Edge Computing-Based Tasks Offloading and Block Caching for Mobile Blockchain

Yong Yan[1], Yao Dai[2, *], Zhiqiang Zhou[3], Wei Jiang[4] and Shaoyong Guo[2]

Abstract: Internet of Things (IoT) technology is rapidly evolving, but there is no trusted platform to protect user privacy, protect information between different IoT domains, and promote edge processing. Therefore, we integrate the blockchain technology into constructing trusted IoT platforms. However, the application of blockchain in IoT is hampered by the challenges posed by heavy computing processes. To solve the problem, we put forward a blockchain framework based on mobile edge computing, in which the blockchain mining tasks can be offloaded to nearby nodes or the edge computing service providers and the encrypted hashes of blocks can be cached in the edge computing service providers. Moreover, we model the process of offloading and caching to ensure that both edge nodes and edge computing service providers obtain the maximum profit based on game theory and auction theory. Finally, the proposed mechanism is compared with the centralized mode, mode A (all the miners offload their tasks to the edge computing service providers), and mode B (all the miners offload their tasks to a group of neighbor devices). Simulation results show that under our mechanism, mining networks obtain more profits and consume less time on average.

Keywords: Edge computing, blockchain, mining offloading, block caching.

1 Introduction

With the rapid development of information and communication technology, Internet of Things (IoT) technology has been applied to the modern society everywhere [Christidis and Devetsikiotis (2016)]. However, there is no trusted platform to protect user privacy, protect information between different IoT domains, and promote edge processing. Since the introduction of Bitcoin from 2009, blockchain has gained a significant popularity in many fields of IoT [Min, Li, Liu et al. (2016)]. Blockchain enables reliable transactions without a centralized management mechanism even if there are unreliable participants in the network [Min, Li, Liu et al. (2016); Swan (2015)]. It can decentralize the credibility because it adopts a gossip peer-to-peer architecture and the transaction information can

---

[1] Electric Power Research Institute, State Grid Zhejiang Electric Power Company, Hangzhou, China.

[2] State Key Laboratory of Networking and Switching Technology Beijing University of Posts and Telecommunications, Beijing, China.

[3] Electric Power Research Institute, State Grid Zhejiang Electric Power Company, Hangzhou, China.

[4] State Grid Corporation of China, Beijing, China.

* Corresponding Author: Yao Dai. Email: daiyao@bupt.edu.cn.

be accessed by every peer [Min, Li, Liu et al. (2016); Samaniego and Deters (2016)]. Meanwhile, the application of cryptography avoids the double-spending and data tampering problem.

Since blockchain can provide secure transactions and protect privacy for smart terminals, we integrate blockchain technology into constructing trusted IoT platforms. However, the application of blockchain is hampered by the challenges of mining process [Cha, Chen, Su et al. (2018)]. During the mining process, participants need to complete a heavy computing processes called the proof of work (PoW), which puts extremely high demands on miners'storage capacity and computing capacity [Baktir, Ozgovde and Ersoy (2016)]. As a new computational paradigm, edge computing has become a promising technology, in which nodes distribute geographic data and aggregate data within its coverage [Wang and Yong (2015)]. Compared with traditional cloud computing, it enables network resources (computing or storage resources) to be closer to the end user, and enables the terminal to authenticate directly at the edge node, thereby supporting blockchain deployment in the IoT network [Chen, Jiao, Li et al. (2016)]. Therefore, we put forward a blockchain framework based on mobile edge computing where the computation-intensive mining tasks can be offloaded to nearby devices or edge computing service provider (ECSP), and the cryptographic hashes of blocks can be cached in the ECSP.

There are already many successful cases of blockchain and edge computing applied to the Internet of Things. For example, Xiong et al. [Xiong, Zhang, Niyato et al. (2018)] proposed the concept of mobile blockchain which combines blockchain with edge computing. It uses edge computing resources to support blockchain operation. They modeled the interactions between miners and ECSP as a game. The blockchain miner calculates the number of resources requested from the edge computing based on the strategy of other miners and the ECSP, and the ECSP calculates the price of the edge computing resource. Han et al. [Han, Su and Chen (2018)] introduce an architecture called Jupiter, an edge computing-based Blockchain platform provide storage for mobile blockchain.

Based on the researches of blockchain combining edge computing, we optimize the allocation of edge computing resources in this paper. In order to make the resources of the edge nodes fully utilized and facilitate the user's offloading process, we provide two offloading methods, offloading to ECSP, or offloading to neighbor devices. In the first way, miners interact with ECSP through the Stackelberg game theory [Qin, Yuan and Wang (2018)] to obtain the optimal number of requested resources, and then, ECSP adjusts the optimal price. In the second way, the edge devices auction their computing resources, and the miners bid for the resources through double auction. What's more, considering that miners may have insufficient storage, they can negotiate the caching price with the ECSP and then cache cryptographic hashes of blocks in ECSP [Liu, Yu, Teng et al. (2018)].

Compared to traditional edge computing, innovations of our work lie in: we support the offloading of blockchain mining tasks to neighbor devices and ECSP, and content caching of cryptographic hashes of blocks in ECSP. The main contributions of this paper are summarized as follows:

- Our mechanism supports two offloading modes. Miners can offload their mining

tasks to neighbor devices and ECSP, making the resources of the edge nodes fully utilized and facilitate the user's offloading process.

- Task offloading between the edge nodes is performed under an auction model. We determine the auction price by ensuring that social welfare is maximum.

- Mining offloading to ECSP is modeled as a Stackelberg game, in which we calculate the profits of both. Then, we analyze the Nash Equilibrium to obtain the optimal resource demand and resource price based on the profit formula.

- Considering that users may have insufficient storage, we propose that they can cache cryptographic hashes of blocks in ECSP.

The rest of this paper is organized as follows: Section 2 presents the related work. Section 3 analyzes the system model. Section 4 gives the optimization of the proposed problem. The experimental results and corresponding discussions are displayed in Section 5, followed by conclusions in Section 6.

2 Related work

Recently, several mechanisms have been proposed to support mobile blockchain mining.

Xiong et al. [Xiong, Zhang, Niyato et al. (2018)] proposed the concept of mobile blockchain, which combines blockchain with edge computing and uses edge computing resources to support blockchain operation. They modeled the interactions between miners and ECSP as a Stackelberg game. First, the blockchain miner calculates the number of resources requested from the edge computing based on the strategy of other miners, and then, the ECSP calculates the optimal price of the edge computing resource.

Han et al. [Han, Su and Chen (2018)] introduced an architecture called Jupiter. It is an edge computing-based Blockchain platform providing storage for mobile blockchain. Their system is consisted of application level, blockchain core level and infrastructure level. The infrastructure level is consisted of multiple servers with permanent nodes to meet the basic needs of the network and mitigate cold-start issues. Blockchain is the core of the entire system, it provides network communication, transaction broadcast and block allocation. The highest application level provides an interface for users to develop blockchain applications.

Auction method has also been widely adopted for resource allocation. An auction method based on deep learning for edge computing resource allocation was proposed in Luong et al. [Luong, Xiong, Wang et al. (2018)]. Jiao et al. [Jiao, Wang, Niyato et al. (2018)] studied the auction mechanism in the mobile blockchain to maximize the network welfare. Social welfare is described as the revenue of the whole blockchain network. When miners decide their bids, the system chooses the winners until social welfare decreases.

These recent researches have achieved great results and been widely adopted. However, they only consider the use of ECSP's resources. Combining the advantages of these studies and joining our innovations, we propose to offload mining tasks to either nearby edge devices or the ECSP and cache the encrypted hashes of blocks in the ECSP.

3 System and network model

3.1 Network model

The network model is shown in Fig. 1, there are some CMNs under the blockchain and this section considers the CMN with miners$\{1, \cdots, M\}$ and resource sharers$\{1, \cdots, S\}$. The expected mining resources of miners is $\{r_1, \cdots r_M\}$ and the resources of resource providers is $\{c_1, \cdots, c_s\}$. Therefore, the hashing power of miner i is $r_i / \sum_{i=0}^{M} r_j$.

The blockchain is consisted of multiple connected blocks, each block containing transaction information and the hash value of the block. Blockchain can be replicated and propagated to all participants in the publicly accessible blockchain network to synchronize globally. Consensus process guarantees blockchain to be confirmed by most users. Users in the network reach the protocol of the new block added to the blockchain. However, the consensus process may not be able to prevent blockchain from being attacked. For example, the attacker creates sham blockchain information and many fake user names in the network. This may lead to a consensus on false transactions generated by an attacker.

Proof of Work (PoW) is a process difficult to produce but easy to verify, which can solve the problem. In a blockchain network with high mining costs, an attacker needs to control 51 percent of the network's computing power to control the blockchain [Cho (2018)]. For the attacker, this is a huge cost, which is difficult to achieve in practice. Considering that the consensus and PoW computing time for a transaction requires a fixed amount of computation, the time is assumed linear to the number of transactions t_i in the block, expressed as $\tau = lt_i$, where l is a parameter determined by the average verification speed of nodes and network scale. Since the generation of new blocks follows a Poisson process, miner I's possibility of mining failure can be described as $P_O = 1 - e^{-\lambda lt_i}$, where λ is a process parameter referring to the complexity of mining a block. Obviously, the probability of miner i successfully mining to generate a block is

$$P = h_i \times (1 - P_O) = \frac{r_i}{\sum_{j \in M} r_j} e^{-\lambda lt_i} \tag{1}$$

Since the blockchain mining process set high requirements for storage space and computing resources, the IoT devices can offload their mining tasks to nearby edge computing nodes or the ECSP. As shown in Fig. 1, adjacent edge computing nodes can form an auction market, and miners can bid for nearby device's resources. For edge computing service, miners can negotiate prices with the ECSP and rent the resources. What's more, the cryptographic hashes of blocks can be cached in the ECSP.

Miner i successfully mining will gain a corresponding mining reward. The reward is consisted of a fixed reward R and a commission reward defined as $r \times t_i$, in which t_i represents the number of mining transactions and r is the reward for unit transaction (reward rate). Additionally, miner is charged for using resources of ECSP and nearby devices. Therefore, miner i's utility is formulated as

$$U_m = (R + rt_i)\frac{r_i}{\sum_{j \in M} r_j} e^{-\lambda lt_i} - pr_i^e - p_i^j(r_i - r_i^e) \tag{2}$$

To sum up, the mechanism works in the following steps：

- The IoT devices execute the transaction and the new transaction will be spread to the

neighbor peer nodes in the blockchain network.

- The nodes accepting the transaction in the entire network execute the consensus algorithm on the block and performs mining by solving PoW.
- The IoT devices offload their mining tasks to nearby edge devices or the ECSP.
- The IoT devices cache the cryptographic hashes of blocks in the ECSP. When they need to check the blockchain information, they download it from the service provider.
- After passing the consensus algorithm processing, the transaction becomes a valid transaction and a new block is generated to confirm the transaction.
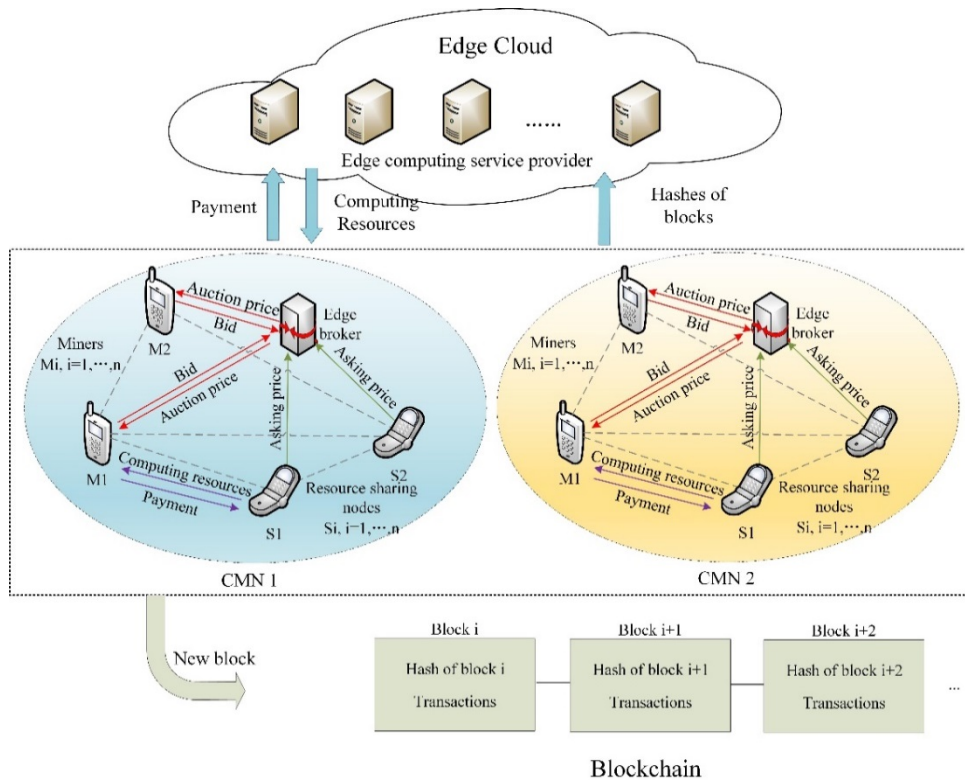

Figure 1: Mining offloading to neighbor devices and ECSP, and block caching in ECSP

3.2 Offloading model

In this subsection, we promote two offloading modes for devices in the IoT blockchain network.

3.2.1 Mining offloading to the neighbor devices

We assume that all devices in the IoT blockchain network have the computing power to execute mining tasks. Therefore, the compute-intensive mining task can be offloaded to the neighbor devices. We use auction methods to solve the problem of competition for

resources between mining devices. Devices that share resources first announce their selling price, and miners bids for the resources. The buyer's bid price $\{b_1, \cdots, b_M\}$ will be sorted in ascending order and the seller's auction price $\{s_1, \cdots, s_S\}$ will be sorted in descending order. If there is a k satisfying the condition that $b_{k-1} < s_{k-1}$ and $b_k > s_k$, the auction price between buyer k and seller k is set to $(s_k + b_k)/2$. The remaining buyers and sellers repeat the above steps.

3.2.2 Mining offloading to ECSP

For the optimization of mining offloading to ECSPs, we model the interactions between the two parties as a Stackelberg game. Miners want to achieve their maximum utility while ECSP is devoted to maximizing profits. Therefore, to adjust the computing resource demands and the resource price, this process is modeled as a Stackelberg Game. The ECSP as the leader first announce the price of the unit resources to miners. Miners as the followers then determine their mining resources demands based on other miners' strategies and prices announced by ECSPs. Depending on the resource allocation, ECO gives the optimal price for maximum profit.

$$U_e = \sum_{i=0}^{M} \mathrm{p} r_i^e - \sum_{i=0}^{M} \mathrm{B} r_i^e \tag{3}$$

3.3 Caching model

When the node does not have enough storage locally, it can choose to upload the cryptographic hash of the block to ECSP. Correspondingly, the payment to the ECSP is linearly related to its storage capability. In this model, we only consider the reward of caching of block cryptographic hashes. Therefore, the reward is linearly related to transaction number. The corresponding problem is similar to the task offloading, so this paper only considered the optimization of the task offloading.

4 Optimization of task offloading

Since miners aim at obtaining the maximum utility and ECSP is devoted to achieve the maximum profit. The interaction between the ECSP and miners is modeled as a two-stage Stackelberg game.

We first analyze miners' mining strategies in Stage II: Given the prices of the ECSP and other miners' strategies, miner i decides on its resource demand.

1) Stage II: Miners' Resource Demand Game: Given the price p decided by the ECSP, the miners compete with each other to maximize their own utility by choosing their own resource demand, which forms the non-cooperative Resource Demand Game. The optimization problem of miners can be described as

$$\boldsymbol{max} \quad U_m = (R + rt_i) \frac{r_i}{\sum_{j \in M} r_j} e^{-\lambda l t_i} - \mathrm{p} r_i^e - p_i^j (r_i - r_i^e) \tag{4}$$

We derive the first and second derivatives of (4) with respect to r_i, which can be written as follows:

$$\frac{\partial U_m}{\partial r_i} = (R + rt_i)e^{-\lambda lt_i}\frac{\sum_{i \neq j} r_i}{(\sum_{i=0}^{M} r_i)^2} - p \tag{5}$$

$$\frac{\partial^2 U_m}{\partial^2 r_i} = -2(R + rt_i)e^{-\lambda lt_i}\frac{\sum_{i \neq j} r_i}{(\sum_{i=0}^{M} r_i)^3} \tag{6}$$

We can see that the first derivatives of (4) is positive and second derivatives of (4) is negative. We have proved that U_m is strictly concave with respect to r_i. Therefore, the Nash equilibrium in the game exists. Further simplifying, the optimal resource demand of miner i is

$$r^* = \frac{N-1}{\sum_{i=0}^{M} \frac{pe^{-\lambda lt_i}}{(R+rt_i)}} \tag{7}$$

2) Stage I: ESP's Profit Maximization: Based on the Nash equilibrium of the resource demand in Stage II, the service provider (leader) can optimize its pricing strategy in Stage I to maximize its profit defined in (3). By substituting (7) into (3), we get the maximum profit of the leader as

$$\textbf{max} \quad U_e = \sum_{i=0}^{M} (p - B)\frac{N-1}{\sum_{i=0}^{M} \frac{pe^{-\lambda lt_i}}{(R+rt_i)}} \tag{8}$$

We derive the first and second derivatives of (8) with respect to p, which can be written as follows:

$$\frac{\partial U_e}{\partial p} = \sum_{i=0}^{M} \frac{B}{p^2}\frac{N-1}{\sum_{i=0}^{M} \frac{e^{-\lambda lt_i}}{(R+rt_i)}} \tag{9}$$

$$\frac{\partial^2 U_e}{\partial^2 p} = -2\sum_{i=0}^{M} \frac{B}{p^2}\frac{N-1}{\sum_{i=0}^{M} \frac{e^{-\lambda lt_i}}{(R+rt_i)}} \tag{10}$$

Since the first derivative of (8) is positive and second derivative of (8) is negative, U_m is strictly concave with respect to p. Therefore, the optimal price exits.

5 Evaluation analysis

In this section, we compare the performance of the proposed distributed mechanism with that of the mode A (all the miners offload their tasks to the ECSP), mode B (all the miners offload their tasks to a group of neighbor devices) and the centralized scheme. Note that the simulation results obtained in this section is the average value of multiple simulations under various parameters. In the simulation, the network coverage radius is set to 500 m, where the service provider density are assumed to be $10^5/m^2$, $10^4/m^2$ and $10^3/m^2$, respectively. The transmit power of the users is 0.1 W, the bandwidth is 20 MHz, the delay threshold is 10 s, the computation intensity is 15000 CPU cycles/bit and the unit price of energy is 0.1 token/J.
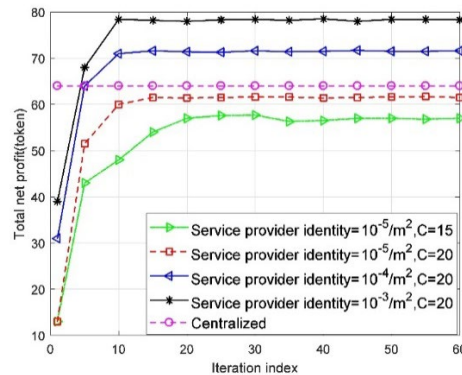
Figure 2: Convergence performance of the proposed mechanism under different service provider densities and storage capacities

First, we compare the convergence performance of the mechanism under different service provider densities and storage capacities. In Fig. 2, the profit increases sharply in the first 10 iterations, and then reaches a stable state in the first 30 iterations, which indicates that our mechanism can rapidly converge. In addition, with the increase in service provider densities, the total benefit increases because miners who choose the A mode can choose closer service provider and shorten the transaction delay, which will decrease energy consumption and increase total net income. What's more, total net profit increases as storage capacity C increases. This is because when the storage capacity increases, the miner has more space to cache the block's encrypted hash, thereby increasing the total net profit. In the decentralized case, the net profit stay the same at different iteration index.

We compare the unloading/caching decisions in different number of miners in Fig. 3. As the number of miners increases and competition for mining becomes more intense, more miners choose to cache the hash of the block at the ECSP. Besides, when the number of miners increases, the percentage of mode B increases. This is because the mining competition becomes fiercer and it is not enough to offloading only to the ECSP. It can be seen that due to the different channel conditions and calculation task sizes of different miners, the optimal strategies made by them to reach the maximum of total net income are not consistent. This is because miners can choose the most beneficial mode for offloading.
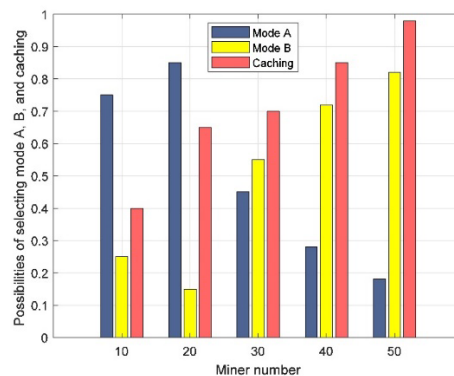

Figure 3: Comparison of offloading/caching decisions under different number of miners
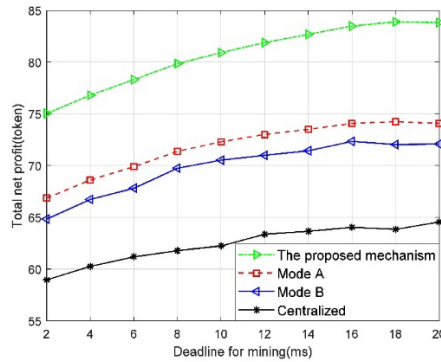
Figure 4: Comparison of total net profit of the four modes

Finally, we compare the delay and net profit of the four modes under different parameters. As shown in Fig. 4, when there is an increase in deadline for mining, the total net profits increase in all modes due to that the mining success rate increases. In addition, our mechanism has the highest net profit, higher than the mode A, B, and the net profit is the lowest in the case of centralized mode.
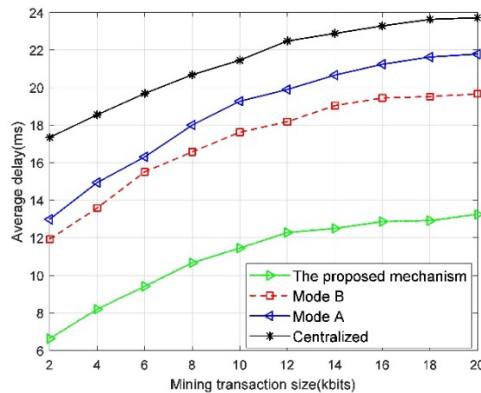

Figure 5: Comparison of average delay of the four modes

Fig. 5. displays that as the mining transaction size increases, the average delay increases. The average delay is minimal under our mechanism and the highest in centralized mode. And due to latency constraints, it is hard for a single resource-constrained ECSP to process a large number of requests from users at once. Therefore, the mining process will take more time, especially when the transaction size is large. However, the mode B scheme in which a group of users shares computing resources to each other can reduce the transmission delay, thereby reducing the overall delay in completing the mining task.

6 Conclusion

In the paper, we integrate blockchain technology into constructing trusted IoT platforms. To address the problem that IoT devices have limited resources to complete mining tasks, we suggest offloading mining tasks to nearby edge computing nodes or the ECSP and caching the encrypted hashes of blocks in the ECSP. Moreover, we model the process of

task offloading and block caching to ensure that both edge devices and ECSP obtain the maximum profit based on game theory and auction theory. Simulation results show that under the proposed mechanism, mining networks obtain more profits and consume less time on average.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Baktir, A. C.; Ozgovde, A.; Ersoy, C. (2017): How can edge computing benefit from software-defined networking: a survey, use cases, and future directions. *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2359-2391.

Cha, S.; Chen, J.; Su, C.; Yeh, K. (2018): A blockchain connected gateway for BLE-based devices in the internet of things. *IEEE Access*, vol. 6, pp. 24639-24649.

Chen, X.; Jiao, L.; Li, W.; Fu, X. (2016): Efficient multi-user computation offloading for mobile-edge cloud computing. *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795-2808.

Cho, H. (2018): Asic-resistance of multi-hash proof-of-work mechanisms for blockchain consensus protocols. *IEEE Access*, vol. 6, pp. 66210-66222.

Christidis, K.; Devetsikiotis, M. (2016): Blockchains and smart contracts for the internet of things. *IEEE Access*, vol. 4, pp. 2292-2303.

Han, S.; Xu, Z.; Chen, L. (2018): Jupiter: a blockchain platform for mobile devices. *IEEE 34th International Conference on Data Engineering*, pp. 1649-1652.

Jiao, Y.; Wang, P.; Niyato, D.; Xiong, Z. (2018): Social welfare maximization auction in edge computing resource allocation for mobile blockchain. *IEEE International Conference on Communications*, pp. 1-6.

Liu, M.; Yu, F. R.; Teng, Y.; Leung, V. C. M.; Song, M. (2018): Computation offloading and content caching in wireless blockchain networks with mobile edge computing. *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 11008-11021.

Luong, N. C.; Xiong, Z.; Wang, P.; Niyato, D. (2018): Optimal auction for edge computing resource management in mobile blockchain networks: a deep learning approach. *IEEE International Conference on Communications*, pp. 1-6.

Min, X.; Li, Q.; Liu, L.; Cui, L. (2016): A permissioned blockchain framework for supporting instant transaction and dynamic block size. *IEEE Trustcom/BigDataSE/ISPA*, pp. 90-96.

Qin, R.; Yuan, Y.; Wang, F. (2018): Research on the selection strategies of blockchain mining pools. *IEEE Transactions on Computational Social Systems*, vol. 5, no. 3, pp. 748-757.

Samaniego, M.; Deters, R. (2016): Blockchain as a service for IoT. *IEEE International Conference on Internet of Things*, pp. 433-436.

Swan, M. (2015): Blockchain: blueprint for a new economy. *O'Reilly Media*.

Wang, L.; Yong, L. (2015): Exploring miner evolution in bitcoin network. *Passive and Active Measurement, Lecture Notes in Computer Science*, vol. 8995.

Xiong, Z.; Zhang, Y.; Niyato, D.; Wang, P.; Han, Z. (2018): When mobile blockchain meets edge computing. *IEEE Communications Magazine*, vol. 56, no. 8, pp. 33-39.