Future Generation Computer Systems 132 (2022) 11–24

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Truthful auction mechanisms for resource allocation in the Internet of Vehicles with public blockchain networks

Jixian Zhang [a], Wenlu Lou [a], Hao Sun [a], Qian Su [a], Weidong Li [b,*]

[a] School of Information Science and Engineering, Yunnan University Kunming, Yunnan 650504, PR China
[b] School of Mathematics and Statistics, Yunnan University Kunming, Yunnan 650504, PR China

ARTICLE INFO

Article history:
Received 16 June 2021
Received in revised form 10 November 2021
Accepted 5 February 2022
Available online 11 February 2022

Keywords:
Blockchain
Internet of vehicles
Edge computing
Resource allocation
Mechanism design

ABSTRACT

Recording vehicle driving data into the blockchain can effectively solve problems of data authenticity and security, which is a focus of research on blockchain and the Internet of Vehicles. However, when using blockchain technology, the proof-of-work completed by vehicles may consume substantial energy and computing resources, which limits the application of blockchain technology in the internet of vehicles environment. Therefore, this paper considers deploying edge computing nodes to support blockchain technology and introducing an auction mechanism to encourage users to record vehicle driving data as miners. This paper proposes two auction mechanisms for the blockchain network formed by edge computing service providers and miners to maximize the social welfare. Specifically, one mechanism is used when the resource demands of the miners are the same, and the other is used when the resource demands of the miners are different. For resource allocation, the former uses the maximum cost maximum flow algorithm to achieve optimal allocation, and the latter uses a heuristic algorithm. For price payment, the former uses the Vickrey–Clarke–Groves mechanism, and the latter uses dichotomy. These two price payment algorithms use critical value theory to calculate the payment price. This paper demonstrates that both are truthful and individually rational. Through experiments, this paper evaluates indicators such as social welfare, satisfaction, and resource utilization. The experiments show that the proposed auction mechanism can effectively maximize social welfare in the blockchain network and provide an effective resource allocation strategy for edge computing service providers.

1. Introduction

In recent years, with the development of the economy, global car ownership has shown a gradually increasing trend. The industrialization and popularization of the Internet of Vehicles (IoV) is of great significance for the construction of a harmonious automobile society and smart cities [1]. For example, the IoV system can collect and save data about the operation of devices. When a vehicle breaks down and causes customer losses, the data can be used to obtain the truth; by collecting the driver's operating data, the driver can be provided with a corresponding driving behavior analysis report to ensure the safety of the vehicle. However, current vehicle driving data are mainly recorded by automobile manufacturers or not recorded, so the integrity and authenticity of the data cannot be guaranteed when using the data. For example, the "Tesla brake failure" incident in China has attracted the attention of many people. A car owner complained about an accident due to a brake failure in a certain Tesla model, and she began to defend her rights. Tesla eventually chose to settle with the owner by repurchasing the vehicle involved in the accident and coming to an agreement with her, but this does not mean Tesla vehicles themselves have no safety hazards. Therefore, it is necessary to introduce a safe and reliable technology to solve the authenticity problems with information records in the IoV environment. The emergence and development of blockchain technology and smart contracts provide feasible solutions for this problem; smart contracts are programs deployed on the blockchain that are run by all nodes in the blockchain network. The result of program execution is obtained by all nodes through a consensus mechanism and is not determined by the execution result of a single node. Therefore, the access control logic of the IoV can be written into a smart contract and deployed to the blockchain. Blockchain nodes determine the users' access rights according to the access control logic predefined by the smart contract, reach a consensus on the results of access control through the consensus mechanism, and store data in a distributed way, which effectively solves the problem of privacy, trust and authenticity in vehicle driving data.

* Corresponding author.
E-mail addresses: denonji@163.com (J. Zhang), louwenlu@mail.ynu.edu.cn (W. Lou), sunhao@mail.ynu.edu.cn (H. Sun), suqian@ynu.edu.cn (Q. Su), weidongmath@126.com (W. Li).

https://doi.org/10.1016/j.future.2022.02.002
0167-739X/

In theory, the integration of the IoV and blockchain technology can greatly improve traffic efficiency and safety [2]. However, the integration of the IoV and blockchain technology has certain prerequisites; that is, the miners in the blockchain network need to solve a preset proof-of-work (PoW) problem to add new data to the blockchain. However, PoW requires considerable central processing unit time and energy and is not suitable for vehicle-mounted mobile devices with limited resources. Edge computing has the advantage of low latency to achieve a shorter response time, as well as the potential to address the concerns of energy consumption, bandwidth burden and security issues [3]. Therefore, it is a reasonable to offload the blockchain task to the edge computing server for execution.

However, the computing resources on the edge computing server are still limited, so a reasonable mechanism needs to be designed to use the resources on the edge server [4]. The auction mechanism can promptly respond to changes in the supply and demand of resources in the market, solve the problem of competitive resource allocation, and provide resources to users at a more reasonable price. Therefore, combining the auction mechanism with the blockchain can attract more people to participate in the blockchain network, which can stabilize the blockchain network as soon as possible.

A very valuable application is the use of blockchain technology to record and save the driving data of vehicles for the realization of intelligent traffic management services. To facilitate the application of blockchain in the IoV system, the advantages of blockchain can be utilized to quickly establish a self-organizing data management platform to support various decentralized applications (DApps). DApp providers use a reward mechanism to motivate people to undertake the task of resource provision and system maintenance. To alleviate the computing bottleneck, a mining task can be offloaded by accessing an edge computing service and allocating the resources on the edge computing server to each miner through the auction mechanism. Since the edge computing service can generate more consensus nodes to perform mining tasks, this will significantly improve the robustness of the blockchain network, thereby attracting more DApp users to join, forming a virtuous circle.

Based on the above background, this paper studies the design of a blockchain-based IoV resource allocation auction mechanism. The major contributions can be summarized as follows:

- Using blockchain technology, the real-time data of vehicle travel in the IoV are recorded, and a mathematical programming model is established between the edge computing service providers (ECSPs) and miners.
- To encourage users to participate in block recording as miners to maximize social welfare, two auction mechanisms, namely, a constant-demand auction mechanism (CDAM) and a multidemand auction mechanism (MDAM), are designed.
- For the CDAM, an optimal resource allocation algorithm based on maximum cost maximum flow and an optimal Vickrey–Clarke–Groves (VCG) price payment algorithm are proposed. In addition, this paper demonstrates the economic characteristics of the CDAM.
- For the MDAM, a resource allocation algorithm based on the degree priority heuristic method and a price payment algorithm based on dichotomy are proposed. In addition, this paper demonstrates the economic characteristics of the MDAM.
- The experiments show that the proposed auction mechanism is better than the first fit and the best fit in terms of social welfare, satisfaction, and resource utilization.

Table 1
Frequently used notation.

Notation	Description
$\mathcal{N}$, N	Set of miners and the total number of miners
$\mathcal{M}$, M	Set of servers and the total number of servers
$0 < \alpha < \beta < 1$	Demand constraints of multidemand miners
D	Total resources of the server
T	Constant bonus from mining a new block
r	Transaction fee rate
λ	Average block time
s_i	Block size of miner i
d_i	Demand of miner i
b_i	Bid of miner i
$\mathbf{X}$, x_{ij}	Resource allocation solution and allocation result

The rest of this paper is organized as follows: Section 2 reviews the recent related work. Section 3 introduces the resource allocation problem of the IoV based on blockchain. Section 4 introduces the mechanism design preliminaries. Section 5 discusses the optimal algorithm of social welfare maximization under the CDAM, as well as the approximate optimal algorithm of social welfare maximization under the MDAM. Section 6 presents the experimental results under the blockchain environment and an economic property analysis of the proposed auction mechanism. Section 7 concludes the paper. Table 1 lists the notation frequently used in this paper.

2. Related work

As cars and highways become increasingly intelligent, an increasing number of cars and roadside infrastructures are being equipped with communication devices. The development of the whole IoV and related applications for the IoV has become an inevitable trend. Resource allocation is an important part of supporting IoV technology, and a large number of researchers worldwide have conducted in-depth research on IoV resource allocation. [5] used the Markov decision process method to determine whether a task should be executed locally or on a mobile edge computing (MEC) server, analyzed the average delay of each task for vehicles and the average power consumption of mobile devices, formulated a power delay constrained minimization problem, and proposed an efficient one-dimensional search algorithm to identify the optimal task scheduling strategy, which reduces the average execution latency. [6] adopted a combination of cloud computing and edge computing to improve the efficiency of edge clouds and proposed an optimization scheme for joint communication and computing resource allocation, which achieved better delay performance than [5]. Based on collaboration between cloud computing and mobile edge computing, [7] proposed dynamic voltage and frequency scaling technology to enable virtual machines to dynamically scale their computing frequency on demand, which effectively improves the computing energy efficiency when processing vehicle tasks. [8] proposed a device-to device (D2D) relay algorithm for joint resource allocation and power control based on energy efficiency. First, resources are allocated to the equivalent D2D relay link to reduce the algorithm complexity and minimize the interference of the D2D link with the cellular link. Then, relay selection is carried out according to the resource allocation result and power control algorithm. As the separated MEC server in [8] can only meet the needs of a few users, [9] therefore proposed a resource allocation scheme in the MEC server collaboration space, establishing an MEC core server, storing and providing the latest resource information for other servers, and optimizing the transmission delay of vehicle offload tasks according to actual applications. However, these papers did not consider the privacy information protection

issues in the IoV or the authenticity of the driving data, so this paper studies these related issues of the IoV based on blockchain.

At present, the emergence of blockchain technology has become a unique and trending but most disruptive technology. Simultaneously, the Internet of Vehicles based on blockchain has also become the focus of academics. Unfortunately, there are potential security threats because various entities in an IoV environment communicate over public channels. As a result, [10] proposed a blockchain-based trust model that curbs malicious activities in the IoV, which substantially guarantees efficient network performance while also ensuring that there is trust among the entities. [11] proposed a novel framework of blockchain-enabled IoV and cooperative positioning for improving vehicular GPS positioning accuracy, system robustness, and security. [12] proposed an improved authentication scheme for IoV based on blockchain technology and verified the feasibility of the scheme in reducing the selfish behavior and malicious attacks in IoV. Furthermore, when some sensing tasks are outsourced to vehicles, they are usually unwilling to participate in sensing tasks. Therefore, [13] proposed a novel time-window-based method to manage the tasks among vehicles and to encourage the vehicles to participate. To incentivize standby miners to participate in block verification, [14] proposed implementing contract theory to model the interaction between active miners and standby miners while considering the security and delay of block verification. However, these works mainly focus on blockchain-based IoV from the perspective of security and pay little attention to the deployment of the blockchain network and the corresponding resource allocation auction mechanism in the IoV.

Using the blockchain-based auction mechanism for resource allocation can enable resource providers to obtain greater benefits while ensuring security, which has been widely used in the field of power and spectrum resource allocation. In addition, parties on both sides of the transaction will consider fairness, efficiency, cost and other factors. [15] proposed an electric vehicle (EV) power trading model based on blockchain and a smart contract. The EV trading parties use the reverse auction mechanism based on a dynamic pricing strategy to complete the transaction matching, which can not only improve the profit of the less competitive power seller but also reduce the cost of the electricity purchaser. To further reduce the transaction cost of blockchain, increase the transaction efficiency, and solve the problem of the lack of privacy protection for a continuous double auction in the existing scheme, [16] proposed a privacy protection scheme of microgrid direct electricity transactions based on consortium blockchain and continuous double auctions. A double auction is an auction in which multiple buyers and sellers seek a price where supply and demand balance. Since the double auction based on secure multiparty computation cannot guarantee its fairness, [17] proposed a blockchain-based fair and secure double auction protocol. In an auction mechanism, VCG is usually used to calculate the optimal price for the user to pay. [18] designed a blockchain-based smart contract system for the trading market of overall power resources to ensure each participant's transaction integrity. [19] developed a private energy auction for blockchain-based microgrid systems (DEAL) and compared DEAL with the VCG auction scenario. The experimental results demonstrate that DEAL outperforms the VCG mechanism by maximizing sellers' revenue while maintaining overall network benefit and social welfare. [20] considered deploying edge computing services to support the mobile blockchain, and proposed an auction-based edge computing resource allocation mechanism for the edge computing service provider. Based on [4,20] focused on the trade-off between the cloud/fog computing service provider and miners, and proposed an auction-based market model for efficient computing resource allocation. This article is inspired by these two articles, but the difference is that the background of our research is in the field of IoV, and the vehicle locations will change in real time. Therefore, it is more important that we consider the deployment constraints between vehicles and ECSs in the problem model.

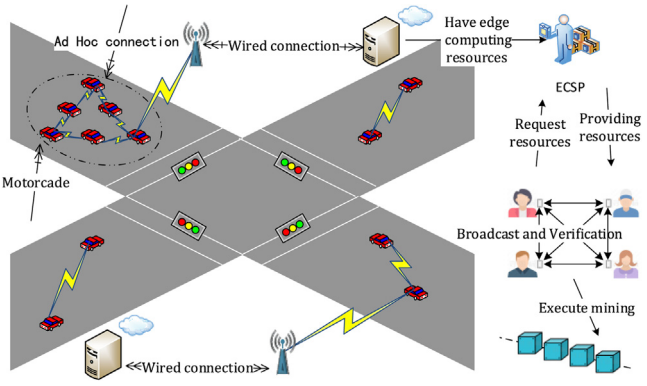

Fig. 1. IoV based on blockchain.

In summary, the current research on the IoV based on blockchain has achieved certain results. However, little attention is paid to the issue of blockchain-based resource allocation and the economic properties of auctions or to allocative externalities in the blockchain network. Therefore, this paper designs two auction mechanisms for resource allocation in the IoV based on blockchain. This paper takes into account the externalities of the blockchain network configuration by means of a resource allocation algorithm to achieve the network economic goal of maximizing social welfare under the two auction mechanisms.

3. Blockchain-based Internet of Vehicles resource allocation problem

This paper uses blockchain to record the real-time driving data of a vehicle to ensure the authenticity of the received data. Assume that in the IoV, each vehicle sends its driving data (such as the driving trajectory, speed, mileage, location, and driver operations) to the cloud or edge server for miners to record on the blockchain. It is worth noting that connecting to cloud servers and edge servers is transparent for users. The difference is that moving vehicles have deployment constraints on edge servers but not for the cloud, which means that any vehicle user can be connected to the cloud at any time; therefore, the cloud platform can be regarded as a special fully connected edge server. A typical IoV scenario is shown in Fig. 1.

3.1. Parameters and definitions

The model in Fig. 1 is based on following assumptions. There are N users acting as miners, where miners $i \in \mathcal{N} = \{1, 2, \ldots, N\}$ and each miner runs a blockchain-based DApp to record and verify the transactional data sent to the blockchain network. Due to the insufficient computing power of miners' devices, the rational miners offload the task of solving the PoW to M edge servers, where edge servers $j \in \mathcal{M} = \{1, 2, \ldots, M\}$. Usually, an edge server is deployed together with a base station, and the signal of the base station determines the coverage area of the edge server, so miners cannot connect to all edge servers. Use $\Delta = \begin{bmatrix} \delta_{11}, \delta_{12}, \ldots, \delta_{1M} \\ \delta_{21}, \delta_{22}, \ldots, \delta_{2M} \\ \ldots\ldots \\ \delta_{N1}, \delta_{N2}, \ldots, \delta_{NM} \end{bmatrix}$ to describe the deployment constraints of the miners, where $\delta_{ij} = 1$ means that miner i's task

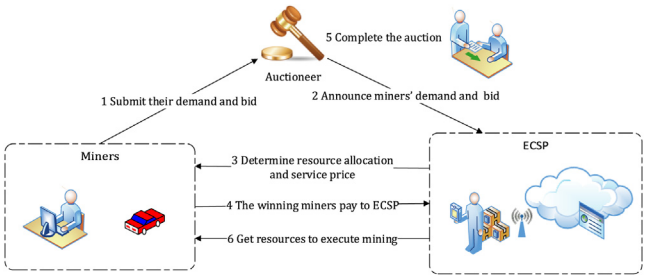

Fig. 2. Auction flowchart.

can be deployed to server j for processing; otherwise, $\delta_{ij} = 0$. Set $\mathcal{C} = \{c_1, c_2, \ldots, c_j, \ldots, c_M\}$ to represent the resources of the server, where c_j represents the capacity of server j and the total resource capacity of all ECSP servers is $D = \sum_{j=1}^{M} c_j$. The ECSP (the seller) sells the computing service, and the miner (the buyer) uses the service by remote access of the edge server. To achieve a dynamic balance between supply and demand in the market and encourage more users to participate in mining, the ECSP uses auctions to allocate resources (the transaction process is shown in Fig. 2). The ECSP first announces the rules of the auction and the available resources to the miners, and then the miners submit their resource demand profiles $\mathcal{D} = \{d_1, d_2, \ldots, d_j, \ldots, d_N\}$ and the corresponding bid profiles $\mathcal{B} = \{b_1, b_2, \ldots, b_j, \ldots, b_N\}$ to the auctioneer. For example, $d_i = 1$ indicates that user i needs 1 unit of computing resources. After receiving the resource demands and bids from the miners, the auctioneer selects the winning miners and notifies all miners of the allocation solution $\mathbf{X} = \begin{bmatrix} x_{11}, x_{12}, \ldots, x_{1M} \\ x_{21}, x_{22}, \ldots, x_{2M} \\ \ldots\ldots \\ x_{N1}, x_{N2}, \ldots, x_{NM} \end{bmatrix}$ and payment price solution $\mathbf{P} = \begin{bmatrix} p_{11}, p_{12}, \ldots, p_{1M} \\ p_{21}, p_{22}, \ldots, p_{2M} \\ \ldots\ldots \\ p_{N1}, p_{N2}, \ldots, p_{NM} \end{bmatrix}$, where $x_{ij} \in \{0, 1\}$.

$x_{ij} = 1$ indicates that miner i's demands are deployed to edge server j for execution, in which case, the edge servers allocate resources to the miners for mining; and $x_{ij} = 0$ denotes no resources are allocated. The payment for a miner who fails the auction is set to zero; i.e., $p_{ij} = 0$ if $x_{ij} = 0$. Assuming that miners are single-minded [21], that is, that each miner accepts only its requested quantity of resources or none. At the end of the auction, the selected miners or winners make the payment according to the price assigned by the ECSP and access the edge computing service to execute mining.

3.2. Problems and models

This paper was inspired by [4] to model the interaction between the ECSP and miners. With allocation x_{ij} and demand d_i, miner i's hash power γ_i can be calculated from [4]:

$$\gamma_i = \frac{\sum_{j=1}^{M} d_i x_{ij}}{D} \tag{1}$$

where $\sum_{i=1}^{N} \gamma_i \leq 1$, $x_{ij} \in \{0, 1\}$, and $D = \sum_{j=1}^{M} c_j$. Hash power indicates the percentage of computing power of each user in the entire blockchain network. The higher the computing power is, the greater the probability of successful mining. During the mining process, miners race to be the first to solve the PoW with the correct hash value and spread the block to reach a consensus. However, it is possible that multiple users have completed the PoW of the block at the same time, but only one user's block can be recorded, and the other blocks need to be discarded (orphan blocks). Therefore, for each user, the block that he solves has a probability of becoming an orphan block. Obviously, the orphaning probability is directly related to the average block time and the block size. The orphaning probability of miner i is defined as follows:

$$P_i^o = 1 - e^{-\frac{1}{\lambda}\tau_i} \tag{2}$$

In formula (2), λ is the average block time. According to the statistics shown in [22], the time τ_i for miner i to propagate the block is linearly related to the block size; i.e., $\tau_i = \xi s_i$, $\xi > 0$, where the block size of each miner is expressed as $\mathcal{S} = \{s_1, s_2, \ldots, s_j, \ldots, s_N\}$. Before mining starts, each miner puts unconfirmed transaction data into its block. When miner i broadcasts its blocks to the blockchain network and reaches a consensus, the time for broadcasting and verifying each transaction is affected by the block size s_i. The first miner who successfully reaches a consensus can obtain a reward R_i, which consists of a constant reward $T \geq 0$ for mining the block and a variable reward rs_i, where r is a predefined transaction fee rate. Therefore, the reward for miner i can be expressed as:

$$R_i = (T + rs_i) P_i(\gamma_i, s_i) \tag{3}$$

where $P_i(\gamma_i, s_i)$ is the probability that miner i receives the reward for contributing a block to the blockchain:

$$P_i(\gamma_i, s_i) = \gamma_i (1 - P_i^0) = \gamma_i e^{-\frac{1}{\lambda}\xi s_i} \tag{4}$$

3.3. Social welfare maximization

When miner i makes a valuation b_i for the computing services of the ECSP, b_i is valuated based on the network effect of the blockchain (the principle of the valuation is: the greater the authenticity and robustness of the blockchain network are, the more valuable the blockchain network itself. Hence, more miners participate in the blockchain network, and the miners can obtain more rewards.) By fitting the experimental data, the network utility function is defined as follows [20]:

$$\omega(d_N) = \frac{1 - e^{-\nu d_N}}{1 + \mu e^{-\nu d_N}}. \tag{5}$$

The network utility function can reflect the stability of the blockchain network, which means that when the number of people is large enough, each user can obtain a greater and more stable revenue. The network effect function is a monotone concave function, where $d_N = \sum_{i=1}^{N} \sum_{j=1}^{M} d_i x_{ij} \in [0, D]$, $\omega \in [0, 1]$, and $\mu, \nu > 0$ is the curve fitting parameter. Since miner i cannot know the number of winning miners or the total allocated resource quantity of the ECSP until the end of the auction, assuming that miner i can only give the bid b_i according to its expected reward and demand without considering network effects and other miners' demands, i.e., setting $\gamma_i = 1$, so miner i's ex ante valuation is:

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

Since miner i makes bid b_i based on the reward R_i it receives, v_i' is the ex ante valuation, so miner i's bid can be considered $b_i = v_i'$. From an economic point of view, this assumption is reasonable because when a user participates in a market behavior, he will first evaluate the most likely revenue from the market and use this to determine his investment.

After the auction is over, the miner receives the allocation result, and miner i's ex post valuation in the context of the overall network effect can be obtained as:

$$v_i'' = v_i' \omega \gamma_i = (T + rs_i) e^{-\frac{1}{\lambda}\xi s_i} \frac{1 - e^{-\nu d_N}}{1 + \mu e^{-\nu d_N}} \frac{\sum_{j=1}^{M} d_i x_{ij}}{D} \tag{7}$$

The ex post valuation means that the value of the user's investment is truly reflected in the market. It is closely related to the stability of the market and the behavior of other users. Therefore, the social welfare maximization problem is transformed into the following nonlinear integer programming problem:

$$\begin{aligned} Objective: \max S &= \sum_{i=1}^{N}\sum_{j=1}^{M} v_i''\delta_{ij} \\ &= \sum_{i=1}^{N}\sum_{j=1}^{M} v_i'\omega\gamma_i\delta_{ij} \\ &= \sum_{i=1}^{N}\sum_{j=1}^{M} b_i \frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}} \frac{\sum_{j=1}^{M} d_i x_{ij}}{D}\delta_{ij} \end{aligned} \tag{8}$$

$$Subject\ to: \sum_{j=1}^{M} x_{ij}\delta_{ij} \le 1 \tag{8a}$$

$$\sum_{i=1}^{N} d_i x_{ij}\delta_{ij} \le c_j \tag{8b}$$

$$x_{ij} \in \{0,1\}\ \forall i \in \mathcal{N},\ j \in \mathcal{M} \tag{8c}$$

Formula (8) indicates that the total social welfare is the product of the user's ex ante valuation, the network utility function, computing power, and deployment constraints. Such a model can more truly reflect the economic behavior in the blockchain network. The above nonlinear integer programming problem can enable the ECSP to obtain the greatest benefits. (8a) means that each miner can only be served by at most one server, (8b) means that the maximum service that each server can provide cannot exceed its own total resources, and (8c) indicates that the problem is an integer programming problem. In the auction mechanism of maximizing social welfare, the first step is to solve the resource allocation problem, and then the price that each winner needs to pay can be calculated on the premise that the result of the allocation is determined. Therefore, in the allocation stage, the ECSP pursues the maximization of social welfare [23], and the bid and estimation of each miner on the corresponding server can be considered components of social welfare. To achieve the maximization of social welfare, two cases are discussed : (1) the design of an auction mechanism under constant demand and (2) the design of an auction mechanism under multiple demands.

4. Mechanism design preliminaries

During the auction process, the miners expect to obtain greater benefits, and assume that each miner may submit untruthful job information. Therefore, it is necessary to ensure that the auction mechanism is truthful; a truthful auction mechanism needs to meet the following definitions [21,23]:

Definition 1 (*Utility Value*)**.** The utility value of user i is $u_i = b_i - p_i$, where b_i is miner i's true valuation and p_i is the payment for miner i calculated by the payment rule P. The objective of each miner during the auction is to maximize its own utility value.

Definition 2 (*Individual Rationality*)**.** A mechanism of individual rationality should be satisfied when the miner submits its truthful demand and its utility value is greater than or equal to zero, that is, $u_i = b_i - p_i \ge 0$. In other words, as long as the miner participates in the auction and reports its job information truthfully, it will never incur losses.

Definition 3 (*Monotonicity*)**.** If miner i wins the auction with demand d_i and bid b_i, then it will also win with any lower demand $d_i' < d_i$ and higher bid $b_i' > b_i$.

Definition 4 (*Critical Value*)**.** If the demand d_i and bid b_i submitted by miner i are allocated, when the demand of miner i does not change, there will be a critical value cv_i. If miner i's bid $b_i > \mathrm{cv}_i$, miner i must be allocated successfully; otherwise, it will not be allocated successfully.

Definition 5 (*Truthfulness*)**.** If the resource allocation algorithm *ALLOC*() in the auction mechanism is monotonic and the price calculation mechanism *PAY*() is based on a critical value, then the mechanism is truthful [3].

5. Blockchain-based internet of vehicles resource allocation auction mechanism design

This paper divides the problem of blockchain-based IoV resource allocation into two categories: constant-demand and multidemand types. Constant demand indicates that the amount of resources required by each miner for mining is the same, and multidemand indicates that the amount of resources required by each miner for mining is different.

5.1. Constant-demand auction mechanism

Let the demand of all miners be $d_i = 1$. Since miner i's bid b_i is equal to its ex ante valuation v_i', the social welfare maximization problem at this time can be transformed into the following nonlinear integer programming problem:

$$Objective: \max S(\mathbf{X}) = \sum_{i=1}^{N}\sum_{j=1}^{M} b_i \frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}} \frac{\sum_{j=1}^{M} x_{ij}}{D}\delta_{ij} \tag{9}$$

$$Subject\ to: \sum_{j=1}^{M} x_{ij}\delta_{ij} \le 1 \tag{9a}$$

$$\sum_{i=1}^{N} x_{ij}\delta_{ij} \le c_j \tag{9b}$$

$$x_{ij} \in \{0,1\}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M} \tag{9c}$$

An improved minimum cost and maximum flow algorithm, that is, the maximum cost and maximum flow algorithm is used to solve the following maximizing social welfare problem:

$$Objective: \max \hat{S}(\mathbf{X}) = \sum_{i=1}^{N}\sum_{j=1}^{M} b_i x_{ij}\delta_{ij} \tag{10}$$

$$Subject\ to: \sum_{j=1}^{M} x_{ij}\delta_{ij} \le 1 \tag{10a}$$

$$\sum_{i=1}^{N} x_{ij}\delta_{ij} \le c_j \tag{10b}$$

$$x_{ij} \in \{0,1\}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M} \tag{10c}$$

Theorem 1. *Under constant demand, the solutions of formulas (9) and (10) are equivalent.*

Proof. In formula (9), $\omega = \frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}}$ is a monotonic concave function, and the maximum cost maximum flow algorithm can ensure that the value of d_N is the maximum value. At the same time, D is constant, so the solution of

$$\max S(\mathbf{X}) = \sum_{i=1}^{N}\sum_{j=1}^{M} b_i \frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}} \frac{\sum_{j=1}^{M} x_{ij}}{D}\delta_{ij}$$

is equivalent to the solution of $\max \hat{S}(\mathbf{X}) = \sum_{i=1}^{N}\sum_{j=1}^{M} b_i x_{ij}\delta_{ij}$ under constant demand.

5.1.1. A simple example

In the maximum cost maximum flow algorithm, first, find a constant $\sigma = 1 + \max_{i\in\mathcal{N}} b_i$ such that each miner's bid becomes $b_i^* = \sigma - b_i \geq 1$ and then use the logic of minimum cost and maximum flow to examine the converted bid of miner i. In this way, the maximum cost and maximum flow can be achieved. The maximum flow ensures that miners can occupy the resources of the ECSP to the greatest extent, and the maximum cost ensures that the ECSP can choose miners with higher bids; thus, the maximum cost and maximum flow can maximize social welfare. Then, the actual bid of miner i is $b_i = \sigma - b_i^*$; that is, σ achieves the mutual conversion of the minimum cost $\sum_{i=1}^{N} b_i^*$ and the maximum cost $\sum_{i=1}^{N} b_i$.

Suppose there are 3 miners and 2 servers and that the capacity of each server is 2. The deployment constraints of the miners are $\Delta = \begin{bmatrix} \delta_{11}, \delta_{12} \\ \delta_{21}, \delta_{22} \\ \delta_{31}, \delta_{32} \end{bmatrix} = \begin{bmatrix} 1, 0 \\ 1, 0 \\ 0, 1 \end{bmatrix}$. The resource demand of each miner for a specific server and the corresponding bid is (d_i, b_i). In this example, the resource demand of the first miner and the corresponding bid is (1,5), the resource demand of the second miner and the corresponding bid is (1,6), the resource demand of the third miner and the corresponding bid is (1,7), and the maximum bid is 7; therefore, $\sigma = 1 + \max_{i\in\mathcal{N}} b_i = 8$, and the miner's bid becomes $b_1^* = 3, b_2^* = 2, b_3^* = 1$. Fig. 3 shows a simple example based on maximum cost and maximum flow. At this time, the maximum flow in the network is $v(f) = f_{s,1} + f_{s,2} + f_{s,3} = 1 + 1 + 1 = 3$, the minimum fee based on the maximum flow is $(b_1^* + b_2^* + b_3^*) \cdot d_i = (3 + 2 + 1) \cdot 1 = 6$, and the actual maximum cost before conversion by σ is $((8 - b_1^*) + (8 - b_2^*) + (8 - b_3^*)) \cdot d_i = (b_1 + b_2 + b_3) \cdot d_i = (5 + 6 + 7) \cdot 1 = 18$.

5.1.2. Maximum cost maximum flow algorithm for maximizing social welfare

Algorithm 1 Constant cost maximum flow algorithm for maximizing social welfare (CDAM)

Input: Miners' demands $\mathcal{D}$; Miners' bids $\mathcal{B}$; Servers' resources $\mathcal{C}$; Deployment constraints Δ;
Output: Resource allocation matrix **X** and service price matrix **P**

1: $\sigma = 1 + \max_{i\in\mathcal{N}} b_i$
2: **for all** $i \in \mathcal{N}, j \in \mathcal{M}$ **do**
3: $x_{ij} \leftarrow 0, p_{ij} \leftarrow 0, b_i \leftarrow \theta - b_i$
4: **end for**
5: $\mathbf{X} \leftarrow maxcost\_maxflow(\mathcal{N})$
6: **for all** $x_{ij} \in \mathbf{X}$ **do**
7: **if** $x_{ij} = 1$ **then**
8: $\mathcal{N}_{-i} \leftarrow \mathcal{N} \setminus i$
9: $\mathbf{X}' \leftarrow maxcost\_maxflow(\mathcal{N}_{-i})$
10: $p_i \leftarrow S(\mathbf{X}') - (S(\mathbf{X}) - b_i\omega\gamma_i)$
11: **end if**
12: **end for**
13: **return X, P**

In Algorithm 1, lines 1–5 obtain the optimal resource allocation and social welfare through the maximum cost maximum flow algorithm, and lines 6–13 determine the price paid by each miner to the ECSP by the VCG price payment algorithm on the basis of optimal allocation, which guarantees that the auction mechanism is truthful. Note that line 9 of the algorithm uses formula (10) to calculate the optimal allocation, and line 10 uses formula (9) to calculate the payment price.

In terms of computational complexity, because the demand of each user is assumed to be 1, the optimal problem can be solved in polynomial time. This conclusion has important guiding significance in practice because if the demand is variable, the problem is NP-hard. An improved *Bellman–Ford* algorithm is used to calculate the maximum cost and maximum flow in line 5. The algorithm computational complexity in the allocation phase is $O(VE) = O((M + N)MN)$, where V is the number of vertices in the maximum cost maximum flow graph and E is the number of edges. When using VCG to solve the payment in lines 6–11, it can be seen that each winning user needs to be calculated separately, so the algorithm complexity is $O((M + N)MN^2)$.

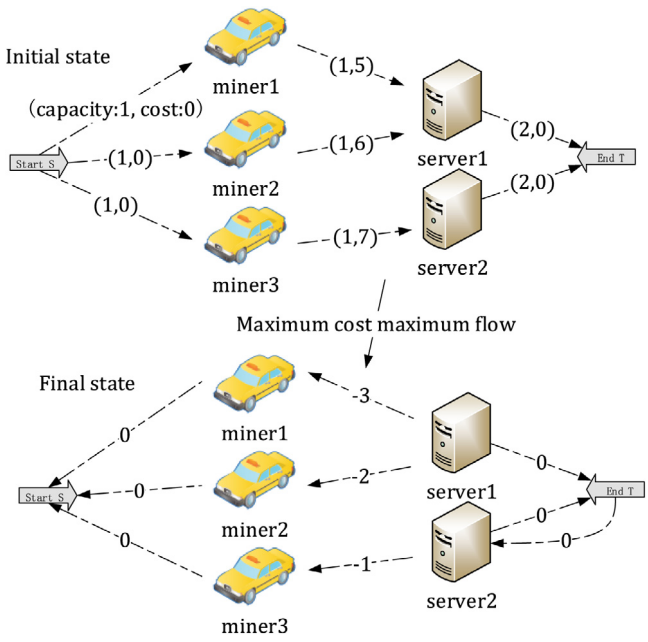

Fig. 3. Resource allocation based on the maximum cost and maximum flow.

Theorem 2. *The CDAM algorithm is individually rational.*

Proof. According to line 5 of Algorithm 1, the maximum cost maximum flow algorithm is implemented to obtain the resource allocation **X**. At this time, the utility u_i of miner i is defined as follows:

$$u_i = b_i - p_i = b_i(1 - \omega\gamma_i) + S(\mathbf{X}) - S(\mathbf{X}') \tag{11}$$

Clearly, $b_i(1 - \omega\gamma_i) \geq 0$ and $S(\mathbf{X}) - S(\mathbf{X}') \geq 0$; therefore, $u_i \geq 0$, and the CDAM algorithm is individually rational.

Theorem 3. *The payment price of the CDAM algorithm is optimal.*

Proof. The VCG mechanism [4] is used to calculate the service price. The formal VCG-based payment function is defined as follows:

$$p_i = \sum_{j\in A(\boldsymbol{\theta}_{-i})} b_j - \sum_{j\in A(\boldsymbol{\theta}), j\neq i} b_j, \ i \in A(\boldsymbol{\theta}) \tag{12}$$

where $A(.)$ represents the optimal allocation function and the output is the winner set. $\boldsymbol{\theta}$ and $\boldsymbol{\theta}_{-i}$ represent the demand information submitted by all the users and the demand information submitted by all users except i respectively. $\sum_{j\in A(\boldsymbol{\theta}_{-i})} b_j$ is the optimal social welfare when user i does not participate in the auction, and $\sum_{j\in A(\boldsymbol{\theta}), j\neq i} b_j$ is the optimal social welfare of all users, except user i. p_i is the final payment price of user i. Line 10 of Algorithm 1 shows that $S(\mathbf{X}')$ is the maximum social welfare of miner i when it does not participate in the auction, $(S(\mathbf{X}) - b_i\omega\gamma_i)$ is the maximum social welfare of miner i when it participates

in the auction minus the maximum social welfare of miner i. It can be known that the resource allocation output $\mathbf{X}$ obtained by Algorithm 1 is optimal, and VCG is an optimal payment solution under optimal allocation. Therefore, the payment price output by Algorithm 1 is optimal.

Theorem 4. *The CDAM algorithm is truthful.*

Proof. Theorem 1 shows that in the case of fixed demand, the optimal allocation can be achieved by using the maximum cost maximum flow algorithm. Using the VCG mechanism based on optimal allocation, the optimal price payment by the miner to the server can be determined. According to Theorem 1 and the properties of the VCG mechanism, the CDAM algorithm is truthful.

5.2. Multidemand auction mechanism

When miner i's demand d_i is different, the maximum social welfare is as follows:

$$Objective: \max S(\mathbf{X}) = \sum_{i=1}^{N}\sum_{j=1}^{M} b_i \frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}} \frac{\sum_{j=1}^{M} d_i x_{ij}}{D} \delta_{ij} \tag{13}$$

$$Subject\ to: \sum_{j=1}^{M} x_{ij}\delta_{ij} \le 1 \tag{13a}$$

$$\sum_{i=1}^{N} d_i x_{ij}\delta_{ij} \le c_j \tag{13b}$$

$$x_{ij} \in \{0,1\}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M} \tag{13c}$$

Since the demands of miners are different, it is reasonable to assume that the ECSP places a restriction on the purchase quantity, i.e., $\alpha D < d_i < \beta D$, where αD and βD are the lower and upper limits on each miner's demand, respectively, and $0 < \alpha < \beta < 1$ are predetermined demand constraint ratios. To achieve the goal of maximizing social welfare, a degree priority heuristic algorithm is designed for maximizing the social welfare of multidemand miners, which includes the resource allocation algorithm MDAM-ALLOC and the price payment algorithm MDAM-PAY. The miners' payment is calculated on the basis of resource allocation. When performing resource allocation, the algorithm defines the priority of the computing service by the server's in-degree definition; in other words, the smaller the server's in-degree is, the higher its priority, and the server with the highest priority first accepts or rejects resource requests according to the demands of the corresponding miners. Moreover, the concept of resource density $f_i = \frac{b_i}{d_i}$ is defined, whereby the server tends to allocate resources to the miners with high resource density.

Regarding Eq. (13), when discussing the auction problem of social welfare maximization for multidemand miners, finding the optimal solution to both the resource allocation problem and payment price problem is NP-hard and cannot be performed in polynomial time [24]. The significance of the design mechanism is that it compares the results of the auction mechanism based on the heuristic method and the auction mechanism based on the first-fit and best-fit algorithms.

5.2.1. A simple example

As shown in Fig. 4 : there are 3 miners and 2 servers, where f_i is the miner's bid density when server j allocates resources to miner i and provides services. The in-degree of the first server is 3 ($l_1 = 3$), and the in-degree of the second server is 2 ($l_2 = 2$), $l_2 < l_1$, so the second server first matches the corresponding miners who need to be allocated resources and provided services. At this time, because both miners (miner 2 and miner 3) have

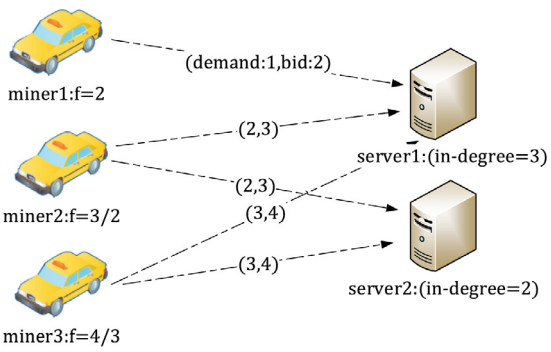

Fig. 4. IoV based on blockchain.

resource demands for the second server, where $f_2 = \frac{b_2}{d_2} = \frac{3}{2}$, $f_3 = \frac{b_3}{d_3} = \frac{4}{3}$, and $f_2 > f_3$, miner 2's unit bid is higher than miner 3's unit bid for the second server. Therefore, miner 2 will obtain the resource allocation and computing services of the second server. That is, $x_{22} = 1$. Since the task demands of each miner are atomic, miner 2 will not obtain the resource allocation and services provided by other servers after receiving the resource allocation and computing services of the second server.

5.2.2. Multidemand auction mechanism allocation algorithm

Algorithm 2 MDAM-ALLOC

Input: Miners' demands $\mathcal{D}$; Miners' bids $\mathcal{B}$; Servers' resources $\mathcal{C}$; Deployment constraints Δ;
Output: Miner resource allocation matrix $\mathbf{X}$

1: Initialize the allocation solution x_{ij} and servers' in-degrees id_j
2: **for all** $j \in \mathcal{M}$ **do**
3: $\quad id_j \leftarrow \sum_{i \in \mathcal{N}} \delta_{ij}$
4: **end for**
5: **for all** $i \in \mathcal{N}$ **do**
6: $\quad f_i \leftarrow \frac{b_i}{d_i}$
7: **end for**
8: Sort all the servers by input degree id_j in nondecreasing order to set $\mathcal{I}$
9: Sort all the miners by bid density f_i in nondecreasing order to set $\mathcal{F}$
10: **for all** $j \in \mathcal{M}$, according to the nondecreasing order in $\mathcal{I}$ **do**
11: $\quad$ **for all** $i \in \{i|\delta_{ij} = 1, i \in \mathcal{N}\}$, according to the nonincreasing order in $\mathcal{F}$ **do**
12: $\quad\quad$ **if** *checkFeasible*$(d_i, c_j) = 1$ and i is not allocated **then**
13: $\quad\quad\quad x_{ij} \leftarrow 1, c_j \leftarrow c_j - d_i$
14: $\quad\quad$ **end if**
15: $\quad$ **end for**
16: **end for**
17: **return X**

Algorithm 2 first calculates the in-degree of each server and arranges the miners in ascending order of resource density $f_{ij} = \frac{b_i}{d_i}$ (lines 5–7); then, the server with the smallest nonzero in-degree allocates resources to miners (lines 10–16). When multiple miners compete for the same server, the miner with the highest resource density will be allocated successfully under the premise of limited server resources, and the algorithm will check the demands from miners who were allocated resources in line 12. If a miner's demand exceeds the server capacity, the allocation will fail by the function *checkFeasible*. Executing Algorithm 2 will yield the allocation solution $\mathbf{X}$.

The payment by the miner to the server is calculated based on the critical value under the premise of resource allocation. The specific design of the payment in a multidemand auction mechanism (MDAM-PAY) is shown in Algorithm 3 below.

Algorithm 3 MDAM-PAY

Input: Miner resource allocation matrix **X** obtained by Algorithm 2 and miners' bids $\mathcal{B}$;
Output: Service price **P**

1: $\varepsilon \leftarrow 10^{-6}$, *for each* $i \in \mathcal{N}$, $p_i \leftarrow 0$
2: **for all** $\{i \mid i \in \mathcal{N}, x_{ij} = 1\}$ **do**
3: $upper \leftarrow b_i, lower \leftarrow 0, b_i \leftarrow \frac{upper+lower}{2}$
4: **while** $((|upper - lower|) > \varepsilon)$ **do**
5: $\mathbf{X}' \leftarrow MDAM - ALLOC(\mathcal{D}, \mathcal{B}, \mathcal{C}, \Delta)$
6: **if** $x_{ij} = 1$ *in* $\mathbf{X}'$ **then**
7: $upper \leftarrow b_i, b_i \leftarrow \frac{upper+lower}{2}$
8: **else**
9: $lower \leftarrow b_i, b_i \leftarrow \frac{upper+lower}{2}$
10: **end if**
11: **end while**
12: $p_i \leftarrow \frac{lower\frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}}d_i}{D}$
13: **end for**
14: **return P**

The third line of the code in Algorithm 3 resets miner i's bid b_i to half of the original bid and then runs the MDAM-ALLOC algorithm. Under the premise that the bids of the other miners remain unchanged, determine whether the miner is still selected by the server after the update. If it is still selected, the bid is decreased again by dichotomy; otherwise, the bid is increased for the calculation of the service price (lines 4–13). *upper* is the upper bound of the dichotomy, and *lower* is the lower bound of the dichotomy. When the difference between the upper and lower bounds is smaller than the predesigned sufficiently small value ε, $p_i = \frac{lower*\frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}}}{D}$ is used as the final price paid by miner i to server j. It is worth noting that the miner's final payment p_i is multiplied by the current network effect. The MDAM-PAY algorithm can ensure that miners must be selected when the resource estimation $b_i > lower$ and must not be selected when $b_i < lower$, which meets the characteristics of the critical value in Definition 4.

Theorem 5. *The MDAM algorithm is individually rational.*

Proof. The MDAM-ALLOC algorithm can yield an approximately optimal allocation of resources, and the MDAM-PAY algorithm makes price payments based on the resource allocation. As seen from the above, the MDAM-PAY algorithm is based on the critical value, so the utility is

$$u_i = b_i - p_i = b_i - \frac{lower\frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}}d_i}{D} \geq b_i - \frac{b_i\frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}}d_i}{D} \geq 0 \quad (14)$$

and the MDAM algorithm is therefore individually rational.

Theorem 6. *The MDAM algorithm is truthful.*

According to Definitions 1, 2, and 3, if the MDAM algorithm is truthful, it must be true that the resource allocation algorithm is monotonic and the price payment algorithm is based on a critical value. The following demonstrates that the MDAM algorithm is truthful.

Proof. In the MDAM-ALLOC algorithm, the priority of the server is first calculated according to the supply and demand relationship between the miners and the server. Then, the server with the highest priority will calculate the bid density f_i of each miner according to the information (d_i, b_i) submitted by the miners and arrange the miners in descending order of bid density f_i. When miner i submits (d_i, b_i) to obtain a resource allocation, submitting a smaller demand d_i' and a larger bid b_i' will increase the resource density f_i of the miner, so miner i can still successfully obtain a resource allocation, which satisfies the monotonicity of Definition 3. The MDAM-PAY algorithm uses a dichotomy based on the critical value of the payment. MDAM-PAY can ensure that resource allocation succeeds when $b_i > lower$ and that resource allocation fails when $b_i < lower$, which meets the critical value characteristics of Definition 4. Therefore, the MDAM algorithm is truthful.

Theorem 7. *The MDAM algorithm runs in polynomial time.*

Proof. When the user's resource demands are inconsistent, the resource allocation problem becomes NP-hard and cannot be solved in polynomial time. In Algorithm 2 MDAM-ALLOC, it is necessary to first sort all edge servers according to the nondecreasing order of in-degrees of each server in line 8 and then sort and allocate resources to the users according to the bid density in nonincreasing order on each server in lines 9–16, so the computational complexity of MDAM-ALLOC is $O(MN^2)$. When using MDAM-PAY to solve the payment, it can be seen that each winning user needs to be calculated separately, so the algorithm complexity is $O(MN^3)$. Therefore, the MDAM algorithm has polynomial time complexity.

6. Experiment and analysis

In this section, the network effect function is first verified through experiments. Then, for constant-demand miners, use CPLEX to verify that the number of winning miners, satisfaction, resource utilization, social welfare and execution time obtained by the improved minimum cost maximum flow (maximum cost maximum flow) algorithm are optimal; for multidemand miners, the optimal solution to both the resource allocation problem and the payment price problem is NP-hard and cannot be solved in polynomial time, and there is no optimal social welfare, so a heuristic algorithm is applied to solve the problem of maximizing social welfare. The hardware configuration of the experimental platform is as follows: the processor is an Intel(R) Core(TM) i5-9400 CPU with 16 GB memory and a 1200 GB hard disk.

6.1. Experimental setting

(1) According to the fitting of the experimental data, the CDAM algorithm sets the network effect parameter to $\mu = 0.1$, $\nu = 0.5 \times 10^{-4}$ and randomly generates values according to a (0,10) uniform distribution to simulate the miners' bids, where the demand d_i of each miner is 1;
(2) According to the fitting of the experimental data, the MDAM algorithm sets the network effect parameter to $\mu = 0.5$, $\nu = 2.5 \times 10^{-4}$ and randomly generates values according to a (5,25) uniform distribution to simulate the miners' bids, where the miners' demands d_i are randomly generated according to a (1,5) uniform distribution;
(3) The CDAM algorithm and MDAM algorithm preset the resource capacity c_j of each server and randomly generate the number of servers that each miner can deploy for in the range of 1 to $M/20$. For example, when there are 50 servers, the number of servers that each miner can deploy for is in the range [1,3];
(4) Generate 50 groups of data for each experiment and use the average of their results for analysis to avoid the random impact of the data.

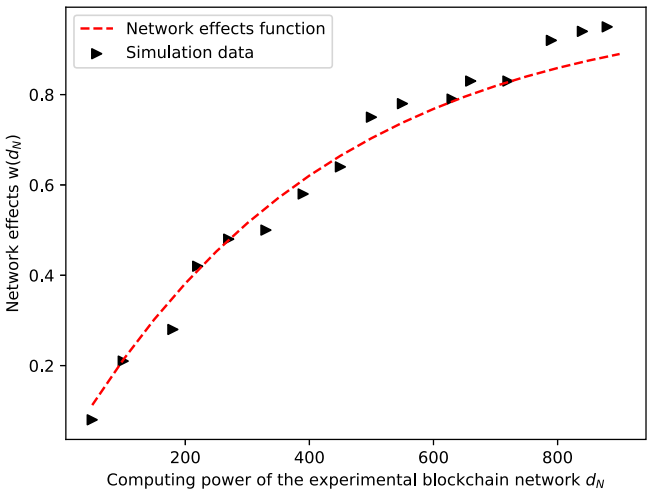

Fig. 5. Estimation of the network effect function $\omega(d_N)$ in (5).

(5) An IBM CPLEX12 runs the program to obtain the optimal solution for resource allocation based on the maximum cost and maximum flow;
(6) C++ is used to implement the CDAM algorithm, MDAM algorithm, first-fit algorithm, and best-fit algorithm.

6.2. Experimental analysis

In the first experiment, we verified the network effect function. An earlier real-world mobile blockchain mining experiment was performed in [25,26]. In the experiment, the network effect function is verified through experimental simulation. It can be seen from Fig. 5 that as the demand of all miners in the blockchain network increases, the computing power of the blockchain network increases, and the network effect shows a gradually increasing trend, which means that as the number of miners participating in the blockchain network increases, the authenticity and robustness of the blockchain network continue to increase, thereby increasing the value of the blockchain network itself. Formula (8) shows that the more participants there are, the more stable the blockchain network, and the resource providers can obtain more social welfare. It can be seen that the network utility function conforms to the trend of diminishing marginal utility. This is in line with our understanding. For a blockchain network, early participants play a greater role in network stability . From formula (11) and formula (14), it can also be seen that these users will obtain greater user utility. With the stability of the blockchain network, the user utility of users who participate in the later period decrease, and the system reaches a stable and sustainable state at this time.

6.2.1. Constant demand experiment

Fig. 6 shows the impact of the number of servers on the experimental indicators. When using the CDAM, the number of miners is fixed at 300, it is easy to see from Fig. 6(a) that social welfare is better than that achieved by the first-fit and best-fit algorithms and shows a trend of first increasing and then decreasing. This is because when the number of miners is fixed, the number of servers is small, and the supply of resources exceeds the demand at the beginning. Additionally, the miners with high bids are quickly allocated resources, and the network effect and hash power increase, so social welfare is on the rise. However, when the number of servers increases to a certain extent, as the total server resources D become larger, the hash power decreases and the network effect remains unchanged, so the social welfare $\sum_{i=1}^{N}\sum_{j=1}^{M} b_i \frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}} \frac{\sum_{j=1}^{M} d_i x_{ij}}{D} \delta_{ij}$ will inevitably decrease. In addition, by comparing the results of CPLEX, it is easy to see that the optimal social welfare can be obtained by using the maximum cost maximum flow in Algorithm 1. Notably, regardless of whether the first-fit algorithm or the best-fit algorithm is implemented, as the number of edge servers increases, the optimal social welfare cannot be obtained.

In Fig. 6(b), it is easy to see that the number of winning miners is greater than that achieved with the first-fit and best-fit algorithms, and it shows a trend of first increasing and then remaining unchanged; this is because when the number of miners is fixed, as the number of servers increases, more miners can be provided with resources. However, when the demand for miners reaches saturation, the number of winning miners will not continue to increase even if the number of servers is increased. In addition, by comparing the results of CPLEX, it is easy to see that the number of winning miners can be maximized by using the maximum cost maximum flow algorithm.

When using the CDAM, it is easy to see from Fig. 6(c) that the miners' satisfaction $\frac{\sum_{j=1}^{M} x_{ij}}{N}$ is higher than that achieved with the first-fit and best-fit algorithms, and the trend of satisfaction and the number of winning miners is the same. In this experiment, satisfaction refers to the percentage of the number of winning miners, so it is positively correlated with the number of winning miners. In addition, by comparing the results of CPLEX, it can be seen that the maximum cost maximum flow algorithm can optimize the satisfaction of miners.

Fig. 6(d) illustrates that the server resource utilization rate is better than that achieved by using the first-fit and best-fit algorithms and shows a trend of first remaining unchanged and then decreasing; this is because when the number of miners is fixed, the resources of the server can fully meet the demands of the miners at the beginning. Therefore, even if the number of servers is increased, their resources will not be used by the miners, resulting in a waste of resources. In addition, by comparing the results of CPLEX, it can be seen that the optimal server resource utilization rate can be achieved by using the maximum cost maximum flow algorithm.

Fig. 6(e) shows the execution time of different algorithms. The execution time of CDAM is the longest, but it is still in polynomial time. The reason is because when CDAM and CPLEX are used to solve the transformed formula (9) integer programming problem, the CPLEX solver will have a greater advantage. On the other hand, the first-fit and best-fit algorithms are implemented based on the heuristic method, so the execution time is shorter. The best-fit algorithm requires multiple iterations, so the execution time is longer than that of the first-fit algorithm.

Fig. 7 shows the impact of the number of miners on the experimental indicators. In this experiment, the impact of the increase in the number of miners is verified, and the number of servers is fixed at 50. As shown in Fig. 7(a), social welfare increases with the increase in the number of miners. This is because when the number of servers is fixed, the total resources D of the server remain unchanged, so the hash power continues to increase. Moreover, the more miners there are, the more demand there will be, and the network effect will increase accordingly; miners with higher bids will be selected, and social welfare will inevitably increase. In addition, by comparing the results of CPLEX, it is easy to see that the social welfare of the maximum cost maximum flow algorithm is better than that of first fit and best fit.

From Fig. 7(b), when the number of servers is fixed, the number of winning miners first increases and then stays unchanged with the increase in the number of miners. This is because when the number of servers is fixed, the total resources D of the servers remain unchanged. When the number of winning miners

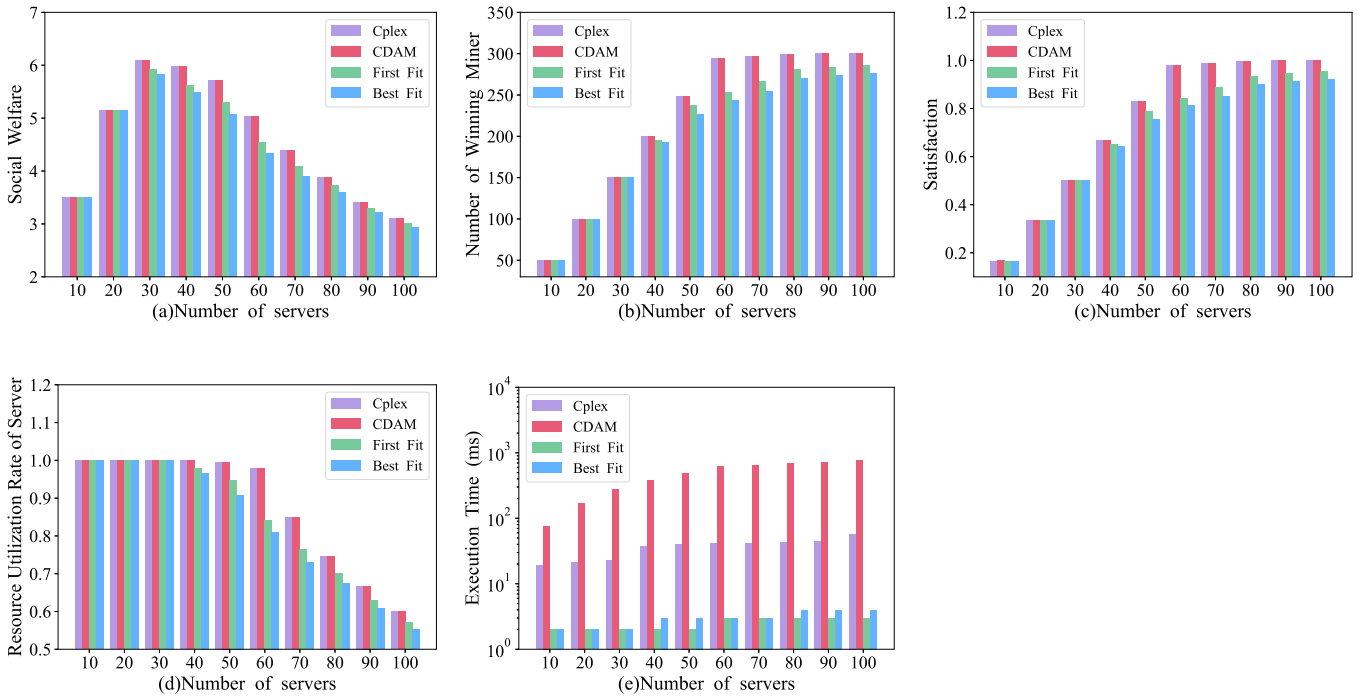

Fig. 6. Comparison of the indicators of the four algorithms when the number of miners is fixed (miner number: 300).

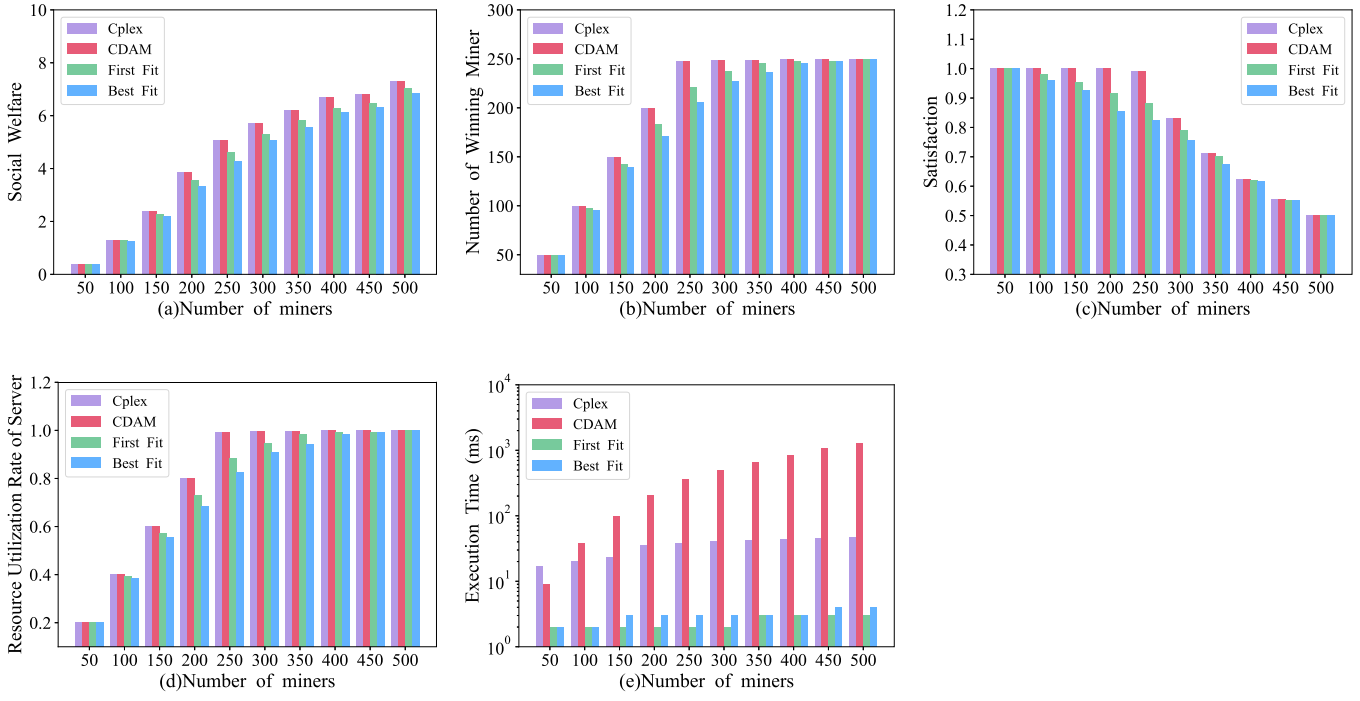

Fig. 7. Comparison of the indicators of the four algorithms when the number of servers is fixed (server number: 50).

increases to a certain level, the total server resources D will be used up, so the final number of winning miners will remain unchanged. In addition, by comparing the results of CPLEX, it is easy to see that the optimal results can be obtained by using the maximum cost maximum flow algorithm and that the number of winning miners is more than that obtained by using the first-fit and the best-fit algorithms.

Fig. 7(c) shows that when the number of servers is fixed, the satisfaction of miners decreases gradually with the increase in the number of miners. This is because when the number of servers is fixed, the rate of increase in the number of winning miners in Fig. 7(c) is less than the rate of increase in the number of total miners in the network, resulting in a decrease in the percentage of winning miners. Moreover, as the number of miners in the network increases, the competition among miners becomes more fierce, and the satisfaction of miners eventually decreases. In addition, by comparing the experiment with CPLEX, it is easy to see that miner satisfaction is higher when using the maximum cost maximum flow algorithm than when using the first-fit and best-fit algorithms.

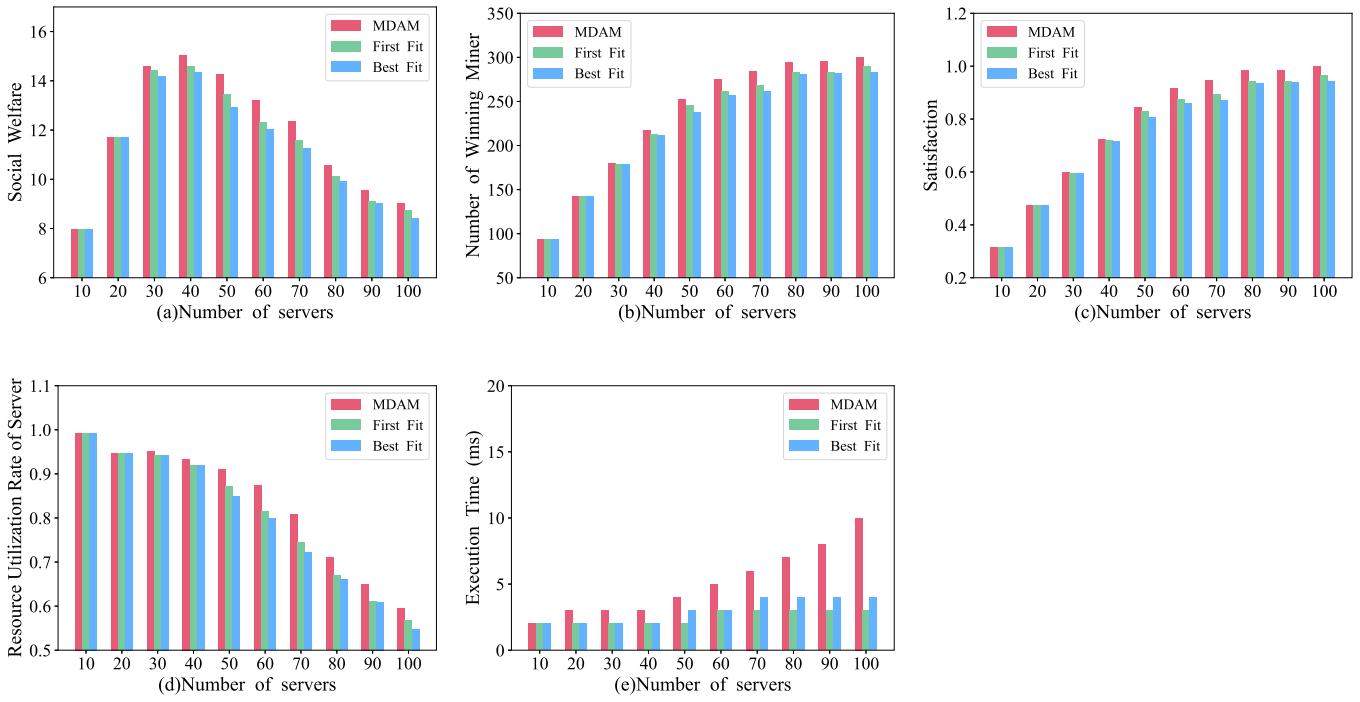

Fig. 8. Comparison of the indicators of the three algorithms when the number of miners is fixed (miner number: 300).

From Fig. 7(d), when the number of servers is fixed, the resource utilization rate of the servers first increases and then remains unchanged as the number of miners increases. This is because when the number of servers is fixed, the total resources D of the servers remain unchanged. Therefore, when the number of winning miners increases to a certain level, it will not increase. In addition, by comparing the results of CPLEX, it is easy to see that the server resource utilization is higher by using the maximum cost maximum flow algorithm than by using the first-fit and the best-fit algorithms.

Fig. 7(e) shows the execution time of different algorithms. The execution time of CPLEX does not change significantly with an increasing miner scale, while the execution time of CDAM increases with an increasing miner scale, but it is still in polynomial time. This result is greatly related to the computational complexity of the CDAM algorithm. At the same time, similar to the previous experiment, the best-fit and first-fit algorithms still have a short running time.

In summary, the social welfare of the CDAM algorithm is the same as the optimal solution, which verifies the correctness of Theorem 1, which transforms a nonlinear optimization problem into an equivalent linear optimization problem. If there is no such conversion, it creates considerable difficulties in the problem solving. The execution time is also within the acceptable range. The second revelation to us is that too much computing power does not create greater social welfare. This is derived from the (1) formula. The total computing power is determined by the sum of the computing power of all servers. This hypothesis is a necessary condition for guaranteeing Theorem 1. In most cases, the total computing power will be less than the sum of user requirements, otherwise the interests of the provider will be damaged. Similarly, too many users will also saturate the system so that it cannot continue to improve social welfare, which means that there is a delicate balance between the number of servers and miners. Finally, the reason why first fit is better than best fit is that best fit generates many fragments in the allocation, which reduces allocation efficiency.

6.2.2. Multidemand experiment

Figs. 8 and 9 below show that all indicators in the multidemand experiment are not compared with CPLEX optimality. This is because the problems of resource allocation and price payment for miners are NP-hard and cannot be solved in polynomial time, so there is no optimal result. In this part, three approaches are compared : MDAM, first-fit and best-fit algorithms.

Fig. 8 shows the impact of the number of servers on the experimental indicators. In this experiment, the number of miners is fixed at 300. It is easy to see from Fig. 8(a) that social welfare is better than that achieved with the first-fit and best-fit algorithms. Fig. 8(a) shows a trend of first increasing and then decreasing, and the reason is the same as that for Fig. 6(a). The difference is that social welfare in the case of multidemand miners is higher than that in the case of constant-demand miners. This is because the demand of each miner in the case of constant-demand miners is 1, and the demand of each miner in the case of multidemand miners is greater than or equal to 1, so the social welfare $\sum_{i=1}^{N}\sum_{j=1}^{M} b_i \frac{1-e^{-\nu d_N}}{1+\mu e^{-\nu d_N}} \frac{\sum_{j=1}^{M} d_i x_{ij}}{D} \delta_{ij}$ in the case of multidemand miners is relatively high. At the same time, because the MDAM-ALLOC algorithm adopts the degree of the nondecreasing order strategy to sort the servers, it is more reasonable, so in terms of social welfare indicators, the MDAM algorithm is better than the first fit and best-fit algorithms.

It is easy to see from Fig. 8(b) that the number of winning miners is greater than that when using the first-fit and best-fit algorithms, and Fig. 8(b) shows a rising trend. This is because when the number of miners is fixed, as the number of servers increases, more miners can be provided with resources. However, because miners have multiple demands at this time, the demand for resources is relatively high. Therefore, the number of winning miners may or may not be saturated. In this case, the advantages of the MDAM algorithm can be seen. When the MDAM algorithm is used in Fig. 8(b), the number of winning miners no longer increases, and the number of winning miners in the network is saturated. Although the number of winning miners

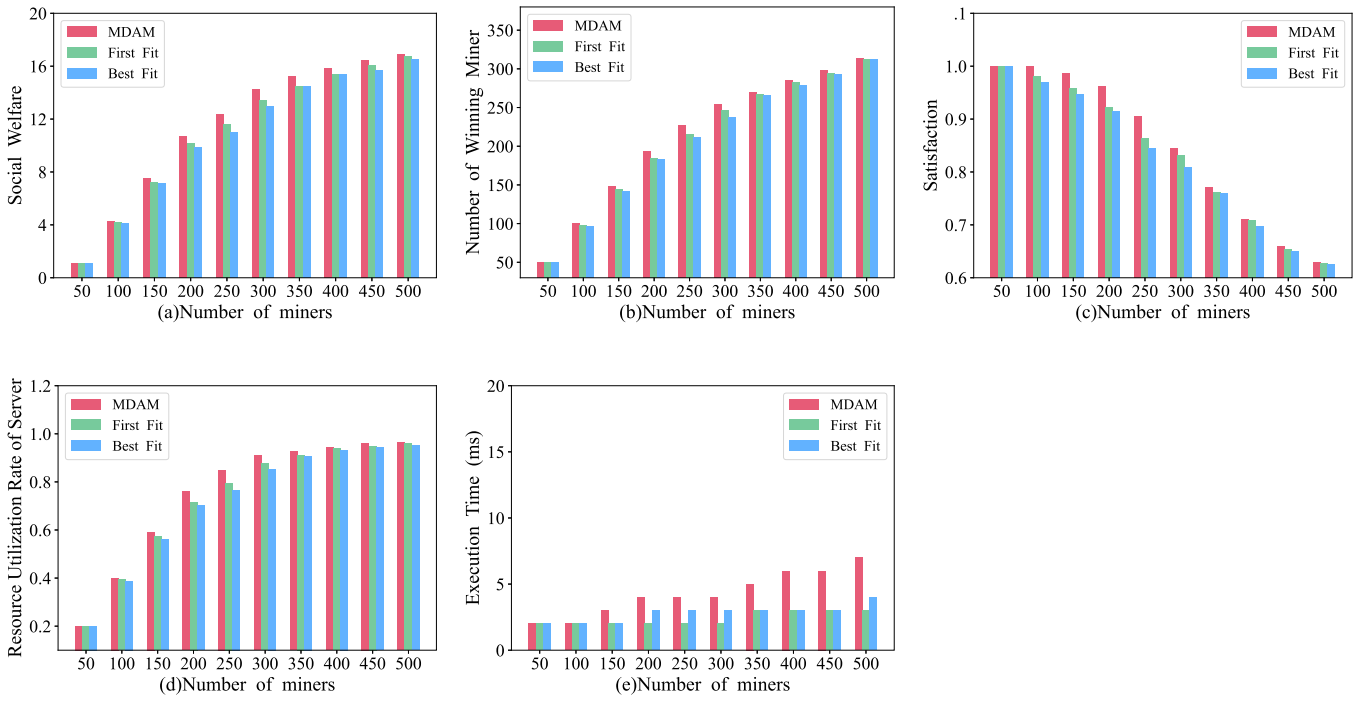

Fig. 9. Comparison of the indicators of the three algorithms when the number of servers is fixed (server number: 50).

also increases using first-fit and best-fit algorithms, they do not reach saturation.

Fig. 8(c) illustrates that the satisfaction $\frac{\sum_{j=1}^{M} x_{ij}}{N}$ of the miners is higher than that when using the first-fit and best-fit algorithms, and the trend of satisfaction and the number of winning miners is the same. The reason is the same as that for Fig. 6(c) . The difference is that the satisfaction and the number of winning miners in the multidemand case are higher than those in the case of constant demand. This is because in the multidemand case, resource allocation is based on the miner's cost effectiveness. In the case of constant demand, resource allocation is based on the miner's bid. Therefore, it is easier to select increasingly suitable miners when allocating resources based on cost effectiveness, and satisfaction is relatively high.

It is easy to see from Fig. 8(d) that the server resource utilization rate is better than that when using the first-fit and best-fit algorithms and shows a trend of first remaining unchanged and then decreasing. The reason is the same as that for Fig. 6(d).

Fig. 8(e) reflects the execution time of different algorithms. Because the three algorithms are heuristic algorithms, they can all be executed in polynomial time. However, because the MDAM computational complexity is $O(MN^3)$, which is higher than that of the first fit and best-fit algorithms, the execution time is slightly longer.

Fig. 9 shows the impact of the number of miners on the experimental indicators. In this experiment, the number of servers is fixed at 50. As shown in Fig. 9(a), social welfare increases with the increase in the number of miners, and social welfare is better when using the MDAM algorithm than when using the first-fit and best-fit algorithms; the reason is the benefit derived from the MDAM degree priority in the ascending ordering strategy in the MDAM-ALLOC algorithm. Meanwhile, this difference is because social welfare in the case of multidemand miners is higher than that in the case of constant-demand miners. This is because the demand of each miner in the case of constant demand is 1, and the demand of each miner in the multidemand case is greater than or equal to 1, so the social welfare $\sum_{i=1}^{N}\sum_{j=1}^{M} b_i \frac{1-e^{-vd_N}}{1+\mu e^{-vd_N}} \frac{\sum_{j=1}^{M} d_i x_{ij}}{D} \delta_{ij}$ in the multidemand case is relatively high.

It is easy to see from Fig. 9(b) that when the number of servers is fixed, the number of winning miners increases with the increase in the number of miners and the number of winning miners is higher when the MDAM algorithm is used than when the first-fit and best-fit algorithms are used. The reason is the same as that for Fig. 7(b).

Fig. 9(c) shows that when the number of servers is fixed, the satisfaction $\frac{\sum_{j=1}^{M} x_{ij}}{N}$ of the miners gradually decreases as the number of miners increases. In addition, when using the MDAM algorithm, satisfaction is better than when using the first match and the best match. The reason is the same as that shown in Fig. 9(a).

As seen from Fig. 9(d), when the number of servers is fixed, the server resource utilization rate increases with the increase in the number of miners. This is because when the number of servers is fixed, the total resources D of the servers remain unchanged. As the number of winning miners increases, the resource demand increases, so the server resource utilization rate increases. In addition, the server resource utilization obtained using MDAM is better than that obtained using the first-fit and best-fit algorithms. This is because the MDAM degree priority in the nondecreasing ordering strategy can allocate resources on each server as reasonably as possible.

Fig. 9(e) reflects the execution time of different algorithms. Because the three algorithms are heuristic algorithms, they can all be executed in polynomial time. However, because the MDAM computational complexity is $O(MN^3)$, which is higher than that of the first fit and best-fit algorithms, the execution time is slightly longer.

In summary, in the MDAM algorithm, the server with a small in-degree is first allocated to ensure allocation efficiency. Additionally, in a specific server, the resource density descending

order is used to ensure that the user with high cost performance is selected first, so the MDAM algorithm is superior to the other two algorithms in various comparisons. Such a monotonic allocation strategy can also ensure that the mechanism is truthful. It is worth noting that the best-fit and first-fit algorithms cannot meet the truthful requirement because users can place their tasks on more favorable servers to obtain greater utility by submitting untruthful resource requirements and bidding. This is also true for CDAM. Additionally, excessive miner participation can intensify resource competition and does not bring about social welfare improvement; the reason is similar to the analysis in the 6.2.1. This finding indicates that there is still a delicate balance between the number of miners and the number of servers.

7. Conclusion

This article innovatively combines edge computing resource allocation, blockchain, the IoV network, and mechanism design from several independent fields and explores and discusses a feasible solution for solving the problem of data credibility in the IoV. Specifically, this paper proposes a blockchain-based IoV resource allocation auction mechanism that can effectively configure the computing resources of the ECSP. The allocative externalities which are particular to blockchain networks are also considered, including competition among the miners and the network effects of the total hash/computing power. For miners under constant demand, this paper proposed an auction mechanism to achieve optimal social welfare and demonstrated that the CDAM algorithm is truthful and individually rational and that resource allocation and price payment are optimal. For miners with multiple demands, the optimal solution for resource allocation and price payment cannot be obtained; thus, optimal social welfare cannot be obtained. Therefore, an approximate optimal algorithm (MDAM) was proposed, which demonstrated that the algorithm is truthful, has individual rationality, and runs in polynomial time.

From a technical perspective, the combination of blockchain and an auction mechanism can promote many unexpectedly good effects. On the one hand, blockchain requires substantial computing resources for distributed data recording. On the other hand, market-oriented blockchain applications require a large number of users to join to form a consensus and achieve the stability of the blockchain network. The auction mechanism can efficiently allocate resources and improve user enthusiasm, which compensates for the shortcomings of blockchain technology. It is foreseeable that in addition to the IoV discussed in this article, the combination of the two will also have many application scenarios in the context of mobile crowdsourcing services and mobile federated learning.

However, there are still some foreseeable limitations. For example, this paper assumes the energy and computational constraints for a PoW-based public blockchain network in an ideal communication environment. For practical system implementation, the communication constraint is an important factor in establishing the mobile blockchain network. One example is that the limited bandwidth of 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. Second, this article assumes that the requirements submitted by users are the same type of resources, but in reality, users may submit requirements for multidimensional resources, such as CPUs, memory and storage. This involves the problem of multidimensional resource allocation. Third, this article only considers static scenarios in the IoV. In practice, the deployment constraints will continue to change during the driving of the vehicle. Finally, in the mechanism design, the use of VCG and dichotomy to calculate the payment price still has the problem of low revenue to the resource provider. Addressing these considerations will make the problem more interesting and closer to the real environment, which is the main direction of our future research.

CRediT authorship contribution statement

Jixian Zhang: Conceptualization, Methodology, Writing – original draft. **Wenlu Lou:** Software, Writing – original draft. **Hao Sun:** Software. **Qian Su:** Methodology. **Weidong Li:** Conceptualization, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported in part by the National Natural Science Foundation of China (Nos. 62062065, 61762091, 61662088, 12071417 and 11663007), a Project of the Natural Science Foundation of Yunnan Province of China (2019FB142 and 2018ZF017), and the Program for Excellent Young Talents, Yunnan, China.

References

[1] S. Liu, L. Liu, J. Tang, B. Yu, Y. Wang, W. Shi, Edge computing for autonomous driving: Opportunities and challenges, Proc. IEEE 107 (8) (2019) 1697–1716.
[2] T. Jiang, H. Fang, H. Wang, Blockchain-based internet of vehicles: Distributed network architecture and performance analysis, IEEE Int. Things J. 6 (3) (2019) 4640–4649.
[3] J. Zhang, X. Yang, N. Xie, X. Zhang, A.V. Vasilakos, W. Li, An online auction mechanism for time-varying multidimensional resource allocation in clouds, Future Gener. Comput. Syst. 111 (2020) 27–38.
[4] Y. Jiao, P. Wang, D. Niyato, K. Suankaewmanee, Auction mechanisms in Cloud/Fog computing resource allocation for public blockchain networks, IEEE Trans. Parallel Distrib. Syst. 30 (9) (2019) 1975–1989.
[5] J. Liu, Y. Mao, J. Zhang, K.B. Letaief, Delay-optimal computation task scheduling for mobile-edge computing systems, in: IEEE International Symposium on Information Theory - Proceedings, Vol. 2016-Augus, 2016, pp. 1451–1455.
[6] J. Ren, G. Yu, Y. He, G.Y. Li, Collaborative cloud and edge computing for latency minimization, IEEE Trans. Veh. Technol. 68 (5) (2019) 5031–5044.
[7] W. Zhang, Z. Zhang, S. Zeadally, H.-C. Chao, V.C.M. Leung, Energy-efficient workload allocation and computation resource configuration in distributed Cloud/Edge computing systems with stochastic workloads, IEEE J. Sel. Areas Commun. 38 (2020) 1118–1132.
[8] C. Tian, Z. Qian, X. Wang, L. Hu, Analysis of joint relay selection and resource allocation scheme for relay-aided D2D communication networks, IEEE ACCESS. 7 (2019) 142715 – 142725.
[9] Y. Yang, Y. Wang, R. Wang, S. Chu, A resource allocation method based on the core server in the collaborative space for mobile edge computing, in: 2018 IEEE/CIC International Conference on Communications, ICCC, 2019, pp. 568–572.
[10] J. Gao, K.O.-B. Obour Agyekum, E.B. Sifah, K.N. Acheampong, Q. Xia, X. Du, M. Guizani, H. Xia, A blockchain-SDN-enabled internet of vehicles environment for fog computing and 5G networks, IEEE Int. Things J. 7 (2020) 4278–4291.
[11] Y. Song, Y. Fu, F.R. Yu, L. Zhou, Blockchain-enabled internet of vehicles with cooperative positioning: A deep neural network approach, IEEE Int. Things J. 7 (2020) 3485–3498.
[12] X. Wang, P. Zeng, N. Patterson, F. Jiang, R. Doss, An improved authentication scheme for internet of vehicles based on blockchain technology, IEEE Access 7 (2019) 45061–45072.
[13] B. Yin, Y. Wu, T. Hu, J. Dong, Z. Jiang, An efficient collaboration and incentive mechanism for internet of vehicles (IoV) with secured information exchange based on blockchains, IEEE Int. Things J. 7 (2020) 1582–1593.
[14] J. Kang, Z. Xiong, D. Niyato, D. Ye, D.I. Kim, J. Zhao, Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory, IEEE Trans. Veh. Technol. 68 (3) (2019) 2906–2920.

[15] H. Liu, Y. Zhang, S. Zheng, Y. Li, Electric vehicle power trading mechanism based on blockchain and smart contract in V2G network, IEEE Access 7 (2019) 160546–160558.
[16] S. Zhang, M. Pu, B. Wang, B. Dong, A privacy protection scheme of microgrid direct electricity transaction based on consortium blockchain and continuous double auction, IEEE Access 7 (2019) 151746–151753.
[17] L. Liu, M. Du, X. Ma, Blockchain-based fair and secure electronic double auction protocol, IEEE Intell. Syst. 35 (2020) 31–40.
[18] T. Ha, D. Lee, C. Lee, S. Cho, VCG auction mechanism based on block chain in smart grid, in: 2021 International Conference on Information Networking, ICOIN, 2021, pp. 465–468.
[19] M.U. Hassan, M.H. Rehmani, J. Chen, DEAL: Differentially private auction for blockchain-based microgrids energy trading, IEEE Trans. Serv. Comput. 13 (2020) 263–275.
[20] Y. Jiao, P. Wang, D. Niyato, Z. Xiong, Social welfare maximization auction in edge computing resource allocation for mobile blockchain, in: IEEE International Conference on Communications, Vol. 2018-May, IEEE, 2018.
[21] N. Nisan, Algorithmic game theory, Commun. Acm 53 (7) (2009) 78–86.
[22] Bolt, Wilko, Bitcoin and cryptocurrency technologies: a comprehensive introduction, J. Econ. Literature 55 (2) (2017) 647.
[23] M.M. Nejad, L. Mashayekhy, D. Grosu, Truthful greedy mechanisms for dynamic virtual machine provisioning and allocation in clouds, IEEE Trans. Parallel Distrib. Syst. 26 (2) (2015) 594–603.
[24] H. Kellerer, U. Pferschy, D. Pisinger, Knapsack Problems, 2004.
[25] K. Suankaewmanee, D.T. Hoang, D. Niyato, S. Sawadsitang, P. Wang, Z. Han, Performance analysis and application of mobile blockchain, in: 2018 International Conference on Computing, Networking and Communications, ICNC 2018, IEEE, 2018, pp. 642–646.
[26] Z. Xiong, S. Feng, D. Niyato, P. Wang, Z. Han, Optimal pricing-based edge computing resource management in mobile blockchain, in: IEEE International Conference on Communications, Vol. 2018-May, IEEE, 2018.

Jixian Zhang born in 1980. He received the M.S. degree in computer science from University of Electronic Science and Technology of China in 2006, and received the Ph.D. degree in computer science from University of Electronic Science and Technology of China in 2010. He is an associate professor at Yunnan University. His research interests include, cloud computing, edge computing and auction mechanism design.

Wenlu Lou was born in 1996. She is currently working toward the M.S. degree in computer software and the theory with the School of Information Science and Engineering, Yunnan University, Kunming, China. Her research interests include cloud/edge computing, blockchain, internet of vehicles and resource allocation.

Hao Sun was born in 1996. He is currently working toward the M.S. degree in the School of Information Science and Engineering, Yunnan University, Kunming, China. His research interests include blockchain, edge computing, resource allocation, auction mechanism design.

Qian Su born in 1983. She received the M.S. degree in computer science from Yunnan University in 2009. She is a lecturer and a Ph.D. candidate at the School of Information Science and Engineering, Yunnan University. Her research interests include edge computing, cloud and edge collaboration.

Weidong Li received the Ph.D. degree in department of mathematics, from Yunnan University in 2010. He is currently a professor at Yunnan University. His main research interests are combinatorial optimization, approximation algorithm, randomized algorithm and cloud computing.