC}_m = \frac{S_m}{\gamma_m}. \tag{8}$$

2) Mining Computation Model: The computation cost of each FN executing the PoW puzzle varies greatly depending on the workload of FNs. The power consumption of an FN is determined mainly by the processor and memory [31]. Assume the processor power of one server m depends on the frequency ν_m and the voltage V_m when computing one block. Thus, the processor power of FN m is defined as $\text{PP}_m = \mathcal{A}_m \nu_m V_m^2$, where $\mathcal{A}_m$ is a constant related to the cores of CPU. Then, the memory power of one FN m is defined as $\text{MP}_m = (1/2)\mathcal{B}_m \nu_m V_m^2$, where $\mathcal{B}_m$ is also a constant decided by FN. Finally, the computation cost CC_m of an FN is shown as follows:

$$\text{CC}_m = \text{PP}_m + \text{MP}_m. \tag{9}$$

To improve the utilization of computing resources, we set the computing resource schedule constraint. Assume d_n is the required quantity of CPU cycles when transaction n is confirmed by an FN m. Thus, the total CPU cycles needed per block by FN m should not exceed that supplied by FN m

$$\sum_{n \in N} x_{nm} d_n \leq v_m. \tag{10}$$

D. Chain Mining Constraints

To optimize the social welfare of miners in a blockchain system, we set the first mining requirement about the size of each transaction, i.e., the size of a transaction being mined by FN m is at least equal to the minimum acceptable transaction size of FN m. Here, we set ϖ_m as the minimum acceptable transaction size of FN m and s_n as the size of transaction n. Then, an infinitely negative number C is assigned in the transaction size constraint, being mainly used to judge whether picking out the transactions. Additionally, we also introduce one event function I_A. Here, when the event A is true, the function $I_A = 1$, and 0 otherwise. Especially, $I_{\{s_n < \varpi_m\}} = 1, x_{nm} = 0$ means when the size of transaction n is lower than ϖ_m of FN m, the transaction n will not be mined by FN m. Furthermore, $I_{\{s_n < \varpi_m\}} = 0, x_{nm} \in \{0, 1\}$ represents when the size of transaction n satisfies ϖ_m of FN m, the transaction n will have chances to be mined by FN m. With the above, the constraint can be given mathematically

$$C\left(1 - I_{\{s_n < \varpi_m\}}\right) + x_{nm} \leq 0. \tag{11}$$

To this end, we set the second mining requirement about the transaction fee, an expense that a business must pay each FN who confirms a transaction. Especially, the transaction fee ρ_n of the transaction n can not be less than the minimum acceptable transaction fee ς_m of FN m. Note that $I_{\{\rho_n < \varsigma_m\}} = 1, x_{nm} = 0$ means if the transaction fee ρ_n is less than the minimum acceptable transaction fee of FN m, it will be dropped by FN m. What is more, $I_{\{\rho_n < \varsigma_m\}} = 0, x_{nm} \in \{0, 1\}$ represents if and only if the transaction fee ρ_n is higher than ς_m, the transaction n can be selected to mine by FN m

$$C\left(1 - I_{\{\rho_n < \varsigma_m\}}\right) + x_{nm} \leq 0. \tag{12}$$

When the transactions are uploaded to the PFN, FNs will compete for the priority right to mine one block. However, only one FN who first mines the block will obtain the reward. It will inevitably cause waste of computing resources of other FNs who mine the same block. Thus, to improve the computing resource utilization efficiency, we set the third mining constraint. Each transaction can be mined at most of D_n FNs during one mining process, given by mathematically

$$\sum_{m \in M} x_{nm} \leq \mathcal{D}_n. \tag{13}$$

IV. Problem Formulation of Social Welfare Maximization

In this section, we first define the service cost of FN m, which consists of transmission cost and computation cost, shown as follows:

$$C_m = \zeta * \mathrm{TC}_m + \vartheta * \mathrm{CC}_m \tag{14}$$

where ζ and ϑ are tunable parameters to keep the balance between the transmission cost and computation cost.

With the above, we define the social welfare of FNs, i.e., the mining expected reward of miners minus the mining service cost. Particularly, the following equation is defined as the objective value of each FN m:

$$\Omega_m = R_m - C_m. \tag{15}$$

To this end, we formulate the social welfare maximization problem as follows. The goal is to maximize the social welfare of all FNs, by taking into account of network resources in the mining phase and mining constraints

$$\max_{x_{nm}} \sum_m \Omega_m$$
$$\text{s.t. } (7) \text{ and } (10)\text{–}(13). \tag{16}$$

V. Two-Sided Matching Game With One-Sided Preference

In this section, we first propose the two-sided matching game with one-sided preference. We then design a DMA for resource management in blockchain applications involving PoW assisted by FNs and PFN.

In this article, considering that matching theory has emerged as a promising technique for wireless resource management integrated with blockchain technology, it not only can take advantage of local information of each player, such as its transaction size, transaction fee, and QoS to design player's preference function but also can achieve the stability and optimality accurately reflecting different player's objective. Besides, it can also capture not only the cooperative interactions between players on different sides but also the competitive interactions between players on the same side. Thus, reaping the benefits of matching theory for wireless networks, we develop a practical DMA based on edge computing in our blockchain system. On the one hand, it eliminates competition among IoT devices. On the other hand, it improves the limited computing resource utilization among FNs of conflicting interests and the probability of successful mining. Furthermore, the two-sided matching game with one-sided preference also captures local information of multiple devices to solve the social welfare maximization problem.

Especially, we first design FNs' preference profiles corresponding to their local information. But since the mining tasks are items that do not have preference choices once being dropped into PFN, the preference profiles of transactions are not considered in this article. Thus, the matching theory is applied to reformulate the social welfare maximization problem into a distributed matching game with one-sided preferences. Finally, we design DMA based on edge computing to maximize the social welfare of FNs under the consideration of limited computing resources and various mining constraints, converging to a stable solution.

A. Matching Game Formulation

According to the local information collected from PFN and FNs, we assume that the FNs have strict priority for transactions, and PFN can schedule the transactions to FNs randomly.

Additionally, in terms of mining requirements, we find that one transaction can be mined by multiple FNs simultaneously and one FN can mine a block consisting of multiple transactions. Therefore, we design a matching game with one-sided preference given by the tuple $(N, M, (\succ_m, \varrho_m, v_m)_{m\in M}, (A_n)_{n\in N})$. Here, $\succ_m$ is the set of strict preference relation of FN m over the transactions and it is a linear ordering over $N \cup \{m\}$. $A_n \subseteq M$ represents a set of acceptable FNs for transaction n, derived by formulas (11) and (12). Formally, we define the matching game as follows.

Definition 1: A matching μ is a mapping from the set $N \cup M$ into the set $N \cup M$, which is an assignment of transactions to FNs such that:

1) $\mu(n) \in M$, if and only if $\mu(m) \in N$;
2) for all $n \in N$, there exists $m \in M$ such that $\mu(n) \in M$;
3) for all $m \in M$, there exists $n \in N$ such that $\mu(m) \in N$;
4) $|[(\sum_n s_n \mu(m))/\gamma_m]| \le \varrho_m$ and $|\sum_n d_n \mu(m)| \le v_m$;
5) $|\varpi_m \mu(n)| \le s_n$, $|\varsigma_m \mu(n)| \le \rho_n$ and $|\sum_m \mu(n)| \le D_n$.

B. Preference Profiles of FNs

In the matching game with one-sided preference, $\phi_n(m)$ denotes the preference profiles of FN m for transaction n. Particularly, the preference profiles are achieved by the estimated profit of FN m when picking the transaction n. It can be seen that the preference function of FN m picking transaction n is expressed as follows:

$$\phi_n(m) = \left\{\rho_n \frac{w_m}{\sum_{m\in M} w_m} e^{-\lambda f s_n}\right\} - \left\{\frac{\zeta}{B_n} + \vartheta f s_n\right\} \tag{17}$$

where B_n is the bandwidth allocated to transaction n by FN m.

C. Distributed Matching Algorithm

To solve the social welfare maximization problem, in terms of a noticeable property of matching theory [32] and blockchain technology, we now develop the DMA based on edge computing to obtain a stable matching result. Before describing the implementation process of DMA, we first define stable matching as follows.

Definition 2 (Stable Matching): A matching μ is stable if no players are matched to an unacceptable partner and there is no unmatched pair (n, m), where each of them either strictly prefers the other to his partner in μ.

DMA is an extension of the deferred acceptance algorithm, proposed by Gale and Shapley [33]. Only when the reduced preference ordering of FNs is empty, the stable matching exists. We now present the algorithm with more details. First, given a instance $\pi = (N, M, \phi_N(M))$ of DMA. Second, notice that the algorithm conducts PFN to manage the schedule of all the transactions, and FN m tentatively chooses the one it likes best and becomes engaged to the corresponding transaction n. Then, FN m will reject the rest of the proposals and declare them to be unacceptable. Besides, we repeat iterations until the algorithm halts when no more proposals can be made. Notice that when existing some FN m who has received a proposal but is not matched in matching μ, the stable matching result will not exist, but we will still repeat it until all the FNs are matched, deriving an engagement mapping, i.e., a stable matching result.

Algorithm 1 DMA

Input: $\pi = (N, M, \phi_N(M))$, γ_k, ϖ_m, ς_m.
Output: μ
Initialize: flag $(m) \leftarrow$ false, $\forall m \in M$;
  transaction set of FN m in the blockchain: $\Re_m = \emptyset$;
repeat
  while transaction n not engaged **do**
    PFN schedules n to at most D_n FNs randomly
    if $s_n \ge \varpi_m$ or $\rho_n \ge \varsigma_m$ **then**
      updating $\Re_m$;
      flag $(m) \leftarrow$ true;
    end if
    if $\phi_n(m) > \phi_{n'}(m), n, n' \in \Re_m$ **then**
      if $|\sum_n x_{nm} s_n| \le \varsigma_m \gamma_m$ or $|\sum_n x_{nm} d_n| \le v_m$ **then**
        $\Re_m = \Re_m / n'$;
      end if
    end if
  end while
until $\phi_n(m) = \emptyset, \forall m \in M, n \in N$;
if every FN is engaged **then**
  μ and new block of each FN received;
else
  return "false"
end if
Output: μ is a stable matching.

In the algorithm, we need to pay attention to two details. First, "deleting the pair (n, m)" means that a transaction should be deleted from the preference ordering of an FN. Second, "flag (m)" represents the matching event about FN m is true or false.

VI. Performance Analysis of DMA

In this section, we will analyze the performance of DMA from two aspects, stability and computational complexity of DMA, respectively.

A. DMA to Find Stable Matching

The optimality property of the stable matching of DMA will be derived with the aid of several lemmas in this section.

Lemma 1: A matching μ is feasible for two-sided matching with one-sided preference if for each $m \in M$, $\mu(m) \neq m$ implies $\mu(m) \in N$.

Proof: Note that for each $n \in N$ and each $m \in M$, $\mu(m) \in N \cup \{m\}$. If $\mu(m) \in N$, then FN m is matched to transactions under μ. If $\mu(m) = m$, then FN m is said to be unmatched under μ. Thus, if $\mu(m) \neq m$, it must satisfy $\mu(m) \in N$. ■

Lemma 2: If the pair (n, m) is deleted during the execution of algorithm DMA, then the pair cannot block any matching pairs that are never deleted.

Before proving Lemma 2, we first introduce the definition of block pair, expressed as follows.

Definition 3 (Block Pair): A pair (n, m) is a blocking pair with respect to matching μ if transaction n and FN m are not engaged together, but they prefer each other to other engaged partners.

Proof: According to the analysis of DMA, notice that under the constraint (13), PFN schedules a sequence of transactions to FNs randomly. For those eligible matchings, according to the preference profiles of FNs, FN makes decision in terms of constraints (11) and (12) in the blockchain. They will delete those pairs that are strict successors of current entries. For example, suppose transaction sequence $(1, 2, 3)$ is available to FN m satisfying $\phi_1(m) > \phi_2(m) > \phi_3(m)$ and $\Omega_m = \phi_1(m) + \phi_2(m) + \phi_3(m)$. If there is another transaction sequence $(1, 2, 4)$ being available to FN m, it satisfies $\phi_1(m) > \phi_2(m) > \phi_4(m)$ and $\phi_4(m) > \phi_3(m)$. Thus, the FN m will delete the pair $(3, m)$ and derive $\Omega'_m = \phi_1(m)+\phi_2(m)+\phi_4(m)$. Finally, we draw a conclusion $\Omega'_m > \Omega_m$ and the deleted pair $(3, m)$ cannot block any matching pairs that are never deleted. ■

Lemma 3: No stable matching is ever deleted during the execution of DMA based on edge computing.

Proof: Suppose that $(\mu(m), m)$ is a stable matching but is deleted during the execution of DMA. For a contradiction, assume $(\mu(m), m)$ is a stable matching for FN m. If there exists another FN denoted by m', it is also becoming engaged to transaction set $\mu(m)$ and satisfying $\Omega_m < \Omega_{m'}$. Assume there is a stable matching for FN m', which strictly prefers $\mu(m')$ to $\mu(m)$. Suppose the social welfare of FN m' for stable matching is $\Omega'_{m'}$, which must satisfy $\Omega_{m'} < \Omega'_{m'}$. Thus, for FN m', its matching $(\mu(m), m')$ would have been deleted before $(\mu(m'), m')$. Since the matching $(\mu(m), m)$ is a stable matching, which will be never deleted, according to Lemma 2, giving a contradiction. ■

Corollary 1: DMA converges to a stable matching.

Proof: According to Lemma 3, note that the output of our DMA algorithm is a stable matching. Let μ be the stable matching by DMA. Assume for contradiction, μ is not a stable matching in all the engagement relations at the termination of DMA. Then, suppose matching μ' is a stable matching and there must be an FN m, who strictly prefers μ' to μ. To this end, due to Lemma 2, matching (μ', m) will be deleted before the FN m is engaged to μ. Finally, we know that no stable matching is deleted during the execution of the algorithm DMA by Lemma 3, giving a contradiction. Thus, we say μ is a stable matching. ■

B. Computational Complexity Analysis of DMA

In this section, we will analyze the computational complexity of Algorithm 1. It can be seen that our designed algorithm always tries to search for a stable matching in the bipartite graph formed by the engagement relation. Considering that our search is limited to some constraints, it becomes difficult to find a standard maximum cardinality matching. However, the key to bound the total iteration times of work done is to analyze the engagement graph of Algorithm 1 in finding maximum cardinality matching, called as the computational complexity of DMA.

Note that with the help of [34], note that for men and women marriage problems, the total number of operations during the execution of the Gale/Shapley algorithm is bounded by a calculation. Especially, constant times the number of pairs deleted. It is clear that the lower bound of the computational complexity of the algorithm of [34] is $O(n^2)$. Inspired by the analysis of the bound in the worst case, we do a similar analysis in the current Algorithm 1. Thus, the computational complexity of DMA is essentially bounded via a calculation, where constant times the number of deleted pairs as well.

Suppose that the total number of iterations is κ, during the ith iteration of the repeat loop, x_i pairs are deleted because those partners are not in both reference ordering set and y_i pairs are participated in the matching process but are deleted because of certain constraints or successors. Thus, we could derive the total deleted pairs $\sum_i^\kappa (x_i + y_i)$, and how to derive the upper bound of the maximum pairs deleted? First, note that the number of all the possible pairs is MN. Second, suppose the minimized transaction size is $s^{\min}$ and the allowable maximized block size is $S^{\max} = \varrho_m * \gamma_m$, thus, we derive the number of one block picking transactions at most $I = (S^{\max}/s^{\min})$. Furthermore, suppose the required computing resource is at least $d^{\min}$, and the maximized computing resource applied by an FN is $D^{\max}$. So the number of one block picking transactions is at most $X = (D^{\max}/d^{\min})$. Simultaneously, according to the transaction fee and transaction size constraint, suppose the number of transactions is dropped from PFN is at most $Y = \max\{\sum_n(\rho_n < \varsigma_m), \sum_n(s_n < \varpi_m)\}$. Thus, when all FNs are matched, the maximum cardinality matching is $\min\{N - Y, M \times \min\{I, X\}\}$. Thus, the maximized number of pairs deleted is $MN - \max\{MI, NX\}$. With above, note that $\sum_i^\kappa (x_i + y_i) \leq MN - \min\{N - Y, M \times \min\{I, X\}\}$. Finally, the computational complexity of the algorithm is $O((MN - \min\{N - Y, M \times \min\{I, X\}\}))$. Hence, the overall computational complexity of Algorithm 1 is $O(MN)$, a lower bound to find a stable matching in this article.

The followings are the proof of the upper bound about DMA, and the upper number of achieving a final stable matching result is bounded by $\{M \times \min\{I, X\} + MN\}$, which contains two parts. Especially, suppose that each FN included in a pair needs to search $\min\{I, X\}$ times to obtain a stable matching result based on the analysis of lower bound. Thus, the number of total searching times for all the FNs is $M \times \min\{I, X\}$. Furthermore, since the FNs need to search for available transactions to form all the possible pairs, and the maximum number of pairs deriving a stable matching is MN. Thus, the upper bound of the computational complexity to find a stable matching is $\{M \times \min\{I, X\} + MN\}$.

VII. Performance Evaluation

In this section, a computer simulation is provided to show the simulation results of the proposed DMA enabled with edge computing. We first present the simulation settings, then in terms of the simulation parameters, simulation results are shown meaningfully.

TABLE I
SIMULATION PARAMETERS

Parameter	Value
The transaction fee of transaction n ρ_n	60-90
The fixed reward for mining a new block R	12.5
A constant parameter reflecting the impact of block size f	0.002
The Poisson parameter of the occurrence of a new block λ	0.07
The available bandwidth of FN m B_m	5MHz-15MHz
The channel gain between PFN and FN H_m	1.0e-16-1.6e-14
The power of noise δ^2	-174dBm/Hz
The maximization tolerant transmission delay of FN m ϱ_m	0.3-1.5s
The parameter to estimate the value of transmission cost ζ	0.0005/per unit cost
The parameter to measure the value of computational cost ϑ	0.002/per unit cost
The transaction size s_n	5000-1000B
The number of CPU cores related with the fog server $\mathcal{A}_m$	10-50
A constant parameter decided by the fog server $\mathcal{B}_m$	10-80
The CPU-cycle frequency of FN m v_m	10-100GHz CPU cycles/s

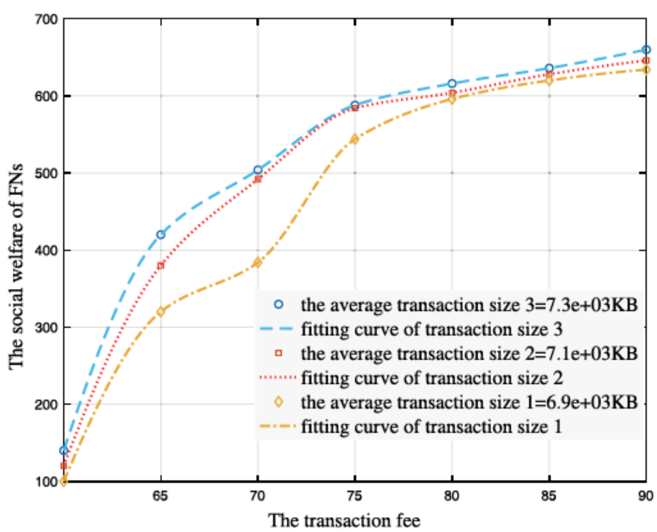

Fig. 2. Social welfare of FNs and the average transaction size with regard to the transaction fee ρ.

A. Simulation Settings

We conduct our simulation on a MATLAB $R2017b$ to round each simulation. Furthermore, other parameter settings are based on the Ethereum, which is illustrated in Table I.

In the wireless network simulation part, assume the network is covered by mixed picocell FNs, femtocell FNs, and microcell FNs coverage radius of which is 1000 m. The average computation power of each FN varies from 300 to 1500 w. Besides, the transmission power of PFN is set to 300 w.

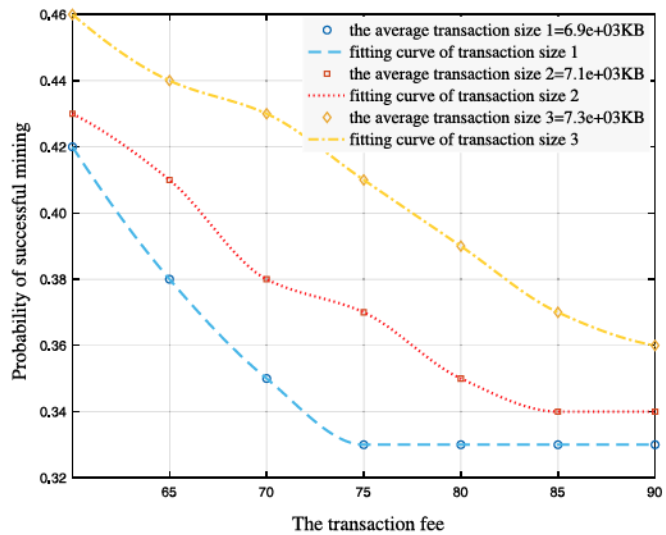

Fig. 3. Probability of successful mining and the average transaction size versus the transaction fee ρ.

Furthermore, to illustrate the impacts of different parameters from our proposed model on the performance, we consider 100 transactions and 20 FNs at first. Then, if there are other demands of simulations, the number of transactions and FNs will vary according to the simulation requests. Then, considering the PoW from blockchain applications assisted by the fog servers, we set the transactions that are released by the parameter $\gamma_k = 5$ Poisson distribution. Finally, some other parameters about the blockchain technology are listed in Table I.

In this article, to verify the performance of our proposed DMA algorithm, we compare the simulation results with the other two existing algorithms: 1) ADMMs-based algorithm [16] and 2) GIA [17]. Besides, we discuss the simulations in different situations with more details and evaluate our proposed algorithm by using the transmission delay of each block, the probability of successful mining, and the social welfare of all FNs.

B. Simulation Results

As is shown in Fig. 2, when IoT devices are connected to FNs, we first explore the effect of the transaction fee parameter on the social welfare of FNs in a blockchain system. As discussed in Fig. 2, it shows that with the increasing of the transaction fee, the social welfare of FNs is also growing quickly. Then, we change each transaction size, and it is obvious that social welfare is also decreasing with the increasing of the average transaction size by observing those three color lines. This is because when the transaction fee is growing, the reward for each FN is also increasing, leading to the growth of the social welfare of the system. Besides, from Fig. 3, as the probability of successful mining is lower, the transaction fee is larger. This is because the variable transaction fee increases the existence of orphan blocks. However, due to the effect of different transaction sizes on transmission delay, the larger the transaction size that has the potential incentive to increase the transmission delay, the lower the probability.

Furthermore, from Figs. 4 and 5, we explore the social welfare of FNs and the probability of successful mining between

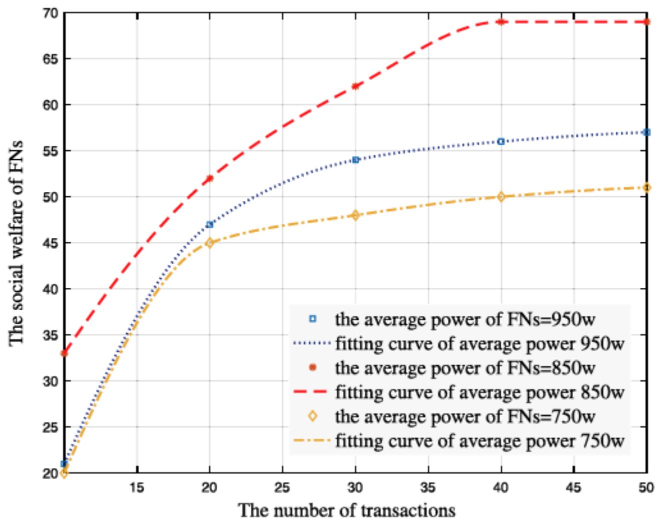

Fig. 4. Social welfare of FNs and the average power of FNs with regard to the number of transactions N.

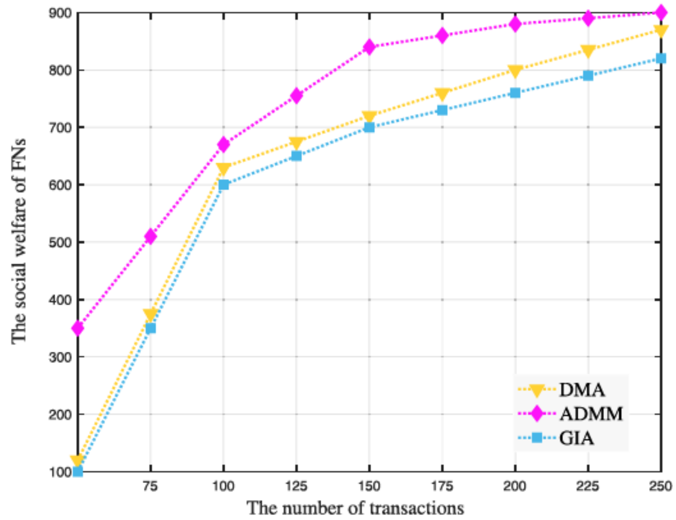

Fig. 6. Social welfare performance of the proposed algorithm compared with ADMM and GIA.

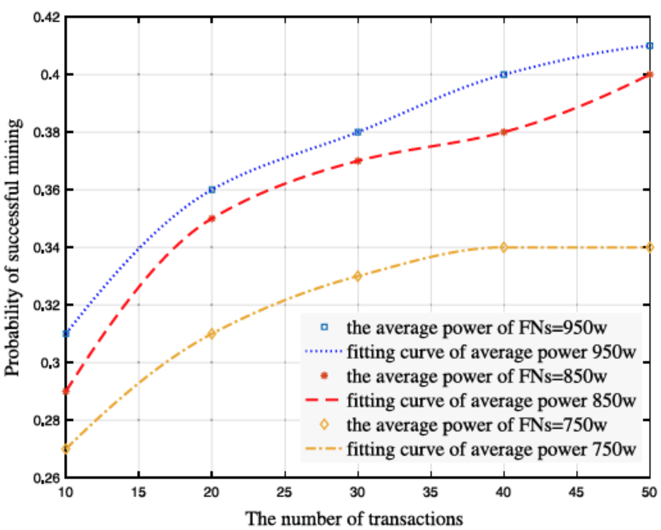

Fig. 5. Probability of successful mining and the average power of FNs versus the number of transactions N.

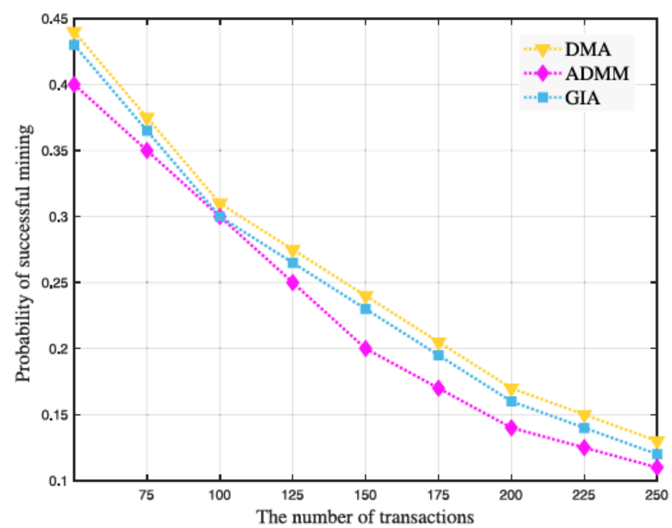

Fig. 7. Probability of successful mining of the proposed algorithm compared with ADMM and GIA.

the number of transactions and the operation power of each FN. We observe that the social welfare and probability increase with the growth in the number of transactions. Because when the number of transactions increases, the number of FNs is fixed. Thus, social welfare will increase to a stable value. Then, with the power of FNs increasing, the hash power of each FN also increases, but the maintenance cost and operation cost of FNs become larger. As shown in Fig. 4, the system intends to set the perfect average power of each FN as 850 w to achieve more social welfare. Additionally, Fig. 5 shows that the increase of the probability is consistent with mining power. Because of the higher hash power incentives to the higher probability of successful mining. Since the block size is limited and the number of transactions is growing, the number of successful mining block will not increase any more during the mining process.

To verify the performance of our proposed algorithm based on edge computing, we investigate the impact of the number of transactions on DMA, ADMM, and GIA, respectively, in a blockchain system, which is shown in Figs. 6 and 7. Especially, we explore the number of transactions from 50 to 250 and set the number of FNs at 20. From Fig. 6, we can observe that the social welfare of the ADMM is higher than that of DMA and GIA, but the probability of successful mining of DMA is the highest. The reason is that the iterative function of GIA is always changing based on its own iterative standard, the randomness of the iterative function of GIA may incentive the reduction of social welfare. For the ADMM, although it does not own the iterative function, the search space is the largest compared with DMA, and the social welfare of ADMM is the highest. Furthermore, Fig. 7 illustrates the variation in the probability associated with the number of transactions. We find that when N increases, the results under those three algorithms are decreasing to a stable state. This is because the number of FNs and block size is limited.

In Fig. 8, we investigate the impact of the number of FNs on the social welfare of DMA compared with ADMM and GIA.

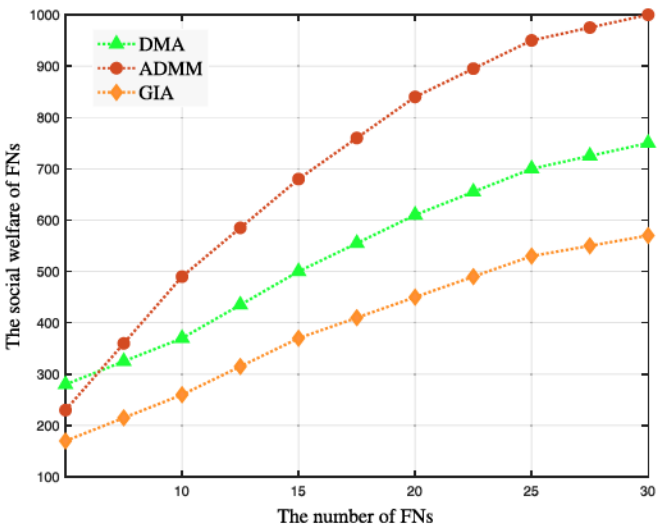

Fig. 8. Impact of the number of FNs M on the social welfare of the proposed algorithm compared with ADMM and GIA.

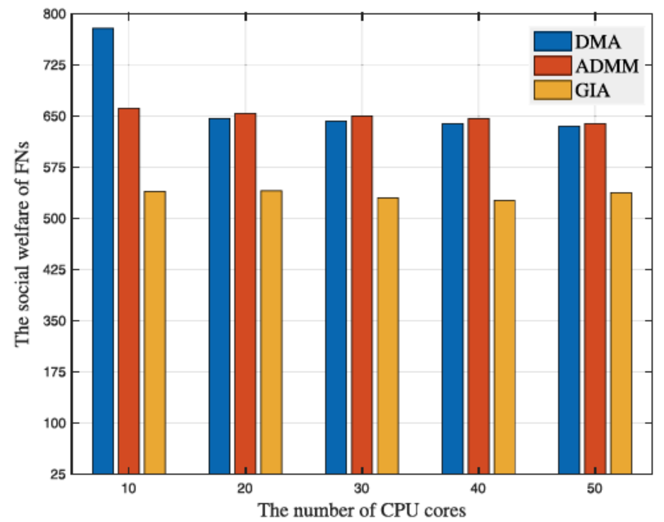

Fig. 10. Social welfare of FNs and the number of CPU cores with DMA, ADMM, and GIA.

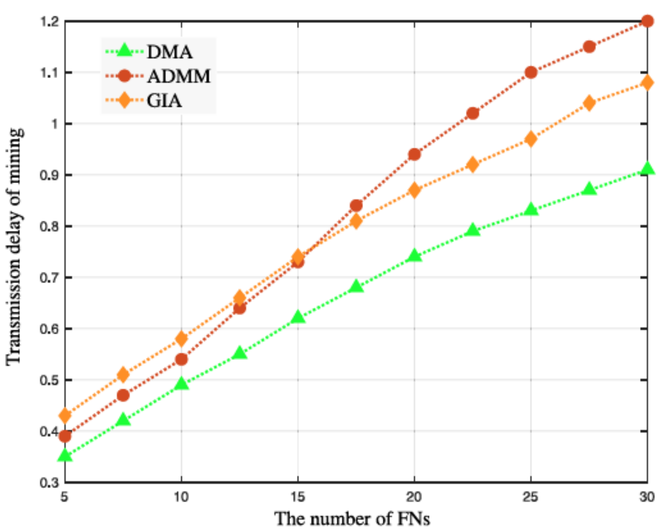

Fig. 9. Impact of the number of FNs M on the transmission delay of the proposed algorithm compared with ADMM and GIA.

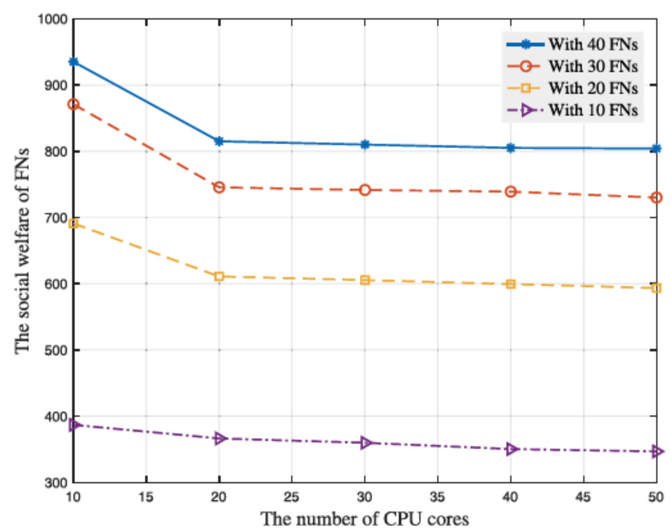

Fig. 11. Social welfare of FNs and the number of FNs with regard to the number of CPU cores.

We first fix the number of transactions to 100, then just observe the variant of the number of FNs. As expected in Fig. 8, it is clear that the social welfare of FNs increases as the number of FNs rises. Especially, DMA can improve the social welfare of FNs at least 50% compared with GIA, but it is lower than ADMM. Finally, Fig. 9 demonstrates the impact of the number of FNs on the transmission delay of DMA, ADMM, and GIA. By comparing curves with different transmission delays, we find that the transmission delay of DMA is the least compared with the other two algorithms. It is of great significance to improve the transmission delay of DMA, because of the high demand for data recording time for intensive IoT applications.

Next, we give an illustration of the social welfare of FNs and the number of CPU cores with three schemes, DMA, ADMM, and GIA, respectively. In Fig. 10, it can be seen that with the number of CPU cores increasing, the social welfare decreases. The reason is that the increasing of consumed operation cost of FNs is higher than that of the revenue of FNs. Thus, it is important to efficiently schedule computing resources. Furthermore, since the number of transactions is fixed, the demanded computing resource and the social welfare of FNs will be stable in the end. In addition, it also finds that the social welfare of DMA is close to that of ADMM, which is far more than GIA. Although the ADMM is better than DMA on social welfare, the transmission delay and the probability of successful mining of DMA is better.

Finally, we examine the impacts brought by the number of FNs and the CPU cores, which are shown in Fig. 11. We find that social welfare increases with the increment of the number of FNs and the decrease in the number of CPU cores. This is because the maintenance costs of servers increase when the CPU cores increase, but the total number of transactions is not changed, the required maximization computing resource is also quantitative. Thus, social welfare will converge to a steady state.

VIII. Conclusion

In this article, we have proposed a social welfare optimized blockchain for fog-enabled IoT networks. In particular, we first formulate a social welfare maximization problem while facing the varieties of mining tasks and heterogeneous resource capabilities at FNs. Then, we jointly optimize the restricted computing resource and various mining requirements of FNs, such as transaction size, transaction fee, transaction utilization, and transmission delay together. Besides, in terms of matching theory, we reformulate the original problem into a two-sided matching game with one-sided preference. Then, we design a DMA algorithm based on edge computing to achieve a suboptimal solution, i.e., the stability. Furthermore, we have performed the MATLAB to validate the proposed theoretical model. Additionally, by conducting the simulations, we have evaluated the network performance, which greatly improves the social welfare of all FNs. For future work, we will further study a two-sided matching market, where the online matching mechanism between the diversified resources of fog/cloud and a variety of computation tasks with IoT devices' features under the condition of the complex network environment.

References

[1] P. Danzi, A. E. Kalor, C. Stefanović, and P. Popovski, "Delay and communication tradeoffs for blockchain systems with lightweight IoT clients," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2354–2365, Apr. 2019.

[2] S. Shen, Y. Han, X. Wang, and Y. Wang, "Computation offloading with multiple agents in edge-computing–supported IoT," *ACM Trans. Sens. Netw.*, vol. 16, p. 8, Dec. 2019.

[3] M. Chamekh, S. E. Asmi, M. Hamdi, and T. Kim, "IoT based tracking system for supply chain management," in *Proc. 6th Int. Conf. Wireless Netw. Mobile Commun. (WINCOM)*, Marrakesh, Morocco, 2018, pp. 1–5.

[4] M. Demir, O. Turetken, and A. Ferwom, "Blockchain and IoT for delivery assurance on supply chain (BIDAS)," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Los Angeles, CA, USA, 2019, pp. 5213–5222.

[5] J. W. Heron, J. Jiang, H. Sun, V. Gezerlis, and T. Doukoglou, "Demand-response round-trip latency of IoT smartgrid network topologies," *IEEE Access*, vol. 6, pp. 22930–22937, 2018.

[6] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan, and X. Chen, "Convergence of edge computing and deep learning: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 869–904, 2nd Quart., 2020.

[7] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2084–2123, 3rd Quart., 2016.

[8] J. H. Jeon, K. Kim, and J. Kim, "Block chain based data security enhanced IoT server platform," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Chiang Mai, Thailand, 2018, pp. 941–944.

[9] Y. Song, Y. Fu, F. R. Yu, and L. Zhou, "Blockchain-enabled Internet of vehicles with cooperative positioning: A deep neural network approach," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3485–3498, Apr. 2020.

[10] M. Singh, A. Singh, and S. Kim, "Blockchain: A game changer for securing IoT data," in *Proc. IEEE 4th World Forum Internet Things (WF-IoT)*, Singapore, 2018, pp. 51–55.

[11] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Netw.*, vol. 33, no. 5, pp. 156–165, Sep./Oct. 2019.

[12] X. Li, X. Wang, P. Wan, Z. Han, and V. C. M. Leung, "Hierarchical edge caching in device-to-device aided mobile networks: Modeling, optimization, and design," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 8, pp. 1768–1785, Aug. 2018.

[13] Z. He, Y. Zhang, B. Tak, and L. Peng, "Green fog planning for optimal Internet-of-Thing task scheduling," *IEEE Access*, vol. 8, pp. 1224–1234, 2019.

[14] M. Debe, K. Salah, M. H. U. Rehman, and D. Svetinovic, "IoT public fog nodes reputation system: A decentralized solution using ethereum blockchain," *IEEE Access*, vol. 7, pp. 178082–178093, 2019.

[15] F. Guo, F. R. Yu, H. Zhang, H. Ji, M. Liu, and V. C. M. Leung, "Adaptive resource allocation in future wireless networks with blockchain and mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1689–1703, Mar. 2020.

[16] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Distributed resource allocation in blockchain-based video streaming systems with mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 695–708, Jan. 2019.

[17] 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.

[18] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-edge computing: Partial computation offloading using dynamic voltage scaling," *IEEE Trans. Commun.*, vol. 64, no. 10, pp. 4268–4282, Oct. 2016.

[19] O. Bibani *et al.*, "A demo of IoT healthcare application provisioning in hybrid cloud/fog environment," in *Proc. IEEE Int. Conf. Cloud Comput. Technol. Sci. (CloudCom)*, Luxembourg City, Luxembourg, Dec. 2016, pp. 472–475.

[20] C. Qiu, H. Yao, C. Jiang, S. Guo, and F. Xu, "Cloud computing assisted blockchain-enabled Internet of Things," *IEEE Trans. Cloud Comput.*, early access, Jul. 23, 2019, doi: 10.1109/TCC.2019.2930259.

[21] D. Wu and N. Ansari, "A cooperative computing strategy for blockchain-secured fog computing," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6603–6609, Jul. 2020.

[22] Y. Jiao, P. Wang, D. Niyato, and K. Suankaewmanee, "Auction mechanisms in cloud/fog computing resource allocation for public blockchain networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 9, pp. 1975–1989, Sep. 2019.

[23] W. Sun, J. Liu, Y. Yue, and P. Wang, "Joint resource allocation and incentive design for blockchain-based mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 19, no. 9, pp. 6050–6064, Sep. 2020.

[24] Y. Yuan, T. Yang, H. Feng, and B. Hu, "An iterative matching-Stackelberg game model for channel-power allocation in D2D underlaid cellular networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 11, pp. 7456–7471, Nov. 2018.

[25] L. Wang, H. Wu, Z. Han, P. Zhang, and H. V. Poor, "Multi-hop cooperative caching in social IoT using matching theory," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2127–2145, Apr. 2018.

[26] T. Sanguanpuak, S. Guruacharya, N. Rajatheva, M. Bennis, and M. Latva-Aho, "Multi-operator spectrum sharing for small cell networks: A matching game perspective," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3761–3774, Jun. 2017.

[27] Y. Chen, Y. Xiong, Q. Wang, X. Yin, and B. Li, "Ensuring minimum spectrum requirement in matching-based spectrum allocation," *IEEE Trans. Mobile Comput.*, vol. 17, no. 9, pp. 2028–2040, Sep. 2018.

[28] B. Jia, H. Hu, Y. Zeng, T. Xu, and Y. Yang, "Double-matching resource allocation strategy in fog computing networks based on cost efficiency," *J. Commun. Netw.*, vol. 20, no. 3, pp. 237–246, Jun. 2018.

[29] P. Liu, J. Li, and Z. Sun, "Matching-based task offloading for vehicular edge computing," *IEEE Access*, vol. 7, pp. 27628–27640, 2019.

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

[31] M. Adhikari and H. Gianey, "Energy efficient offloading strategy in fog-cloud environment for IoT applications," *Internet of Things*, vol. 6, Jun. 2019, Art. no. 100053.

[32] G. Haeringer and V. Iehle, "Two-sided matching with one-sided preferences," in *Proc. 15th ACM Conf. Econo. Comput.*, Apr. 2014, p. 353.

[33] D. Gale and L. S. Shapley, "College admissions and the stability of marriage," *Amer. Math. Monthly*, vol. 69, no. 1, pp. 9–15, 1962.

[34] C. Ng and D. S. Hirschberg, "Lower bounds for the stable marriage problem and its variants," *SIAM J. Comput.*, vol. 19, pp. 71–77, Feb. 1990.

Lichao Yang received the B.S. degree in mathematics and statistics from Zhengzhou University, Zhengzhou, China, in 2015. She is currently pursuing the Ph.D. degree with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China.

She is currently a visiting Ph.D student with the University of Texas at Arlington, Arlington, TX, USA. Her current research interests include 5G small cell networks, fog and edge computing, online multiple-dimension resource management, and blockchain.

Ming Li (Associate Member, IEEE) received the B.E. degree in electrical engineering from Sun Yat-sen University, Guangzhou, China, in 2007, the M.E. degree in electrical engineering from Beijing University of Posts and Telecommunications, Beijing, China, in 2010, and the Ph.D. degree in electrical and computer engineering from Mississippi State University, Starkville, MS, USA, in 2014.

She is currently an Assistant Professor with the Department of Computer Science and Engineering, University of Texas at Arlington, Arlington, TX, USA. Before that, she was an Assistant Professor with the Department of Computer Science and Engineering, University of Nevada Reno, Reno, NV, USA. Her research has been funded by NSF, DoT, and NIH. Her research interests include Internet of Things, mobile computing, wireless networks, smart health, smart city, and privacy-preserving computing.

Dr. Li is a recipient of the prestigious awards of NSF CRII and CAREER. Her work won Best Paper Awards in IEEE Global Communications Conference (Globecom'15) and IEEE International Conference on Dependable, Autonomic and Secure Computing (DASC'7), respectively. She is a member of ACM.

Heli Zhang (Member, IEEE) received the B.S. degree in communication engineering from Central South University, Changsha, China, in 2009, and the Ph.D. degree in communication and information System from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2014.

She is currently an Associate Professor with the School of Information and Communication Engineering, BUPT. She participated in many National projects funded by National Science and Technology Major Project, National 863 High-tech and National Natural Science Foundation of China, and cooperated with many Corporations in research. Her research interests include heterogeneous networks, long-term evolution/fifth generation, and Internet of Things.

Dr. Zhang has been the reviewer for *IEEE Wireless Communications*, *IEEE Communication Magazine*, IEEE Transactions on Vehicular Technology, IEEE Communication Letters, and IEEE Transactions on Networking.

Hong Ji (Senior Member, IEEE) received the B.S. degree in communications engineering and the M.S. and Ph.D. degrees in information and communications engineering from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 1989, 1992, and 2002, respectively.

In 2006, she was a Visiting Scholar with the University of British Columbia, Vancouver, BC, Canada. She is currently a Professor with BUPT. She has authored more than 300 journal/conference papers. Several of her papers had been selected for best paper. Her research interests include wireless networks and mobile systems, including cloud computing, machine learning, intelligent networks, green communications, radio access, ICT applications, system architectures, management algorithms, and performance evaluations.

Prof. Ji has guest-edited *International Journal of Communication Systems* (Wiley) Special Issue on Mobile Internet: Content, Security and Terminal. She has served as the Co-Chair for Chinacom'11, and a member of the Technical Program Committee of WCNC'19/15/14/12, Globecom'17/16/15/14/13/12/11/10, ISCIT'17, CITS'16/15/12, WCSP'15, ICC'20/13/12/11, ICCC'13/12, PIMRC'12/11, IEEE VTC'12S, and Mobi-World'11. She is serving on the editorial boards for the IEEE Transactions on Green Communications and Networking and *International Journal of Communication Systems* (Wiley).

Mingyan Xiao (Graduate Student Member, IEEE) received the B.E. degree from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2014, and the M.S. degree from the National University of Defense Technology, Changsha, China, in 2017. She is currently pursuing the Ph.D. degree with the Department of Computer Science, University of Texas at Arlington, Arlington, TX, USA.

Her recent research interests are in the area of resource allocation and management in wireless networks, and data-driven security and privacy.

Xi Li (Member, IEEE) received the B.E. and Ph.D. degrees in communication and information system from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2005 and 2010, respectively.

In 2017 and 2018, she was a Visiting Scholar with the University of British Columbia, Vancouver, BC, Canada. She is currently a Professor with the School of Information and Communication Engineering, BUPT. She has published more than 100 papers in international journals and conferences. Her current research interests include resource management and intelligent networking in next generation networks, the Internet of Things, and cloud computing.

Prof. Li has also served as a TPC Member of IEEE WCNC 2012/2014/ 2015/2016/2019, PIMRC 2012/2017/2018/2019, GLOBECOM 2015/2017/2018, ICC 2015/2016/2017/2018/2019, Infocom 2018, and CloudCom 2013/2014/2015, the Chair of Special Track on cognitive testbed in CHINACOM 2011, the Workshop Chair of IEEE GreenCom 2019, and a peer reviewer of many academic journals.