Auction Mechanisms in Cloud/Fog Computing Resource Allocation for Public Blockchain Networks

Yutao Jiao, Ping Wang, *Senior Member, IEEE*, Dusit Niyato, *Fellow, IEEE*, and Kongrath Suankaewmanee

Abstract—As an emerging decentralized secure data management platform, blockchain has gained much popularity recently. To maintain a canonical state of blockchain data record, proof-of-work based consensus protocols provide the nodes, referred to as miners, in the network with incentives for confirming new block of transactions through a process of "block mining" by solving a cryptographic puzzle. Under the circumstance of limited local computing resources, e.g., mobile devices, it is natural for rational miners, i.e., consensus nodes, to offload computational tasks for proof of work to the cloud/fog computing servers. Therefore, we focus on the trading between the cloud/fog computing service provider and miners, and propose an auction-based market model for efficient computing resource allocation. In particular, we consider a proof-of-work based blockchain network, which is constrained by the computing resource and deployed as an infrastructure for decentralized data management applications. Due to the competition among miners in the blockchain network, the allocative externalities are particularly taken into account when designing the auction mechanisms. Specifically, we consider two bidding schemes: the constant-demand scheme where each miner bids for a fixed quantity of resources, and the multi-demand scheme where the miners can submit their preferable demands and bids. For the constant-demand bidding scheme, we propose an auction mechanism that achieves optimal social welfare. In the multi-demand bidding scheme, the social welfare maximization problem is NP-hard. Therefore, we design an approximate algorithm which guarantees the truthfulness, individual rationality and computational efficiency. Through extensive simulations, we show that our proposed auction mechanisms with the two bidding schemes can efficiently maximize the social welfare of the blockchain network and provide effective strategies for the cloud/fog computing service provider.

Index Terms—Blockchain, auction, cloud/fog computing, social welfare, pricing, proof of work, game theory

1 Introduction

By contrast to traditional currencies, cryptocurrencies are traded among participants over a peer-to-peer (P2P) network without relying on third parties such as banks or financial regulatory authorities [1]. As the backbone technology of decentralized cryptocurrencies, blockchain has also heralded many applications in various fields, such as finance [2], Internet of Things (IoT) [3] and resource offloading [4]. According to the market research firm Tractica's report, it is estimated that the annual revenue for enterprise applications of blockchain will increase to $19.9 billion by 2025 [5]. Essentially, blockchain is a tamperproof, distributed database that records transactional data in a P2P network. The database state is decentrally maintained, and any member node in the overlay blockchain network is permitted to participate in the state maintenance without identity authentication. The transactions among member nodes are recorded in cryptographic hash-linked data structures known as *blocks*. A series of confirmed blocks are arranged in chronological order to form a sequential chain, hence named *blockchain*. All member nodes in the network are required to follow the Nakamoto consensus protocol [1] (or other protocols alike), to agree on the transactional data, cryptographic hashes and digital signatures stored in the block to guarantee the integrity of the blockchain.

The Nakamoto consensus protocol integrates a critical computing-intensive process, called *Proof-of-Work (PoW)*. In order to have their local views of the blockchain accepted by the network as the canonical state of the blockchain, consensus nodes (i.e., block miners) have to solve a cryptographic puzzle, i.e., find a nonce to be contained in the block such that the hash value of the entire block is smaller than a preset target. This computational process is called *mining*, where the consensus nodes which contribute their computing power to mining are known as *miners*. Typically, the mining task for PoW can be regarded as a tournament [6]. First, each miner collects and verifies a certain number of unconfirmed transaction records which are aggregated into a new block. Next, all miners chase each other to be the first one to obtain the

- *Y. Jiao, D. Niyato, and K. Suankaewmanee are with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798. E-mail: {yjiao001, dniyato}@ntu.edu.sg, kongrath.jojo@gmail.com.*
- *P. Wang is with the Lassonde School of Engineering, York University, Toronto, ON M3J 1P3, Canada. E-mail: wangping@ntu.edu.sg.*

Manuscript received 30 Apr. 2018; revised 9 Feb. 2019; accepted 11 Feb. 2019. Date of publication 13 Mar. 2019; date of current version 7 Aug. 2019. (Corresponding author: Ping Wang.) Recommended for acceptance by O. Rana. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TPDS.2019.2900238

1045-9219 © 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

desired nonce value as the PoW solutions for the new block which combines the collected transactional data[1] and block metadata. Once the PoW puzzle is solved, this new block will be immediately broadcasted to the entire blockchain network. Meanwhile, the other miners receive this message and perform a chain validation-comparison process to decide whether to approve and add newly generated block to the blockchain. The miner which successfully has its proposed block linked to the blockchain will be given a certain amount of rewards, including a fixed bonus and a variable transaction fee, as an incentive of mining.

Since no prior authorization is required, the permissionless blockchain is especially suitable for serving as a platform for decentralized autonomous data management in many applications. Some representative examples can be found in data sharing [7], electricity trading in smart grid [8] and personal data access control [9]. Apart from the feature of public access, permissionless blockchains have the advantage in quickly establishing a self-organized data management platform to support various decentralized applications (DApps). This is a breakthrough in production relations in that people can independently design smart contracts and freely build decentralized applications themselves without the support or permission from trusted intermediaries. By the PoW-based Nakamoto consensus protocol, people are encouraged to become consensus nodes, i.e., miners, with the mining reward. Unfortunately, solving the PoW puzzle needs continuous, high computing power which mobile devices and IoT devices cannot afford. As the number of mobile phone users is forecasted to reach nearly 5 billion[2] in 2019, it is expected that DApps would usher in explosive growth if mobile devices can join in the mining and consensus process and self-organize a blockchain network to support DApps [10]. To alleviate the computational bottleneck, the consensus nodes can access the cloud/fog computing service [11] to offload their mining tasks, thus enabling blockchain-based DApps. As the cloud/fog computing service can breed more consensus nodes to be able to execute the mining task, it would significantly improve the robustness of the blockchain network and then raise the valuation of DApps, which further attracts more DApp users to join, forming a virtuous circle.

In this paper, we mainly investigate the trading between the cloud/fog computing service provider (CFP) and the computationally lightweight devices, i.e., *miners*. From the system perspective, we aim to maximize the *social welfare* which is the total utility of the CFP and all miners in the blockchain network. The social welfare can be interpreted as the system efficiency [12]. For an efficient and sustainable business ecosystem, there are some critical issues about cloud/fog resources allocation and pricing for the service provider. First, which miner can be offered the computing resources? Too many miners will cause service congestion and incur high operation cost to the service provider. By contrast, a very small group of miners may erode the integrity of the blockchain network. Second, how to set a reasonable service price for miners such that they can be incentivized to undertake the mining tasks? The efficient method is to set up an auction where the miners can actively submit their bids to the CFP for decision making. We should also consider how to make miners truthfully expose their private valuations. A miner's valuation on the computing service is directly related to its privately collected transactional data which determines its expected reward from the blockchain. To address the above questions, we propose an auction-based cloud/fog computing resource market model for blockchain networks. Moreover, we design truthful auction mechanisms for two different bidding schemes. One is the *constant-demand scheme* where the CFP restricts that each miner can bid only for the same quantity of computing resources. The other one is the *multi-demand scheme* where miners can request their demands and express the corresponding bids more freely. The major contributions of this paper can be summarized as follows:

- In the auction-based cloud/fog computing resources market, we take the competition among miners [13] and network effects of blockchain by nature [14] into consideration. We study the auction mechanism with allocative externalities[3] to maximize the social welfare.
- From the perspective of the CFP, we formulate social welfare maximization problems for two bidding schemes: constant-demand scheme and multi-demand scheme. For the constant-demand bidding scheme, we develop an optimal algorithm that achieves optimal social welfare. For the multi-demand bidding scheme, we prove that the formulated problem is NP-hard and equivalent to the problem of non-monotone submodular maximization with knapsack constraints. Therefore, we introduce an approximate algorithm that generates sub-optimal social welfare. Both the algorithms are designed to be truthful, individually rational and computationally efficient.
- Based on the real-world mobile blockchain experiment, we define and verify two characteristic functions for system model formulation. One is the hash power function that describes the relationship between the probability of successfully mining a block and the corresponding miner's computing power. The other one is the network effects function that characterizes the relationship between security of the blockchain network and total computing resources invested into the network.
- Our simulation results show that the proposed auction mechanisms not only help the CFP make practical and efficient computing resource trading strategies, but also offer insightful guidance to the blockchain developer in designing the blockchain protocol.

To the best of our knowledge, this is the first work that investigates resource management and pricing for blockchain networks in the auction-based market. This paper is an extended version of our conference paper [15]. In [15], we considered only the miners with constant demand and did not perform the real-world experiment to verify the network effects function.

1. We refer to all transaction records stored in the block simply as transactional data in the rest of the paper.

2. https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide

3. The allocative externalities occur when the allocation result of the auction affects the valuation of the miners.

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

TABLE 1
Frequently Used Notations

Notation	Description
$\mathcal{N}$, N	Set of miners and the total number of miners
$\mathcal{M}$	Set of winners, i.e., the selected miners by the auction
$\mathbf{d}$, d_i	Miners' service demand profile and miner i's demand for cloud/fog computing resource
$\mathbf{b}$, b_i	Miners' bid profile and miner i's bid for its demand d_i
$\mathbf{x}$, x_i	Resource allocation profile and allocation result for miner i
$\mathbf{p}$, p_i	Price profile and cloud/fog computing service price for miner i
γ_i	Miner i's hash power
T, r	Fixed bonus from mining a new block and the transaction fee rate
s_i	Miner i's block size
λ	Average block time
D	Total supply of computing resources from CFP
$w(\cdot)$	Network effects function
q	Quantity of computing resource required by constant-demand miner
β	Demand constraint ratio for multi-demand miner

The rest of this paper is organized as follows. Section 2 reviews related work. The system model of cloud/fog computing resource market for blockchain networks is introduced in Section 3. Section 4 discusses the constant-demand bidding scheme and the optimal algorithm for social welfare maximization. In Section 5, the approximate algorithm for multi-demand bidding scheme is presented in detail. Experimental results of mobile blockchain and the performance analysis of the proposed auction mechanisms are presented in Section 6. Finally, Section 7 concludes the paper. Table 1 lists notations frequently used in the paper.

2 Related Work

As the core part of the blockchain network, creating blocks integrates the distributed database (i.e., ledger), the consensus protocol and the executable scripts (i.e., smart contract) [16]. From the perspective of data processing, a DApp is essentially developed on the basis of smart contracts and transactional data stored in the blockchain. DApps usually use the distributed ledger to monitor the state/ownership change of the tokenized assets. The execution of smart contracts is driven by the transaction/data change to determine the blockchain state transition autonomously, e.g., the asset re-distribution among the DApp users [3], [16]. With the public blockchain, DApps do not have to rely on a centralized infrastructure and intermediary that supports ledger maintenance and smart contracts execution with dedicated storage and computing resources. Instead, DApp providers adopt the token-based reward mechanisms which incentivize people to undertake the tasks of resource provision and system maintenance. In this way, the functionalities of DApps can be freely activated and realized among transaction issuing/validation, information propagation/storage and consensus participation [16], [17]. Therefore, the public blockchain network is a suitable platform for incentive-driven Distributed Autonomous Organization (DAO) systems. To date, a line of literature study the DAO in wireless networking based on the public blockchain. The authors in [4] established a trading platform for Device-to-Device (D2D) computation offloading based on a dedicated cryptocurrency network. They introduced smart contract-based auctions between neighbor D2D nodes to execute resource offloading and offload the block mining tasks to the cloudlets. The authors in [18] adopted a PoW-based public blockchain as the backbone of a P2P file storage market, where the privacy of different parties in a transaction is enhanced by the techniques such as ring signatures and one-time payment addresses. When identity verification is required for market access granting, e.g., in the scenarios of autonomous network slice brokering [19] and P2P electricity trading [8], the public blockchain can be adapted into consortium blockchain by introducing membership authorizing servers with little modification to the consensus protocols and the smart contract design.

Recently, there have already been some studies on the blockchain network from the point of game theory. The authors in [20] proposed a game-theoretic model where the occurrence of working out the PoW puzzle was modeled as a Poisson process. Since a miner's expected reward largely depends on the block size, each miner's response is to choose a reasonable block size before mining for its optimal expected reward. An analytical Nash equilibrium in a two-player case was discussed. In [21], the authors presented a cooperative game model to investigate the mining pool. In the pool, miners form a coalition to accumulate their computing power for steady rewards. Nevertheless, these works mainly focused on the block mining strategies and paid little attention to the deployment of the blockchain network for developing DApps and corresponding resource allocation problems. As a branch of the game theory, the auction mechanism has been widely used to deal with resource allocation issues in various areas, such as mobile crowdsensing [22], [23], [24], cloud/edge computing [25], [26], and spectrum trading [27]. In [24], the authors proposed incentive mechanisms for efficient mobile task crowdsourcing based on reverse combinatorial auctions. They considered data quality constraints in a linear social welfare maximization problem. The authors in [25] designed optimal and approximate strategy-proof mechanisms to solve the problem of physical machine resource management in clouds. They formulated the problem as a linear integer program. In [26], the authors proposed an auction-based profit maximization model for hierarchical mobile edge computing. Unfortunately, it did not take any economic properties, e.g., incentive compatibility, into account. While guaranteeing the strategyproofness, the authors in [27] investigated the problem of redistributing wireless channels and focused on the social welfare maximization. They not only considered strategyproofness, but also took the channel spatial reusability, channel heterogeneity and bid diversity into account. However, in their combinatorial auction setting, the bidder's requested spectrum bundle is assumed to be always truthful. In fact, none of these works can be directly applied to allocating computing resources for the blockchain mainly due to its unique architecture. In the blockchain network, the allocative externalities [28], [29] should be particularly taken into consideration. For example, besides its own received computing resources, each miner also cares much about the other miners' computing power.

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

In our paper, the social welfare optimization in the multi-demand bidding scheme is proved to be a problem of non-monotone submodular maximization with knapsack constraints, which has not been well studied in auction mechanism design to date. The most closely related papers are [23] and [30] in mobile crowdsourcing. In [23], the authors presented a representative truthful auction mechanism for crowdsourcing tasks. They studied a non-monotone submodular maximization problem without constraints. In [30], the authors formulated a monotone sub-modular function maximization problem when designing a truthful auction mechanism. The total payment to the mobile users is constrained by a fixed budget. Technically, the algorithms in aforementioned works cannot be applied in our models. In addition, the authors in [31] used deep learning to recover the classical optimal auction for revenue maximization and applied it in the edge computing resources allocation in mobile blockchain. However, it only considers one unit of resource in the auction.

3 System Model: Blockchain Mining and Auction Based Market Model

3.1 Cloud/Fog Computing Resource Trading

Our system model is built under the assumptions that 1) the public blockchain network adopts the classical PoW consensus protocol [1], 2) miners do not use their own devices, e.g., computationally lightweight or mobile devices, to execute the mining tasks. We consider a scenario where there is one CFP and a community of miners $\mathcal{N} = \{1, \ldots, N\}$. Each miner runs a blockchain-based DApps to record and verify the transactional data sent to the blockchain network. Due to insufficient energy and computing capacity of their devices, the miners offload the task of solving PoW to nearby cloud/fog computing service which is deployed and maintained by the CFP. To perform the trading, the CFP launches an auction. The CFP first announces auction rules and the available service to miners. Then, the miners submit their resource demand profile $\mathbf{d} = (d_1, \ldots, d_N)$ and corresponding bid profile $\mathbf{b} = (b_1, \ldots, b_N)$ which represents the valuations of their requested resources. After having received miners' demands and bids, the CFP selects the winning miners and notifies all miners the allocation $\mathbf{x} = (x_1, \ldots, x_N)$ and the service price $\mathbf{p} = (p_1, \ldots, p_N)$, i.e., the payment for each miner[4]. We assume that miners are single minded [32], that is, each miner only accepts its requested quantity of resources or none. The setting $x_i = 1$ means that miner i is within the winner list and allocated resources for which it submits the bid, while $x_i = 0$ means no resource allocated. The payment for a miner which fails the auction is set to be zero, i.e., $p_i = 0$ if $x_i = 0$. At the end of the auction, the selected miners or winners make the payment according to the price assigned by the CFP and access the cloud/fog computing service.

3.2 Blockchain Mining with Cloud/Fog Computing Service

With the allocation x_i and demand d_i, miner i's hash power γ_i can be calculated from

$$\gamma_i(\mathbf{d}, \mathbf{x}) = \frac{d_i x_i}{d_{\mathcal{N}}}, \tag{1}$$

which is a linear fractional function. The function depends on other miners' allocated computing resources and satisfies $\sum_{i \in \mathcal{N}} \gamma_i = 1$. $d_{\mathcal{N}} = \sum_{i \in \mathcal{N}} d_i x_i$ is the total quantity of allocated resources. The hash power function $\gamma_i(\mathbf{d}, \mathbf{x})$ is verified by a real-world experiment as presented later in Section 6.

Before executing the miner selection by the auction, each miner has collected unconfirmed transactional data into its own block. We denote each miner's *block size*, i.e., the total size of transactional data and metadata, by $\mathbf{s} = (s_1, \ldots, s_N)$. In the mining tournament, the generation of new blocks follows a Poisson process with a constant mean rate $\frac{1}{\lambda}$ throughout the whole blockchain network [33]. λ is also known as the *average block time*. If the miner i finds a new block, the time for propagation and verification of transactions in the block is dominantly affected by s_i. The first miner which successfully has its block reach consensus can receive a *token reward* R. The token reward is composed of a *fixed bonus* $T \geq 0$ for mining a new block and a variable transaction fee $t_i = rs_i$ determined by miner i's block size s_i and a predefined *transaction fee rate* r [20]. Thus, miner i's token reward R_i can be expressed as follows:

$$R_i = (T + rs_i)\mathbb{P}_i(\gamma_i(\mathbf{d}, \mathbf{x}), s_i), \tag{2}$$

where $\mathbb{P}_i(\gamma_i(\mathbf{d}, \mathbf{x}), s_i)$ is the probability that miner i receives the reward for contributing a block to the blockchain.

We note that obtaining the reward rests with successful mining and instant propagation. Miner i's probability of discovering the nonce value P_i^m is equal to its hash power γ_i, i.e., $P_i^m = \gamma_i$. However, a lucky miner may even lose the tournament if its broadcast block is not accepted by other miners at once, i.e., failing to reach consensus. The newly mined block that cannot be added onto the blockchain is called *orphan block* [20]. A larger block needs more propagation and verification time, thus resulting in larger delay in reaching consensus. As such, a larger block size means a higher chance that the block suffers orphaned. According to the statistics displayed in [34], miner i's block propagation time τ_i is linear to the block size, i.e., $\tau_i = \xi s_i$. ξ is a constant that reflects the impact of s_i on τ_i. Since the arrival rate of new blocks follows the Poisson distribution, miner i's orphaning probability is:

$$P_i^o = 1 - \mathrm{e}^{-\frac{1}{\lambda}\tau_i}. \tag{3}$$

Substituting τ_i, we can express $\mathbb{P}_i$ as follows:

$$\mathbb{P}_i(\gamma_i(\mathbf{d}, \mathbf{x}), s_i) = P_i^m(1 - P_i^o) = \gamma_i \mathrm{e}^{-\frac{1}{\lambda}\xi s_i}. \tag{4}$$

3.3 Business Ecosystem for Blockchain Based DApps

Here, we describe the business ecosystem for blockchain based DApps in Fig. 1. In developing a blockchain based DApps, there exists a blockchain developer which is responsible for designing or adopting the blockchain operation protocol. The developer specifies the fixed bonus T, the transaction fee rate r and so on. Through adjusting the difficulty of finding the new nonce, the blockchain developer

4. Throughout this paper, the terms price and payment are used interchangeably.

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

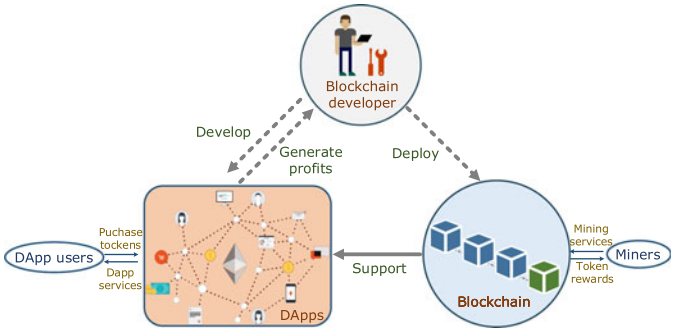


Fig. 1. Business ecosystem for blockchain based DApps.

keeps the average block time λ at a reasonable constant value. To support the DApps, in the deployed blockchain network, miners perform mining and token reward, i.e., R, is used to incentivize them. The reward may come from the token that DApps users pay to the blockchain network.

When bidding for computing resources, miners always evaluate the value of the tokens. In fact, the intrinsic value of tokens depends on the trustworthiness and robustness, i.e., the value of the blockchain network itself. From the perspective of trustworthiness, the PoW-based blockchain is only as secure as the amount of computing power dedicated to mining tasks [14]. This results in positive network effects [14] in that as more miners participate and more computing resources are invested, the security of the blockchain network is improved, and hence the value of a reward given to miners increases. A straightforward example is that if the robustness of the blockchain network is very low, i.e., vulnerable to manipulation (51 percent attack, double-spending attack, etc.), that means this blockchain is insecure and cannot support any decentralized application effectively. Naturally, this blockchain network losses its value and its distributed tokens (including the rewards to miners) would be worthless. On the contrary, if there are many miners and computing resources invested, the blockchain would be more reliable and secure [35]. Thus, users would trust it more and like to use its supported decentralized applications through purchasing the tokens and then miners would also gain more valuation on their received tokens (reward). To confirm this fact, we conduct a real-world experiment (see Section 6.1) to evaluate the value of the tokens and the reward by examining the impact of the total computing power on preventing double-spending attacks. By curve fitting of the experimental data, we define the network effects by a non-negative utility function as follows:

$$w(\pi) = a_1\pi - a_2\pi e^{a_3\pi}, \tag{5}$$

where $\pi = \frac{d_{\mathcal{N}}}{D} \in [0,1]$ is the normalized total computing power of the blockchain network. $d_{\mathcal{N}} = \sum_{i\in\mathcal{N}} d_i x_i$ is the total quantity of allocated computing resources, and D is the maximum quantity that CFP can supply. $a_1, a_2, a_3 > 0$ are curve fitting parameters and this network effects function in the feasible domain is monotonically increasing with a diminishing return.

3.4 Miner's Valuation on Cloud/Fog Computing Resources

In the auction, a miner's bid represents the valuation of computing resources for which it demands. Since miner i cannot know the number of winning miners and the total quantity of allocated resources until the end of auction, we assume that miner i can only give the bid b_i according to its expected reward R_i and demand d_i without considering network effects and other miners' demands, i.e., setting $w(d_{\mathcal{N}}) = 1$ and $\sum_{j\in\mathcal{N}\setminus\{i\}} d_j x_j = 0$. In other words, miner i has an *ex-ante* valuation v_i' which can be written as ($P_i^m = \gamma_i = 1$)

$$v_i' = R_i d_i = (T + rs_i)e^{-\frac{1}{\lambda}\xi s_i} d_i. \tag{6}$$

Here, we assume that R_i represents the miner i's valuation for one unit computing resource and d_i is decided according to miner i's own available budget. Since our proposed auction mechanisms are truthful (to be proved later), b_i is equal to the true *ex-ante* valuation v_i', i.e., $b_i = v_i'$.

After the auction is completed, miners receive the allocation result, i.e., $\mathbf{x}$, and are able to evaluate the network effects. Hereby, miner i has an *ex-post* valuation v_i'' as follows:

$$\begin{aligned} v_i'' &= v_i' w(\pi)\gamma_i(\mathbf{d},\mathbf{x}) \\ &= \frac{d_i^2 x_i}{d_{\mathcal{N}}}(a_1\pi - a_2\pi e^{a_3\pi})(T + rs_i)e^{-\frac{1}{\lambda}\xi s_i} \\ &= \frac{d_i^2 x_i}{D}\left(a_1 - a_2 e^{a_3\frac{d_{\mathcal{N}}}{D}}\right)(T + rs_i)e^{-\frac{1}{\lambda}\xi s_i}. \end{aligned} \tag{7}$$

3.5 Social Welfare Maximization

The CFP selects winning miners, i.e., winners, and determines corresponding prices in order to maximize the social welfare. Let c denote the unit cost of running the cloud/fog computing service, so the total cost to the CFP can be expressed by $C(d_{\mathcal{N}}) = cd_{\mathcal{N}} = \sum_{i\in\mathcal{N}} cd_i x_i$. Thus, we define the social welfare of the blockchain network S as the difference between the sum of all miners' ex-post valuations and the CFP's total cost, i.e.,

$$\begin{aligned} S(\mathbf{x}) &= \sum_{i\in\mathcal{N}} v_i'' - C(d_{\mathcal{N}}) \\ &= \sum_{i\in\mathcal{N}} \frac{d_i^2 x_i}{D}\left(a_1 - a_2 e^{a_3\frac{d_{\mathcal{N}}}{D}}\right)(T + rs_i)e^{-\frac{1}{\lambda}\xi s_i} \\ &\quad - cd_{\mathcal{N}}. \end{aligned} \tag{8}$$

Therefore, the primary objective of designing the auction mechanism is to solve the following integer programming:

$$\begin{aligned} \max_{\mathbf{x}} \quad S(\mathbf{x}) = \sum_{i\in\mathcal{N}} \Bigg(& \frac{d_i^2 x_i}{D}\left(a_1 - a_2 e^{\frac{a_3}{D}\sum_{i\in\mathcal{N}} d_i x_i}\right) \\ & (T + rs_i)e^{-\frac{1}{\lambda}\xi s_i}\Bigg) - \sum_{i\in\mathcal{N}} cd_i x_i, \end{aligned} \tag{9}$$

$$\text{s.t.} \quad \sum_{i\in\mathcal{N}} d_i x_i \le D, \tag{10}$$

$$x_i \in \{0,1\}, \forall i \in \mathcal{N}, \tag{11}$$

where (10) is the constraint on the quantity of computing resources that the CFP can offer. In the next two sections, we consider two types of bidding schemes in the auction

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

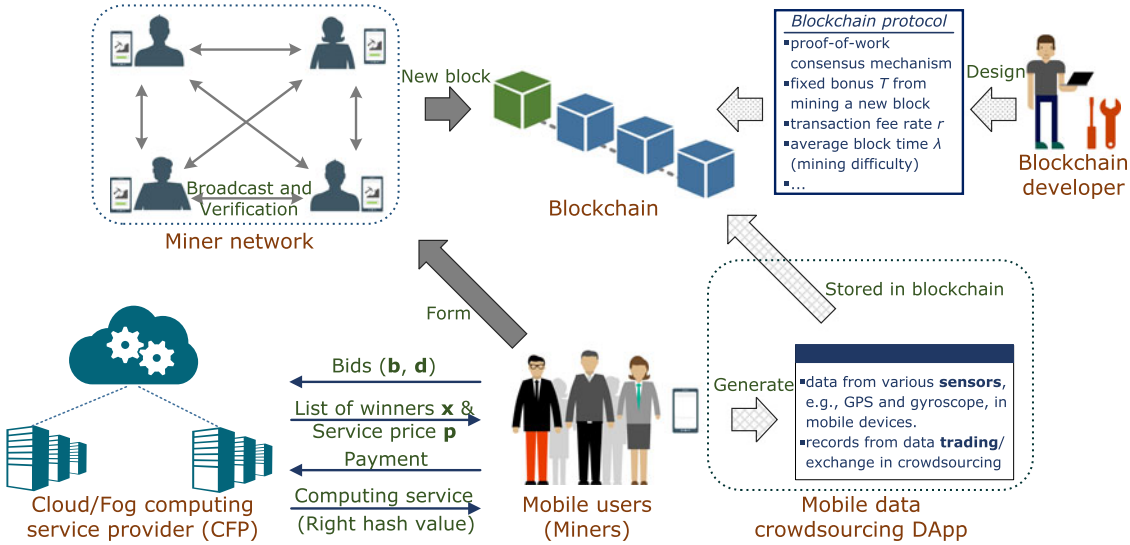


Fig. 2. An example mobile data crowdsourcing application illustrating the system model and the cloud/fog computing resource market for blockchain networks.

design: constant-demand bidding scheme and multi-demand bidding scheme. Accordingly, there are two types of miners: constant-demand miners and multi-demand miners. We aim to maximize the social welfare, while guaranteeing the truthfulness, individual rationality, and computational efficiency.

3.6 Example Application: Mobile Data Crowdsourcing

As shown in Fig. 2, we take an example of mobile data crowdsourcing to illustrate the use of our model and and to demonstrate an effectiveness of the related concepts. Initially, there are a group of mobile users. Each of the mobile users can be either a worker that collects data from the sensors in its mobile device or a requester that wants to buy the sensing data from other users (workers). However, there is often no trusted or authorized crowdsourcing platform to process the data trading and record the transactions. Moreover, no mobile user has enough trust, right, or capability to establish and operate such a centralized platform. In this case, a viable solution is to design and deploy a blockchain based crowdsourcing DApp by a blockchain developer. Based on the designed protocol, the mobile users can utilize the available cloud/fog computing resources to self-organize a reliable blockchain network. Thus, their data trading activities can be facilitated by the established decentralized crowdsourcing platform with smart contracts.

The blockchain developer adopts the PoW protocol and sets the parameters, such as the fixed reward T, the transaction fee rate r and the average block time λ. Due to limited energy and computational capability, mobile users (miners) need to buy computing resources from the CFP through an auction process and then join the miner network. Before the auction begins, miner i may possess a certain amount of data to be stored in the blockchain and knows its block size s_i. According to (6), the miner i will evaluate its expected reward and the ex-ante value v_i' of the computing resources based on the protocol parameters, its block size and demand. Next, the miner i submits the bid b_i and the demand d_i to the CFP. Using our proposed auction algorithm, the CFP can select the winning miners, i.e., the allocation x_i, and determine the price p_i to maximize the social welfare. Meanwhile, it can guarantee the miner's truthfulness and non-negative utility which is the difference between the ex-post valuation v_i'' and its payment p_i. Once the auction ends, the winning miners which are allocated the computing resources form a miner network. With the CFP service in solving the PoW puzzle and calculating the hash values, the winning miners can start the mining and consensus process to verify and contribute new blocks containing the crowdsourced data and corresponding transaction records to the blockchain. For more details about the blockchain-based crowdsourcing, please refer to [36].

4 Auction-Based Mechanism for Constant-Demand Miners

In this section, we first consider a simple case where all miners submit bids for the same quantity of computing resources. Here, each miner's demand is q units, i.e., $d_i = q \in (0, D), \forall i \in \mathcal{N}$. Thus, the optimization problem for the CFP can be expressed as follows:

$$\max_{\mathbf{x}} \quad S(\mathbf{x}) = \sum_{i\in\mathcal{N}} \left(\frac{q^2 x_i}{D} \left(a_1 - a_2 \mathrm{e}^{\frac{a_3}{D}\sum_{i\in\mathcal{N}} q x_i} \right) (T + r s_i) \mathrm{e}^{-\frac{1}{\lambda} c s_i} \right) - \sum_{i\in\mathcal{N}} c q x_i, \tag{12}$$

$$\text{s.t.} \quad \sum_{i\in\mathcal{N}} q x_i \le D, \tag{13}$$

$$x_i \in \{0, 1\}, \forall i \in \mathcal{N}. \tag{14}$$

The first proposed truthful auction for Constant-Demand miners in Blockchain networks (CDB auction), as presented

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

in Algorithm 1, is an optimal one and its rationale is based on the well-known Myerson's characterization [37] provided in Theorem 1.

Algorithm 1. CDB Auction

Input: Miners' bid profile **b** and demand profile **d**.
Output: Resource allocation **x** and service price **p**.

1: **begin**
2: **for each** $i \in \mathcal{N}$ **do**
3: $x_i \leftarrow 0, p_i \leftarrow 0$
4: **end for**
5: Sort bids **b** in descending order.
6: $j \leftarrow \arg\max_{j \in \mathcal{N}} b_j$
7: $\mathcal{M} \leftarrow \{j\}, S \leftarrow \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q}{D}}\right) b_j - cq$
8: **while** $\mathcal{M} \neq \mathcal{N}$ **and** $|\mathcal{M}| \leq D$ **do**
9: $j \leftarrow \arg\max_{j \in \mathcal{N} \setminus \mathcal{M}} b_j$
10: $\mathcal{M}_t \leftarrow \mathcal{M} \cup \{j\}$
11: $S_t \leftarrow \sum_{i \in \mathcal{M}_t} \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{a_3 q |\mathcal{M}_t|}\right) b_i - cq|\mathcal{M}_t|$
12: **if** $S_t < S$ **or** $S_t < 0$ **then**
13: **break**
14: **end if**
15: $\mathcal{M} \leftarrow \mathcal{M} \cup \{j\}$
16: **end while**
17: **for each**$i \in \mathcal{M}$ **do**
18: $x_i \leftarrow 1, \mathcal{N}_{-i} \leftarrow \mathcal{N} \setminus \{i\}, \mathcal{M}_{-i} \leftarrow \mathcal{M} \setminus \{i\}$
19: $j \leftarrow \arg\max_{j \in \mathcal{N}_{-i}} b_j$
20: $\mathcal{M}' \leftarrow \{j\}, S' \leftarrow \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q |\mathcal{M}'|}{D}}\right) b_j - cq$
21: **while** $\mathcal{M}' \neq \mathcal{N}$ **and** $|\mathcal{M}'| \leq D$ **do**
22: $j \leftarrow \arg\max_{i \in \mathcal{N}_{-i} \setminus \mathcal{M}'} b_j$
23: $\mathcal{M}'_t \leftarrow \mathcal{M}' \cup \{j\}$
24: $S'_t \leftarrow \sum_{i \in \mathcal{M}'_t} \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q |\mathcal{M}'_t|}{D}}\right) b_i - cq|\mathcal{M}'_t|$
25: **if** $S'_t < S'$ **or** $S'_t < 0$ **then**
26: **break**
27: **end if**
28: $\mathcal{M}' \leftarrow \mathcal{M}'_t, S' \leftarrow S'_t$
29: **end while**
30: $p_i = S' - \sum_{i \in \mathcal{M}_{-i}} \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q |\mathcal{M}_{-i}|}{D}}\right) b_i - cq|\mathcal{M}_{-i}|$
31: **end for**
32: **End**

Theorem 1 ([32, Theorem 13.6]). *An auction mechanism is truthful if and only if it satisfies the following two properties:*

1) *Monotonicity: If miner i wins the auction with bid b_i, then it will also win with any higher bid $b'_i > b_i$.*
2) *Critical payment: The payment by a winner is the smallest value needed in order to win the auction.*

As illustrated in Algorithm 1, the CDB auction consists of two consecutive processes: winner selection (lines 5-16) and service price calculation (lines 17-31). The winner selection process is implemented with a greedy method. For the convenience of later discussion, we define a set of winners as $\mathcal{M}$. Adding a miner i in $\mathcal{M}$ means setting $x_i = 1$. Thus, we transform the original problem in (12), (13), and (14) to an equivalent set function form as follows:

$$\max_{\mathcal{M} \subseteq \mathcal{N}} \quad S(\mathcal{M}) = \sum_{i \in \mathcal{M}} \left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q |\mathcal{M}|}{D}}\right) \frac{q b_i}{D} - cq|\mathcal{M}|, \tag{15}$$

$$\text{s.t.} \quad q|\mathcal{M}| \leq D, \tag{16}$$

where $|\mathcal{M}|$ represents the cardinality of set $\mathcal{M}$ which is the number of winners in $\mathcal{M}$ and $b_i = v'_i = (T + rs_i)\mathrm{e}^{-\frac{1}{\lambda} \xi s_i} q$. In the winner selection process (lines 5-11), miners are first sorted in a descending order according to their bids. Then, they are sequentially added to the set of winners $\mathcal{M}$ until the social welfare $S(\mathcal{M})$ begins to decrease. Finally, the set of winners $\mathcal{M}$ and the allocation **x** are output by the algorithm.

Proposition 1. *The resource allocation* **x** *output by Algorithm 1 is globally optimal to the social welfare maximization problem given in (12), (13), and (14).*

Proof. With the proof by contradiction, this result follows from Claim 1. □

Claim 1. Let $\mathcal{M}_A$ be the solution output by Algorithm 1 on input **b**, and $\mathcal{M}_O$ be the optimal solution. If $\mathcal{M}_A \neq \mathcal{M}_O$, then we can construct another solution $\mathcal{M}^*_O$ whose social welfare $S(\mathcal{M}^*_O)$ is even larger than the optimal social welfare $S(\mathcal{M}_O)$.

Proof. We assume $b_1 \geq \cdots \geq b_N$ and $\mathcal{M}_A \neq \mathcal{M}_O$. Next, we consider two cases.

1) Case 1: $\mathcal{M}_O \subset \mathcal{M}_A$. According to Algorithm 1, it is obvious that we can construct a solution $\mathcal{M}^*_O$ with higher social welfare by adding a member from $\mathcal{M}_A$ to $\mathcal{M}_O$.
2) Case 2: $\mathcal{M}_O \not\subset \mathcal{M}_A$. Let m be the first element (while-loop lines 7-14) that $m \notin \mathcal{M}_O$. Since m is maximal (b_m is minimal by assumption), we have $1, \ldots, m-1 \in \mathcal{M}_O$ and the corresponding set of winning bids $\mathbf{b}_{\mathcal{M}_O} = \{b_1, \ldots, b_{m-1}, b'_m, b'_{m+1}, \ldots, b'_{|\mathcal{M}_O|}\}$, where the bids $\{b_1, \ldots, b'_{|\mathcal{M}_O|}\}$ are listed in the descending order. Meanwhile, Algorithm 1 chooses $\mathbf{b}_{\mathcal{W}_A} = \{b_1, \ldots, b_{m-1}, b_m, b_{m+1}, \ldots, b_{|\mathcal{M}_A|}\}$ and there must be $b_m > b'_j$ for all $j \geq m$. In particular, we have $b_m > b'_m$. Hence, we define $\mathbf{b}_{\mathcal{M}^*_O} = \mathbf{b}_{\mathcal{M}_O} \cup \{b_m\} \setminus \{b'_m\}$, i.e., we obtain $\mathbf{b}_{\mathcal{M}^*_O}$ by removing b'_m and adding b_m to $\mathbf{b}_{\mathcal{M}_O}$. Thus, the social welfare of $\mathbf{b}_{\mathcal{W}^*_O}$ is calculated as follows:

$$S(\mathcal{M}^*_O) = S(\mathcal{M}_O) + \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q |\mathcal{M}|}{D}}\right)(b_m - b'_m).$$

As $b_m - b'_m > 0$, $(a_1 - a_2 \mathrm{e}^{\frac{a_3 q |\mathcal{M}|}{D}}) \frac{q}{D} > 0$ and $|\mathcal{M}^*_O| = |\mathcal{M}_O|$, $S(\mathcal{M}^*_O)$ is strictly larger than $S(\mathcal{M}_O)$. This is in contradiction to that $\mathcal{M}_O$ is the optimal solution and thus proves the claim. □

We apply Vickrey–Clarke–Groves (VCG) mechanism [38] in the service price calculation. In lines 16-30, for each iteration, we exclude one selected miner from the set of winners and re-execute the winner selection process to calculate the social cost of the miner as its payment. The VCG-based payment function is defined as follows:

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

$$p_i = S(\mathcal{M}_{\mathcal{N}\setminus\{i\}}) - S(\mathcal{M}_{\mathcal{N}} \setminus \{i\}), \tag{17}$$

where $S(\mathcal{M}_{\mathcal{N}\setminus\{i\}})$ is the optimal social welfare obtained when the selected miner i is excluded from the miner set $\mathcal{N}$, and $S(\mathcal{M}_{\mathcal{N}} \setminus \{i\})$ is the social welfare of the set of winners which is obtained by removing miner i from the optimal winner set selected from $\mathcal{N}$.

Proposition 2. *The CDB auction (Algorithm 1) is truthful.*

Proof. Since the payment calculation in the algorithm relies on the VCG mechanism, it directly satisfies the second condition in Theorem 1 [32]. For the first condition about monotonicity in Theorem 1, we need to show that if a winning miner i raises its bid from b_i to b_i^+ where $b_i^+ > b_i$, it still stays in the winner set. We denote the original winner set by $\mathcal{M}$ and the new winner set by $\mathcal{M}_+$ after miner i changes its bid to b_i^+. The original set of bids is $\mathbf{b} = \{b_1, \ldots, b_i, \ldots, b_N\} (i \leq |\mathcal{M}|)$ sorted in the descending order. In addition, we define $S(\mathbf{b}_{\mathcal{K}}) = S(\mathcal{K}), \forall \mathcal{K} \subseteq \mathcal{N}$ which means the social welfare of a set of bids is equal to that of the set of corresponding miners. We discuss the monotonicity in two cases.

1) Case 1: $b_{i-1} \geq b_i^+ \geq b_i \geq b_{i+1}$. The new set of ordered bids is $\mathbf{b}^+ = \{b_1, \ldots, b_{i-1}, b_i^+, b_{i+1}, \ldots, b_N\}$. We have

$$\begin{aligned} S(\{b_1, \ldots, b_i^+\}) &= \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q i}{D}}\right)\left(\sum_{j=1}^{i-1} b_j + b_i^+\right) - cqi \\ &> S(\{b_1, \ldots, b_i\}) = \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q i}{D}}\right)\sum_{j=1}^{i} b_j - cqi. \end{aligned} \tag{18}$$

The social welfare of the new set of bids $\{b_1, \ldots, b_i^+\}$ is larger than that of the original set of bids $\{b_1, \ldots, b_i\}$, which guarantees b_i^+ being in the set of winning bids.

2) Case 2: $b_{k-1} \geq b_i^+ \geq b_k \geq \cdots \geq b_i$, $1 < k < i$. The new set of ordered bids is $\mathbf{b}^+ = \{b_1, \ldots, b_{k-1}, b_i^+, b_k, \ldots, b_{i+1}, \ldots, b_N\}$. We have

$$\begin{aligned} S(\{b_1, \ldots, b_{k-1}, b_i^+\}) =& \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q k}{D}}\right)\left(\sum_{j=1}^{k-1} b_j + b_i^+\right) \\ &- cqk, \end{aligned} \tag{19}$$

$$S(\{b_1, \ldots, b_{k-1}, b_k\}) = \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q k}{D}}\right)\sum_{j=1}^{k} b_j - cqk, \tag{20}$$

$$\begin{aligned} &S(\{b_1, \ldots, b_{k-1}\}) \\ &= \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q(k-1)}{D}}\right)\sum_{j=1}^{k-1} b_j - cq(k-1). \end{aligned} \tag{21}$$

As the coefficient $\frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q|\mathcal{M}|}{D}}\right)$ in $S(\mathcal{M})$ is a monotonically decreasing function of $\mathcal{M}$, increasing b_i may change the set of winners $\mathcal{M}$ and reduce the number of winning miners. However, the first i bids $\{b_1, \ldots, b_{k-1}, b_k, \ldots, b_i\}$ in the original set of bids $\mathbf{b}$ have already won the auction, so we have $S(\{b_1, \ldots, b_{k-1}, b_k\}) > S(\{b_1, \ldots, b_{k-1}\})$. From the following inequation (22),

$$\begin{aligned} S(\{b_1, \ldots, b_{k-1}, b_k\}) &= \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q k}{D}}\right)\left(\sum_{j=1}^{k-1} b_j + b_k\right) \\ &< \frac{q}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 q k}{D}}\right)\left(\sum_{j=1}^{k-1} b_j + b_i^+\right) = S(\{b_1, \ldots, b_{k-1}, b_i^+\}), \end{aligned} \tag{22}$$

the proof can be finally concluded by

$$S(\{b_1, \ldots, b_{k-1}, b_i^+\}) > S(\{b_1, \ldots, b_{k-1}\}), \tag{23}$$

which implies that b_i^+ still remains the bid of a winner in the auction. □

Proposition 3. The CDB auction (Algorithm 1) is computationally efficient and individually rational.

Proof. Sorting the bids has the complexity of $O(N\log N)$. Since the number of winners is at most $\min(\frac{D}{q}, N)$, the time complexity of the winner selection process (while-loop, lines 7-15) is $O(\min^2(\frac{D}{q}, N))$. In each iteration of the payment calculation process (lines 16-30), a similar winner selection process is executed. Therefore, the whole auction process can be performed in polynomial time with the time complexity of $O(\min^3(\frac{D}{q}, N) + N\log N)$.

According to Proposition 1 and the properties of the VCG mechanism [38], the payment scheme in Algorithm 1 guarantees the individual rationality. □

5 Auction-based Mechanisms for Multi-Demand Miners

In this section, we investigate a more general scenario where miners request multiple demands of cloud/fog computing resources.

5.1 Social Welfare Maximization for the Blockchain Network

We first investigate the winner selection problem defined in (9), (10), and (11) from the perspective of an optimization problem. Evidently, it is a nonlinear integer programming problem with linear constraints, which is NP-hard to obtain the optimal solution. Naturally, we can find an approximate method with a lower bound guarantee. Similar to Section 4, the original problem is rewritten as a subset function form

$$\begin{aligned} \max_{\mathcal{M}\subseteq\mathcal{N}} \quad & S(\mathcal{M}) = \sum_{i\in\mathcal{M}} \frac{d_i}{D}\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{i\in\mathcal{M}} d_i}{D}}\right) b_i \\ & - c\sum_{i\in\mathcal{M}} d_i, \end{aligned} \tag{24}$$

$$s.t. \quad \sum_{i\in\mathcal{M}} d_i \leq D, \tag{25}$$

where $S(\mathcal{M})$ is the social welfare function of the selected set of winners $\mathcal{M}$ and $b_i = v_i' = (T + rs_i)\mathrm{e}^{-\frac{1}{\lambda}\xi_{s_i}} d_i$. This form means that we can view it as a subset sum problem [39]. We assume that there is at least one miner i such that

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

$S(\{i\}) > 0$. Additionally, although the miners can submit demands that they want instead of the same constant quantity of computing resources, it is reasonable to assume that the CFP puts a restriction on the purchase quantity, i.e., $\beta_1 D < d_i \leq \beta_2 D$, where $\beta_1 D$, $\beta_2 D$ are respectively the lower and upper limit on each miner's demand, and $0 < \beta_1 < \beta_2 < 1$ are predetermined demand constraint ratios. Clearly, $S(\emptyset) = 0$.

Definition 1 (Submodular Function [40]). *Let $\mathcal{X}$ be a finite set. A function $f : 2^{\mathcal{X}} \to \mathbb{R}$ is submodular if*

$$f(\mathcal{A} \cup \{x\}) - f(\mathcal{A}) \geq f(\mathcal{B} \cup \{x\}) - f(\mathcal{B}), \tag{26}$$

for any $\mathcal{A} \subseteq \mathcal{B} \subseteq \mathcal{X}$ and $x \in \mathcal{X} \setminus \mathcal{B}$, where $\mathbb{R}$ is the set of reals. A useful equivalent definition is that f is submodular if and only if the derived set-function

$$f_x(\mathcal{A}) = f(\mathcal{A} \cup \{x\}) - f(\mathcal{A}) \qquad (\mathcal{A} \subseteq \mathcal{X} \setminus \{x\}), \tag{27}$$

is monotonically decreasing for all $x \in \mathcal{X}$.

Proposition 4. *The social welfare function $S(\mathcal{M})$ in (24) is submodular.*

Proof. By Definition 1, we need to show that $S_u(\mathcal{M})$ in (30) is monotonically decreasing, for every $\mathcal{M} \subseteq \mathcal{N}$ and $u \in \mathcal{N} \setminus \mathcal{M}$.

$$S_u(\mathcal{M}) = S(\mathcal{M} \cup \{u\}) - S(\mathcal{M}) \tag{28}$$

$$\begin{aligned} &= \sum_{i \in \mathcal{M} \cup \{u\}} \frac{d_i}{D} \left(a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{i \in \mathcal{M} \cup \{u\}} d_i}{D}} \right) b_i \\ &\quad - \sum_{i \in \mathcal{M}} \frac{d_i}{D} \left(a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{i \in \mathcal{M}} d_i}{D}} \right) b_i - c d_u \end{aligned} \tag{29}$$

$$\begin{aligned} &= \underbrace{\left(\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{i \in \mathcal{M} \cup \{u\}} d_i}{D}} \right) - \left(a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{i \in \mathcal{M}} d_i}{D}} \right) \right) \sum_{i \in \mathcal{M}} \frac{d_i b_i}{D}}_{①} \\ &\quad + \underbrace{\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{i \in \mathcal{M} \cup \{u\}} d_i}{D}} \right) \frac{d_u b_u}{D} - c d_u}_{②} \end{aligned} \tag{30}$$

Let $g(z) = a_1 - a_2 \mathrm{e}^{\frac{a_3}{D} z}$, where $z \in \mathbb{R}^+$. Then, the first derivative and second derivative of $g(z)$ are expressed respectively as follows:

$$\frac{\mathrm{d}g(z)}{\mathrm{d}z} = -\frac{a_2 a_3}{D} \mathrm{e}^{\frac{a_3}{D} z}, \frac{\mathrm{d}^2 g(z)}{\mathrm{d}z^2} = -\frac{a_2 a_3^2}{D^2} \mathrm{e}^{\frac{a_3}{D} z}. \tag{31}$$

Because $a_2, a_3, D > 0$, we have $-\frac{a_2 a_3}{D} \mathrm{e}^{\frac{a_3}{D} z} < 0$ and $-\frac{a_2 a_3^2}{D^2} \mathrm{e}^{\frac{a_3}{D} z} < 0$, which indicates that $g(z)$ is monotonically decreasing and concave.

Next, we discuss the monotonicity of $S_u(\mathcal{M})$ in (30). Note that expanding $\mathcal{M}$ means increasing the total quantity of allocated resources $d_{\mathcal{M}} = \sum_{i \in \mathcal{M}} d_i$. Substituting $z = d_{\mathcal{M}}$ and $z = d_{\mathcal{M} \cup \{u\}}$ into $g(z)$, we observe that $g(d_{\mathcal{M} \cup \{u\}}) - g(d_{\mathcal{M}}) = g(\sum_{i \in \mathcal{M} \cup \{u\}} d_i) - g(\sum_{i \in \mathcal{M}} d_i) = (a_1 - a_2 \mathrm{e}^{\frac{a_3}{D} \sum_{i \in \mathcal{M} \cup \{u\}} d_i}) - (a_1 - a_2 \mathrm{e}^{\frac{a_3}{D} \sum_{i \in \mathcal{M}} d_i}) < 0$ is decreasing and negative due to $d_{\mathcal{M}} < d_{\mathcal{M} \cup \{u\}}$ and the monotonicity and concavity of $g(z)$. Additionally, it is clear that when $\mathcal{M}$ expands, $\sum_{i \in \mathcal{M}} d_i b_i > 0$ is positive and increasing. Therefore, ① in (30) is proved to be monotonically decreasing. Because $g(z)$ is monotonically decreasing, it is straightforward to see that ② in (30) is also monotonically decreasing with the expansion of $\mathcal{M}$. Finally, we can conclude that $S_u(\mathcal{M})$ is monotonically decreasing, thus proving the submodularity of $S(\mathcal{M})$. □

It is worth noting that there is a constraint in (10), also called a knapsack constraint. This constraint not only affects the resulting social welfare and the number of the selected miners in the auction, but also needs a careful auction mechanism design to guarantee the truthfulness. Essentially, the optimization problem appears to be a *non-monotone submodular maximization with knapsack constraints*. It is known that there is a $(0.2 - \eta)$-approximate algorithm which applies the fractional relaxation and local search method [41, Figure 5]. $\eta > 0$ is a preset constant parameter that specifies the approximation ratio $(0.2\text{-}\eta)$. For the ease of expression, we name this approximate algorithm as FRLS algorithm. In general, the FRLS algorithm first solves a linear relaxation of the original integer problem using local search, and then it rounds the obtained fractional solution to an integer value. However, the algorithm requires the objective function to be non-negative. To address this issue, let $H(\mathcal{M}) = S(\mathcal{M}) + c \sum_{i \in \mathcal{N}} d_i$. Clearly, $H(\mathcal{M}) \geq 0$ for any $\mathcal{M} \subseteq \mathcal{N}$ and it remains submodular since $c \sum_{i \in \mathcal{N}} d_i$ is a constant. Additionally, maximizing $S(\mathcal{M})$ is equivalent to maximizing $H(\mathcal{M})$. Hence, we attempt to design the FRLS auction which selects the winner based on the FRLS algorithm and let service price $p_i = b_i$. As to the specific input to the FRLS algorithm, it takes 1 as the number of knapsack constraints, the normalized demand profile $\frac{\mathbf{d}}{D}$ as its knapsack weights parameter, η as the approximate degree, and $H(\mathcal{M})$ as the value oracle which allows querying for function values of any given set. The FRLS auction is computationally efficient, as the running time of the FRLS algorithm is polynomial [41]. Furthermore, miners just need to pay their submitted bids to the CFP and cannot suffer deficit, so the FRLS auction also satisfies the individual rationality requirement. However, we find that FRLS auction cannot guarantee truthfulness. The corresponding proof is omitted due to space constraints.

5.2 Multi-Demand Miners in Blockchain Networks (MDB) Auction

Although the FRLS auction is capable solving the social welfare maximization problem approximately, it is not realistic to be directly applied in a real market since it cannot prevent the manipulation of bids by bidders, i.e., lacking of truthfulness. As mentioned before, we aim to design an auction mechanism that not only achieves a good social welfare, but also possesses the desired properties, including computational efficiency, individual rationality and truthfulness. Therefore, we present a novel auction mechanism for Multi-Demand miners in Blockchain networks (MDB auction). In this auction, the bidders are limited to be single-minded in the combinatorial auctions. That is, we can assume safely that the mechanism always allocates to the

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

winner i exactly the d_i items that it requested and never allocates anything to a losing bidder. The design rationale of the MDB auction relies on Theorem 2.

Theorem 2. *([42]) In the multi-unit and single minded setting, an auction mechanism is truthful if it satisfies the following two properties:*

1) *Monotonicity: If a bidder i wins with bid (d_i, b_i), then it will also win with any bid which offers at least as much price for at most as many items. That is, bidder i will still win if the other bidders do not change their bids and bidder i changes its bid to some (d_i', b_i') with $d_i' \leq d_i$ and $b_i' \geq b_i$.*
2) *Critical payment: The payment of a winning bid (d_i, b_i) by bidder i is the smallest value needed in order to win d_i items, i.e., the infimum of b_i' such that (d_i, b_i') is still a winning bid, when the other bidders do not change their bids.*

5.2.1 Auction Design

Before presenting the MDB auction, we first introduce the *marginal social welfare density*. It is the density of miner i's marginal social welfare contribution to the existing set of winners $\mathcal{M}$, which is defined as follows:

$$\begin{aligned} S_i'(\mathcal{M}) &= \frac{S_i(\mathcal{M})}{d_i} = \frac{S(\mathcal{M}\cup\{i\}) - S(\mathcal{M})}{d_i} \\ &= \underbrace{\frac{\left(a_2 \mathrm{e}^{\frac{a_3 \sum_{j\in\mathcal{M}} d_j}{D}} - a_2 \mathrm{e}^{\frac{a_3 \sum_{j\in\mathcal{M}\cup\{i\}} d_j}{D}}\right) \sum_{j\in\mathcal{M}} d_j b_j}{D d_i}}_{①} \\ &\quad + \underbrace{\left(a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{j\in\mathcal{M}\cup\{i\}} d_j}{D}}\right) \frac{b_i}{D} - c}_{②}. \end{aligned} \tag{32}$$

For the sake of brevity, we simply call it *density*.

As illustrated in Algorithm 2, the MDB auction allocates computing resources to miners in a greedy way. According to the density, all miners are sorted in a non-increasing order:

$$S_1'(\mathcal{M}_0) \geq S_2'(\mathcal{M}_1) \geq \cdots \geq S_i'(\mathcal{M}_{i-1}) \geq \cdots \geq S_N'(\mathcal{M}_{N-1}). \tag{33}$$

The ith miner has the maximum density $S_i'(\mathcal{M}_{i-1})$ over $\mathcal{N} \setminus \mathcal{M}_{i-1}$ where $\mathcal{M}_{i-1} = \{1, 2, \ldots, i-1\}$ and $\mathcal{M}_0 = \emptyset$. From the sorting, the MDB auction finds the set of winners $\mathcal{M}_{L_m}$ containing L_m winners, such that $d_{\mathcal{M}_{L_m}} \leq D$, $S_{L_m}'(\mathcal{M}_{L_m-1}) \geq 0$ and $S_{L_m+1}'(\mathcal{M}_{L_m}) < 0$ (lines 6-13).

To determine the service price for each winner $i \in \mathcal{M}_{L_m}$ (lines 14-36), the MDB auction re-executes the winner selection process and similarly sorts other winners in $\mathcal{N}_{-i} = \mathcal{N} \setminus \{i\}$ as follows:

$$S_{i_1}'(\mathcal{T}_0) \geq S_{i_2}'(\mathcal{T}_1) \geq \cdots \geq S_{i_k}'(\mathcal{T}_{k-1}) \geq \cdots \geq S_{i_{N-1}}'(\mathcal{T}_{N-2}), \tag{34}$$

where $\mathcal{T}_{k-1}$ denotes the first $k-1$ winners in the sorting and $\mathcal{T}_0 = \emptyset$. From the sorting, we select the first L_p winners where the L_pth winner is the last one that satisfies $S_{i_{L_p}}'(\mathcal{T}_{L_p-1}) \geq 0$ and $d_{\mathcal{T}_{L_p-1}} \leq D - d_i$. Let $\widetilde{S}$ denote the (L_p+1)th winner's virtual density. If the (L_p+1)th winner has a negative density on $\mathcal{T}_{L_p}$, i.e., $S_{i_{L_p+1}}'(\mathcal{T}_{L_p}) < 0$, or its demand is larger than that of winner i, i.e., $d_{L_p+1} > d_i$, we set $\widetilde{S} = 0$. Otherwise, $\widetilde{S} = S_{i_{L_p+1}}'(\mathcal{T}_{L_p})$. Meanwhile, Algorithm 2 forms a price list $\mathbf{L} = \{S_{i_1}'(\mathcal{T}_0), \ldots, S_{i_{L_p}}'(\mathcal{T}_{L_p-1}), \widetilde{S}\}$ containing (L_p+1) density values. According to the list, we find the winner i's minimum bid b_i' such that $S_i'(\mathcal{T}_{k-1}) \geq S_{i_k}'(\mathcal{T}_{k-1}), \exists k \in \{0, 1, \ldots, L_p\}$ or $S_i'(\mathcal{T}_{L_p}) \geq \widetilde{S}$. Here, b_i' is called miner i's ex-ante price, which is the payment without considering the allocative externalities. Then, we set $p_i = \left(a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{j\in\mathcal{M}_{L_m}} d_j}{D}}\right) \frac{b_i'}{D}$ as the winner i's final payment.

Algorithm 2. MDB Auction

Input: Miners' demand profile **d** and bid profile **b**.
Output: Resource allocation **x** and service price profile **p**.

1: **begin**
2: **for each** $i \in \mathcal{N}$ **do**
3: $x_i \leftarrow 0, p_i \leftarrow 0$
4: **end for**
5: $\mathcal{M} \leftarrow \varnothing, d \leftarrow 0$
6: **while** $\mathcal{M} \neq \mathcal{N}$ **do**
7: $j \leftarrow \arg\max_{i\in\mathcal{N}\setminus\mathcal{M}} S_i'(\mathcal{M})$
8: **if** $d + d_j > D$ **or** $S_j'(\mathcal{M}) < 0$ **then**
9: **break**
10: **end if**
11: $\mathcal{M} \leftarrow \mathcal{M} \cup \{j\}$
12: $d \leftarrow d + d_j$
13: **end while**
14: **for each** $i \in \mathcal{M}$ **do**
15: $x_i \leftarrow 1, \mathcal{N}_{-i} \leftarrow \mathcal{N} \setminus \{i\}$
16: $\mathcal{T}_0 \leftarrow \varnothing, d' \leftarrow 0, k \leftarrow 0, L_p \leftarrow 0$
17: **while** $\mathcal{T}_k \neq \mathcal{N}_{-i}$ **do**
18: $i_{k+1} \leftarrow \arg\max_{l\in\mathcal{N}_{-i}\setminus\mathcal{T}_k} S_l'(\mathcal{T}_k)$
19: $b_{i_{k+1}}' \leftarrow \arg_{b_i\in\mathbb{R}^+} S_i'(\mathcal{T}_k) = S_{i_{k+1}}'(\mathcal{T}_k)$
20: **if** $d' + d_{i_{k+1}} > D$ **or** $S_{i_{k+1}}'(\mathcal{T}_k) < 0$ **then**
21: **break**
22: **else if** $d' + d_{i_{k+1}} \leq D - d_i$ **then**
23: $L_p \leftarrow L_p + 1$
24: **end if**
25: $\mathcal{T}_{k+1} \leftarrow \mathcal{T}_k \cup \{i_{k+1}\}, d' \leftarrow d' + d_{i_{k+1}}$
26: $k \leftarrow k + 1$
27: **end while**
28: **if** $S_{i_{L_p+1}}'(\mathcal{T}_{L_p}) < 0$ **or** $d_{i_{L_p+1}} > d_i$ **then**
29: $\widetilde{S} \leftarrow 0$
30: **else**
31: $\widetilde{S} \leftarrow S_{i_{L_p+1}}'(\mathcal{T}_{L_p})$
32: **end if**
33: $b_{i_{L_p+1}}' \leftarrow \arg_{b_i\in\mathbb{R}^+} S_i'(\mathcal{T}_{L_p}) = \widetilde{S}$
34: $b_i' \leftarrow \min_{k\in\{0,1,\ldots,L_p+1\}} b_{i_k}'$
35: $p_i \leftarrow (a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{j\in\mathcal{M}} d_j}{D}}) \frac{b_i'}{D}$
36: **end for**
37: **end**

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

5.2.2 Properties of MDB Auction

We show the computational efficiency (Proposition 5), the individual rationality (Proposition 6), and the truthfulness (Proposition 7) of the MDB auction in the following.

Proposition 5. *MDB auction is computationally efficient.*

Proof. In Algorithm 2, finding the winner with the maximum density has the time complexity of $O(N)$ (line 7). Since the number of winners is at most N, the winner selection process (the while-loop lines 6-13) has the time complexity of $O(N^2)$. In the service price determination process (lines 14-36), each for-loop executes similar steps as the while-loop in lines 6-13. Hence, lines 14-36 have the time complexity of $O(N^3)$ in general. Hence, the running time of Algorithm 2 is dominated by the for-loop, which is bounded by polynomial time $O(N^3)$. □

Proposition 6. *MDB auction is individually rational.*

Proof. Let i_i be the miner i's replacement which appears in the ith place in the sorting (34) over $\mathcal{N}_{-i}$. Since miner i_i would not be in the ith place if winner i is considered, we have $S'_{i_i}(\mathcal{T}_{i-1}) \leq S'_i(\mathcal{T}_{i-1})$. Note that Algorithm 2 chooses the minimum bid b'_i for miner i, which means that given the bid b'_i, miner i's new density $S''_i(\mathcal{T}_{i-1})$ at least satisfies $S''_i(\mathcal{T}_{i-1}) \leq S'_{i_i}(\mathcal{T}_{i-1}) \leq S'_i(\mathcal{T}_{i-1})$. According to the definition of the density in (32), $S'_i(\mathcal{T}_{i-1})$ is a monotonically increasing function of b_i. Hence, we have $b_i - b'_i \geq 0$ as $S'_i(\mathcal{T}_{i-1}) \geq S''_i(\mathcal{T}_{i-1})$. Therefore, the final payment for miner i is not more than its ex-post valuation, i.e., $p_i = \left(a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{j \in \mathcal{M}_{L_m}} d_j}{D}}\right) \frac{b'_i}{D} \leq v''_i = \left(a_1 - a_2 \mathrm{e}^{\frac{a_3 \sum_{j \in \mathcal{M}_{L_m}} d_j}{D}}\right) \frac{b_i}{D}$. Thus, the individual rationality of MDB auction is ensured. □

Proposition 7. *MDB auction is truthful.*

Proof. Based on Theorem 2, it suffices to prove that the selection rule of the MDB auction is monotone, and the ex-ante payment b'_i is the critical value for winner i to win the auction.

We first discuss the monotonicity of the MDB auction in terms of winner i's bid and demand subsequently. Recalling the density $S'_i(\mathcal{M})$ in Equation (32), it is clear that $S'_i(\mathcal{M})$ is a monotonically increasing function of miner i's bid b_i. As miner i takes the ith place in the sorting (33), when winner i raises its bid from b_i to b_i^+, it at least has a new larger density $S'_{i^+}(\mathcal{T}_{i-1}) > S'_i(\mathcal{T}_{i-1}) \geq 0$. Because of the submodularity of $S(\mathcal{M})$, miner i can only have a larger density when it is ranked higher in the sorting, i.e., $S'_{i^+}(\mathcal{M}_{i-k}) > S'_{i^+}(\mathcal{M}_{i-1}) \geq 0, \forall k \in \{2, 3, \ldots, i\}$. Therefore, miner i with a higher bid can always win the auction. Similarly, when it comes to miner i's demand d_i, we only need to show that $S'_i(\mathcal{M})$ is a monotonically decreasing function of d_i. Let

$$h(z) = \frac{a_4\left(1 - \mathrm{e}^{\frac{a_3}{D}z}\right)}{z}, \tag{35}$$

where $z \in \mathbb{R}^+$ and all parameters are positive. The first derivative of $h(z)$ is

$$\frac{\mathrm{d}h(z)}{\mathrm{d}z} = -\frac{a_4(\frac{a_3}{D}\mathrm{e}^{\frac{a_3}{D}z}z + 1 - \mathrm{e}^{\frac{a_3}{D}z})}{z^2}. \tag{36}$$

Since the first derivative of $(\frac{a_3}{D}\mathrm{e}^{\frac{a_3}{D}z}z + 1 - \mathrm{e}^{\frac{a_3}{D}z})$ is $\frac{a_3^2}{D^2}\mathrm{e}^{\frac{a_3}{D}z}z > 0$, we can have $\frac{\mathrm{d}h(z)}{\mathrm{d}z} < 0$ with $a_3, a_4, D, z > 0$. Thus, $h(z)$ is monotonically decreasing with z. By substituting $z = d_i$, we can easily observe that ① in (32) is a monotonically decreasing function with respect to d_i. Finally, $S'_i(\mathcal{M})$ is proved to be monotonically decreasing with d_i since ② in (32) is clearly a monotonically decreasing function of d_i as well.

Next, we prove that b'_i is the critical ex-ante payment. This means that bidding lower $b_i^- < b'_i$ can lead to miner i's failure in the auction. Given that d_i is fixed, we note that b'_i is the minimum bid such that miner i's new density $S''_i(\mathcal{T}_k)$ is no more than any value in the kth place in the sorting (34), where $k \in \{0, 1, \ldots, L_p - 1\}$. If miner i submits a lower bid b_i^-, it must be ranked after the L_pth winner in (34) due to submodularity of $S(\mathcal{M})$. Then, its density has to be compared with $\tilde{S}$. Considering the $(L_p + 1)$th winner in the sorting (34), if its density $S'_{i_{L_p+1}}(\mathcal{T}_{L_p}) \geq 0$ and $d_{i_{L_p+1}} \leq d_i$, $\tilde{S}$ is set to be $S'_{i_{L_p+1}}(\mathcal{T}_{L_p})$. In this case, miner i with bid b_i^- cannot take the $(L_p + 1)$th place as its new density is $S''_i(\mathcal{T}_{L_p}) < S'_i(\mathcal{T}_{L_p}) \leq \tilde{S} = S'_{i_{L_p+1}}(\mathcal{T}_{L_p})$. Also, it no longer can win the auction by taking the place after the $(L_p + 1)$th because the remaining supply $D - d_{\mathcal{T}_{L_p+1}}$ cannot meet its demand d_i, i.e., $D - d_{\mathcal{T}_{L_p+1}} < d_i$. If $S'_{i_{L_p+1}}(\mathcal{T}_{L_p}) < 0$ or $d_{i_{L_p+1}} > d_i$, $\tilde{S}$ is just set to be 0. Apparently, b_i^- is not a winning bid as $S''_i(\mathcal{T}_{L_p}) < b'_i = \tilde{S} = 0$. □

6 Experimental Results and Performance Evaluation

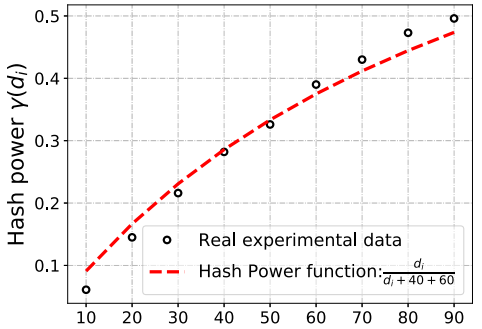
In this section, we first perform experiments to verify the proposed hash power function and network effects function. Then, from simulation results, we examine the performance of the proposed auction mechanisms in social welfare maximization and provide useful decision making strategies for the CFP and the blockchain developer.

6.1 Verification for Hash Power Function and Network Effects Function

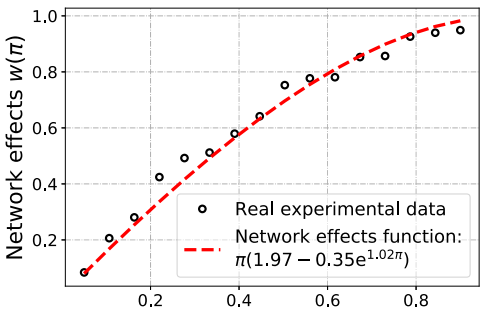
Similar to the experiments on mobile blockchain mining in [10], [43], we design a mobile blockchain client application in the Android platform and implement it on each of three mobile devices (miners). The client application can not only record the data generated by internal sensors or the transactions of the mobile P2P data trading, but also allows each mobile device to be connected to a computing server through a network hub. The miners request the computing service from the server. Then, the server allocates the computing resources and starts mining the block for the miners. At the server side, the each miner's CPU utilization rate is managed and measured by the Docker platform.[5] In

5. https://www.docker.com/community-edition.

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.



(a) Miner i's computing resouces demand d_i



(b) Normalized computing power of the experimental blockchain network π

Fig. 3. Estimation of (a) the hash power function $\gamma(d_i)$ in (1) and (b) the network effects function $w(\pi)$ in (5).

our experiment, all mining tasks (solving the PoW puzzle) are under Go-Ethereum[6] blockchain framework.

To verify the hash power function in (1), we vary the service demand of one miner i in terms of CPU utilization, i.e., d_i, while fixing the other two miners' service demand at 40 and 60. Here, the total amount of computing resources is $d_{\mathcal{N}} = d_i + 40 + 60$. Besides, we initially broadcast 10 same transaction records to the miners in the network so that all mined blocks have the same size. Fig. 3a shows the change of the hash power, i.e., the probability of successfully mining a block with different amount of computing resources. We note that the hash power function defined in (1) can well fit the real experimental results.

To verify the network effects function in (5), we investigate the capability of the blockchain to prevent the double-spending attacks. We add a malicious miner with fixed computing powers, i.e., an attacker performing double-spending attacks, to the blockchain network. Then, we conduct several tests by varying the CPU resources of the other miners, i.e., the sum of existing honest miners' computing resources $d_{\mathcal{N}}$, to measure the probability of the successful attacks. Specifically we count the number of fake blocks which successfully join the chain every 10,000 blocks generated in each test. Based on the above results, we finally calculate the proportion of the genuine blocks every 10,000 blocks (i.e., each data point in the Fig. 3b) as the security measure or the network effects of the blockchain network. As illustrated in Fig. 3b, it is evident that the network effects function in (5) also well fits the real experiment results. Based on the experiments, we set $a_1 = 1.97, a_2 = 0.35, a_3 = 1.02$ in the following simulations.

6. https://ethereum.github.io/go-ethereum.

TABLE 2
Default Parameter Values

Parameters	Values	Parameters	Values
N	300	T	12.5
r	0.007	λ	15
c	0.001	q	10
a_1	1.97	β_1, β_2	0, 0.02
a_2	0.35	ξ	0.001
a_3	1.02	D	1000

6.2 Numerical Results

To demonstrate the performance of the proposed auction mechanisms and the impacts of various parameters on the social welfare of the blockchain network, we consider a set of N miners, e.g., mobile users in a PoW-based blockchain application supported by the CFP. Each miner's block size is uniformly distributed over (0,1024]. Instead of being restricted to submit a constant demand as in the CDB auction, each miner in the MDB auction and FRLS auction can choose its desired demand which follows the uniform distribution over $[\beta_1 D, \beta_2 D]$. Except Fig. 6a, each measurement is averaged over 600 instances and the associated 95 percent confidence interval is given. We can find that the confidence intervals are very narrowly centered around the mean. The default parameter values are presented in Table 2. Note that setting $q = 10$, $\beta_1 = 0$ and $\beta_2 = 0.02$ means the expected demand of miners in the MDB auction is equal to the constant demand of miners in the CDB auction. Hence, we can compare the performance of both proposed auction mechanisms.

6.2.1 Evaluation of MDB Auction versus FRLS Auction in Terms of Social Welfare Maximization

We evaluate the performance of the MDB auction in maximizing the social welfare by comparing it with the FRLS auction. Table 3 shows the social welfare obtained by the MDB auction and the FRLS auction. The social welfare generated from the MDB auction is lower than that from the FRLS auction when dealing with a small number of miners. As the group of interested miners grows, the MDB auction can achieve slightly larger social welfare although it has to preserve necessary economic properties, including individual rationality and truthfulness. The main reason is that the FRLS auction is an algorithm which only provides a theoretical lower bound guarantee in the worst case for approximately maximizing the social welfare, and may have more severe performance deterioration when interested miners become more.

6.2.2 Impact of the Number of Miners N

Besides the social welfare, we introduce the satisfaction rate, i.e., the percentage of winners selected from all interested miners, as another metric. Here, we compare the social

TABLE 3
MDB Auction versus FRLS Auction in Social Welfare Maximization

Number of miners	10	15	20	25
MDB auction	33.954	50.368	65.421	80.135
FRLS auction	34.656	49.935	65.060	79.853

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

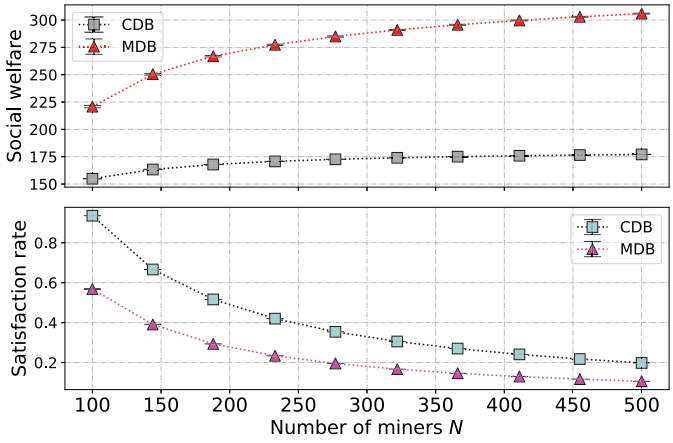


Fig. 4. Impact of the number of miners N.

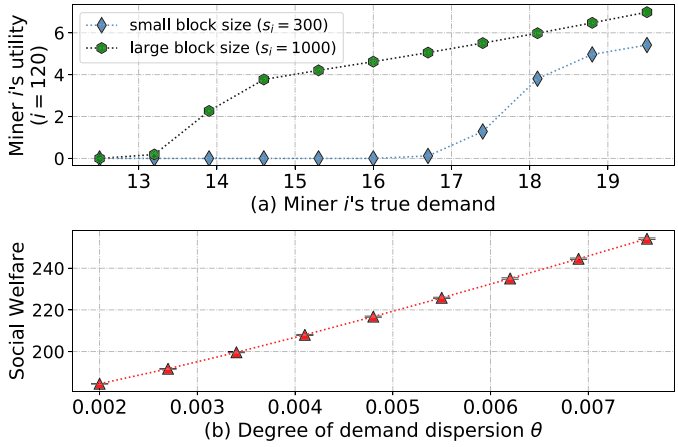


Fig. 6. Relationship between miner i's ($i = 120$) utility and its true demand, and the impact of the degree of demand dispersion θ.

welfare as well as the satisfaction rate of the CDB auction and the MDB auction with various number of miners, as shown in Fig. 4. From Fig. 4, we observe that the social welfare S in both auction mechanisms increases as the base of interested miners becomes larger. We observe that the satisfaction rate decreases and the rise of the social welfare also slows down with the increase of N. The main reason is that the competition among miners becomes more obvious when more miners take part in the auction, and, with more winners selected by the auction, the subsequent winner's density decreases due to the network effects. When choosing between the CDB auction and the MDB auction, Fig. 4 clearly shows that there is a tradeoff between the social welfare and the satisfaction rate. The MDB auction can help the CFP achieve more social welfare than the CDB auction because of its advantage in relaxing restrictions on miners' demand. However, the CDB is relatively more fair because the MDB auction allows miners with large demand to take up more computing resources and this leads to a lower satisfaction rate.

6.2.3 *Impact of the Unit Cost c, the Fixed Bonus T, the Transaction Fee Rate r and the Block Time λ*

The CFP organizes the auction and cares about the unit cost of the computing resource. It is obvious from Fig. 5a that as the computing resources become expensive, the social welfare in each auction mechanism decreases linearly. The blockchain developer may be more interested in optimizing the blockchain protocol parameters, including the fixed reward, the transaction fee rate and the block time. In Figs. 5b, 5c, and 5d, we study their impacts on the social welfare of the blockchain network. Figs. 5b and 5c illustrate that if the blockchain developer raises the fixed bonus T or the transaction fee rate r, higher social welfare will be generated nearly in proportion. This is because miner's valuation increases with higher T and r, according to the definition in (6). Moreover, by increasing T and r, we observe that the difference of the social welfare between the CDB auction and the MDB auction amplifies. The reason is that raising T and r can significantly improve the valuation of miner i which possesses large block size s_i and high demand d_i. As shown in Fig. 5d, when the blockchain developer raises the difficulty of mining a block, i.e., extending the block time λ, the social welfare goes up. This is because a long block time λ gives the miner which has solved the PoW puzzle a higher probability to successfully propagate the new block and reach consensus. However, different from adjusting T and r, the marginal gains in social welfare gradually become smaller if the blockchain developer continues to increase the difficulty of the blockchain mining. This is mainly due to that the increasing value of λ has less impact on the miner's valuation, as can be seen from the Equations (4) and (6).

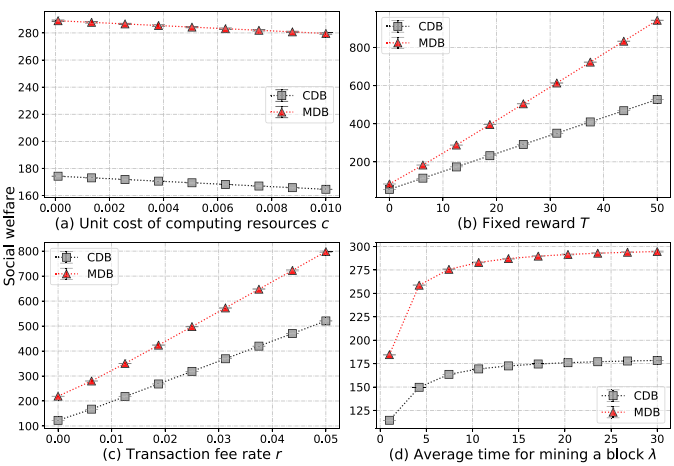


Fig. 5. Impact of unit cost c, fixed bonus T, transaction fee rate r and block time λ.

6.2.4 *Miner's Utility and Individual Demand Constraints in the MDB Auction*

In the MDB auction, we randomly choose a miner (ID = 120) to see its utility which is defined by the difference between its ex-post valuation and its payment, i.e., $v''_{120} - p_{120}$. The miner's block size is respectively at a low level ($s_{120} = 300$) and a high level ($s_{120} = 1000$). We investigate the impact of the miner's true demand on its utility, which also reflects the impact of its available budget. Fig. 6a shows that when miner 120' s true demand rises, its utility initially stays at 0 and then suddenly increases. This indicates that only when the miner's demand is above a threshold, it can be selected as the winner by the MDB auction, i.e., x_i changes immediately from 0 to 1, obtains the computing resources and finally has a positive utility. Otherwise, the miner would not be allocated the resources, i.e., $x_i = 0$, and then both its ex-post valuation and

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

payment should be 0 according to the MDB auction algorithm, which results in zero utility. Additionally, if the miner's generated block is larger, it can obtain higher utility with the same true demand. This implies that miners with large block size and high demand are easier to be selected by the MDB auction for social welfare maximization.

In Fig. 6b, we investigate the impact of the demand constraints on the social welfare in the MDB auction. To fix the miner's expected demand at q, we set demand constraints $\beta_1 D = q - \theta D$ and $\beta_2 D = q + \theta D$ where $\theta \in [0, \min(\frac{q}{D}, 1 - \frac{q}{D})]$ characterizes the degree of demand dispersion. It is clear that social welfare increases as the degree of demand dispersion rises and miners have more freedom to submit their desired demands.

7 Conclusions

In this paper, we have investigated the cloud/fog computing services that enable blockchain-based DApps. To efficiently allocate computing resources, we have presented an auction-based market model to study the social welfare optimization and considered allocative externalities that particularly exist in blockchain networks, including the competition among the miners as well as the network effects of the total hash power. For miners with constant demand, we have proposed an auction mechanism (CDB auction) that achieves optimal social welfare. For miners with multiple demands, we have transformed the social welfare maximization problem to a non-monotone submodular maximization with knapsack constraints problem. Then, we have designed two efficient mechanisms (FRLS auction and MDB auction) maximizing social welfare approximately. We have proven that the proposed CDB and MDB auction mechanisms are truthful, individually rational and computationally efficient and are able to solve the social welfare maximization problem.

In this work, we consider the energy and computational constraints for PoW-based public blockchain network while assuming an ideal communication environment. For practical system implementation, the communication constraint is actually an important factor in establishing the mobile blockchain network. An example is that the limited bandwidth for each miner's mutual wireless communication will not only affect each miner's utility, but also have an adverse impact on the block broadcasting process and the throughput of the whole blockchain network. For the future work, we will take the complicated communication environment into account, and design new spectrum allocation algorithms for more efficient and practical blockchain system.

Acknowledgments

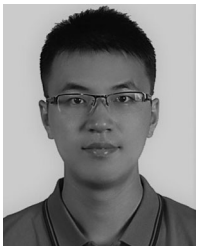
This work was supported in part by WASP/NTU M4082187 (4080), Singapore MOE Tier 1 under Grant 2017-T1-002-007 RG122/17, MOE Tier 2 under Grant MOE2014-T2-2-015 ARC4/15, Singapore NRF2015-NRF-ISF001-2277, and Singapore EMA Energy Resilience under Grant NRF2017EWT-EP003-041.

References

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
[2] Y. Guo and C. Liang, "Blockchain application and outlook in the banking industry," *Financial Innovation*, vol. 2, no. 1, 2016, Art. no. 24.
[3] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the Internet of Things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.
[4] D. Chatzopoulos, M. Ahmadi, S. Kosta, and P. Hui, "FlopCoin: A cryptocurrency for computation offloading," *IEEE Trans. Mobile Comput.*, vol. 17, no. 5, pp. 1062–1075, May 2018.
[5] "Blockchain for enterprise applications," *Tech. Rep.*, Tractica, https://www.tractica.com/research/blockchain-for-enterprise-applications/. The latest publication date is 3Q 2018.
[6] J. Garay, A. Kiayias, and N. Leonardos, "The bitcoin backbone protocol: Analysis and applications," in *Proc. Annu. Int. Conf. Theory Appl. Cryptographic Techn.*, 2015, pp. 281–310.
[7] H. Shafagh, L. Burkhalter, A. Hithnawi, and S. Duquennoy, "Towards blockchain-based auditable storage and sharing of IoT data," in *Proc. Cloud Comput. Secur. Workshop*, 2017, pp. 45–50.
[8] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain, "Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3154–3164, Dec. 2017.
[9] G. Zyskind, O. Nathan, et al., "Decentralizing privacy: Using blockchain to protect personal data," in *Proc. IEEE Secur. Privacy Workshops*, 2015, pp. 180–184.
[10] 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.
[11] Q. Li, L. Zhao, J. Gao, H. Liang, L. Zhao, and X. Tang, "SMDP-based coordinated virtual machine allocations in cloud-fog computing systems," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1977–1988, Jun. 2018.
[12] X. Zhang, Z. Huang, C. Wu, Z. Li, and F. C. Lau, "Online auctions in IaaS clouds: Welfare and profit maximization with server costs," *IEEE/ACM Trans. Netw.*, vol. 25, no. 2, pp. 1034–1047, Apr. 2017.
[13] A. Kiayias, E. Koutsoupias, M. Kyropoulou, and Y. Tselekounis, "Blockchain mining games," in *Proc. ACM Conf. Econ. Comput.*, 2016, pp. 365–382.
[14] C. Catalini and J. S. Gans, "Some simple economics of the blockchain," *Tech. Rep.*, National Bureau of Economic Research, 2016, http://www.nber.org/papers/w22952.
[15] Y. Jiao, P. Wang, D. Niyato, and Z. Xiong, "Social welfare maximization auction in edge computing resource allocation for mobile blockchain," in *Proc. IEEE Next Generation Netw. Internet Symp.*, May 2018, pp. 1–6.
[16] D. T. T. Anh, M. Zhang, B. C. Ooi, and G. Chen, "Untangling blockchain: A data processing view of blockchain systems," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 7, pp. 1366–1385, Jul. 2018.
[17] F. Tschorsch and B. Scheuermann, "Bitcoin and beyond: A technical survey on decentralized digital currencies," *IEEE Commun. Surv. Tut.*, vol. 18, no. 3, pp. 2084–2123, Jul.–Sep. 2016.
[18] H. Kopp, D. Mödinger, F. Hauck, F. Kargl, and C. Bösch, "Design of a privacy-preserving decentralized file storage with financial incentives," in *Proc. IEEE Eur. Symp. Secur. Privacy Workshops*, 2017, pp. 14–22.
[19] J. Backman, S. Yrjölä, K. Valtanen, and O. Mämmelä, "Blockchain network slice broker in 5G: Slice leasing in factory of the future use case," in *Proc. Internet Things Bus. Models Users Netw.*, Nov. 2017, pp. 1–8.
[20] N. Houy, "The bitcoin mining game," *Ledger*, vol. 1, pp. 53–68, 2016.
[21] Y. Lewenberg, Y. Bachrach, Y. Sompolinsky, A. Zohar, and J. S. Rosenschein, "Bitcoin mining pools: A cooperative game theoretic analysis," in *Proc. Int. Conf. Auton. Agents Multiagent Syst.*, 2015, pp. 919–927.
[22] X. Zhang, Z. Yang, W. Sun, Y. Liu, S. Tang, K. Xing, and X. Mao, "Incentives for mobile crowd sensing: A survey," *IEEE Commun. Surv. Tut.*, vol. 18, no. 1, pp. 54–67, Jan.–Mar. 2016.
[23] D. Yang, G. Xue, X. Fang, and J. Tang, "Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones," *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1732–1744, Jun. 2016.
[24] H. Jin, L. Su, D. Chen, K. Nahrstedt, and J. Xu, "Quality of information aware incentive mechanisms for mobile crowd sensing systems," in *Proc. 16th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2015, pp. 167–176.
[25] L. Mashayekhy, M. M. Nejad, and D. Grosu, "Physical machine resource management in clouds: A mechanism design approach," *IEEE Trans. Cloud Comput.*, vol. 3, no. 3, pp. 247–260, Jul.–Sep. 2015.
[26] A. Kiani and N. Ansari, "Toward hierarchical mobile edge computing: An auction-based profit maximization approach," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 2082–2091, Dec. 2017.
[27] Z. Zheng, F. Wu, and G. Chen, "A strategy-proof combinatorial heterogeneous channel auction framework in noncooperative wireless networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 6, pp. 1123–1137, Jun. 2015.

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.

[28] M. Salek and D. Kempe, "Auctions for share-averse bidders," in *Proc. Int. Workshop Internet Netw. Econ.*, 2008, pp. 609–620.
[29] P. Jehiel and B. Moldovanu, "Efficient design with interdependent valuations," *Econometrica*, vol. 69, no. 5, pp. 1237–1259, 2001.
[30] D. Zhao, X.-Y. Li, and H. Ma, "How to crowdsource tasks truthfully without sacrificing utility: Online incentive mechanisms with budget constraint," in *Proc. IEEE Conf. Comput. Commun.*, 2014, pp. 1213–1221.
[31] N. C. Luong, D. Niyato, P. Wang, and Z. Xiong, "Optimal auction for edge computing resource management in mobile blockchain networks: A deep learning approach," in *Proc. IEEE Int. Conf. Commun.*, May 2018, pp. 1–6.
[32] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic Game Theory*, vol. 1. Cambridge, U.K.: Cambridge Univ. Press, 2007.
[33] D. Kraft, "Difficulty control for blockchain-based consensus systems," *Peer-to-Peer Netw. Appl.*, vol. 9, no. 2, pp. 397–413, 2016.
[34] A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. Princeton, NJ, USA: Princeton Univ. Press, 2016.
[35] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Trans. Depend. Sec. Comput.*, vol. 15, no. 5, pp. 840–852, Sep./Oct. 2018.
[36] M. Li, J. Weng, A. Yang, W. Lu, Y. Zhang, L. Hou, L. Jia-Nan, Y. Xiang, and R. Deng, "CrowdBC: A blockchain-based decentralized framework for crowdsourcing," *IEEE Trans. Parallel Distrib. Syst.*, 2018, doi: 10.1109/TPDS.2018.2881735.
[37] R. B. Myerson, "Optimal auction design," *Math. Operations Res.*, vol. 6, no. 1, pp. 58–73, 1981.
[38] V. Krishna, *Auction Theory*. Cambridge, MA, USA: Academic Press, 2009.
[39] J. C. Lagarias and A. M. Odlyzko, "Solving low-density subset sum problems," *J. ACM*, vol. 32, no. 1, pp. 229–246, 1985.
[40] L. Lovász, "Submodular functions and convexity," in *Mathematical Programming The State of the Art*, Berlin, Germany: Springer, 1983, pp. 235–257.
[41] J. Lee, V. S. Mirrokni, V. Nagarajan, and M. Sviridenko, "Non-monotone submodular maximization under matroid and knapsack constraints," in *Proc. 41st Annu. ACM Symp. Theory Comput.*, 2009, pp. 323–332.
[42] N. Nisan, "Chapter 9-Algorithmic mechanism design: Through the lens of multiunit auctions," *Handbook of Game Theory with Economic Applications*, vol. 4, Amsterdam, Netherlands: Elsevier, 2015, pp. 477–515.
[43] K. Suankaewmanee, D. T. Hoang, D. Niyato, S. Sawadsitang, P. Wang, and Z. Han, "Performance analysis and application of mobile blockchain," in *Proc. Int. Conf. Comput. Netw. Commun.*, Mar. 2018, pp. 642–646.



Yutao Jiao is currently working toward the PhD degree in the School of Computer Science and Engineering, Nanyang Technological University (NTU). His research interests include the Internet of Things (IoT), artificail intelligence (AI), blockchain and the economics in the area of big data.

Ping Wang (M'08-SM'15) received the bachelor's and master's degrees from the Huazhong University of Science and Technology, in 1994 and 1997, respectively, and the PhD degree from the University of Waterloo, Canada, in 2008, all in electrical engineering. She is currently an associate professor at York University. Prior to that, she was with Nanyang Technological University, Singapore, from 2008 to 2018. Her research interests are mainly in radio resource allocation, network design, performance analysis and optimization for heterogeneous wireless networks. She received the Best Paper Awards from IEEE Wireless Communications and Networking Conference (WCNC) 2012 and IEEE International Conference on Communications (ICC) 2007. She is a senior member of the IEEE.

Dusit Niyato (M'09-SM'15-F'17) received the BEng degree from the King Mongkut's Institute of Technology Ladkrabang (KMITL), Thailand, in 1999, and the PhD degree in electrical and computer engineering from the University of Manitoba, Canada, in 2008. He is currently a professor with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His research interests are in the areas of energy harvesting for wireless communication, the Internet of Things, and sensor networks. He is a fellow of the IEEE.

Kongrath Suankaewmanee received the bachelor's degree in computer engineering from the King Mongkut's University of Technology Thonburi (KMUTT), Thailand, in 2011. He is currently working at Computer Networks and Communications Lab, School of Computer Science and Engineering, Nanyang Technological University (NTU), Singapore. His research interests are in the area of web technologies, blockchain technologies, and smart grid systems.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.

Authorized licensed use limited to: Universidade Tecnologica Federal do Parana. Downloaded on May 27,2023 at 13:09:59 UTC from IEEE Xplore. Restrictions apply.