BCC: Blockchain-Based Collaborative Crowdsensing in Autonomous Vehicular Networks

Yilong Hui, *Member, IEEE*, Yuanhao Huang, Zhou Su, *Senior Member, IEEE*, Tom H. Luan, *Senior Member, IEEE*, Nan Cheng, *Member, IEEE*, Xiao Xiao, and Guoru Ding, *Senior Member, IEEE*

***Abstract*—The vehicular crowdsensing, which benefits from edge computing devices (ECDs) distributedly selecting autonomous vehicles (AVs) to complete the sensing tasks and collecting the sensing results, represents a practical and promising solution to facilitate the autonomous vehicular networks (AVNs). With frequent data transaction and rewards distribution in the crowdsensing process, how to design an integrated scheme which guarantees the privacy of AVs and enables the ECDs to earn rewards securely while minimizing the task execution cost (TEC) therefore becomes a challenge. To this end, in this article, we develop a blockchain-based collaborative crowdsensing (BCC) scheme to support secure and efficient vehicular crowdsensing in AVNs. In the BCC, by considering the potential attacks in the crowdsensing process, we first develop a secure crowdsensing environment by designing a blockchain-based transaction architecture to deal with privacy and security issues. With the designed architecture, we then propose a coalition game with a transferable reward to motivate AVs to cooperatively execute the crowdsensing tasks by jointly considering the requirements of the tasks and the available sensing resources of AVs. After that, based on the merge and split rules, a coalition formation algorithm is designed to help each ECD select a group of AVs to form the optimal crowdsensing coalition (OCC) with the target of minimizing the TEC. Finally, we evaluate the TEC of the task and the rewards of the ECDs by comparing the proposed scheme with other schemes. The results show that our scheme can lead to a lower TEC for completing crowdsensing tasks and bring higher rewards to ECDs than the conventional schemes.**

***Index Terms*—Autonomous vehicular networks (AVNs), blockchain, coalition game, vehicular crowdsensing.**

Manuscript received May 12, 2021; revised July 4, 2021; accepted August 4, 2021. Date of publication August 17, 2021; date of current version March 7, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61901341, Grant 62071356, and Grant U1808207; in part by the China Postdoctoral Science Foundation under Grant 2021TQ0260; in part by the National Natural Science Foundation of Shaanxi Province under Grant 2020JQ-301; in part by XAST under Grant 095920201322; and in part by the Fundamental Research Funds for the Central Universities of Ministry of Education of China under Grant XJS200109 and Grant JB210113. *(Corresponding author: Zhou Su.)*

Yilong Hui, Yuanhao Huang, Nan Cheng, and Xiao Xiao are with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: ylhui@xidian.edu.cn; huangyh@stu.xidian.edu.cn; nancheng@xidian.edu.cn; xiaoxiao@xidian.edu.cn).

Zhou Su is with the School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200444, China, and also with the School of Cyber Science and Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: zhousu@ieee.org).

Tom H. Luan is with the School of Cyber Engineering, Xidian University, Xi'an 710071, China (e-mail: tom.luan@xidian.edu.cn).

Guoru Ding is with the College of Communications Engineering, Army Engineering University, Nanjing 210007, China (e-mail: dr.guoru.ding@ieee.org).

Digital Object Identifier 10.1109/JIOT.2021.3105547

I. Introduction

With the rapid development of smart cities, the intelligent transportation system (ITS) is gradually reforming the original transportation structure [1]. Autonomous driving, as an indispensable part of ITS, is expected to improve the driving experience by reducing traffic jams and enhancing traffic efficiency [2]–[5]. With autonomous driving, the ITS will be easier to be realized since autonomous vehicles (AVs) are more controllable and flexible than traditional vehicles. Consequently, the incidence of traffic congestion and accidents caused by human factors can be significantly reduced [6]. In addition, with the adoption of the autonomous driving system, drivers and passengers have the choice of taking more time on enjoying the journey [7].

The AVs, which are equipped with cameras, lidar, and various sensor devices, can sense the traffic environment and collect the data in the smart city. In this way, the valuable information in the city, such as traffic flow and the icing on the roads [8], can be analyzed and used to support the ITS, which has led to the paradigm of vehicular crowdsensing with the support of autonomous vehicular networks (AVNs). In the AVNs, as shown in Fig. 1, there are a number of edge computing devices (ECDs) deployed by different organizations and institutions to undertake various crowdsensing tasks. The ECDs can be seen as the roadside infrastructures which are equipped with edge computing servers [9]–[12]. Therefore, an ECD can receive a crowdsensing task request from a service requester (SR) and publish the task to a group of AVs. By doing this, both the service undertaker (i.e., ECD) and the service performer (i.e., AVs) can obtain the rewards for completing the tasks.

In the crowdsensing process, a group of AVs can execute the task collaboratively to decrease the task execution time. For example, to sense the icing on a road, the road segment can be divided according to the number of lanes and assigned to different AVs. However, the AVs, which are equipped with various types of sensors, have different sensing accuracy and time and therefore declare different rewards for completing the crowdsensing tasks. On the other hand, the crowdsensing tasks, which aim to support various vehicular applications in the AVNs, typically have different completion requirements. Considering these features, how to optimally select a group of AVs to complete the crowdsensing tasks thus becomes a challenge. In addition, as we know, the AVNs are open-access

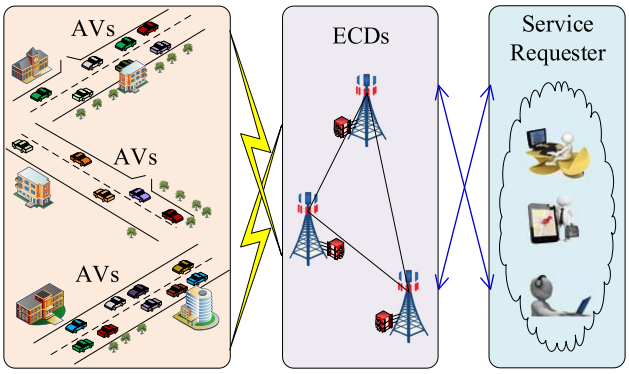


Fig. 1. Crowdsensing system in AVNs.

networks, there are privacy disclosure and network attack problems caused by the data transaction and reward distribution in the vehicular crowdsensing process. For instance, an AV may refuse to contribute its resources if its privacy is compromised [13]. Furthermore, the potential security threats can also decrease the motivation of both ECDs and AVs, which will further reduce the performance of the vehicular crowdsensing system.

To address the aforementioned challenges, a collection of works based on reputation and trust has been studied [14]–[19]. However, the schemes based on reputation and trust may result in a severe monopoly problem if high reputation users collude with each other. Moreover, it is difficult for new nodes to join the crowdsensing process and obtain rewards [15]. Recently, the application of blockchain technology has been widely studied to provide a secure environment for vehicular services [20]–[23]. However, few of them concurrently consider the following three aspects for supporting the vehicular crowdsensing in the AVNs: 1) with the frequent data transaction and reward distribution among the participants in the networks, how to guarantee the privacy of AVs and satisfy the security and nonrepudiation requirements of task transactions in the crowdsensing process. To this end, the first goal of our work is to design a transaction architecture to protect the privacy of AVs and ensure the security and nonrepudiation of transactions; 2) by jointly considering the sensing accuracy, time, and cost, how to develop a model through which the AVs are motivated to cooperate with each other to improve the sensing efficiency. To address this challenge, we need to develop a cooperative model for AVs and design an effective incentive mechanism to encourage AVs to participate in the crowdsensing process; and 3) based on the requirements of the published tasks and the available sensing resources of AVs, how to minimize the task execution cost (TEC) by selecting the optimal AVs to collaboratively execute the task. Consequently, the last goal of this article is to design an algorithm to optimally select AVs to execute the crowdsensing task with the target of minimizing the TEC. From the above analysis, we can conclude that an integrated secure crowdsensing scheme needs to be designed to guarantee the privacy of AVs and enable the ECDs to earn rewards securely while minimizing the TEC.

To this end, in this article, we propose a blockchain-based collaborative crowdsensing (BCC) scheme to support the vehicular crowdsensing in AVNs. Specifically, based on the attack models in the crowdsensing process, we first design the blockchain-based transaction architecture in order to ensure a secure trading environment through which the privacy of AVs and the security of the transactions can be ensured. Then, by jointly considering the requirements of the published tasks and the available sensing resources of AVs, a coalition game model is presented to formulate the interactions among the AVs which intend to collaboratively execute the crowdsensing tasks. After that, based on the merge and split rules, a coalition formation algorithm is designed to select a group of AVs to form the optimal crowdsensing coalition (OCC) with the target of minimizing the TEC. Finally, we analyze the security of the BCC based on the attack models and evaluate the performance of the BCC using extensive simulations. We show that the BCC can not only effectively defend against the attacks but also lead to the lowest TEC and bring the highest rewards to the ECDs by comparing with other schemes. The main contributions of this article are threefold.

1) *Transaction Architecture Establishment:* We propose an integrated transaction architecture for achieving secure vehicular crowdsensing in AVNs. With the designed architecture, the participants (i.e., SRs, ECDs, and AVs) in the BCC can trade crowdsensing tasks securely in an environment without requiring mutual trust.
2) *Coalition Game Formulation:* With the secure transaction architecture, we develop a coalition game with a transferable reward to model the interactions among the AVs in the crowdsensing process, where the AVs are motivated by the transferable reward to collaborate with each other on executing the crowdsensing tasks.
3) *Coalition Formation Algorithm Design:* Based on the merge and split rules, a coalition formation algorithm is designed by jointly considering the requirements of the published tasks and the available sensing resources of AVs. With this algorithm, a group of AVs can be selected to form the OCC to minimize the TEC.

The remainder of this article is organized as follows. Section II reviews the related works. Section III introduces the system model. Section IV presents the proposed BCC scheme in detail. Section V analyzes the security and evaluates the performance of the proposed crowdsensing scheme. Section VI closes this article with the conclusion.

II. Related Work

A. Crowdsensing Schemes

Zhu *et al.* [13] proposed a trust-based crowdsensing scheme in vehicular networks, where the malicious vehicles and trustworthy vehicles can be segregated by using the designed scheme. Although this work provides a solution for reliable crowdsensing, it still needs to design an incentive mechanism to encourage participants to carry out crowdsensing tasks. Different from this work, we distribute rewards to participants through smart contracts, where the AVs are encouraged by the rewards to cooperate with each other to complete the task. Xiao *et al.* [24] analyzed the crowdsensing system in vehicular networks and formulated the interactions between a server and vehicles as a vehicular crowdsensing game. However, the

system lacks the consideration of the security threats and the reward models for developing a secure crowdsensing environment. Unlike this work, we design the BCC scheme to ensure the safety of the transaction while maximizing the rewards of the participants. Ni *et al.* [25] discussed the security, privacy, and fairness of crowdsensing in vehicular networks, where the solutions to achieve security assurance, privacy preservation, and incentive fairness are introduced in detail. However, this work does not develop a mechanism that can not only achieve privacy-preserving data collection but also realize verifiable reward distribution. In this article, we achieve this goal by designing a blockchain-based architecture and a coalition formation algorithm to obtain a secure trading environment and fair reward distribution. Based on the crowdsensing technology, Wang *et al.* [26] studied a framework for real-time traffic management in vehicular networks, where the effectiveness of the framework is evaluated by using a real-world taxi trajectory. However, the framework ignores the security issues in the crowdsensing process. In this article, we analyze various security problems and integrate the blockchain and coalition game to develop a low cost and secure environment to facilitate vehicular crowdsensing. Campioni *et al.* [27] proposed several heuristics algorithms to study two recruitment problems for vehicular crowdsensing. Compared with other algorithms, the designed algorithms can obtain near-optimal solutions. However, this work only intends to select participants to obtain the most sensing data with the lowest cost. In contrast, in our article, we design a more complete framework to execute crowdsensing services by jointly considering the requirements of the task, the available resources, the cost, and the task execution time, as well as the security issues in the crowdsensing process.

B. Blockchain

Huang *et al.* [28] proposed a blockchain-based crowdsensing scheme in industrial systems, where the miners are exploited to verify the sensory data. However, this work ignores the dynamic cooperation among participants. On the contrary, we focus on the cooperation among AVs and propose the BCC scheme to facilitate the crowdsensing services. With the designed secure scheme, a group of AVs can be selected to complete the tasks cooperatively. Kong *et al.* [29] presented a privacy-preserving and verifiable sensory data-sharing scheme with a permissioned blockchain. However, this work does not select the optimal vehicles to execute the task. In contrast, this article not only designs a secure transaction architecture but also designs an algorithm according to the coalition game, aiming at selecting the optimal vehicles to minimize the TEC. Liang *et al.* [30] proposed a novel crowdsensing scheme that offers a solution for the confidentiality and integrity of sensing data. Although the scheme can effectively ensure the fairness of transactions, it needs to design an algorithm to reduce the cost of tasks and increase the enthusiasm of workers. Different from [30], this article studies vehicular crowdsensing based on the blockchain architecture to effectively promote the cooperation among AVs and reduce the TEC. Wang *et al.* [31] proposed a blockchain-based scheme for nondeterministic cooperation in a vehicular crowdsensing system. With this scheme, real-time tasks can be directly assigned to one or more vehicle teams by the platform. However, the proposed algorithm can only reduce the cost of the team, where the requirement of task execution time is ignored. On the contrary, we propose an algorithm based on a coalition game, where the cost, accuracy, and time requirements of tasks are comprehensively considered to obtain the OCC. Hu *et al.* [32] proposed a novel blockchain-based framework to secure the sensing process. Moreover, to provide a fair incentive mechanism, the authors designed a sensory data market based on a three-stage Stackelberg game. However, this work does not consider the requirements of the tasks and the available sensing resources of participants. Unlike this work, our article jointly considers the requirements of the task and the available resources of AVs to execute the task based on the designed coalition game with a transferable reward.

C. Coalition Game

Chen *et al.* [33] proposed a cooperative content transmission scheme in which the cooperation between a vehicle and its neighbors is formulated as a coalitional game. However, the designed scheme does not consider the security problems to facilitate vehicular content distribution. Unlike this work, the coalition game in this article is designed to facilitate crowdsensing services in AVNs based on a blockchain architecture. To help vehicles select the optimal access networks, Xu *et al.* [34] presented a network selection scheme based on a coalition formation game, where a fast convergence algorithm is designed to obtain the final state of the coalition structure. Different from this work, we design a coalition game and analyze the cost and rewards based on different crowdsensing task requirements, which can effectively improve the rewards of the participants. Based on a coalition formation game, Saad *et al.* [35] developed a cooperative scheme to promote the cooperation among ECDs. By using simulations, it is shown that the proposed scheme can improve the rewards of each ECD. However, this work does not consider the attack behaviors of the participants. In this article, we develop a secure trading environment with the adoption of blockchain and smart contract to deal with various attacks in the task execution process. Zhou *et al.* [36] developed a coalition formation game-based resource allocation scheme in vehicular networks. The numerical results demonstrate that the designed scheme has a lower delay than the other heuristic schemes. Different from [36], the coalition game in this article is designed to integrate with the proposed secure transaction architecture. With the designed game model, the AVs are motivated by the transferable reward to join the crowdsensing process. With heterogeneous access networks, Hui *et al.* [37] proposed an optimal access control scheme for vehicles, where a coalition formation game is designed to formulate the cooperation among vehicles to reduce the content download cost. However, this work ignores the privacy of vehicles and the security issues. In contrast, a secure trading environment is developed in this article for the crowdsensing system with the adoption of blockchain and smart contracts.

III. System Model

In this section, we introduce the system model of the BCC which consists of the network model, task model, blockchain model, and attack model, respectively.

A. Network Model

The AVNs consist of register authority (RA), SRs, ECDs, and AVs.

RA: Similar to [38]–[40], the RA can be regarded as the trusted registration center which means that it cannot be compromised. Each entity in the AVNs can generate a pair of keys by connecting to the RA using a secure channel. In addition, the RA owns a secure database to store AVs' registration information so that the security and reliability of data storage can be ensured [13].

SRs: In the AVNs, the SR refers to the individual, organization, or institution which intends to collect the useful data and has registered to the RA. Let $N = \{1, \ldots, n, \ldots, N\}$ denote the set of SRs in the networks. For SR n, it can entrust a crowdsensing task to one of the ECDs. After the crowdsensing task is completed, the SR then collects the task result and distributes the corresponding rewards to the selected ECD.

ECDs: The ECDs, which are at the edge of the AVNs, are deployed along the roadsides with the target of making profits. The set of ECDs in the AVNs is denoted by $J = \{1, \ldots, j, \ldots, J\}$. For ECD j, it registers to the RA for obtaining its certificate and a pair of keys. After becoming the legitimate entity, it can undertake the crowdsensing tasks from the SRs and then publish the tasks to the AVs. In addition, each ECD can collect the sensed data from AVs within its communication coverage and output the final sensing result.

AVs: The AVs in the AVNs can be used for completing the crowdsensing task after registering to the RA. The set of AVs in the AVNs is denoted as $I = \{1, \ldots, i, \ldots, I\}$. Equipped with an onboard unit (OBU) and various sensor devices, each AV can execute the crowdsensing task published by an ECD and then upload the task result to earn rewards. Produced by different companies, the AVs in the networks are typically equipped with various types and models of sensors. Let $K = \{1, \ldots, k, \ldots, K\}$ be the set of the crowdsensing tasks in the AVNs. For AV i, its sensing accuracy of crowdsensing task k that is published by ECD j is denoted as $a_{i_{j_k}}$. If the AV does not equip the sensor to complete task k, we have $a_{i_{j_k}} = 0$. In addition, AVs spend different sensing time in terms of a given task because different sensors have different sensing performances. Let $t_{i_{j_k}}$ ($t_{k,\min} \leq t_{i_{j_k}} \leq t_{k,\max}$) denote the sensing time that AV i spent on completing crowdsensing task k. It is the AV's estimated time cost to complete the task based on the task requirements and sensor performance. In addition, we define $\overline{t_{i_{j_k}}}$ as the real sensing time that AV i spent on completing the task. Then, the time error between the estimated time $t_{i_{j_k}}$ and the real sensing time $\overline{t_{i_{j_k}}}$ can be expressed as $\overline{t_{i_{j_k}}} - t_{i_{j_k}}$. If the time error is no higher than the maximum value of the time error Δt (i.e., $\overline{t_{i_{j_k}}} - t_{i_{j_k}} \leq \Delta t$), we believe that the estimated sensing time is reasonable. Conversely, if the time error is higher than Δt (i.e., $\overline{t_{i_{j_k}}} - t_{i_{j_k}} > \Delta t$), it will be regarded as a malicious AV. Caused by different sensing time $t_{i_{j_k}}$ and sensing accuracy $a_{i_{j_k}}$, AVs usually declare different rewards to complete the crowdsensing task. Therefore, we define the reward per unit sensing time declared by AV i as

$$r_{i_{j_k}} = \delta_{i_{j_k}} \left(a_{i_{j_k}} + \log\left(\frac{t_{k,\max} - t_{i_{j_k}}}{t_{k,\max} - t_{k,\min}} \right) \right) \tag{1}$$

where $\delta_{i_{j_k}}$ refers to the AV's preference for the crowdsensing task. It is a parameter set by AV i to control the requested rewards. If an AV intends to obtain higher rewards, it can increase $\delta_{i_{j_k}}$. On the contrary, if an AV is interested in the task, it can reduce $\delta_{i_{j_k}}$ to increase the probability of participating in the crowdsensing process.

B. Crowdsensing Task Model

In the AVNs, if the SR intends to collect useful data, the SR can entrust an ECD to complete its requested crowdsensing task. For crowdsensing task $k (k \in K)$ which is requested by SR $n (n \in N)$, it can be modeled as a tuple $\{n_k, R_{n_k}, T_{n_k}, A_{n_k}\}$, where n_k is the description of the crowdsensing task. T_{n_k} is the requested deadline to complete the crowdsensing task. R_{n_k} is the reward for completing the task which is paid by the SR who entrusts the ECD to publish the crowdsensing task. A_{n_k} is the task sensing accuracy determined by the SR. The task result with a sensing accuracy that is lower than A_{n_k} is useless to the SR. For crowdsensing task k, it can be further divided into several subtasks and executed by different AVs with the target of reducing the task completion time [41]. For example, in order to obtain the high-definition map of a given road with four lanes, we can divide the crowdsensing task into four subtasks, where each subtask is in charge of sensing the traffic environment and collecting the map data of one lane.

C. Blockchain Model

Blockchain can be regarded as a distributed ledger to record the transactions of crowdsensing services.

Initialization: In the AVNs, the blockchain networks are initialized by the RA [42]. Specifically, the RA generates the system parameters and its keys (i.e., private key SK_{RA} and public key PK_{RA}), where the private key SK_{RA} is kept secret and the other parameters are published to the AVNs.

Registration: After the initialization, the SRs, ECDs, and AVs then can register to the RA and obtain their public keys, private keys, digital certificates (DCs), and wallet addresses to become legitimate entities. Here, we take AV i for example to describe the registration process that includes the following steps.

1) AV i generates its public key PK_i, private key SK_i, and its wallet address.
2) AV i binds its identity in the real world and sends its identity information along with its public key PK_i to the RA.
3) The RA checks the AV's identity. If the identity is valid, the RA signs the AV's public key PK_i with its private key SK_{RA} to issue a certificate for the AV. As such, AV i can obtain its DC as

$$DC_i = \left\langle PK_i || \mathrm{Sign}_{SK_{RA}}(PK_i) || T_{DC_i} \right\rangle \tag{2}$$

where $\text{Sign}_{SK_{RA}}(PK_i)$ is the digital signature function and T_{DC_i} is the timestamp of the generation time of the AV's certificate. In this way, all the participants in the AVNs can audit the DC of AV i by using the RA's public key.

Operation: After becoming legitimate entities, each participant in the AVNs needs to put its deposit. If the participant has an attack behavior in the crowdsensing process, the deposit will be deducted as a penalty. In addition, the keys and certificates of the attacker will be revoked by the RA. Once the registration process is complete, the blockchain starts to operate. In the AVNs, the legitimate ECDs are the authorized nodes that maintain the whole blockchain ledger and have the right to build new blocks. In contrast, the AVs in the AVNs only have the right to request and download the blockchain ledger. If an AV intends to obtain the latest blockchain ledger, it can download the ledger from its connected ECD.

D. Attack Model

In the crowdsensing process, both AVs and ECDs are vulnerable to probable attackers. The security threats that exist in the crowdsensing process include three aspects, i.e., privacy disclosure, malicious ECDs, and malicious AVs [22].

Privacy Disclosure: In this article, the privacy of AVs refers to the location and identity information. In the crowdsensing process, if an AV connects with the ECD to obtain the crowdsensing task or upload the task result using wireless links, the private information of the AV then can be obtained by the attackers to make further analysis [43]. Specifically, the eavesdroppers may listen on communication channels to capture the messages in the crowdsensing process. As a result, it is possible for the eavesdroppers to guess the locations and identify personal preferences of AVs from the visiting frequency of points of interest [44]. In addition, an attacker may discover the AV's true identity through analyzing different data associated with the same anonymous identity, thereby compromising the privacy of the AVs [45]. For example, if an AV uploads its results of subtasks without any privacy protection, the identity of the AV may be inferred by the malicious adversaries from sensitive information.

Malicious ECDs: In this article, the malicious ECDs have the following two types of threats: 1) a malicious ECD may declare that it does not obtain the task result. Consequently, it refuses to pay the rewards to the AVs which provide their resources to complete the task and 2) a malicious ECD does not assign the crowdsensing task to the OCC to minimize the TEC. For example, the ECD may randomly select a group of AVs to execute the task, where its energy and computing resources spent on obtaining the OCC can be saved.

Malicious AVs: In the crowdsensing process, the malicious AVs have the following three types of threats: 1) a malicious AV pretends to declare that it does not obtain the rewards for completing the crowdsensing task after uploading the task result; 2) a malicious AV attempts to upload fake or wrong sensing data to obtain the rewards without any contributions to the crowdsensing task; and 3) a malicious AV intends to upload the result of its subtask multiple times to acquire more rewards.

IV. Blockchain-Based Collaborative Crowdsensing Scheme in AVNs

In this section, we first introduce the transaction architecture of the BCC proposed in this article. After that, we model the interactions among the AVs involved in the BCC as a coalition formation game with a transferable reward, where a coalition formation algorithm is designed to obtain the OCC with the target of minimizing the TEC.

A. Transaction Architecture

It is universal that the same task/subtask is requested more times in AVNs. Therefore, we use parameter ϱ to represent whether the ECD has already obtained the result of the task/subtask. Specifically, $\varrho = 1$ if the ECD caches the task result and $\varrho = 0$ otherwise. If $\varrho = 0$, the ECD will select AVs to execute subtasks, generate the final task result, and send the result to the SR. On the contrary, if $\varrho = 1$, the ECD will directly send the task result to the SR to obtain the reward. Here, we consider the situation where $\varrho = 0$ to introduce the designed transaction architecture. In the BCC, as shown in Fig. 2, the transaction architecture includes the following steps.

1) Sign Smart Contract: A smart contract is a protocol designed to execute a contract automatically when given conditions are satisfied. In the BCC, three types of smart contracts are deployed in advance.

Smart Contract Signed Between the SR and Its Selected ECDs: In the networks, the SR can sign the smart contracts with a number of ECDs in advance. If the SR intends to complete a crowdsensing task, it can select an ECD and initiate a request containing task information to trigger the smart contract signed by the SR and the selected ECD. This contract takes effect when the new block that packages this transaction is added to the blockchain. Along with this, the reward will be transferred from the SR's wallet to the selected ECD's wallet by executing this contract.

Smart Contract Signed by All the ECDs: The smart contract signed by all the ECDs is always valid and aims to reward the ECDs which contribute to the system. Specifically, after generating a block, the ECD that builds this block and the ECDs which participate in publishing the task will obtain the corresponding rewards. Namely, this contract will be executed automatically to transfer the rewards from the wallets of the SRs to the wallet of the ECD that generates the block and the wallets of the ECDs that are entrusted by these SRs to publish tasks in this period of time.

Smart Contract Signed Between the ECD and the AVs: Similar to the smart contract signed between the SR and the ECD, each AV can select an ECD to sign a smart contract in advance. If the ECD has a crowdsensing task that needs to allocate, it can select AVs that have signed the smart contracts to form the optimal OCC. This contract takes effect when the new block that packages the transactions between the ECD and the selected AVs is added to the blockchain. The reward will be transferred from the wallet of the ECD to the wallets of the AVs that contribute to the crowdsensing task.

2) Upload Task Request: When SR n selects ECD j to publish crowdsensing task k, it first generates a task

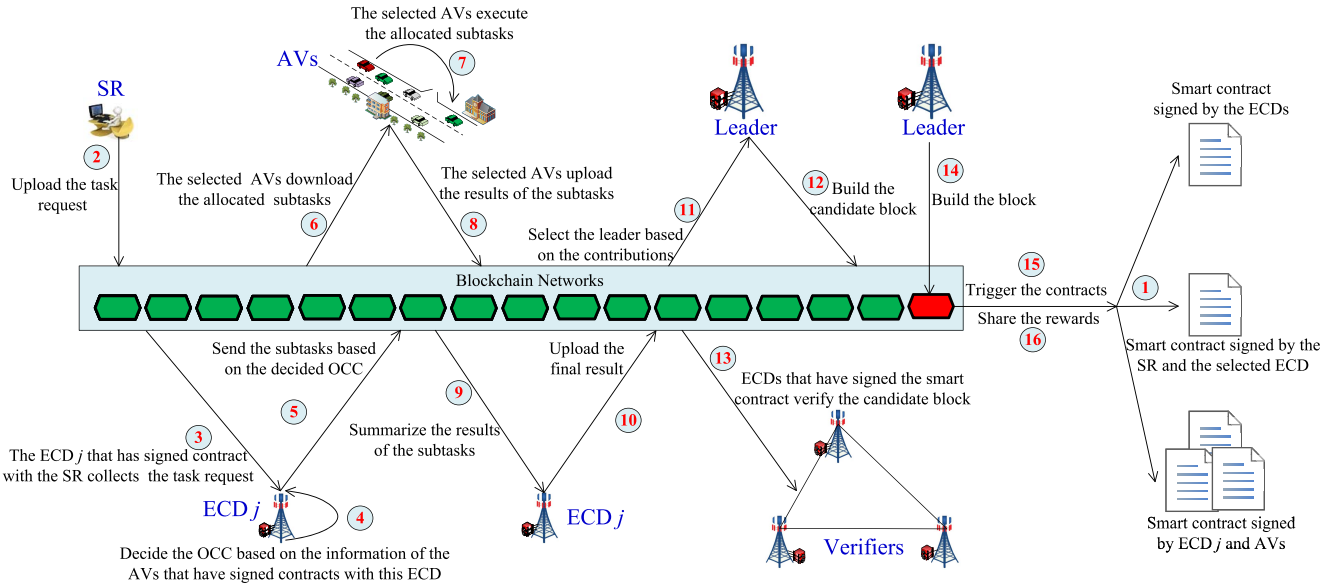


Fig. 2. Blockchain-based secure crowdsensing in AVNs.

request

$$\mathrm{req}_n = \left\langle n_k || R_{n_k} || T_{n_k} || A_{n_k} || T_{\mathrm{req}_{n_k}} \right\rangle. \quad (3)$$

Based on the task request, the SR then signs the request with its private key and encrypts it using the ECD's public key, shown as

$$\mathrm{E}_{PK_j}(\mathrm{req}_n) = \mathrm{E}_{PK_j}\left(\left\langle \mathrm{req}_n || \mathrm{Sign}_{SK_n}(\mathrm{H}(\mathrm{req}_n)) || DC_n \right\rangle\right) \quad (4)$$

where $\mathrm{E}_{PK_j}(\cdot)$ is the encryption function. $\mathrm{H}(\cdot)$ is the hash function to obtain the digest of the task request. After this, SR n uploads the request message of the crowdsensing task to the blockchain networks.

3) Collect Task Request: ECD j collects the task request and verifies the DC of SR n using the SR's public key PK_n. If DC_n is valid which means that SR n is a legitimate node, the ECD then further verifies the signature of the SR and the request of the crowdsensing task. Specifically, if $\mathrm{H}(\mathrm{req}_n)$ calculated by ECD j is equal to $\mathrm{DE}_{PK_n}(\mathrm{Sign}_{SK_n}(\mathrm{H}(\mathrm{req}_n)))$, the request of the crowdsensing task then can be regarded as complete and has not been tampered with, where $\mathrm{DE}_{PK_n}(\cdot)$ is the decryption function.

4) Decide the OCC: Let $I^*(I^* \subseteq I)$ denote the set of AVs that have signed smart contracts with the ECD and can be used for executing the crowdsensing task. Namely, we have $a_{i_{j_k}} \geq A_{n_k} \quad \forall i \in I^*$, based on the request of the crowdsensing task and the resources of the AVs in set I^*, the ECD then applies the coalition formation game with a transferable reward to form the OCC. The AVs which are in the OCC will be selected to execute the crowdsensing task collaboratively with the target of minimizing the TEC. The detail of the coalition formation game is designed in Section IV-B.

5) Upload the Subtasks: Denote by $I^*_{j_k}(I^*_{j_k} \subseteq I^*)$ the set of AVs in the OCC, where each AV in set $I^*_{j_k}$ can be decided to execute a subtask. The ECD then generates a subtask message for AV $i(i \in I^*_{j_k})$, shown as

$$\mathrm{SubM}_{j_k} = \left\langle \mathrm{req}_n || \mathrm{Sign}_{SK_n}(\mathrm{H}(\mathrm{req}_n)) || i_{j_k} || I^*_{j_k} || T_{\mathrm{SubM}_{j_k}} \right\rangle \quad (5)$$

where i_{j_k} is the description of the subtask allocated to AV i. $|I^*_{j_k}|$ is the number of AVs in set $I^*_{j_k}$. Then, ECD j signs the subtask message using its private key and encrypts it using the AV's public key, we have

$$\mathrm{E}_{PK_i}(\mathrm{SubM}_{j_k}) = \mathrm{E}_{PK_i}\Big(\big\langle \mathrm{SubM}_{j_k} || \mathrm{Sign}_{SK_j}(\mathrm{H}(\mathrm{SubM}_{j_k})) || DC_j \big\rangle\Big). \quad (6)$$

After this, the ECD uploads the subtask message to the blockchain networks.

6) Download the Subtasks: The AVs that are selected by the ECD to execute the crowdsensing task then collaboratively download the subtasks from the blockchain networks.

7) Execute the Subtasks: After receiving and verifying $\mathrm{E}_{PK_i}(\mathrm{SubM}_{j_k})$, the AV starts to execute the subtask.

8) Upload the Results of the Subtasks: If AV i completes its subtask, it generates a result message, shown as

$$\mathrm{ResultM}_{i_{j_k}} = \Big\langle \mathrm{SubR}_{i_{j_k}} || \mathrm{Sign}_{SK_i}(\mathrm{H}(\mathrm{SubR}_{i_{j_k}})) || \mathrm{SubM}_{j_k} || \mathrm{Sign}_{SK_j}(\mathrm{H}(\mathrm{SubM}_{j_k})) || T_{\mathrm{ResultM}_{i_{j_k}}} \Big\rangle. \quad (7)$$

Then, the AV signs the result message using its private key and encrypts it using the ECD's public key, we have

$$\mathrm{E}_{PK_j}(\mathrm{ResultM}_{i_{j_k}}) = \mathrm{E}_{PK_j}\Big(\mathrm{ResultM}_{i_{j_k}} || \mathrm{Sign}_{SK_i}(\mathrm{H}(\mathrm{ResultM}_{i_{j_k}})) || DC_i \Big). \quad (8)$$

After that, the AV uploads the encrypted message of the subtask result to the blockchain networks.

9) Summarize the Results: After receiving the uploaded result message, ECD j first decrypts and then verifies the subtask result. If the result is valid, the ECD summarizes the results of the subtasks.

10) Upload the Final Result: The ECD outputs the final result of the crowdsensing task. Then, the ECD signs the final result message using its private key and encrypts it using the SR's public key. After that, the ECD uploads the final result message to the blockchain networks.

11) Select the Leader: For a period of time, the transactions generated in the BCC are collected by each ECD in the AVNs. Therefore, all the legitimate ECDs have the right to be selected as the leader for building a new block. The leader which will obtain the rewards for generating the block is selected based on its stake, namely, its contributions to the crowdsensing services. Specifically, the ECD that has the most contribution to the crowdsensing system in the period of time will be selected as the leader. The contribution of ECD j can be calculated by

$$\ell_j = \frac{\sum_{j_k=1}^{j_{k^*}} x_{j_k}}{\sum_{j=1}^{J}\sum_{j_k=1}^{j_{k^*}} x_{j_k}} \tag{9}$$

where j_{k^*} is the number of crowdsensing tasks published by ECD j. x_{j_k} is the number of subtasks allocated to the selected AVs. In this way, the leader can be selected by

$$j^* = \arg\max_j \{\ell_1, \ldots, \ell_j, \ldots, \ell_J\}. \tag{10}$$

12) Build Candidate Block: The leader, i.e., ECD j^*, then builds a candidate block that packs up all the transactions over this period as a Merkle tree structure. Then, ECD j^* signs the candidate block $\mathrm{C_{block}}$ with its private key and generates a message for verification, shown as

$$\mathrm{C_{block}M} = \left\langle \mathrm{C_{block}} || \mathrm{Sign}_{SK_{j^*}}(\mathrm{H}(\mathrm{C_{block}})) || DC_{j^*} || T_{\mathrm{C_{block}M}} \right\rangle. \tag{11}$$

13) Verify the Candidate Block: For the ECDs except leader j^*, they act as the verifiers in the BCC to audit the candidate block. The verification process, which is also known as the consensus process, aims to make all the ECDs accept the candidate block. Here, we resort to the practical Byzantine fault tolerance (PBFT) consensus mechanism to complete the verification process. We assume that the number of ECDs in the AVNs is no smaller than $3\Gamma + 1$, where Γ is the maximum number of malicious ECDs. Based on PBFT, the consensus process has the following steps.

1) The leader j^* broadcasts the candidate block message $\mathrm{C_{block}M}$ to the verifiers in the AVNs.
2) Each verifier audits $\mathrm{C_{block}M}$. In addition, in this step, each ECD verifies the TEC determined by leader j^*. Specifically, each ECD randomly selects a group of AVs to form a set $I_{j_k}^*$ and calculates the TEC. If the calculated TEC is no smaller than the TEC determined by leader j^*, the TEC is verified. Otherwise, the ECD broadcasts a warning message to all the verifiers in the AVNs.
3) Each verifier encapsulates the verification result into the message if the message is valid and broadcasts the message to the leader and other verifiers.
4) If the number of verification messages received by an ECD is no smaller than $2\Gamma + 1$, it generates a commit message and broadcasts the message to other ECDs.
5) If the number of commit messages received by an ECD is no smaller than $2\Gamma + 1$, the ECD considers that the candidate block is verified and the consensus is reached.

14) Build the Block: If the candidate block is verified, namely, the consensus process is finished, a new block then can be built in the BCC. Along with this, each ECD in the AVNs adds the candidate block $\mathrm{C_{block}}$ to its blockchain ledger.

15) Trigger the Contracts: After building the new block, the three types of smart contracts signed by different participants will be triggered.

16) Share the Rewards: Once the smart contracts are triggered, the rewards of the participants in the BCC will be distributed. Specifically, we allocate the rewards to the AVs and ECDs that contribute to the crowdsensing system. After a block is generated, the total rewards $\sum_{j=1}^{J}\sum_{j_k=1}^{j_{k^*}} R_{j_k}$ are divided into two parts, where R_{j_k} is the reward for completing the task requested by SR n (i.e., $R_{j_k} = R_{n_k}$). The first part of the total rewards is the rewards of AVs, it can be expressed as

$$R_{\mathrm{AVs}} = \sum_{j=1}^{J}\sum_{j_k=1}^{j_{k^*}}\sum_{i_{j_k}=1}^{I_{j_k}^*} r_{i_{j_k}} \frac{t_{i_{j_k}}}{|I_{j_k}^*|} \tag{12}$$

where $\sum_{i_{j_k}=1}^{I_{j_k}^*} r_{i_{j_k}}(t_{i_{j_k}}/|I_{j_k}^*|)$ is the reward that paid to the AVs in set $I_{j_k}^*$, i.e., the TEC of the crowdsensing task.

The second part of the total rewards is the rewards of ECDs, shown as

$$\begin{aligned} R_{\mathrm{ECDs}} &= \sum_{j=1}^{J}\sum_{j_k=1}^{j_{k^*}} R_{j_k} - R_{\mathrm{AVs}} \\ &= \sum_{j=1}^{J}\sum_{j_k=1}^{j_{k^*}} \left(R_{j_k} - \sum_{i_{j_k}=1}^{I_{j_k}^*} r_{i_{j_k}} \frac{t_{i_{j_k}}}{|I_{j_k}^*|} \right) \end{aligned} \tag{13}$$

where $R_{j_k} - \sum_{i_{j_k}=1}^{I_{j_k}^*} r_{i_{j_k}}(t_{i_{j_k}}/|I_{j_k}^*|)$ are the remaining rewards that will be distributed by the ECDs. For the rewards of ECDs, we consider the contribution of different ECDs in the crowdsensing system and further divide the rewards as follows.

1) *The ECD j^* That Generates the New Block:* For this period of time, the block is generated by ECD j^*. Therefore, ECD j^* can obtain the rewards for generating the new block, shown as

$$R_{j^*} = \mu R_{\mathrm{ECDs}} = \mu \sum_{j=1}^{J}\sum_{j_k=1}^{j_{k^*}} \left(R_{j_k} - \sum_{i_{j_k}=1}^{I_{j_k}^*} r_{i_{j_k}} \frac{t_{i_{j_k}}}{|I_{j_k}^*|} \right) \tag{14}$$

where $0 < \mu < 1$.

2) *The ECD j Which Publishes the Crowdsensing Task by Determining the OCC of the Task:* For each ECD $j (j \in J)$, it can obtain the rewards to compensate for the resource consumption of publishing tasks. We have

$$R_j = \frac{\sum_{j_k=1}^{j_{k^*}} x_{j_k}}{\sum_{j=1}^{J}\sum_{j_k=1}^{j_{k^*}} x_{j_k}} \times (1-\mu) R_{\mathrm{ECDs}}$$

$$= \frac{\sum_{j_k=1}^{j_k*} x_{j_k}}{\sum_{j=1}^{J}\sum_{j_k=1}^{j_k*} x_{j_k}} \times (1-\mu)\sum_{j=1}^{J}\sum_{j_k=1}^{j_k*}\left(R_{j_k} - \sum_{i_{j_k}=1}^{I_{j_k}^*} r_{i_{j_k}} \frac{t_{i_{j_k}}}{|I_{j_k}^*|}\right). \quad (15)$$

B. Coalition Game Analysis

After introducing the transaction architecture of the proposed BCC, we then model the task allocation process charged by ECD j as a coalition formation game to minimize the TEC, i.e., $\sum_{i_{j_k}=1}^{I_{j_k}^*} r_{i_{j_k}}(t_{i_{j_k}}/|I_{j_k}^*|)$. In order to minimize the rewards paid for executing the crowdsensing task, ECD j follows the rational principle in determining the OCC. For the AVs in set I^*, they have different sensing times and thus declare different rewards for completing task k. On the one hand, an AV may not complete the crowdsensing task within T_{n_k} individually, it can form a coalition with other AVs to collaboratively execute the task to decrease the task execution time and obtain rewards. On the other hand, the coalition formed by different AVs causes different TEC. Therefore, the problem boils down to finding a group of AVs that can complete the task collaboratively within T_{n_k} while leading to the lowest TEC. It can be formulated as

$$\begin{aligned} \min\{\digamma_{j_k}\} &= \min\left\{\sum_{i_{j_k}=1}^{I_{j_k}^*} r_{i_{j_k}} \frac{t_{i_{j_k}}}{|I_{j_k}^*|}\right\} \\ \text{s.t. } & C1: a_{i_{j_k}} \geq A_{n_k} \\ & C2: \max\left\{\frac{t_{i_{j_k}}}{|I_{j_k}^*|}, i \in I^*\right\} \leq T_{n_k} \end{aligned} \quad (16)$$

where $\max\{(t_{i_{j_k}}/|I_{j_k}^*|), i \in I^*\} \leq T_{n_k}$ ensures that the crowdsensing task can be completed within the required deadline. To solve this problem, ECD j models the interactions among the AVs in set I^* as a coalition formation game with a transferable reward, where the AVs in set $I_{j_k}^*$ can be determined by obtaining the OCC of the game. In addition, for the malicious AV in set I^* which pretends to upload fake or wrong sensing data, we design a fault tolerance method to defend against this attack. Specifically, based on the total number of subtasks, the ECD randomly selects some subtasks to be repeatedly assigned to different AVs. In this way, some of the same subtasks are completed by different AVs. By comparing the sensing data uploaded by the AVs, the malicious AV with this attack can be easily found. If an AV uploads false or wrong sensing data, its deposit stored in RA will be used to compensate for the loss of the task publisher. As the deposit is much larger than the rewards obtained by participating in the crowdsensing process, the proposed method thus can efficiently defend against this attack.

Definition 1: For any set $I^{'}(I^{'} \subseteq I^*)$, it can be called a coalition and expressed as $\langle I^{'}, R(I^{'})\rangle$, where $R(I^{'})$ is the TEC of the coalition.

Especially, the coalition I^* is called the grand coalition in the BCC, where the grand coalition is the OCC if the proposed coalition game is superadditive and its core is not empty.

Definition 2: For a coalition game with a transferable reward, it is said to be superadditive if any two coalitions $I^{'}(I^{'} \subseteq I^*)$ and $I^{''}(I^{''} \subseteq I^*)$ satisfy $I^{'} \cap I^{''} = \emptyset$ and $R(I^{'}) + R(I^{''}) \geq R(I^{'} \cup I^{''})$.

From Definition 2, we can know that in a superadditive coalition game, if a coalition is divided into two disjoint small coalitions, the sum of the TEC of the two small coalitions is always no less than the original coalition.

Theorem 1: The coalition game for obtaining the lowest TEC in the BCC is nonsuperadditive.

The proof of this theorem is given in Appendix A.

Definition 3: For a reward vector, it is said to be a partition if the vector satisfies the group rationality and individual rationality concurrently.

Definition 4: For a reward vector of AVs $\{R(\{1\}), \ldots, R(\{i\}), \ldots, R(\{I_{j_k}^*\})\}$, it is said to be group rational if $\sum_{i=1}^{I_{j_k}^*} R(\{i\}) = R(I_{j_k}^*)$.

Definition 5: For a reward vector of AVs $\{R(\{1\}), \ldots, R(\{i\}), \ldots, R(\{I_{j_k}^*\})\}$, it is said to be individually rational if $R(\{i\}) \geq R\{i\}$.

In Definition 5, $R\{i\}$ is the reward of AV i which completes the crowdsensing task individually. This definition indicates that if a reward vector of a coalition is individually rational, the reward obtained by each AV in the coalition is no smaller than acting alone.

Definition 6: For a coalition game, the set of all the available partitions is called the core of the game.

Theorem 2: The core of the coalition game for obtaining the TEC in the BCC is empty.

The proof of this theorem is given in Appendix B.

Based on the given definitions and theorems, it can be concluded that the coalition game in the BCC is nonsuperadditive. In addition, the core of the game is empty. It means that the grand coalition composed of all the AVs cannot lead to the lowest TEC. Therefore, we propose a coalition formation algorithm to obtain the optimal OCC by considering the following two cases.

Case 1: All the AVs in set I^* can complete the crowdsensing task within T_{n_k} individually.

In this case, we have $\max\{t_{i_{j_k}}, i \in I^*\} \leq T_{n_k}$, it means that all the AVs in set I^* can complete the crowdsensing task within T_{n_k} individually. Therefore, we only need to consider the reward declared by each AV in I^*, where the optimal sensing coalition can be decided in a greedy manner to allocate the crowdsensing task. In other words, the optimal sensing coalition only has one AV, which is selected by

$$i^* = \arg\min_i \left\{r_{i_{j_k}} t_{i_{j_k}}, i \in I^*\right\} \quad (17)$$

where $r_{i_{j_k}} t_{i_{j_k}}$ is the reward of AV i for completing the crowdsensing task. Based on (17), we then can obtain the OCC $I_{j_k}^* = \{i^*\}$ and the lowest TEC $r_{i_{j_k}^*} t_{i_{j_k}^*}$.

Case 2: At least one AV in set I^* cannot complete the crowdsensing task within T_{n_k} individually.

In this case, there is at least one AV that cannot complete the crowdsensing task within T_{n_k} individually. Therefore, it needs to cooperate with other AVs to execute the task and obtain the rewards. Different from case 1, both the task execution time

Algorithm 1 Coalition Formation Algorithm for AVs in Set I^*

Input: $r_{i_{j_k}}$, $t_{i_{j_k}}$,$\forall i \in I^*$
if $I^* \neq \emptyset$ **then**
  if $\max\{t_{i_{j_k}}, i \in I^*\} \leq T_{n_k}$ **then**
    Case 1:
    $i^* \leftarrow \arg\min_i \{r_{i_{j_k}} t_{i_{j_k}}, i \in I^*\}$
    $I^*_{j_k} \leftarrow \{i^*\}$, $F_{j_k} \leftarrow r_{i^*_{j_k}} t_{i^*_{j_k}}$
  else
    Case 2:
    Sort the elements in set R_{I^*} in ascending order
    if $t_1 \leq T_{n_k}$ **then**
      $I^*_{j_k} \leftarrow \{1\}$, $F_{j_k} \leftarrow r_1 t_1$
    else
      judge $= 0$, partition the AVs by $\{1, ..., i, ..., I^*\}$
      for $\psi = 1; \psi \leq I^*$ **do**
        for $\phi = \psi + 1; \phi \leq I^*$ **do**
          if $\max\left\{\frac{t_\psi}{\phi-\psi+1}, ..., \frac{t_\phi}{\phi-\psi+1}\right\} > T_{n_k}$ **then**
            $I_\psi \cup \{\phi\} \leftarrow$ merge $\leftarrow \{I_\psi \setminus \{\phi\}, \{\phi\}\}$
            if $\phi = I^*$ **then**
              for $\phi = \psi + 1; \phi \leq I^*$ **do**
                $I_\psi \cup \{\phi\} \rightarrow$ split $\rightarrow \{I_\psi \setminus \{\phi\}, \{\phi\}\}$
              end for
            end if
          else
            $I^*_{j_k} \leftarrow I_\psi \cup \{\phi\}$, judge $= 1$
          end if
          if judge > 0 **then**
            break
          end if
        end for
        if judge > 0 **then**
          break
        end if
      end for
      $F_{j_k} \leftarrow \sum_{i_{j_k}=1}^{I^*_{j_k}} \frac{r_{i_{j_k}} t_{i_{j_k}}}{|I^*_{j_k}|}$
      if judge $= 0$ **then**
        The ECD cancels the crowdsensing task
      end if
    end if
  end if
else
  The ECD cancels the crowdsensing task
end if
Output: The OCC $I^*_{j_k}$ and the lowest TEC F_{j_k}

and the declared task rewards of the AVs need to be considered in achieving the OCC. Let $R_{I^*} = \{r_1 t_1, \ldots, r_i t_i, \ldots, r_{I^*} t_{I^*}\}$ denote the set of rewards that AVs are independent to execute the crowdsensing task. The ECD sorts the elements in the set in ascending order. If the first element (i.e., $r_1 t_1$) satisfies $t_1 \leq T_{n_k}$, then this case is the same as case 1, where the OCC can be decided by (17). Otherwise, the ECD starts the coalition formation process with the adoption of the merge and split rules, where the specific description of the proposed coalition formation process for obtaining the OCC can be seen in Algorithm 1. Based on the OCC, if the crowdsensing task can be completed by the AVs and the TEC is no larger than R_{n_k}, the ECD then allocates the task to the AVs in the OCC. Otherwise, the task will be canceled.

V. Security Analysis and Performance Evaluation

A. Security Analysis

The proposed BCC can secure the crowdsensing services in the AVNs by defending against the threats mentioned in Section III-D. The details about the performance of the proposed BCC are discussed as follows.

Privacy Protection: In the BCC scheme, the location information of AVs is only used for local sensing and will not be disclosed to others [46]. In addition, the ECD only collects the sensing results of AVs through blockchain networks and does not monitor the local sensing process. When ECD selects AVs to perform subtasks, the information about AVs only includes the sensing price and the sensing accuracy, which effectively prevents the attack behavior of eavesdroppers. For example, when a malicious attacker obtains the result of subtasks uploaded by an AV, it cannot obtain any information about how the AV performs subtasks. This is because the uploaded result does not contain any information about the location of AVs. Especially, a position blur mechanism can be used to convert the exact location of an AV to grid coordinates and obfuscates it by using pseudorandom functions and Bloom filters (BFs) [38]. For identity information, we use the key as a pseudonym, where each transaction is associated with a pair of pseudonyms to identify the sender and receiver. If the attacker only obtains the pseudonyms of the sender and the receiver, it is difficult to associate the specific identity of AVs with the transaction. Furthermore, each AV can use different pseudonyms to perform different crowdsensing tasks. In this way, the malicious attacker can only trace back the corresponding transaction record after obtaining the pseudonym. In other words, the attacker can neither associate it with the identity information of the AV nor continuously monitor the AV's transaction process.

Resilience to Malicious ECDs:

1) In the crowdsensing process, a malicious ECD may publish a crowdsensing task while declaring that it does not obtain the result of the published crowdsensing task. With the adoption of BCC, the rewards of the AVs are obtained after the new block is built, where all the transactions are recorded in the blockchain that is maintained by all the ECDs in the AVNs. In addition, the rewards are charged by the smart contract, where the smart contract is executed automatically after the new block is generated.
2) The OCC determined by an ECD will be further verified by other ECDs in case of this ECD is a malicious node. If the OCC determined by the leader is not verified which means that the OCC cannot lead to the lowest TEC, the deposit of the ECD will be forfeited, where the deposit is much larger than the rewards. Therefore, the threat of a malicious ECD with this type can be effectively eliminated.

TABLE I
SIMULATION PARAMETERS

Parameters	Values
The requested deadline to complete the task: T_{n_k}	0.1hour
Task sensing accuracy required by the SR: A_{n_k}	0.75-0.95
Rewards for completing the task: R_{n_k}	2
Minimum sensing time: $t_{k,\min}$	0.1hour
Maximum sensing time: $t_{k,\max}$	0.5-1hour
AV's preference for the task: $\delta_{i_{j_k}}$	0.1-1
Sensing accuracy of AV i: $a_{i_{j_k}}$	0.1-1
Total number of ECDs: J	41
Maximum number of malicious verifiers: Γ	0-12
Computing resources of ECD j: c_j	10-30GHz
Candidate block size: S	6-10MB
Maximum block interval: t_0	10s
Data transmission rate between j' and j: $\gamma_{j,j'}$	10-100Mbps
Resources for generating/verifying MACs: α	1MHz

Resilience to Malicious AVs:

1) Similar to the case where a malicious ECD refuses to pay the rewards to the AV which uploads the result of the allocated subtask, the malicious AV which declares that it does not obtain the rewards after uploading the task result to the ECD can also be eliminated through the execution of the smart contract.
2) For the malicious AV which pretends to upload fake or wrong sensing data, we design the fault tolerance method to defend against this threat. Specifically, the ECD selects a number of subtasks and allocates each of them to different AVs. In this way, the AV which uploads the fake or wrong sensing data must bear the risk of losing its deposit. Thus, the malicious AV with this type can be eliminated.
3) In the BCC, the AVs which participate in the crowdsensing process are recorded in the blockchain networks. In other words, only these AVs can obtain their rewards after uploading the task result and each of them can only obtain the rewards once. Therefore, the BCC can defend against the malicious AV which attempts to upload the result of its subtask multiple times to obtain more rewards.

B. Performance Evaluation

We study the performance of the BCC scheme using simulations operated on MATLAB. The number of ECDs in the networks is 41, where the maximum number of malicious ECDs changes from 0 to 12 for simulation. The computing resources of each ECD ranges from 10 to 30 GHz. The block size lies in [6, 10] MB and the data transmission rate between two ECDs is selected from [10, 100] Mb/s [47]. As discussed in Section IV, the architecture designed in this article is distributed, where each ECD follows the first-in–first-out rule to publish crowdsensing tasks. Therefore, we consider the scenario that SR *n* entrusts ECD *j* to publish crowdsensing task *k*, where the deadline for the task (i.e., T_{n_k}) is 0.1 h. The sensing accuracy requested by the task (i.e., A_{n_k}) changes from 0.75 to 0.95 for simulation. The number of AVs that can be selected to execute the crowdsensing task ranges from 15 to 90. The time that an AV completes the crowdsensing task individually is randomly selected from [0.1, $t_{k,\max}$] h, where the value of $t_{k,\max}$ varies from 0.5 to 1 h in the simulation to evaluate the performance of the BCC. Parameters used in the simulation are summarized in Table I. With this scenario, we evaluate the complexity of the proposed algorithm, the delay caused by the blockchain, the TEC of the task, and the rewards obtained by ECDs.

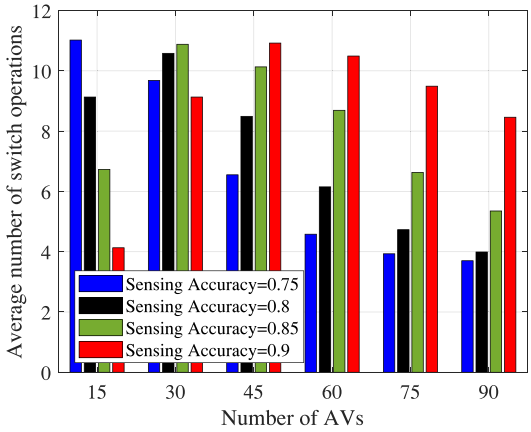


Fig. 3. Average number of switch operations by changing the number of AVs and the requested sensing accuracy.

The Complexity of the Algorithm: The complexity of the coalition game algorithm can be analyzed by calculating the number of switch operations [48]–[52]. Therefore, we carry out a simulation to evaluate the complexity of the proposed algorithm (i.e., the number of switch operations) by changing the number of AVs and the requested sensing accuracy. It can be seen in Fig. 3, the number of switch operations first increases and then decreases with the increase of the number of AVs. This is because when the number of AVs is small (i.e., 15), only a small number of AVs can participate in the game, resulting in a small number of switch operations. When the number of AVs is large (i.e., 90), it is easier for AVs to form a coalition that meets the requirements, thus reducing the number of switch operations. In addition, we can see from the figure that when the number of AVs is small (i.e., 15), the high sensing accuracy requirement (i.e., 0.9) will lead to the lowest number of switch operations. Besides, we can know from this figure that the algorithm can converge to the Nash stable coalition partition through the limited number of switch operations.

The Delay Caused by Blockchain: Similar to [47], the maximum block interval t_0 in our BCC scheme is 10 s and the candidate block size changes from 6 to 10 MB. With this configuration, we then evaluate the delay caused by the blockchain. Specifically, the average time includes the block generation time (i.e., block interval) and the block validation time, where the validation process can be divided into two parts, i.e., message delivery and message verification. Therefore, based on [47] and [53], we carry out a simulation to evaluate the delay caused by the blockchain which consists of message delivery delay t_D and validation delay t_V. As discussed in Section IV-A, the candidate block is verified by the PBFT mechanism including the following phases. In the *Pre-prepare* phase, the ECD j^*, as the leader of the consensus process, broadcasts the message of the candidate block $C_{\text{block}}M$ to other ECDs (i.e., the verifiers) in the AVNs. In

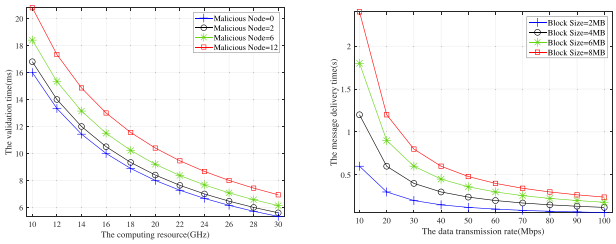


Fig. 4. (a) Validation delay by changing the computing resources of ECDs and the number of malicious ECDs. (b) Message delivery delay by changing the maximum data transmission rate and the block size.

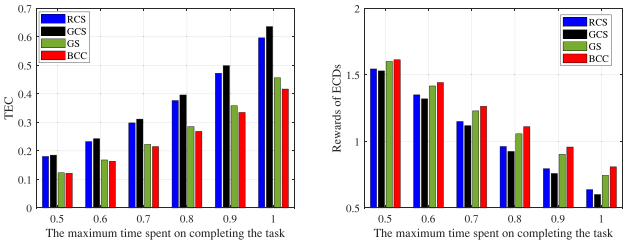


Fig. 5. (a) TEC of the task by changing the maximum time spent on completing the task. (b) Rewards of ECDs by changing the maximum time spent on completing the task.

this phase, the leader generates $J-1$ message authentication codes (MACs) to send the message and each verifier needs to verify one MAC. For the *Prepare* phase, each verifier encapsulates the verification result into the message if the message is valid. Then, the verifier broadcasts the message to the leader and other verifiers. In this phase, each verifier generates $J-1$ MACs and verifies $J-2$ MACs when they receive them. In contrast, the leader needs to verify $J-1$ MACs received from all the verifiers. In the *Commit* phase, the verifier will broadcast the commit message to the other ECDs. Thus, the leader and all the verifiers first send and then receive $J-1$ commit messages, which need to generate $J-1$ MACs and verify $J-1$ MACs.

Therefore, we can conclude that the leader and each verifier need to complete $4(J-1)$ MAC operations in each consensus process. Concerning the case with Γ faulty ECDs, a normal ECD will process at most two messages from a malicious node per message [47]. By considering this fact, the computing resources requested to complete the consensus process is $4(J+\Gamma-1)\alpha$, where α is the resources for generating/verifying MACs. Thus, the validation time t_V can be calculated by

$$t_V = \max_j \left\{ \frac{4(J+\Gamma-1)\alpha}{c_j}, j \in J \right\}. \tag{18}$$

In addition, in the three-stage consensus process, any message should be delivered within the limited time t_0. Hence, the message delivery time t_D of $\mathrm{C_{block}M}$ can be calculated by

$$t_D = \min\left\{ \max_{j \neq j^*} \frac{S}{\gamma_{j,j^*}}, t_0 \right\} + \min\left\{ \max_{j' \neq j, j^*} \frac{S}{\gamma_{j,j'}}, t_0 \right\} + \min\left\{ \max_{j' \neq j} \frac{S}{\gamma_{j,j'}}, t_0 \right\}. \tag{19}$$

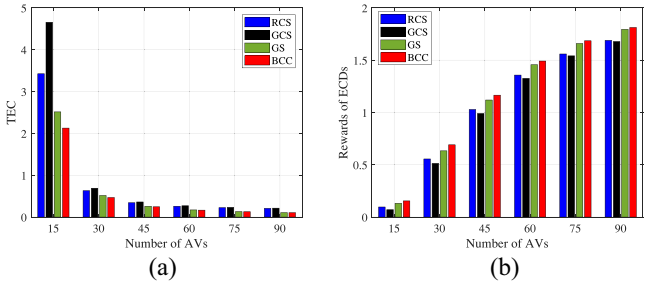
Based on the above analysis, we then simulate the validation delay and the message delivery delay. Fig. 4(a) shows the validation delay by changing the maximum computing resources of ECDs and the number of malicious ECDs. From this figure, we can see that the validation can be carried out more quickly with the increase of computing resources. In addition, if there are more malicious ECDs in the consensus process, the validation delay will be significantly increased. Fig. 4(b) is the message delivery delay by changing the maximum data transmission rate and the block size. We can see from this figure that the message delivery delay decreases significantly with the increase of the data transmission rate between ECDs. In addition, when the block size becomes large, more time will be taken to transmit the three-stage messages. Therefore, as we can see from the figure, the message delivery delay increases with the increase of the block size.

The TEC of the Task and the Rewards of the ECDs: We evaluate the TEC of the task and the rewards of the ECDs by changing different parameters (i.e., the number of AVs, the sensing accuracy requested by the crowdsensing task, and the maximum time that an AV completes the task in the networks). The schemes used in the simulations are detailed as follows.

1) *The Random Coalition Scheme (RCS):* The ECD randomly selects an AV to form the coalition. If the AV in the coalition cannot complete the task, the ECD selects a new AV to merge in the coalition. Repeat this process until the crowdsensing task can be completed by the AVs in the coalition.
2) *The Grand Coalition Scheme (GCS):* The ECD selects all the AVs in set I^* to form the grand coalition to execute the crowdsensing task.
3) *The Greedy Scheme (GS):* The ECD selects AVs to form the coalition in a greedy manner. Specifically, the ECD selects the AV which declares with the lowest reward to form the coalition. After this, the ECD removes the selected AV from set I^*. If the AV in the coalition cannot complete the task, the ECD then selects a new AV which declares with the lowest reward from set I^* to merge in the coalition. Repeat this process until the crowdsensing task can be completed by the AVs in the coalition.
4) *The BCC:* The ECD selects the AVs in set I^* with the adoption of the proposed BCC to form the OCC.

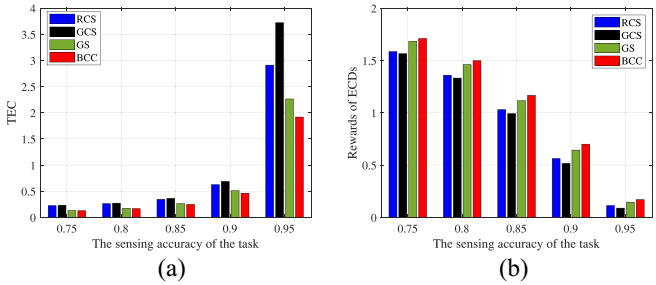
Fig. 5(a) depicts the TEC of the task by changing the maximum time spent on completing the crowdsensing task. We can see from this figure that the proposed BCC can lead to the lowest TEC for completing the crowdsensing task compared with the conventional schemes. With the increase of the maximum time spent on completing the task, the differences between the proposed BCC and other schemes become large. In addition, with the increase of the maximum time to complete the task, the average time of AVs to complete the task becomes long. Along with this, the probability that the task can be completed by AVs becomes low. Consequently, the TEC in all the schemes increases with the increase of the maximum time spent on completing the task.

Fig. 5(b) shows the rewards of the ECDs in the networks by changing the maximum time spent on completing the crowdsensing task. From this figure, it can be seen that the proposal



(a) (b)

Fig. 6. (a) TEC of the task by changing the number of AVs. (b) Rewards of ECDs by changing the number of AVs.



(a) (b)

Fig. 7. (a) TEC of the task by changing the sensing accuracy requested by the task. (b) Rewards of ECDs by changing the sensing accuracy requested by the task.

can result in the highest rewards for ECDs compared with other schemes. Furthermore, we can see from this figure that the rewards of ECDs decrease with the increase of the maximum time spent on completing the task in all the schemes. The reason for this is that with the increase of the maximum time spent on completing the task, each AV needs a longer time to complete the task individually. As the deadline for completing the task does not change, the probability to complete the task by the AVs becomes low which decreases the rewards of the ECDs in the AVNs.

Fig. 6(a) shows the TEC of the task by changing the number of AVs. From this figure, we can see that the BCC can lead to the lowest TEC for completing the task no matter the number of AVs is large or small. This is because the BCC can select the optimal AVs to form the OCC for completing the task. Furthermore, we can see that the TEC for completing the task decreases with the increase of the number of AVs. This is because the increase of the number of AVs leads to the fact that more AVs with a low sensing time can be selected to form the OCC to minimize the TEC. Therefore, the TEC decreases with the increase of the number of AVs.

Fig. 6(b) depicts the rewards of the ECDs in the networks by changing the number of AVs. With different numbers of AVs, we can see from this figure that the BCC can obtain higher rewards than the other schemes. The reasons for this are as follows. First, the probability that the task can be completed by AVs increases with the increase of the number of AVs. In this way, more rewards can be obtained by ECDs. Second, as we can see from Fig. 6(a), the TEC for completing the task decreases with the increase of the number of AVs. As a result, more rewards for completing the task then can be distributed to the ECDs which contribute their resources for publishing the task and generating the new block.

Fig. 7(a) shows the TEC of the task by changing the sensing accuracy requested by the task. By comparing with other schemes, it can be seen in this figure that the BCC can lead to the lowest TEC. In addition, with the increase of the sensing accuracy requested by the task, only a small number of AVs can be selected to form the coalition to complete the task collaboratively. The probability to complete the crowdsensing task thus becomes low. On the other hand, for the AVs with a sensing accuracy that is larger than the requested sensing accuracy, they may request a high reward for executing the crowdsensing task, resulting in a high TEC for completing the crowdsensing task.

Fig. 7(b) shows the rewards of the ECDs in the networks by changing the sensing accuracy requested by the crowdsensing task. With the designed BCC, the OCC for completing the task can be formed by the AVs to minimize the TEC. As a result, we can see from this figure that the BCC can obtain the highest rewards compared with other schemes. In addition, the rewards of ECDs decrease with the increase of the sensing accuracy requested by the task. When the requested sensing accuracy is 0.95, a large number of AVs cannot be used to form the OCC. Along with this, the probability that the available AVs can complete the crowdsensing task within the deadline becomes low. Consequently, compared with other values, the rewards of the ECDs in the AVNs are lower when the requested sensing accuracy is 0.95.

VI. Conclusion

In this article, we have proposed BCC to support vehicular crowdsensing in AVNs. In the BCC, we have designed the blockchain-based transaction architecture to guarantee the privacy of AVs and enable ECDs to earn rewards securely. With the blockchain architecture, we have designed a coalition game model to formulate the interactions among the AVs, where the transferable rewards can motivate AVs to collaboratively execute the crowdsensing tasks. Based on the game model, a coalition formation algorithm has been developed to select a group of AVs to form the OCC with the target of minimizing the TEC. By analyzing the security and evaluating the performance of the proposed BCC, we have shown that BCC can effectively defend against the attacks and outperform the conventional schemes in terms of the TEC for completing the task and the rewards obtained by the ECDs.

For further work, based on the various behaviors of the AVs in the networks, we plan to design vehicle classification schemes to further improve the performance of vehicular crowdsensing. In addition, by considering the frequent data sharing in the AVNs, the integration of digital twin technology and vehicular crowdsensing will be considered to reduce the overhead of the crowdsensing system.

Appendix A
Proof of Theorem 1

We consider two coalitions $I^{'}(I^{'} \subseteq I^{*})$ and $I^{''}(I^{''} \subseteq I^{*})$, where both of them only have one AV in the coalition, namely,

$I^{'} = \{i^{'}\}$ and $I^{''} = \{i^{''}\}$. Obviously, we have $I^{'} \cap I^{''} = \emptyset$. The two coalitions can form a new coalition, denoted as $\bar{I} = I^{'} \cup I^{''} = \{i^{'}, i^{''}\}$. We assume that both the two AVs can complete the crowdsensing task individually. In other words, we have $t_{i_{j_k}^{'}} \leq T_{n_k}$ and $t_{i_{j_k}^{''}} \leq T_{n_k}$. We further assume $t_{i_{j_k}^{'}} < t_{i_{j_k}^{''}}$ and $r_{i_{j_k}^{'}} < r_{i_{j_k}^{''}}$. As such, the AV $i^{'}$ which declares a lower reward will be selected to execute the crowdsensing task. The reward of AV $i^{'}$ can be calculated by $R(I^{'}) = R(\{i^{'}\}) = r_{i_{j_k}^{'}} t_{i_{j_k}^{'}}$. Accordingly, the reward of AV $i^{''}$ in coalition $I^{''}$ is zero, shown as $R(I^{''}) = R(\{i^{''}\}) = 0$. Then, we calculate the reward of coalition $\bar{I}$, shown as

$$R(I^{'} \cup I^{''}) = \frac{r_{i_{j_k}^{'}} t_{i_{j_k}^{'}} + r_{i_{j_k}^{''}} t_{i_{j_k}^{''}}}{|\bar{I}|} = \frac{r_{i_{j_k}^{'}} t_{i_{j_k}^{'}} + r_{i_{j_k}^{''}} t_{i_{j_k}^{''}}}{2}. \tag{20}$$

Based on (20), we define Δ as

$$\Delta = R(I^{'}) + R(I^{''}) - R(I^{'} \cup I^{''}). \tag{21}$$

By substituting (20) into (21), we have

$$\Delta = R(I^{'}) + R(I^{''}) - R(I^{'} \cup I^{''}) = r_{i_{j_k}^{'}} t_{i_{j_k}^{'}} + 0 - \frac{r_{i_{j_k}^{'}} t_{i_{j_k}^{'}} + r_{i_{j_k}^{''}} t_{i_{j_k}^{''}}}{2} = \frac{r_{i_{j_k}^{'}} t_{i_{j_k}^{'}} - r_{i_{j_k}^{''}} t_{i_{j_k}^{''}}}{2}. \tag{22}$$

As $t_{i_{j_k}^{'}} < t_{i_{j_k}^{''}}$ and $r_{i_{j_k}^{'}} < r_{i_{j_k}^{''}}$, we have $\Delta < 0$, which means that $R(I^{'}) + R(I^{''}) < R(I^{'} \cup I^{''})$. Based on Definition 2, we thus can conclude that the coalition game is nonsuperadditive. The theorem is proved. ■

APPENDIX B
PROOF OF THEOREM 2

Based on Definition 3, we can know that a partition should satisfy the group rationality and individual rationality concurrently. Here, we prove the theorem according to the individual rational condition. To prove the individual rationality, we continue to consider the two coalitions (i.e., $I^{'} (I^{'} \subseteq I^{*})$ and $I^{''} (I^{''} \subseteq I^{*})$), where the assumptions are the same with the proof of Theorem 1. As such, AV $i^{'}$ can complete the task individually and obtain the reward $R(I^{'}) = R(\{i^{'}\}) = r_{i_{j_k}^{'}} t_{i_{j_k}^{'}}$. If the two coalitions form a new coalition, the reward of AV $i^{'}$ becomes $R(\{i^{'}\}) = (r_{i_{j_k}^{'}} t_{i_{j_k}^{'}}/2)$. Obviously, $r_{i_{j_k}^{'}} t_{i_{j_k}^{'}} > (r_{i_{j_k}^{'}} t_{i_{j_k}^{'}}/2)$, which means that if the two coalitions are merged, the reward of AV $i^{'}$ will become lower than acting alone. Based on Definitions 5 and 6, we therefore can conclude that the core of the proposed coalition game is empty. The theorem is proved. ■

ACKNOWLEDGMENT

The authors would like to thank the Editor and the anonymous reviewers for their careful reading and valuable suggestions that helped to improve the quality of this manuscript.

REFERENCES

[1] Y. Hui, Z. Su, and T. H. Luan, "Unmanned era: A service response framework in smart city," *IEEE Trans. Intell. Transp. Syst.*, early access, Feb. 18, 2021, doi: 10.1109/TITS.2021.3058385.

[2] S. Zhang, J. Chen, F. Lyu, N. Cheng, W. Shi, and X. Shen, "Vehicular communication networks in the automated driving era," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 26–32, Sep. 2018.

[3] J. Wang, J. Liu, and N. Kato, "Networking and communications in autonomous driving: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1243–1274, 2nd Quart., 2019.

[4] H. Peng, Q. Ye, and X. S. Shen, "SDN-based resource management for autonomous vehicular networks: A multi-access edge computing approach," *IEEE Wireless Commun.*, vol. 26, no. 4, pp. 156–162, Aug. 2019.

[5] Z. Su, Y. Hui, and T. H. Luan, "Distributed task allocation to enable collaborative autonomous driving with network softwarization," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 10, pp. 2175–2189, Oct. 2018.

[6] R. Hussain and S. Zeadally, "Autonomous cars: Research results, issues, and future challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1275–1313, 2nd Quart., 2019.

[7] S. Kuutti, S. Fallah, K. Katsaros, M. Dianati, F. Mccullough, and A. Mouzakitis, "A survey of the state-of-the-art localization techniques and their potentials for autonomous vehicle applications," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 829–846, Apr. 2018.

[8] N. Cheng *et al.*, "Big data driven vehicular networks," *IEEE Netw.*, vol. 32, no. 6, pp. 160–167, Nov./Dec. 2018.

[9] Y. Hui, Z. Su, T. H. Luan, and C. Li, "Reservation service: Trusted relay selection for edge computing services in vehicular networks," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 12, pp. 2734–2746, Dec. 2020.

[10] H. Peng and X. Shen, "Deep reinforcement learning based resource management for multi-access edge computing in vehicular networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 4, pp. 2416–2428, Oct.–Dec. 2020.

[11] N. Cheng *et al.*, "Air-ground integrated mobile edge networks: Architecture, challenges, and opportunities," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 26–32, Aug. 2018.

[12] Y. Hui, Z. Su, T. H. Luan, and J. Cai, "Content in motion: An edge computing based relay scheme for content dissemination in urban vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 8, pp. 3115–3128, Aug. 2019.

[13] L. Zhu, C. Zhang, C. Xu, and K. Sharif, "RTSense: Providing reliable trust-based crowdsensing services in CVCC," *IEEE Netw.*, vol. 32, no. 3, pp. 20–26, May/Jun. 2018.

[14] A. Kaci and A. Rachedi, "MC-track: A cloud based data oriented vehicular tracking system with adaptive security," in *Proc. IEEE GLOBECOM*, 2020, pp. 1–6.

[15] C. Huang *et al.*, "RepChain: A reputation-based secure, fast, and high incentive blockchain system via sharding," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4291–4304, Mar. 2021.

[16] Y. Yahiatene and A. Rachedi, "Towards a blockchain and software-defined vehicular networks approaches to secure vehicular social network," in *Proc. IEEE CSCN*, 2020, pp. 1–7.

[17] A. Kaci and A. Rachedi, "Toward a machine learning and software defined network approaches to manage miners' reputation in blockchain," *J. Netw. Syst. Manag.*, vol. 28, pp. 478–501, Jul. 2020.

[18] J. Yu, D. Kozhaya, J. Decouchant, and P. Esteves-Verissimo, "RepuCoin: Your reputation is your power," *IEEE Trans. Comput.*, vol. 68, no. 8, pp. 1225–1237, Aug. 2019.

[19] A. Kaci and A. Rachedi, "PoolCoin: Toward a distributed trust model for miners' reputation management in blockchain," in *Proc. IEEE CCNC*, 2020, pp. 1–6.

[20] J. Kang *et al.*, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4660–4670, Jun. 2019.

[21] D. B. Rawat, R. Doku, A. Adebayo, C. Bajracharya, and C. Kamhoua, "Blockchain enabled named data networking for secure vehicle-to-everything communications," *IEEE Netw.*, vol. 34, no. 5, pp. 185–189, Sep./Oct. 2020.

[22] Z. Su, Y. Wang, Q. Xu, M. Fei, Y.-C. Tian, and N. Zhang, "A secure charging scheme for electric vehicles with smart communities in energy blockchain," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4601–4613, Jun. 2019.

[23] T. Jiang, H. Fang, and H. Wang, "Blockchain-based Internet of Vehicles: Distributed network architecture and performance analysis," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4640–4649, Jun. 2019.

[24] L. Xiao, T. Chen, C. Xie, H. Dai, and H. V. Poor, "Mobile crowdsensing games in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1535–1545, Feb. 2018.
[25] J. Ni, A. Zhang, X. Lin, and X. Shen, "Security, privacy, and fairness in fog-based vehicular crowdsensing," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 146–152, Jun. 2017.
[26] X. Wang *et al.*, "A city-wide real-time traffic management system: Enabling crowdsensing in social Internet of Vehicles," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 19–25, Sep. 2018.
[27] F. Campioni, S. Choudhury, K. Salomaa, and S. G. Akl, "Improved recruitment algorithms for vehicular crowdsensing networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1198–1207, Feb. 2019.
[28] J. Huang *et al.*, "Blockchain-based mobile crowd sensing in industrial systems," *IEEE Trans. Ind. Informat.*, vol. 16, no. 10, pp. 6553–6563, Oct. 2020.
[29] Q. Kong, L. Su, and M. Ma, "Achieving privacy-preserving and verifiable data sharing in vehicular fog with blockchain," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 8, pp. 4889–4898, Aug. 2021.
[30] Y. Liang, Y. Li, and B. Shin, "Faircs-blockchain-based fair crowdsensing scheme using trusted execution environment," *Sensors*, vol. 20, no. 11, pp. 1–15, Jun. 2020.
[31] J. Wang, X. Feng, T. Xu, H. Ning, and T. Qiu, "Blockchain-based model for nondeterministic crowdsensing strategy with vehicular team cooperation," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8090–8098, Sep. 2020.
[32] J. Hu, K. Yang, K. Wang, and K. Zhang, "A blockchain-based reward mechanism for mobile crowdsensing," *IEEE Trans. Comput. Soc. Syst.*, vol. 7, no. 1, pp. 178–191, Feb. 2020.
[33] C. Chen, J. Hu, T. Qiu, M. Atiquzzaman, and Z. Ren, "CVCG: Cooperative V2V-aided transmission scheme based on coalitional game for popular content distribution in vehicular ad-hoc networks," *IEEE Trans. Mobile Comput.*, vol. 18, no. 12, pp. 2811–2828, Dec. 2019.
[34] K. Xu, K.-C. Wang, R. Amin, J. Martin, and R. Izard, "A fast cloud-based network selection scheme using coalition formation games in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 11, pp. 5327–5339, Nov. 2015.
[35] W. Saad, Z. Han, A. Hjorungnes, D. Niyato, and E. Hossain, "Coalition formation games for distributed cooperation among roadside units in vehicular networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 1, pp. 48–60, Jan. 2011.
[36] Z. Zhou, H. Yu, C. Xu, Y. Zhang, S. Mumtaz, and J. Rodriguez, "Dependable content distribution in D2D-based cooperative vehicular networks: A big data-integrated coalition game approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 953–964, Mar. 2018.
[37] Y. Hui, Z. Su, T. H. Luan, and J. Cai, "A game theoretic scheme for optimal access control in heterogeneous vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 12, pp. 4590–4603, Dec. 2019.
[38] C. Zhang *et al.*, "Bsfp: Blockchain-enabled smart parking with fairness, reliability and privacy protection," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 6578–6591, Jun. 2020.
[39] Y. Yao, X. Chang, J. Misic, V. Misic, and L. Li, "Bla: Blockchain-assisted lightweight anonymous authentication for distributed vehicular fog services," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3775–3784, Apr. 2019.
[40] C. Zhang, L. Zhu, C. Xu, and K. Sharif, "PRVB: Achieving privacy-preserving and reliable vehicular crowdsensing via blockchain oracle," *IEEE Trans. Veh. Technol.*, vol. 70, no. 1, pp. 831–843, Jan. 2021.
[41] Y. Zhang, B. Di, P. Wang, J. Lin, and L. Song, "HetMec: Heterogeneous multi-layer mobile edge computing in the 6G era," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4388–4400, Apr. 2020.
[42] Y. Hui *et al.*, "Secure and personalized edge computing services in 6G heterogeneous vehicular networks," *IEEE Internet Things J.*, early access, Mar. 15, 2021, doi: 10.1109/JIOT.2021.3065970.
[43] A. B. T. Sherif, K. Rabieh, M. M. E. A. Mahmoud, and X. Liang, "Privacy-preserving ride sharing scheme for autonomous vehicles in big data era," *IEEE Internet Things J.*, vol. 4, no. 2, pp. 611–618, Apr. 2017.
[44] J. Ni, K. Zhang, Y. Yu, X. Lin, and X. Shen, "Privacy-preserving smart parking navigation supporting efficient driving guidance retrieval," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6504–6517, Jul. 2018.
[45] Y. Li and B. Hu, "A consortium blockchain-enabled secure and privacy-preserving optimized charging and discharging trading scheme for electric vehicles," *IEEE Trans. Ind. Informat.*, vol. 17, no. 3, pp. 1968–1977, Mar. 2021.
[46] Q. Yang and H. Wang, "Privacy-preserving transactive energy management for IoT-aided smart homes via blockchain," *IEEE Internet Things J.*, vol. 8, no. 14, pp. 11463–11475, Jul. 2021.
[47] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Performance optimization for blockchain-enabled Industrial Internet of Things (IIoT) systems: A deep reinforcement learning approach," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3559–3570, Jun. 2019.
[48] Y. Wang, Y. Niu, H. Wu, Z. Han, B. Ai, and Q. Wang, "Sub-channel allocation for device-to-device underlaying full-duplex mmWave small cells using coalition formation games," *IEEE Trans. Veh. Technol.*, vol. 68, no. 12, pp. 11915–11927, Dec. 2019.
[49] N. Zhao, H. Wu, and Y. Chen, "Coalition game-based computation resource allocation for wireless blockchain networks," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8507–8518, Oct. 2019.
[50] Q. Pham, H. T. Nguyen, Z. Han, and W. Hwang, "Coalitional games for computation offloading in noma-enabled multi-access edge computing," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 1982–1993, Feb. 2020.
[51] Y. Wang, H. Wu, Y. Niu, Z. Han, B. Ai, and Z. Zhong, "Coalition game based full-duplex popular content distribution in mmWave vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13836–13848, Nov. 2020.
[52] Y. Chen, B. Ai, Y. Niu, K. Guan, and Z. Han, "Resource allocation for device-to-device communications underlaying heterogeneous cellular networks using coalitional games," *IEEE Trans. Wireless Commun.*, vol. 17, no. 6, pp. 4163–4176, Jun. 2018.
[53] B. Cao, X. Wang, W. Zhang, H. Song, and Z. Lv, "A many-objective optimization model of Industrial Internet of Things based on private blockchain," *IEEE Netw.*, vol. 34, no. 5, pp. 78–83, Sep./Oct. 2020.

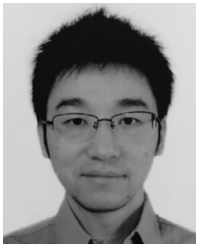
Yilong Hui (Member, IEEE) received the Ph.D. degree in control theory and control engineering from Shanghai University, Shanghai, China, in 2018.

He is currently a Lecturer with the State Key Laboratory of Integrated Services Networks and the School of Telecommunications Engineering, Xidian University, Xi'an, China. He has published over 30 scientific articles in leading journals and international conferences. His research interests include wireless communication, vehicular networks, and autonomous driving.

Dr. Hui was a recipient of the Best Paper Award of International Conference WiCon2016 and IEEE Cyber-SciTech2017.

Yuanhao Huang received the B.Eng. degree in communication and information system from Xidian University, Xi'an, China, in 2020, where he is currently pursuing the master's degree with the School of Telecommunications Engineering.

His current research interests include blockchain and vehicular networks.



Zhou Su (Senior Member, IEEE) received the Ph.D. degree from Waseda University, Tokyo, Japan, in 2003.

His research interests include multimedia communications, wireless communications, and network traffic.

Dr. Su received the Best Paper Award of the International Conference IEEE BigdataSE2019, the IEEE ComSoc GCCTC2018, the IEEE CyberSciTech2017, and the Funai Information Technology Award for Young Researchers in 2009. He is a TPC Member of some flagship conferences, including the IEEE INFOCOM, the IEEE ICC, and the IEEE Globecom. He is the Chair of the Multimedia Services and Applications Over Emerging Networks Interest Group (MENIG), the IEEE ComSoc Society, and the Multimedia Communications Technical Committee. He also served as the Co-Chair for several international conferences, including the IEEE Vehicular Technology Conference Spring 2016 and the IEEE Consumer Communications and Networking Conference 2011. He is an Associate Editor of the IEEE OPEN JOURNAL OF COMPUTER SOCIETY and *IET Communications*.

Tom H. Luan (Senior Member, IEEE) received the B.E. degree in electrical and computer engineering from Xi'an Jiaotong University, Xi'an, China, in 2004, the master's degree in electrical and computer engineering from Hong Kong University of Science and Technology, Hong Kong, in 2007, and the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2012.

He is a Professor with the School of Cyber Engineering, Xidian University, Xi'an. He has authored/coauthored around 70 journal papers and 40 technical papers in conference proceedings, and awarded one U.S. patent. His research mainly focuses on the content distribution and media streaming in the vehicular ad hoc networks and peer-to-peer networking, and protocol design and performance evaluation of wireless cloud computing and fog computing.

Nan Cheng (Member, IEEE) received the B.E. and M.S. degrees from the Department of Electronics and Information Engineering, Tongji University, Shanghai, China, in 2009 and 2012, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, in 2016.

He worked as a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada, from 2017 to 2019. He is currently a Professor with the State Key Laboratory of ISN and the School of Telecommunications Engineering, Xidian University, Xi'an, China. His current research focuses on B5G/6G, space–air–ground integrated network, big data in vehicular networks, and self-driving system. His research interests also include performance analysis, opportunistic communication, and application of AI for vehicular networks.

Xiao Xiao received the B.S. degree in control technology and instrumentation from Xidian University, Xi'an, China, in 2004, and the Ph.D. degree in measuring and testing technologies and instruments from Zhejiang University, Hangzhou, China, in 2009.

He is currently a Lecturer with the School of Telecommunications Engineering, Xidian University. He was a Research Fellow with the Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville, TN, USA, from 2019 to 2020. His research interests include intelligent transportation system, computer vision, and deep learning.

Guoru Ding (Senior Member, IEEE) received the B.S. degree (Hons.) in electrical engineering from Xidian University, Xi'an, China, in 2008, and the Ph.D. degree (Hons.) in communications and information systems from the College of Communications Engineering, Army Engineering University, Nanjing, China, in 2014.

He is currently an Associate Professor with the College of Communications Engineering, Army Engineering University. From 2015 to 2018, he was a Postdoctoral Research Associate with the National Mobile Communications Research Laboratory, Southeast University, Nanjing. His research interests include cognitive radio networks, massive MIMO, machine learning, and data analytics over wireless networks.

Dr. Ding has received the Excellent Doctoral Thesis Award of the China Institute of Communications in 2016, the Alexander von Humboldt Fellowship in 2017, the Excellent Young Scientist of Wuwenjun Artificial Intelligence in 2018, and the 14th IEEE COMSOC Asia–Pacific Outstanding Young Researcher Award in 2019. He was a recipient of the Natural Science Foundation for Distinguished Young Scholars of Jiangsu Province, China, and six best paper awards from international conferences, such as the IEEE VTC-FALL 2014. He has served as a Guest Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (special issue on spectrum sharing and aggregation in future wireless networks). He is currently an Associate Editor of the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING and a Technical Editor of the IEEE 1900.6 Standard Association Working Group.