Blockchain-Based Secure Crowdsourcing in Wireless IoT

Daquan Feng, Long Zhang, Shengli Zhang, Qihui Wu, Xianggen Xia

***Abstract*—With the explosive growth of interconnected smart devices and sensors, the Internet has been entering the Internet of things (IoT) era and revolutionizing many aspects of our daily life. Meanwhile, crowdsourcing has been considered as a promising technology to realize collaborative intelligence. Therefore, more and more IoT-based crowdsourcing applications are emerged to take advantages of the widely distributed IoT devices to sense, collect, and analyze data with the aim to solve complex and nontrivial tasks. However, there exist many technical challenges to be addressed in the IoT-based crowdsourcing, such as security, privacy, and incentive provision. In this paper, we propose a blockchain-based architecture as an integrated solution to realize the secure and trustworthy crowdsourcing in wireless IoT. We first overview the challenges in the traditional crowdsourcing system. Then, we briefly introduce the background of the blockchain and smart contract, and propose a blockchain-based crowdsourcing architecture. In particular, we elaborate the utilization of smart contract on the specific phases of crowdsourcing. By deploying the smart contract instance, we confirm the proposed blockchain-based architecture is feasible.**

***Keywords*—crowdsourcing, blockchain, wireless IoT, security, privacy, incentive**

Manuscript received Dec. 18, 2021; revised Jan. 08, 2022; accepted Jan. 15, 2022. This work was supported in part by the National Key R&D Program of China under Grant 2020YFB1807601, the National Natural Science Foundation of China under Grant 62171291, Guangdong Provincial Department of Science and Technology under Project No.2020B1212030002, Shenzhen Science, and Technology Program under Grant JCYJ20210324095209025, and Guangxi Science and Technology Base and Talent Special Project under Grand AD19110042. The associate editor coordinating the review of this paper and approving it for publication was S. Zhou.

D. Q. Feng, S. L. Zhang. Guangdong Province Engineering Laboratory for Digital Creative Technology, Shenzhen University, Shenzhen 518060, China (e-mail: fdquan@szu.edu.cn; zsl@szu.edu.cn).

L. Zhang. National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: zhanglong3211@yeah.net).

Q. H. Wu. College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China (e-mail: wuqihui2014@sina.com).

X. G. Xia. Department of Electrical and Computer Engineering, University of Delaware, Newark, DE 19716, USA (e-mail: xianggen@udel.edu).

I. Introduction

With the rapid evolution of wireless communications and the continued advances in smart devices, more and more physical objects with certain sensing, computing, and communication capacities, such as sensors, robots, vehicles, meters, wearables, and other consumer electronics, are connected to the Internet and lead to the Internet of things (IoT)[1,2]. The IoT enables the communication and interaction between people and things, and between things themselves, and thus revolutionizes many aspects of our daily life, including healthcare, home automation, retail business, transportation, environment monitoring, agricultural production, and manufacturing operations[3]. At the same time, it is widely accepted that the connected devices will continue to rise exponentially in the coming years, and we are entering the IoT era[4]. According to the recent report from International Data Corporation (IDC), the global IoT spending will surpass $1 trillion in 2020[5].

Billions or even trillions of smart devices and ubiquitous wireless connections have spurred the combination of the IoT and crowdsourcing which leverages a large amount of participants to achieve a cumulative output. In this context, IoT can be exploited to implement computationally difficult or other intractable tasks that are far beyond the scope of small scale devices deployed by a certain person, company, or community in a local area. In brief, the combination can provide us a promising solution with improved costs, speed, quality, flexibility, scalability, or diversity in various fields, including marketing information collection, healthcare data integration, air/water quality detection, intelligent transportation, and emergency preparedness[3,6]. Moreover, the combination of IoT and crowdsourcing can bring significant social benefits with respect to environment protection, disease prevention, smart city construction and public safety management besides the traditional economic rewards.

To obtain the great potential benefits of crowdsourcing in

wireless IoT, various applications, and systems as well as research projects have been developed[7]. However, there are still a number of technical challenges to be addressed, like security, privacy, and incentive provision with regard to user recruitment, data sensing, transmission, process, analysis, and mining, as well as utilization[8].

In this article, we will focus on state-of-the-art research in protocols, algorithms, and schemes for secure and trustworthy crowdsourcing in wireless IoT. To overcome the shortcomings in the traditional centralized crowdsourcing systems, we propose a blockchain-based architecture as an integrated solution to realize secure and trustworthy crowdsourcing in wireless IoT. The key idea is to build a decentralized crowdsourcing architecture to promise the data security and personal privacy and promote active participation of users.

The main contributions of this paper are as follows:

• Firstly, we analyze the security, privacy, and incentive challenges in the traditional crowdsourcing system, and enumerate some current widely used feasible schemes.

• Secondly, we propose a blockchain-based architecture to solve the problems in the traditional crowdsourcing system. According to the initiative of data aggregation, the blockchain-based secure crowdsourcing can be divided into two modes, namely, crowdsourcer-to-crowdsourcee mode and crowdsourcee-to-crowdsourcer mode. And the smart contract is introduced to automate the whole crowdsourcing operation process.

• Then, we deploy a voting smart contract on Ethereum to valid the proposed blockchain-based architecture for crowdsourcing, and the experimental results show that our idea is feasible.

• Finally, we summarize some potential technology challenges when combining the blockchain technology with crowdsourcing.

The rest of this article is organized as follows. We review the related work in section II. In section III, the specific security and privacy challenges of crowdsourcing in wireless IoT are introduced. In section IV, blockchain-based secure architecture for crowdsourcing is described. Simulation and analysis are shown in section V. Finally, conclusions are presented in section VI.

II. Related Work

Currently, many works have dedicated to the security, privacy, and incentive issues in the crowdsourcing[9]. In general, these works can be classified as centralized, distributed, and blockchain-based methods with respect to the architecture design.

For the centralized paradigm, the authors of Ref. [10] designed a trustworthy crowdsourcing model in social Internet of things (SIoT) in which a reputation-based auction mechanism is used for winner selection and payment determination to overcome some security threats, i.e., the denial of service (DoS) and distributed DoS (DDoS) attacks. However, it totally depends on the social cloud to realize data computing and storage functions, and the trustworthiness of sensed data, transmission, and computing results is vulnerable due to the non-cooperative sensing entities. Ref. [11] proposed a game-theoretic approach to solve the trustworthiness and truthfulness challenges among users and maximize the platform and users' utility. To recruit adequate users, some incentive mechanisms have been proposed to compensate for the users, such as the offline mechanisms[3,12] and the dynamic online mechanism[13,14]. However, these works also expose some problems. For instance, the offline incentive mechanism may make the system lack flexibility while the dynamic online mechanism may not recruit sufficient active and qualified users in time[15].

For the distributed paradigm, Ref. [16] designed a fully distributed auction mechanism between multiple crowdsourcers and crowdsourcees in a sensing market. In Ref. [17], the authors aimed to use crowdsourcing behaviours to realize energy management, and proposed an aggregated model to make various operators coordinate and collaborate with each other on a diverse collection of distributed energy resources (DERs), load, and storage. Although they have made full use of the advantages of distributed architecture, the device privacy and data authenticity have not been considered.

For the blockchain-based crowdsourcing, Ref. [18] proposed blockchain-based credit and arbitration mechanism to motivate crowdsourcers to select suitable workers and fairly evaluate the data submitted by crowdsourcees. In Refs. [19,20], the authors proposed a blockchain-powered crowdsourcing method (BPCM) to preserve the participants' privacy in mobile environment. Ref. [21] introduced some real-life examples to present the benefits of blockchain bringing for the crowdsourcing use cases. Nevertheless, most of the existing works are only focused on parts of the challenges in crowdsourcing from single dimension, resulting in the lack of a comprehensive system architecture and fine-grained process. Thus, to make full use of blockchain technology, we propose a blockchain-based architecture and combine the smart contract to describe the business logic for crowdsourcing.

III. Security, Privacy, and Incentive Challenges of Crowdsourcing in Wireless IoT

The traditional crowdsourcing model usually consists of three parts: a crowdsourcer, a large group of crowdsourcees, and crowdsourcing processing system, as shown in Fig. 1. The crowdsourcer is the initiator of the crowdsourcing and

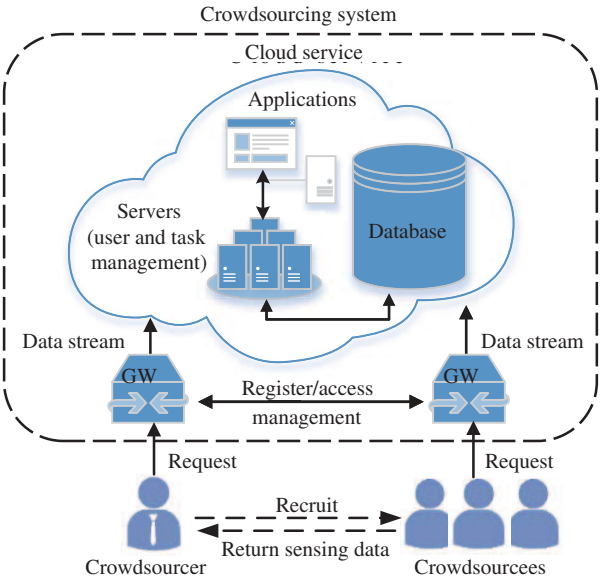


Fig. 1 Traditional crowdsourcing model

responsible for recruiting the crowdsourcees to participate in the tasks. Besides, the crowdsourcer also processes the sensing data for business or other specific purposes. Accordingly, the crowdsourcees are stimulated by the incentive reward to work for the crowdsourcer. The crowdsourcing processing system is composed of gateway and cloud service, in which the cloud service includes various application platforms for the sake of data visualization, high-performance data computing, and converting servers, and large capacity database. Specifically, the gateway is in charge of the user access control while the cloud service is the intermediary place for data interactions, such as user (crowdsourcer/crowdsourcee) registration and update, task management, data invoking, authorization, and storage. Since the IoT devices are normally with limited computation and storage capacity due to the size and memory constraints, the utilization of cloud service can mitigate this problem. However, this centralized cloud service also brings many challenges on the aspects of security, privacy, and incentive. In the following, we will introduce the above challenges.

A. Security Challenge

For the crowdsourcing in wireless IoT, it usually goes through four stages: user recruitment, data sensing, data transmission, and data processing. All the four stages may involve security challenges due to the openness of crowdsourcing system and the massive data input and access by the widely distributed wireless IoT devices.

In the user recruitment stage, a unique identity is necessary for each user to join the crowdsourcing tasks. Since there are a number of widely distributed users in a large area to provide sensing services, a user may disguise himself with a fake identity and then does not provide any service or provide only junk data. Even worse, the malicious user can launch Sybil attack[22] which usually happens in large-scale P2P systems, hostile nodes may counterfeit multiple fictitious identities to subvert the system and achieve a disproportionate influence. Besides, the legal users may suffer from spamming and privacy exposure[23]. Sybil attack can be easily carried out but hard to be detected. It has eroded many distributed systems, such as vehicular Ad-hoc network (VANET), distributed voting, routing, storage[24] and mobile social systems[25]. Although the traditional identity and access management can be leveraged to defense against Internet protocol (IP) spoofing and IP forgery[26], it will cost significant signaling overhead and still be vulnerable to this identity fraud. Therefore, it is critical to design an efficient and effective authenticate scheme embedded in the crowdsourcing systems. By doing so, the users' behaviours can be trailed and audited, and thus the user will be more likely to behave loyally for the incentive reward.

In the data sensing stage, sensors' malfunction or misunderstanding of sensing tasks can render inaccurate data perception and collection. In addition, selfish legitimate users may submit faulted but authenticated data to obtain more benefits. Even worse, adversarial users may pollute data collection and even manipulate the output of the crowdsourcing system by inserting a large amount of junk data. The quality of sensing data directly determines the success or failure of the crowdsourcing task and affects, the subsequent decision making. Therefore, it is also imperative to develop a scheme to detect the anomalous data and prevent the pollution of the data statistics as well as punish the malicious users.

In the data transmission stage, the crowdsourcing system can be influenced by both external and internal malicious adversaries. The former refers to those users who have not been admitted to participate in the system eavesdrop the communications of the benign users and interfere the data transmission process. In this case, physical layer security is a major concern to threatened wireless communications. In particular, the eavesdroppers are unauthorized receivers that jam and intercept communications between benign users, which poses a challenge to reliable and robust transmission[27]. Note that there are many technologies to enable secure data transmission[28,29], thus this paper omits this part and focuses on the design of the whole system architecture. The latter refers to the legitimate users of the system, who collude to delay sending data to degrade the unity of crowdsourcing system. The internal malicious adversaries might also refuse to forward datas, especially in the limited resource environment. Moreover, malicious adversaries may launch the DoS attack and DDoS attack to threaten the security of system. DoS attack is a cyber-attack in which the malicious user intentionally sending an overwhelming amount of fake requests to oversaturate the capacity (e.g., computing,

caching, and communication) of the target service provider. It will cause that the normal requests are unable to be processed, resulting denial-of-service of the legitimate users. The common DoS attacks include smurf attack, ping flood, and ping of death[30]. DDoS attack is an evolution of the DoS attack. In general, DDoS attack utilizes many eroded users to hinder the normal services of legitimate users by flooding or crashing the target service provider. All these issues will degrade the unity of the sensing tasks and the network performance.

In the data processing stage, some computing entities may not work effectively and even attack the crowdsourcing system viciously. On one hand, some selfish computing entities may be reluctant to provide computing services in order to save energy resources. On the other hand, some malicious participants may directly provide invalid processing results to the service provider to disturb the normal operation of the system[10]. Sometimes, an honest user may also provide inaccurate computing results due to the misunderstanding of task requirements. What more serious is that, since the crowdsourcing system generally relies on the centralized cloud to process data[26], the computing results may be manipulated by the adversary for some purposes. In the centralized mode, the sensing entities are not able to protect the data by themselves, since the data processing is untouchable and non-transparent to all the users. Thus, it is also urgent to design a more transparent processing system so that the flow of sensing data can be visible by all the users.

To protect the security of system against the above described threats, some security-related services and mechanisms have been introduced in the literature including identity authentication, network security protocol, automated execution, and fault tolerance and resilience[8,10,31].

Therefore, it is necessary to secure all the four stages in the process of crowdsourcing and coordinating the diverse security mechanisms in different stages. In addition, to make the security stronger, it should be better to propose a holistic security mechanism that connects all parts into an integrated one with integration and interoperability.

B. Privacy Challenge

Privacy leakage is also a major challenge of IoT-based crowdsourcing system since both IoT deployment and crowdsourcing would aggravate the risk of personal data collection. The internal adversaries and honest but curious users can extract some individual sensitive information by analyzing the information tagged with the sensory data. For example, when a user participates in the noise detection task, its geographical location can be speculated through the submitted sensing data; the daily routines (e.g., working and shopping) can be analyzed by the traffic flow monitoring in the vehicle-based task[32]. Although these information is based on the user's spatio-temporal information, it can potentially disclose the user's living habits and personal preference. Thus, some privacy-protection mechanisms have been proposed, e.g., authorization and access management and data anonymization such as group signature[33] and K-anonymity[34].

In brief, a privacy-preserving crowdsourcing system should guarantee that users' individual privacy information is unlinkable with their sensing data tagged with spatio-temporal information. Once the privacy is disclosed, the system should be able to find the offender and penalize the misbehaviour.

C. Incentive Challenge

Incentive is another major concern in the IoT-based crowdsourcing system. For a successful crowdsourcing task, recruiting adequate users to participate and providing the high-quality sensory data is an important prerequisite. However, collecting sensory data is a costly activity for the users, especially for IoT devices with limited resource. The cost usually involves energy, network source (e.g., communication, caching, and computing), and time, and desponds on the difficulty of the sensing tasks. Except for the physical resource consumption, security threat, and privacy leakage would also affect the users' enthusiasm. Therefore, traditional voluntary crowdsourcing mechanisms may be not able to attract adequate users to work for the crowdsourcing tasks[3]. In addition, most of them are based on the centralized crowdsourcing architecture and depend on the support from a third party to establish trustworthiness between the crowdsourcer and the crowdsourcees. As a result, some new defects arise to be addressed, such as subject arbitration, high service fee and system vulnerability.

Therefore, on one hand, the incentive mechanism should attract enough users to participate in the crowdsourcing task and stimulate them to come out with desired products; on the other hand, it should also embed in fairness, low service fee, interference resistant as well as security and privacy-preservation.

D. Summary

The emergence of crowdsourcing paradigm brings new opportunity for collaborative intelligence to solve the computationally difficult or intractable tasks and also promotes the usage of users' idle resource. At the IoT era, any device with certain sensing, computing, and communication capacities can be a potential sensing user, which facilitates the development of crowdsourcing system. However, there also exist many challenges to be addressed, i.e., security, privacy, and incentive issues. In Tab. 1, we summarize these challenges and also give some potential solutions. In general, it is imperative to design an all-in-one secure and high-efficiency structure that considers the following requirements.

Robustness: The normal operation of the crowdsourcing system should not be affected by a single point of failure. That

Tab. 1 Challenges of crowdsourcing in wireless IoT

Challenge	Description	Potential solution
Security	User recruitment stage: sybil attack	Design an efficient and effective authenticate scheme
	Data sensing stage: inaccurate data perception and junk data collection	Develop a scheme to detect the anomalous data and prevent the pollution of the data statistics
	Data transmission stage: communications eavesdropping, DoS and DDoS attacks	Design a decentralized, self-organizing communication system
	Data processing stage: inaccurate or wrong computing results, untouchable and non-transparent data processing	Utilize data encryption technique: hash function and asymmetric encryption
Privacy	External adversaries	Utilize authorization and access management
	Internal adversaries and honest but curious users	Utilize data anonymization: group signature, K-anonymity, and other cryptography techniques
Incentive	Inadequate users' participation, high service fee, security threat, and privacy leakage	Design a fairness, low service fee, interference resistant as well as security and privacy-preservation incentive mechanism

is to say, the system should have a certain fault tolerance to resist against Sybil attack and DDoS attack.

Reliability: The reliability includes data authenticity, integrity, and immutability. Data authenticity refers to that the sensing data should not be corrupted from the original. Data integrity refers to that the sensing data should not be lost and damaged during the transmission and processing. Data immutability guarantees that the data has not be modified or altered during the processing.

Traceability: It is required that the sensing data can be traced, in case the submitted sensing data is maliciously utilized or tampered during the crowdsourcing process.

Fairness: The incentive mechanism should promise the fairness between crowdsourcer and crowdsourcee. The crowdsourcer pays rationally to get satisfying sensing data, and the crowdsourcee is rewarded in the proportion of their contributions.

Scalability: Due to the dynamic of user participation, it requires that the crowdsourcing system should be resilient to the change of scale. Meanwhile, the scalability is also affected by the incentive mechanism in some degree.

Cost-effect: Due to the large-scale applications and frequent transactions in wireless IoT, the crowdsourcing should be cost-effective and benefit both crowdsourcer and crowdsourcee.

IV. Blockchain and Smart Contract for Secure Crowdsourcing

A. Background of Blockchain and Smart Contract

Blockchain: Blockchain makes new solutions to the problems existing in the traditional crowdsourcing system, and helps to build a reliability, fair and secure crowdsourcing system. Blockchain records transaction data in blocks which are composed of the block header and block body. The block header is the container of the verification information for the block, including block version, timestamp, target hash, nonce, parent block hash, and Merkle root. The block body is used for storing transaction data, and also has a transaction counter to limit the block size. Each block connects with the others by embedding the hash of the previous block in sequential order, and thus makes transaction data immutable. The encryption mechanism containing hash operation and timestamp information enhances the reliability of transaction data.

In addition, different from the traditional centralized authority architecture, blockchain uses consensus mechanisms for distributed participates to reach agreement in the presence of faults. Particularly, consensus mechanisms use predefined protocols for untrustworthy nodes to mine, validate and insert the right block into the chain to maintain the security and legitimation of contents in the blocks. In this way, it can effectively solve the Byzantine generals problem in P2P networks. Currently, the widely used consensus mechanisms include proof-of-work (PoW)[35] and proof-of-stake (PoS)[36]. With the rise of blockchain technology, many new consensus mechanisms which make some improvement in terms of the energy consumption, scalability, and throughput have been proposed, e.g., proof of activity[37] , proof of burn[38], and Bitcoin-NG[39].

Smart contract: Smart contract is a sequence of self-executing and trustworthy computer-coded program proposed by Nick Szabo in 1994[40]. The major components of smart contract include address, value, function, and state. When crowdsourcees submit invoking transactions to the address of smart contract, all peers in the network will execute the programs on the basis of transaction value and the global state to realize the corresponding function, such as user identity management, task distribution and data evaluation[41]. In this way, smart contract can replace a trusted third party to verify and validate the contract between transaction counterparties.

In the crowdsourcing, smart contract can be designed and embedded into the blockchain system to simplify the working procedure. There is a unique address for the invoking transaction to trigger the execution of the smart contract. In other words, users can achieve verification and validation functions on blockchain by sending transactions to invoke the specific smart contract. Once the smart contract is triggered by the valid transactions, it will run automatically on the nodes in the blockchain network. The execution results will be verified by the other nodes, and the valid ones will be stored as a part of block content for future reference and audit. Therefore, smart contract can act as a coordinator which can help not only to realize auto-execute function but also to make an effective decision.

B. Blockchain-Based Secure Crowdsourcing

Crowdsourcing is benefited from the collaborative intelligence of unacquainted users. In view of the security, privacy and incentive challenges discussed in section II, we propose a blockchain- (BC-) based secure crowdsourcing architecture to deal with these issues. In the following, we will introduce the BC-based secure crowdsourcing in detail.

Fig. 2 illustrates the new framework of the BC-based crowdsourcing system. Compared with the traditional crowdsourcing model, the BC network embedded with smart contract is added as a self-executing and trustworthy agent to be in charge of the data verification. In particular, the peers in the BC network can be the crowdsourcer and other crowdsourcees who are committed to this task. They locally and independently run the smart contract, and it will not affect the global operation when there appears a failed node. Certainly, the BC is regarded as a distributed database for storing crowdsourcing tasks and metadata of the original sensing data, which can also help to improve the scalability of the system. The metadata includes ownership, timestamp, data hash value, and pointer, etc. The data hash ensures the immutability of the sensing data stored in the off-chain database and the data pointer is used to expedite the process of data retrieve.

Considering that most users in the IoT-based crowdsourcing system are only with limited energy reserve and storage space, the cloud will still play the role of server and off-chain database to compute and store the original sensing data. It is worth mentioning that the homomorphic encryption[42,43] will be used to protect the data security and privacy during the data processing operation in the cloud. With the help of homomorphic encryption, cloud provides the ability to compute the sensing data without first decrypting it. Therefore, it can reduce the computation and storage pressure and also alleviate the obstacle of message synchronization for the miners in the blockchain, meanwhile protecting the privacy of user data. In general, the BC network helps to improve the security and reliability and the cloud reduces the pressure on node computing and storage.

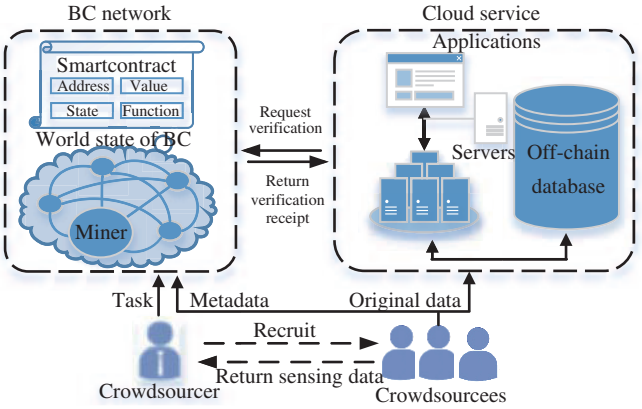


Fig. 2 BC-based secure crowdsourcing

In order to ensure the crowdsourcees to participate in crowdsourcing activity honestly, reputation value h_j is used to measure the crowdsourcee's trustworthiness after successful completion of each transaction, which is also used as the threshold of participating sensing task. The reputation value h_j is set as an initial value when the users register, and update when the crowdsourcing task completes. The update rule of reputation value h_j is based on the positive and negative evaluation results of sensing data quality. If the evaluation result of the sensing data output is positive, i.e., $R_{W_j} = P$, then it will increase by one. Otherwise, the reputation value will decrease by one.

$$h_j = \begin{cases} h_j + 1, & R_{W_j} = P; \\ h_j - 1, & R_{W_j} = N \text{ or invalid.} \end{cases} \tag{1}$$

Besides the function of reward, reputation value can also prevent the transaction forging and resist the malicious attack (e.g., DDoS and Sybil attacks) through filtrating the crowdsourcee's quality. In addition, reputation value can also play as a symbol to build the trustiness between the crowdsourcer and crowdsourcee. The higher the reputation value a user has, the more likely that the data its contributing will be trusted and approved by the miners.

Regarding to the security service, blockchain uses an asymmetric cryptographic technique to realize entity authentication (i.e., signature and verification) and data confidentiality (i.e., encryption and decryption). There are a pair of keys in the asymmetric cryptography: private key and public key. Particularly, private key is kept secretly by the user to encrypt/decrypt the message while public key is distributed over the system for users to verify the message from the source user. As illustrated in Fig. 3, a crowdsourcee digitally signs the sensing data with its private key $K^s_{W_j}$. Each user with public key $K^p_{W_j}$ can authenticate and verify the crowdsourcee. Data confidentiality has a similar procedure. The crowd-

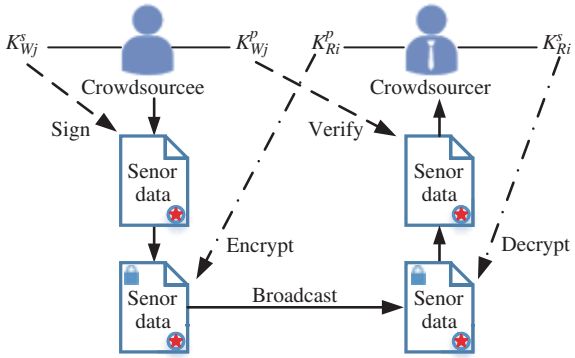


Fig. 3 Entity authentication and data confidentiality

sourcee encrypts the signed sensing data with the crowdsourcer's public key $K^p_{R_i}$. Only the crowdsourcer or other users have the crowdsourcer's private key $K^s_{R_i}$ information to decrypt the data.

V. Smart Contract on Crowdsourcing

The smart contract plays a key role in the proposed BC-based secure crowdsourcing. They process transactions automatically with deterministic output in the trustless environment. The overall BC-based secure crowdsourcing system includes four phases, i.e., user registration, task publication, data acceptance and data evaluation. According to the initiative of data aggregation, the BC-based secure crowdsourcing can be divided into two modes, namely, crowdsourcer-to-crowdsourcee mode and crowdsourcee-to-crowdsourcer mode. In the following, we will introduce the two modes with respect to the above four phases in detail.

A. Mode 1: Crowdsourcer-to-Crowdsourcee

In the crowdsourcer-to-crowdsourcee mode, the crowdsourcer takes the initiative to assign tasks to the crowdsourcees to prevent the remaining tasks in the system from piling up due to over saturation, so that the tasks can be processed in time. The crowdsourcer $R = \{1,2,\cdots,i\}$ takes the initiative to post tasks $T = \{T_1,T_2,\cdots,T_i\}$ and recruit crowdsourcees $W = \{1,2,\cdots,j\}$ to work. The incentive mechanism models the interaction between the crowdsourcer and crowdsourcee as a reverse auction. The crowdsourcer recruits crowdsourcees to work on the sensed task according to the submitted bid before a specified deadline. The qualified crowdsourcees submit the required sensing data to get a reward. The whole workflow is shown in Fig. 4, including four phases: user registration, task publication, data acceptance and data evaluation. In each phase, the corresponding smart contracts are triggered according to the flow of events, so as to execute the specific functions. In the following, we introduce the detailed procedures.

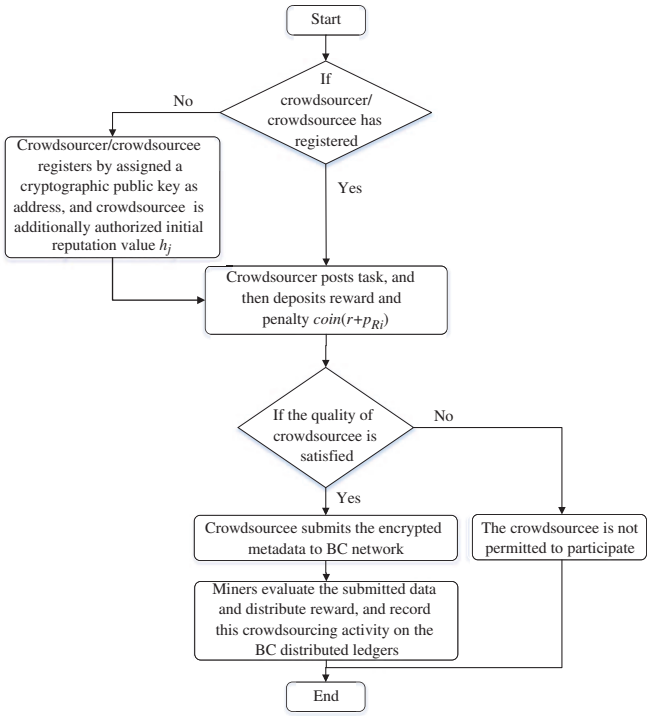


Fig. 4 BC-based crowdsourcing workflow (crowdsourcer-to-crowdsourcee)

1) User Registration: In this phase, the user (crowdsourcer R_i and crowdsourcee W_j) registers or updates the device identity in the blockchain network, and the contract is illustrated in Algorithm 1. For the first-time access, a user is required to register. User's information will be sent to the BC network to request a public key as address. The address does not include any identity information, which is just used for further reward or punishment according to the user's behavior and thus realizes user anonymization. In addition, the crowdsourcee will be given an initial reputation value h_j as introduced before. In the other cases, if a user has registered, it should query the BC to get the old identity registration information first and then request the key generator to update keys (i.e., assign a new user address) before initiating the new crowdsourcing task. The update of address can avoid address tracking in the case that a user always uses the same address, and thus help to improve the privacy level.

2) Task Publication: The task publication contract is illustrated in Algorithm 2. It is used for crowdsourcer to post tasks in the network. After the registration, the crowdsourcer will post a task by issuing a transaction tx_{T_i}, which includes task T_i, deadline time t_d, the maximum crowdsourcee number λ, reputation value threshold h for crowdsourcee and other special rules. Moreover, we introduce the concept of virtual coin $coin(\cdot)$ circulated in the BC network. The virtual coin is published by the crowdsourcing cloud severs and users can exchange it with currency. If the crowdsourcer deposits monetary reward (i.e., $coin(r)$ and $coin(p_{R_i})$) successfully, T_i

Algorithm 1 User registration contract

while each user R_i or W_j **do**
  if (reg_type = new) **then**
    R_iID / W_jID ← new ID (a unique user ID address is assigned by the key generator) ;
    tx_{R_i} ← set (R_iID) / tx_{W_j} ← set ($W_jID \cup h_j$) (create a new asset);
  end if
  if (reg_type = update) **then**
    $Query$ (tx_{R_i} / tx_{W_j}) (get the old identity registration information);
    tx_{R_i} ←set (update R_iID) / tx_{W_j} ←set (update $W_jID \cup h_j$); (update the old asset);
  end if
end while
return tx_{R_i} / tx_{W_j};

Algorithm 2 Task publication contract

while each task transaction tx_{T_i} **do**
  if (Deposit $coin(r + p_{R_i})$ successfully in the BC network) **then**
    Broadcast T_i in the BC network;
  else
    R_i posts task unsuccessfully;
    goto final;
  end if
end while
$j_{max} \leftarrow \lambda$;
$j \leftarrow 0$;
return $E(\cdot)$;

will be broadcasted in the BC network to recruit the crowdsourcees. Deposit $coin(r)$ is used to award crowdsourcee's contribution. $coin(p_{R_i})$ is defined as crowdsourcer's penalty to safeguard the right of good crowdsourcees, in case the crowdsourcer denies the crowdsourcees' contribution. Last, it will generate a task evaluation function $E(\cdot)$ to compare the match degree between the task T_i and sensing data D_{W_j}, and as a criterion to assign a reward for the workers.

3) Data Acceptance: The data acceptance contract is illustrated in Algorithm 3. It is used to provide the provenance of the sensing data which is recorded and stored on the BC network and off-chain database, respectively. When a crowdsourcee has accessed the BC network successfully, it will choose one task (e.g., T_i) and submit the bid $\beta = (t_a, p_j(T_i))$, where t_a ($t_a \leqslant t_d$) is the active time of crowdsourcee; $p_j(T_i)$ is the payment for crowdsourcee W_j to perform the task T_i; $v_j(\cdot)$ is a value function for the crowdsourcer to calculate the task value of crowdsourcee. Let $v_j(T_i) - p_j(T_i)$ denote the marginal value of the crowdsourcer to recruit crowdsourcee W_j to join the task which is the task value subtracting the payment. If the marginal value is non-negative, the crowdsourcee can be recruited to join the task. Similarly, the crowdsourcees are also required to deposit $coin(p_{W_j})$ to ensure the quality of the sensing data. In this way, if an attacker creates multiple virtual identities to maliciously hinder the BC network, it will lose a large amount of deposit. Therefore, it can greatly reduce the potential DDoS attack and Sybil attack and improve

Algorithm 3 Data acceptance contract

while T_i is available and now $< t_d$ **do**
  W_j submits bid $\beta = (t_a, p_j(T_i))$ for the task T_i;
  if ($j > \lambda$) **then**
    W_j cannot receive T_i;
    goto final;
  end if
  if ($t_a \geqslant t_d$) **then**
    W_j cannot receive T_i;
    goto final;
  end if
  if ($h_j < h$) **then**
    W_j cannot receive T_i;
    goto final;
  end if
  if ($v(T_i) - p_j(T_i) < 0$) **then**
    W_j cannot receive T_i;
    goto final;
  end if
  Deposit $coin(p_{W_j})$ in the BC network;
  $j = j++$;
  $Signature_{(D_{W_j})} \leftarrow$ Digital signature on D_{W_j} with $K^s_{W_j}$;
  $D^{encrypted}_{W_j} \leftarrow Encrypt\ (D_{W_j}, Signature_{(D_{W_j})})$ with $K^p_{R_i}$;
  if ($now \geqslant t_d$) **then**
    $D^{encrypted}_{W_j}$ cannot be submitted for timeout;
    goto final;
  end if
  $D^{hash}_{W_j} \leftarrow Hash(D^{encrypted}_{W_j})$;
  $D^{pointer}_{W_j} \leftarrow Aggregate(D^{encrypted}_{W_j})$;
  $t_s \leftarrow now$ (assign the current time to be submission time t_s);
end while
return $D^{hash}_{W_j}, D^{pointer}_{W_j}$;

the security of the crowdsourcing system. The procedure of selecting crowdsourcee according to task requirements will be executed iteratively until all tasks are assigned or deadline t_d is reached.

The data hash $D^{hash}_{W_j}$ and pointer $D^{pointer}_{W_j}$ will be stored on the BC network. $D^{hash}_{W_j}$ is used for data verification and retrieval in the BC network while $D^{pointer}_{W_j}$ is used for the data retrieve in the off-chain database. In this way, it becomes flexible to track data and control data lifecycle (i.e., data utilization, data availability, and data existence).

4) Data Evaluation: The data evaluation contract is illustrated in Algorithm 4. It is used to evaluate sensing data and distribute reward. This procedure is activated by miners when the deadline time has arrived. Crowdsourcee will obtain the corresponding reward if the authenticity of the contributed data is confirmed by the miners. If the data is verified as hostile or misleading data, the system will carry out punishment mechanism on the corresponding crowdsourcee (i.e., reduce reputation value h_j and confiscate $coin(p_{W_j})$). The rest of the reward $coin(r_{R_i})$ and the penalty $coin(p_{R_i})$ deposited by the crowdsourcer will be returned to his wallet W_{R_i}. In the end, the crowdsourcer's wallet W_{R_i}, crowdsourcee's wallet W_{W_j} and the reputation value h_j will be updated according to

Algorithm 4 Data evaluation contract

while each sensing data D_{W_j} **do**
  while $(t_s + t_c \leqslant t_d)$ **do**
    if (Verify $D_{W_j}^{hash}$ with $K_{W_j}^{p}$ is success) **then**
      evaluation result $\leftarrow E(D_{W_j})$;
      if evaluation result $(Q(D_{W_j}) = H)$ **then**
        $h_j \leftarrow h_j + 1$;
        $W_{W_j} \leftarrow coin(r/\lambda) + coin(p_{W_j})$;
      else
        evaluation result $(Q(D_{W_j}) = L)$;
        $h_j \leftarrow h_j - 1$;
        $W_{W_j} \leftarrow coin(p_{W_j})$;
      end if
    else
      verification failed;
      D_{W_j} is infeasible;
      $h_j \leftarrow h_j - 1$;
      $W_{W_j} \leftarrow 0$; (confiscate penalty deposit $coin(p_{W_j})$);
    end if
  end while
end while
$W_{R_i} \leftarrow$ set $(coin(r_{R_i} \cup p_{R_i}))$;
Update W_{R_i}, W_{W_j}, h_j;
return $profile\ (D_{W_j})$;

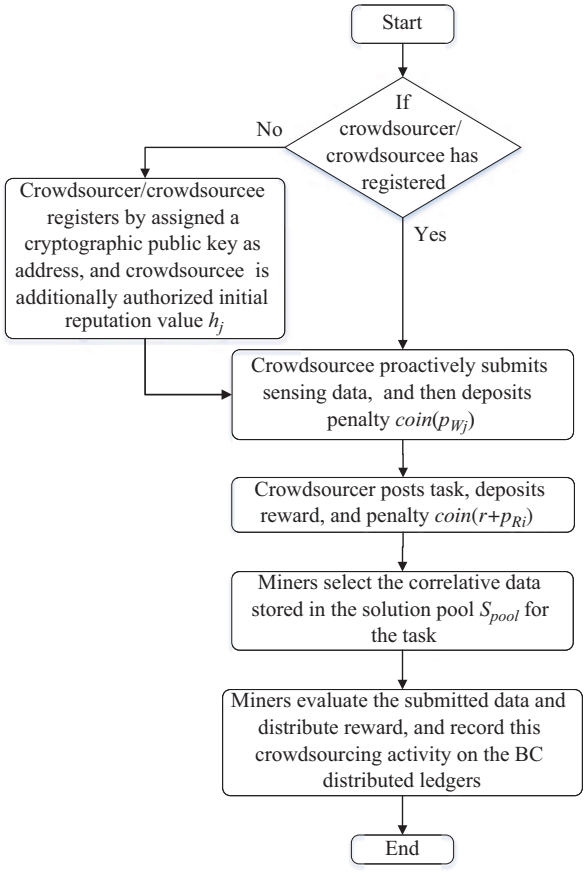


Fig. 5 BC-based crowdsourcing workflow (crowdsourcee-to-crowdsourcer)

the performance in this crowdsourcing activity. Besides, the task solution $profile\ (D_{W_j})$ will be recorded on the distributed ledgers in the BC network.

Algorithm 5 Data submission contract

while each crowdsourcee transaction tx_{W_j} **do**
  if tx_{W_j} is in the BC network **then**
    W_j submits encrypted sensing data $D_{W_j}^{encrypted}$;
    $D_{W_j}^{hash} \leftarrow Hash(D_{W_j}^{encrypted})$;
    $D_{W_j}^{pointer} \leftarrow Aggregate(D_{W_j}^{encrypted})$;
    Deposit $(coin(p_{W_j}) \cup h_j)$ (crowdsourcee deposits monetary penalty and reputation value);
    Put $D_{W_j}^{encrypted}$ into solution pool S_{pool};
    $t_s \leftarrow now$;
  else
    tx_{W_j} is not in the BC network;
    goto final;
  end if
end while
return $D_{W_j}^{hash}$, $D_{W_j}^{pointer}$, t_s;

B. Mode 2: Crowdsourcee-to-Crowdsourcer

In the crowdsourcee-to-crowdsourcer mode, the crowdsourcees take the initiative to choose tasks according to their abilities and interests, collecting and submitting sensing data for the coming reward, which is subsidized by the future data requester, i.e., crowdsourcers. The workflow is shown in Fig. 5, and it also includes four phases, i.e., user registration, data submission, data selection, and data evaluation.

1) User Registration: Similar to the crowdsourcer-to-crowdsourcee mode, it is also necessary for users to register before taking part in the crowdsourcing task. In this phase, the user registration contract is the same as Algorithm 1 in the crowdsourcer-to-crowdsourcee mode.

2) Data Submission: The data submission contract is illustrated in Algorithm 5. The permitted crowdsourcee will take the initiative to submit encrypted sensing data by utilizing idle resource for the potential reward. The sensing data includes data description, type, size, and ownership. Also, the crowdsourcee is required to deposit $coin(p_{W_j})$ as penalty, in case the submitted data is misleading or malicious. Then, the sensing data will be put into solution pool S_{pool} according to its type. The data hash value $D_{W_j}^{hash}$ and data pointer $D_{W_j}^{pointer}$ will be stored on the distributed ledgers for verification and retrieve.

3) Data Selection: The data selection contract is illustrated in Algorithm 6. The crowdsourcer first posts task T_i in the crowdsourcing application. And then, miners evaluate task T_i and feedback deposit requirement. When the crowdsourcer R_i deposits successfully, miners will select and verify the correlative data in the solution pool S_{pool}.

4) Data evaluation: The data evaluation contract is illustrated in Algorithm 7. For each selected crowdsourcee, it will bid γ for the data it has sensed. Miners will verify and evaluate the quality of the data and output the evaluation result $Q(D_{W_j})$. If the quality of the data result is high, the crowdsourcee will

Algorithm 6 Data selection contract

while each task transaction tx_{T_i} **do**
  if (Deposit $coin(r+p_{R_i})$ successfully in the BC network) **then**
    while $(j \leqslant \lambda)$ **do**
      Select and verify data using $D_{W_j}^{hash}$, $D_{W_j}^{pointer}$;
      if data type is the unexpected **then**
        D_{W_j} can not be adopted;
        $W_{W_j} \leftarrow coin(p_{W_j})$;
        goto final;
      end if
      if $h_j < h$ **then**
        D_{W_j} can not be adopted;
        $W_{W_j} \leftarrow coin(p_{W_j})$;
        goto final;
      end if
      if $t_s > t_d$ **then**
        D_{W_j} can not be adopted;
        $W_{W_j} \leftarrow coin(p_{W_j})$;
        goto final;
      end if
      if data size is over **then**
        D_{W_j} can not be adopted;
        $W_{W_j} \leftarrow coin(p_{W_j})$;
        goto final;
      end if
      $j = j++$;
    end while
  else
    R_i deposits unsuccessfully;
    goto final;
  end if
end while
return W;

be rewarded in reputation value and virtual coin. Otherwise, the crowdsourcee will not be rewarded and even be punished. Last, both the crowdsourcer's and crowdsourcees' wallets and the reputation value h_j will be updated, and the task solution profile D_{W_j} will be recorded on the distributed ledgers in the BC network.

C. Summary

To address the security, privacy, and incentive challenges in the traditional centralized crowdsourcing system, we propose a BC-based crowdsourcing architecture, which can lead to many benefits as shown in Tab. 2.

For the security issue, the proposed BC method eliminates the dependence of the centralized certificate authorities and thus reduces the risk of being manipulated by central cloud. The trustiness of the sensing data can also be improved by the distributed consensus and cryptographic technology. Therefore, DDoS and Sybil attacks can be resolved to a certain degree. In addition, the BC technology has provided several open-source implementations based on smart contract, such as Ethereum and Hyperledger, which are cost-effective and efficient for applications.

For the privacy issue, due to the decentralized architecture, the data access control list (ACL) is totally defined by the user.

Algorithm 7 Data evaluation contract

while each W_j in the W **do**
  Bid γ for sensing data D_{W_j};
  Miners verify and evaluate $D_{W_j}^{hash}$;
  evaluation result $\leftarrow Q(D_{W_j})$;
  if evaluation result $(Q(D_{W_j}) = H)$ **then**
    $h_j \leftarrow h_j + 1$;
    $W_{W_j} \leftarrow coin(r/\lambda) + coin(p_{W_j})$;
  else
    evaluation result $(Q(D_{W_j}) = L)$;
    $h_j \leftarrow h_j - 1$;
    $W_{W_j} \leftarrow coin(p_{W_j})$;
  end if
end while
$W_{R_i} \leftarrow$ set $(coin(r_{R_i} \cup p_{R_i}))$;
Update W_{R_i}, W_{W_j}, h_j;
return $profile(D_{W_j})$;

Thus, any unauthorized users or cloud owner will not be allowed to obtain the data. Similar to the pseudo-anonymous solution of bitcoin, the hashes can be used to hide the sensing crowdsourcee's real identify. Therefore, the real identity of the user is invisible to others so as to achieve the prevention of privacy disclosure. Moreover, smart contract can be used to define the privacy requirements and lifecycle of the collected data, including the acquisition, utilization, and the deletion of the data. Therefore, the network is more flexible.

On the incentive side, visual token, and reputation value are explored to stimulate the miners to verify transactions actively and continuously. In addition, compared with the traditional crowdsourcing system, we replace the certificate authority with the BC which can reduce the intermediate service fee. Thus, with the proposed BC-based crowdsourcing architecture, we can fulfill sensing tasks with high accuracy while minimizing the cost.

Though the BC-based crowdsourcing architecture can provide many benefits with respect to robustness, reliability, traceability, fairness, scalability, and cost, there remain some challenges to be further addressed, such as consensus mechanism vulnerability and legal issues.

Consensus mechanism plays the key role for the trustless participants to achieve consensus in a distributed crowdsourcing network. And it is also responsible for system security while defending against malicious attacks. For example, the widely used PoW consensus mechanism can avoid the double spending. However, the PoW will cause a high energy consumption, which is a challenge for the wireless IoT applications. With the burgeoning development of blockchain technology, many new consensus mechanisms are proposed in order to make some optimization on PoW and PoS algorithms to improve the performance in energy consumption, scalability and throughput. However, some may be immature and still need a large amount of practice to verify their effectiveness and feasibility.

Tab. 2 Benefits of BC-based crowdsourcing solution

Security requirements	Existing problem	BC-based solutions
Robustness	Centralized problems such as, single point of failure, DDoS, and Sybil attack	Distributed state replication mechanism, virtual coin, and reputation deposit mechanisms
Reliability	Data leakage, privacy disclosure	Digital signature and public key encryption techniques
Traceability	Data storage chaos, Data tampering	Merkel root and chained data structure
Fairness	Subjective arbitration	Majority approving mechanism and smart contract evaluation and decision
Scalability	Constrained access control	Flexible turing-complete programs in BC and off-chain storage
Cost-effect	High service fee of intermediary system	Micro verification fee of miners

In addition, due to the lack of the relevant regulations, the emergence of blockchain technology may cause legal inapplicability. In detail, smart contract is the set of rules in virtual world, when bound with the real-world parties, the legal enforceability is limited. Therefore, existing laws have loopholes and inconsistencies with the management of blockchain, and it is necessary to establish a sound legal system to control operations.

VI. Simulation and Analysis

In this paper, the simulation experiment scenario is set as crowdsourcing voting. The crowdsourcing process is written into the Ballot smart contract using Solidity programming language. Through compiling and debugging on Remix IDE (an online Ethereum smart contract browser editor), we have completed the deployment and interaction of the contract on the Ethernet test network Ropsten with the help of the browser plug-in Metamask.

A. Crowdsourcing Voting

Firstly, the creator of the contract chairperson puts forward the voting proposal and requirements, and the contract will grant reputation value and a certain amount of tokens to the newly registered users participating in the voting. Then, the registered users can vote on the proposal, and use the weight to show their support of each proposal. The higher the weight, the more people supported the proposal. Finally, the smart contract allocates corresponding rewards to the users who actively participate in the crowdsourcing voting, including increasing the reputation and assigning tokens. A proposal is considered valid only if both the number of votes cast and the number of supporters are not less than the starting criteria.

B. Contract Deployment

After the voting stage, we compile the Ballot contract on the Remix IDE. Then, we process the contract deployment. the deploy and run transactions interface is shown in Fig. 6. We select an Metamask account to initiate the contract whose address is "0x6f9db78ea07d99cd786721892f505a8d50dd582". Then, we deploy five 32 bytes proposal names for voters

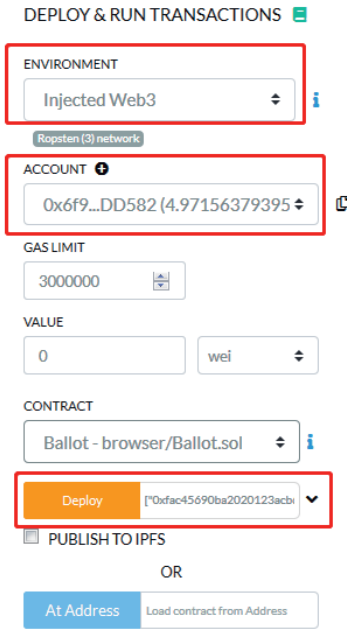


Fig. 6 Contract deployment interface

to vote. After successful deployment of proposals, the chairperson can give right to voters who satisfy the requirements defined on the smart contract.

C. Results Analysis

As shown in Fig. 7 and Fig. 8 the crowdsourcing voting smart contract has the following functions:

1) Authorizing voting address: Only users authorized by the contract creator have the voting right, otherwise their voting will not be counted.

2) Inter account transferring: Smart contract will transfer rewards for the account corresponding to the winning proposal.

3) Voting: Authorized users can vote for alternative proposals.

4) Accessing to contract execution information: Including contract creator, proposal status (name and number of votes), voter information (voting status, reputation value, weight, number of submitted proposals, and balance), and winning

Fig. 7 Contract control interface

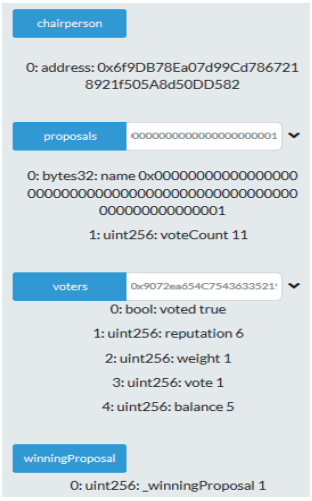


Fig. 8 Contract information acquisition interface

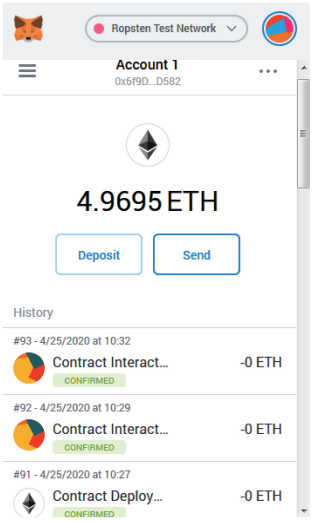


Fig. 9 Transaction history interface

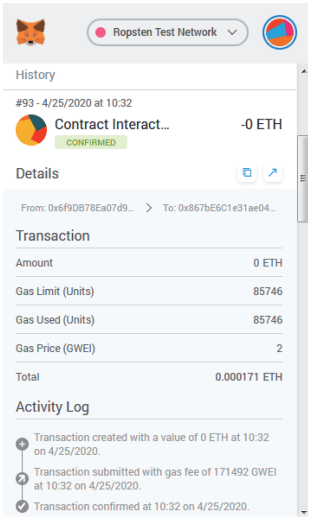


Fig. 10 Transaction detail interface

proposal.

According to the contract, the reputation value of users who can participate in voting should not be less than 6 and the balance should not be less than 5. The users that meet these requirements will be authorized to vote with a weight of 1, and only has one voting opportunity. Defining there are 20 users to vote, and the proposal with more than half of the votes will win. Fig. 7 shows the contract control function while Fig. 8 shows the voting result: the proposal No. 1 won 11 votes, becoming the winning proposal. Fig. 9 shows the transaction history of Account 1, reflecting that transaction information of any account can be queried. Fig. 10 presents the transaction details including transaction address, gas information, and activity log.

Through the actual deployment of smart contract instance, we confirm that the proposed blockchain-based architecture owning the following features. (i) Transaction transparency: Smart contract makes the whole voting process automatic and transparent, without third-party intervention, ensuring the fairness of the transaction. (ii) System security: The fully distributed architecture can prevent system crash caused by single point of failure. The reputation mechanism proposed in this paper can select suitable users and reduce the probability of Sybil and DDoS attacks. (iii) Privacy protection: Users use anonymous address instead of real identity information to transact, preventing personal privacy disclosure. (iv) Information traceability: All transaction information can be traced through hash value and account address, preventing malicious tampering of transaction information.

VII. Conclusion

In this paper, we have first briefly overviewed the security, privacy, and incentive challenges in the traditional crowdsourcing system. To address these challenges, we have then proposed a BC-based crowdsourcing system to build a distributed trusted system. The detailed procedures of smart contract for both the crowdsourcer-to-crowdsourcee mode and the crowdsourcee-to-crowdsourcer mode are introduced. By deploying the smart contract instance, the proposed blockchain-based architecture is proved to be feasible. Even though the BC-based crowdsourcing solution can lead to many benefits with respect to robustness, reliability, traceability, fairness, scalability, and cost, there still remain some challenges and unpredictable difficulties to be solved for practical applications with massive connections/nodes and/or time-sensitive nature. Therefore, we have also identified some potential topics including consensus mechanism and legal system for further study.

References

[1] MENEGHELLO F, CALORE M, ZUCCHETTO D, et al. IoT: Internet of threats? A survey of practical security vulnerabilities in real IoT devices[J]. IEEE Internet of Things Journal, 2019, 6(5): 8182-8201.

[2] FRUSTACI M, PACE P, ALOI G, et al. Evaluating critical security issues of the IoT world: present and future challenges[J]. IEEE Internet of Things Journal, 2017, 5(4): 2483-2495.

[3] YANG D, XUE G, FANG X, et al. Incentive mechanisms for crowdsensing: crowdsourcing with smartphones[J]. IEEE/ACM Transactions on Networking, 2015, 24(3): 1732-1744.

[4] AKSU H, BABUN L, CONTI M, et al. Advertising in the IoT era: vision and challenges[J]. IEEE Communications Magazine, 2018, 56(11): 138-144.

[5] IDC. Worldwide semiannual Internet of things spending guide[EB].

[6] TINATI R, MADAAN A, HALL W. The role of crowdsourcing in the emerging Internet-of-things[C]//Proceedings of the 26th International Conference on World Wide Web Companion. [S.l.:s.n.], 2017: 1669-1672.

[7] ZIEGLER S, CRETTAZ C, HAZAN M, et al. Combining Internet of things and crowdsourcing for pervasive research and end-user centric experimental infrastructures (IoT Lab)[J]. 2017.

[8] GISDAKIS S, GIANNETSOS T, PAPADIMITRATOS P. Security, privacy, and incentive provision for mobile crowd sensing systems[J]. IEEE Internet of Things Journal, 2016, 3(5): 839-853.

[9] LIU J, SHEN H, NARMAN H S, et al. A survey of mobile crowdsensing techniques: a critical component for the Internet of things[J]. ACM Transactions on Cyber-Physical Systems, 2018, 2(3): 1-26.

[10] WANG K, QI X, SHU L, et al. Toward trustworthy crowdsourcing in the social Internet of things[J]. IEEE Wireless Communications, 2016, 23(5): 30-36.

[11] POURYAZDAN M, FIANDRINO C, KANTARCI B, et al. Game-theoretic recruitment of sensing service providers for trustworthy cloud-centric Internet-of-things (IoT) applications[C]//2016 IEEE Globecom Workshops (GC Wkshps). Piscataway: IEEE Press, 2016: 1-6.

[12] KOUTSOPOULOS I. Optimal incentive-driven design of participatory sensing systems[C]//2013 Proceedings IEEE INFOCOM. Piscataway: IEEE Press, 2013: 1402-1410.

[13] GAO L, HOU F, HUANG J. Providing long-term participation incentive in participatory sensing[C]//2015 IEEE Conference on Computer Communications (INFOCOM). Piscataway: IEEE Press, 2015: 2803-2811.

[14] HAN K, ZHANG C, LUO J, et al. Truthful scheduling mechanisms for powering mobile crowdsensing[J]. IEEE Transactions on Computers, 2015, 65(1): 294-307.

[15] XU J, GUAN C, WU H, et al. Online incentive mechanism for mobile crowdsourcing based on two-tiered social crowdsourcing architecture[C]//2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON). Piscataway: IEEE Press, 2018: 1-9.

[16] LUO P, ZHU Y, PENG J, et al. A distributed auction approach to crowdsourced sensing over smartphones[C]//2016 IEEE 22nd International Conference on Parallel and Distributed Systems (ICPADS). Piscataway: IEEE Press, 2016: 1-7.

[17] LIANG B, LIU W, SUN L, et al. An aggregated model for energy management considering crowdsourcing behaviors of distributed energy resources[J]. IEEE Access, 2019, 7: 145757-145766.

[18] DING Y, CHEN Z, LIN F, et al. Blockchain-based credit and arbitration mechanisms in crowdsourcing[C]//2019 3rd International Symposium on Autonomous Systems (ISAS). Piscataway: IEEE Press, 2019: 490-495.

[19] XU X, LIU Q, ZHANG X, et al. A blockchain-powered crowdsourcing method with privacy preservation in mobile environment[J]. IEEE Transactions on Computational Social Systems, 2019, 6(6): 1407-1419.

[20] WU Y, TANG S, ZHAO B, et al. BPTM: blockchain-based privacy-preserving task matching in crowdsourcing[J]. IEEE Access, 2019, 7: 45605-45617.

[21] KOGIAS D G, LELIGOU H C, XEVGENIS M, et al. Toward a blockchain-enabled crowdsourcing platform[J]. IT Professional, 2019, 21(5): 18-25.

[22] DOUCEUR J R. The sybil attack[C]//International workshop on peer-to-peer systems. Berlin: Springer, 2002: 251-260.

[23] ZHANG K, LIANG X, LU R, et al. Sybil attacks and their defenses in the Internet of things[J]. IEEE Internet of Things Journal, 2014, 1(5): 372-383.

[24] RAJAN A, JITHISH J, SANKARAN S. Sybil attack in IOT: modelling and defenses[C]//2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI). Piscataway: IEEE Press, 2017: 2323-2327.

[25] LIANG X, LIN X, SHEN X S. Enabling trustworthy service evaluation in service-oriented mobile social networks[J]. IEEE Transactions on Parallel and Distributed Systems, 2013, 25(2): 310-320.

[26] KSHETRI N. Can blockchain strengthen the Internet of things?[J]. IT Professional, 2017, 19(4): 68-72.

[27] WEI Z, MASOUROS C, LIU F. Secure directional modulation with few-bit phase shifters: optimal and iterative-closed-form designs[J]. IEEE Transactions on Communications, 2020, 69(1): 486-500.

[28] BUTUN I, ÖSTERBERG P, SONG H. Security of the Internet of things: vulnerabilities, attacks, and countermeasures[J]. IEEE Communications Surveys and Tutorials, 2019, 22(1): 616-644.

[29] SENGUPTA J, RUJ S, BIT S D. A comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT[J]. Journal of Network and Computer Applications, 2020, 149: 102481.

[30] NIKOLOV L G. On the contemporary cybersecurity threats[J]. Security and Future, 2017, 1(3): 111-113.

[31] ROMAN R, LOPEZ J, MAMBO M. Mobile edge computing, fog et al.: a survey and analysis of security threats and challenges[J]. Future Generation Computer Systems, 2018, 78: 680-698.

[32] GISDAKIS S, MANOLOPOULOS V, TAO S, et al. Secure and privacy-preserving smartphone-based traffic information systems[J]. IEEE Transactions on Intelligent Transportation Systems, 2014, 16(3): 1428-1438.

[33] CHAUM D, HEYST E. Group signatures[C]//Workshop on the Theory and Application of of Cryptographic Techniques. Berlin: Springer, 1991: 257-265.

[34] SWEENEY L. K-anonymity: a model for protecting privacy[J]. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 2002, 10(5): 557-570.

[35] LI Y, CAO B, PENG M, et al. Direct acyclic graph-based ledger for Internet of things: performance and security analysis[J]. IEEE/ACM Transactions on Networking, 2020, 28(4): 1643-1656.

[36] CAO B, LI Y, ZHANG L, et al. When Internet of things meets blockchain: challenges in distributed consensus[J]. IEEE Network, 2019, 33(6): 133-139.

[37] BENTOV I, LEE C, MIZRAHI A, et al. Proof of activity: extending bitcoin's proof of work via proof of stake [extended abstract] y[J]. ACM SIGMETRICS Performance Evaluation Review, 2014, 42(3): 34-37.

[38] Slimcoin: a peer-to-peer crypto-currency with proof-of-burn. [EB].

[39] EYAL I, GENCER A E, SIRER E G, et al. Bitcoin-NG: a scalable blockchain protocol[C]//13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16). [S.l.:s.n.], 2016: 45-59.

[40] SZABO N. Smart contracts[EB].

[41] LI M, WENG J, YANG A, et al. CrowdBC: a blockchain-based decentralized framework for crowdsourcing[J]. IEEE Transactions on Parallel and Distributed Systems, 2018, 30(6): 1251-1266.

[42] ARCHER D, CHEN L, CHEON J H, et al. Applications of homomorphic encryption[C]//Crypto Standardization Workshop, Microsoft Research. [S.l.:s.n.], 2017: 14.

[43] CHEON J H, KIM J. A hybrid scheme of public-key encryption and somewhat homomorphic encryption[J]. IEEE Transactions on Information Forensics and Security, 2015, 10(5): 1052-1063.

[44] WANG G. On the security of a group signature scheme with forward security[C]//International Conference on Information Security and Cryptology. Berlin: Springer, 2003: 27-39.

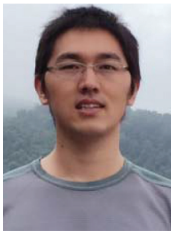
[45] NAGENDRA K S, APARNA. R. Sensitive attributes based privacy preserving in data mining using k-anonymity[J]. International Journal of Computer Applications, 2013, 84(13):1-6.

[46] MOHAN P, PADMANABHAN V N, RAMJEE R. Nericell: rich monitoring of road and traffic conditions using mobile smart-

phones[C]//Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems. New York: ACM, 2008: 323-336.

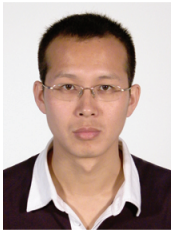
[47] RANA R K, CHOU C T, KANHERE S S, et al. Ear-phone: an end-to-end participatory urban noise mapping system[C]//Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks. Piscataway: IEEE Press, 2010: 105-116.

About the Authors



Daquan Feng received the Ph.D. degree in Information Engineering from the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu, China, in 2015. From 2011 to 2014, he was a visiting student with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA. After graduation, he was a Research Staff with the State Radio Monitoring Center, Beijing, China, and then a Postdoctoral Research Fellow with Singapore University of Technology and Design, Singapore. Since 2016, he has been with the College of Electronics and Information Engineering, Shenzhen University, as an Assistant Professor and then Associate Professor. His research interests include URLLC communications, MEC, and massive IoT networks. Dr. Feng is an Associate Editor of IEEE Communications Letters.

Long Zhang received the M.E. degree in Information and Communication Engineering from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2019. He is currently pursuing the Ph.D. degree with the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu, China. His research interests include next generation mobile networks and the Internet of things.



Shengli Zhang [corresponding author] received the B.Eng. degree in Electronic Engineering and the M.Eng. degree in Communication and Information Engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2002 and 2005, respectively. He received the Ph.D. in the Department of Information Engineering, the Chinese University of Hong Kong (CUHK), Hong Kong, China, in 2008. After that, he joined the Communication Engineering Department, Shenzhen University, Shenzhen, China, where he is a full Professor now. From 2014 to 2015, he was a Visiting Associate Professor at Stanford University. Shengli Zhang is the Pioneer of physical-layer network coding (PNC). He has published over 20 IEEE top journal papers and ACM top conference papers, including IEEE JSAC, IEEE TWC, IEEE TMC, IEEE TCom and ACM Mobicom. His research interests include blockchain, physical layer network coding, and wireless networks. He is a Senior Member of IEEE, severed as an Editor for IEEE TVT, IEEE WCL and IET Communications. He has also severed as TPC Member in several IEEE conferences.

Qihui Wu received the B.S. degree in Communications Engineering and the M.S. and the Ph.D. degrees in Communications and Information Systems from the Institute of Communications Engineering, Nanjing, China, in 1994, 1997, and 2000, respectively. After graduation, he worked at the PLA University of Science and Technology, Nanjing, China. Since 2016, he has been with Nanjing University of Aeronautics and Astronautics (NUAA) and appointed as Changjiang Distinguished Professorship. Currently, he is the Vice-Principal of NUAA. His academic contributions of cognitive radio have been demonstrated in over 200 publications with more than 4 000 citations, where over ten articles are honored as the ESI highly cited article. Furthermore, he has been invited to present keynotes and has been awarded a number of distinctions, such as the IEEE Signal Processing Society's 2015 Young Author Best Paper Award, the 14 th IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award, and the First Prize of the Natural Science Award by China Institute of Electronics/Communications in 2020 and 2017, respectively. Moreover, the broad impacts of his research have been widely publicized with more than 30 invited talks in various international conferences. He has acted as a TPC and a General Chair of the international conferences, such as IEEE VTC, IEEE ICC, IEEE WCSP, and took part in a wide range of IEEE activities. He is also directing the Key Laboratory of Ministry of Industry and Information Technology of China, working on more than ten major research projects in the field of cognitive radio network, intelligent space control of electromagnetic spectrum and massive UAV cluster, sponsored by, such as NFSC. His innovations have been authorized by more than 20 national and international patents and applied in, such as Beidou Satellite and lunar exploration programs. He is an IET fellow.

Xianggen Xia received the B.S. degree in Mathematics from Nanjing Normal University, Nanjing, China, and the M.S. degree in Mathematics from Nankai University, Tianjin, China, and the Ph.D. degree in Electrical Engineering from the University of Southern California, Los Angeles, USA, in 1983, 1986, and 1992, respectively.

He was a Senior/Research Staff Member at Hughes Research Laboratories, Malibu, California, USA, during 1995-1996. In September 1996, he joined the Department of Electrical and Computer Engineering, University of Delaware, Delaware, USA, where he is the Charles Black Evans Professor. His current research interests include space-time coding, MIMO and OFDM systems, digital signal processing, and SAR and ISAR imaging. Dr. Xia is the author of the book titled Modulated Coding for Intersymbol Interference Channels (New York, Marcel Dekker, 2000).

Dr. Xia received the National Science Foundation (NSF) Faculty Early Career Development (CAREER) Program Award in 1997, the Office of Naval Research (ONR) Young Investigator Award in 1998, and the Outstanding Overseas Young Investigator Award from the National Nature Science Foundation of China in 2001. He received the 2019 Information Theory Outstanding Overseas Chinese Scientist Award, the Information Theory Society of Chinese Institute of Electronics. Dr. Xia has served as an Associate Editor for numerous international journals including IEEE Transactions on Signal Processing, IEEE Transactions on Wireless Communications, IEEE Transactions on Mobile Computing, and IEEE Transactions on Vehicular Technology. Dr. Xia is Technical Program Chair of the Signal Processing Symp., Globecom 2007 in Washington D.C. and the General Co-Chair of ICASSP 2005 in Philadelphia.