BSIF: Blockchain-Based Secure, Interactive, and Fair Mobile Crowdsensing

Weizheng Wang, *Graduate Student Member, IEEE*, Yaoqi Yang, Zhimeng Yin, Kapal Dev, *Senior Member, IEEE*, Xiaokang Zhou, *Member, IEEE*, Xingwang Li, *Senior Member, IEEE*, Nawab Muhammad Faseeh Qureshi, *Senior Member, IEEE*, and Chunhua Su

***Abstract*—Given the explosive growth of portable devices, mobile crowdsensing (MCS) is becoming an essential approach that fully utilizes pervasive idle resources to accomplish sensing tasks. The traditional MCS relies on the centralized server for task handle is susceptible to a single point of failure. Targeting this security issue, researchers have proposed a series of blockchain-based MCS. However, nodes in the blockchain suffer from high computation cost for data processing. Simultaneously, most blockchain-based MCS systems lack an efficient incentive mechanism for service requesters and workers. In this work, we integrate the smart contract and mobile devices to establish a secure, interactive, and fair blockchain-based MCS system called BSIF. To prevent illegitimate participants, BSIF requests all users to verify their identities using private keys from the registration phase. In the case of worker location privacy leakage, the location-based symmetric key generator is adopted to coordinate a session key for target range worker selection. Besides, we transfer the data evaluation process to the requester side (e.g., a personal computer), reducing computation cost in the blockchain nodes. Due to the homomorphic feature of the Paillier Cryptosystem and common interest, the requester cannot violate the directives from the blockchain. Subsequently, the Stackelberg game is adopted to investigate the participation level of the workers and the fair reward mechanism for the requesters to achieve a dynamic balance. Finally, the security analysis and performance evaluation demonstrate that our BSIF can defend against possible adversaries while significantly cutting overhead and giving participants the utmost incentive.**

***Index Terms*—Mobile crowdsensing (MCS), security and privacy, blockchain, Stackelberg game.**

I. Introduction

Mobile crowdsensing (MCS) proposed by Ganti et al. [1] in 2011, leverages a large number of individual mobile devices (e.g., smartphones, wearables , and tablet computers) owned by individuals to capture and share surrounding data based on sensing task content. As the statistic data provided by the International Telecommunications Union (ITU),[1] the total number of mobile cellular subscriptions has reached 7.325 billion at the end of 2020. Moreover, the computational performance of mobile devices has obtained remarkable improvement benefiting from the emergence of AI-based chips [2]. Given these two trends, MCS plays an essential role in mobile resource coordination and utilization to outsource sensing tasks for cost-effective impact. In the example of the real-time traffic monitoring system, vehicles can upload their current GPS locations and running speeds to the cloud server through nearby roadside units (RSUs), thus generating a traffic condition map for congestion alleviation [3]. Besides, a model of city area can also be efficiently reconstructed in terms of photos collected by the tourists [4].

Although MCS brings great convenience to all walks of life, some security and privacy issues also rise to the surface. First and foremost, as shown in Fig. 1, the centralized server is responsible for task publishing and collected data processing between task requesters and mobile workers in the traditional MCS scenario. However, once the adversaries compromise the central server, all the task contents and sensory data will be leaked without suspense. Especially in the healthcare scenario, this potential risk discourages the positive attitude of all participants since their private health status can be disclosed, such as the health status and blood pressure recorded in the iWatch [5]. Therefore, to tackle the above-centralized challenge, blockchain-based MCS systems have emerged in recent years. Decentralized network, transparency transaction,

[1] https://data.worldbank.org/indicator/IT.CEL.SETS?end=2020&start=2020

Manuscript received 16 March 2022; revised 15 June 2022; accepted 30 June 2022. Date of publication 17 October 2022; date of current version 22 November 2022. This work was supported in part by NSFC under Project 62102332 and Project CityU APRC 9610491 and in part by the Project of Zhejiang Lab under Project 119001-AC2202. *(Corresponding authors: Yaoqi Yang; Zhimeng Yin; Xiaokang Zhou; Nawab Muhammad Faseeh Qureshi.)*

Weizheng Wang is with the Department of Computer Science, City University of Hong Kong, Hong Kong SAR, China, and also with the Zhejiang Lab, Hangzhou, Zhejiang 310000, China (e-mail: weizheng.wang@ieee.org).

Yaoqi Yang is with the Graduate School, Army Engineering University of PLA, Nanjing 210000, China (e-mail: yaoqi_yang@yeah.net).

Zhimeng Yin is with the Department of Computer Science, City University of Hong Kong, Hong Kong SAR, China (e-mail: zhimeyin@cityu.edu.hk).

Kapal Dev is with the Department of Computer Science, Munster Technological University, Cork, T12 P928 Ireland, and also with the Institute of Intelligent Systems, University of Johannesburg, Johannesburg 2006, South Africa (e-mail: kapal.dev@ieee.org).

Xiaokang Zhou is with the Faculty of Data Science, Shiga University, Hikone 522-8522, Japan, and also with the RIKEN Center for Advanced Intelligence Project, RIKEN, Tokyo 103-0027, Japan (e-mail: zhou@biwako.shiga-u.ac.jp).

Xingwang Li is with the Department of Physics and Electronic Information Engineering, Henan Polytechnic University, Jiaozuo 454000, China (e-mail: lixingwang@hpu.edu.cn).

Nawab Muhammad Faseeh Qureshi is with the Department of Computer Education, Sungkyunkwan University, Seoul 16419, South Korea (e-mail: faseeh@skku.edu).

Chunhua Su is with the Division of Computer Science, The University of Aizu, Fukushima 965-8580, Japan (e-mail: chsu@u-aizu.ac.jp).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSAC.2022.3213306.

Digital Object Identifier 10.1109/JSAC.2022.3213306

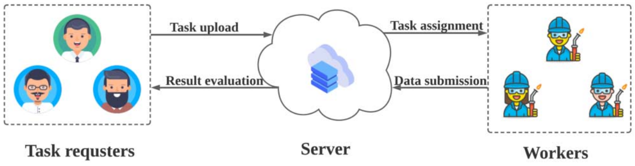


Fig. 1. Traditional MCS framework.

and unalterable record are the three prominent features of blockchain, which establishes trust for all parts, following with the single point of failure defense. Particularly, the smart contract attached to the blockchain can be used to automatically form a confident agreement between requesters and workers with the aid of lines of code.

Despite some MCS frameworks built on blockchain have been implemented [6], [7], [8], the data evaluation process is usually handled by the access nodes. The low throughput and slow transaction processing period of most blockchains hinder this preference. For example, Bitcoin takes 10 minutes to form a block and Ethereum takes about 14 seconds to form a block. During peak times, it may take longer for these blocks to form. Hence, as long as the task requesters are willing to undertake some verification operation, the computation pressure can be lightened in the blockchain.

Other than the centralized server issue in the MCS, communication data amid nodes should also be protected appropriately, owning to the insecure channel. For instance, if nodes submit the unmodified coordinates to an untrusted service provider, the node location will be tracked, leading to a potential leakage of more sensitive information such as name, address and password. As usual, the privacy-preserving MCS is divided into two categories: tasking and reporting process. Regarding tasking process, pseudonymity and obfuscation are the two common methods to conceal the real identity of participants by adopting pseudo ID or adding some data scrambling for the current location. For the reporting process, the privacy of transmitted data needs to be guarded by using cryptographic techniques, selective sensing, and patch cloaking algorithms [9]. Nevertheless, the communication and computation costs are the two crucial indicators when designing the privacy-preserving MCS.

Fairness is another challenge that needs to be well addressed. Specifically, existing systems aim to achieve a dynamic equilibrium between offered rewards given by the task requesters and participation level of mobile workers, suggesting that service providers acquire accurate sensing data at a minimum cost and workers receive the high return by contributing the least efforts. Although a few works [10], [11] have studied the incentive mechanism and privacy protection for MCS separately, the joint game theory-based incentive design and security enhancement of MCS is still absent.

In this paper, in order to tackle the above-mentioned posers, we propose a blockchain-based secure, interactive, and fair MCS called BSIF. The traditional centralized server has been replaced by the blockchain, which can decrease unnecessary risks such as single point of failure and server compromised attack in the MCS. On the other hand, we also consider the costly computation resource of blockchain node, so a portion of data evaluation process has been migrated into task requesters benefiting from the homomorphic encryption of Paillier Cryptosystem. Simultaneously, without the location report requirements, a novel location-based symmetric key generator is introduced to search workers in a target range so that the location exposure issue can be eliminated. Furthermore, to filter unqualified workers, task requesters are also able to set the reputation threshold for the published task. Finally, based on the incentive mechanism in the Stackelberg game, we establish the dynamic reward and participation level determination models, which aim at forming an optimal incentive strategy. The contributions of this paper are listed as follows:

1) To alleviate centralized issues caused by the traditional MCS server, we integrate smart contracts and sensor nodes to construct a blockchain-based MCS framework that strengthens the storage security of sensory data. Moreover, task requesters undertake the data evaluation process in order to shorten the entire period.
2) To achieve the privacy-preserving task allocation and reaction, we utilize a location-based symmetric key generator and task reputation threshold setting to encrypt task content and select qualified workers in the target area, respectively.
3) We model the interaction process between the service provider (SP) and mobile workers (MWs) as a two-stage Stackelberg game and analyze each stage systematically through backward induction. And we also investigate two types of incentive mechanisms for the crowdsensing platform, i.e., the Stackelberg game-based discriminatory incentive mechanism and the Stackelberg game-based uniform incentive mechanism, respectively.
4) To verify the reliability and effectiveness of our proposed BSIF, we conduct the elaborate security analysis and experiment evaluation. The outcome presents our BSIF can meet both requirements.

The remainder of this article is organized as follows. In Section II, we discuss related work about blockchain, privacy-preservation and fairness-oriented MCS, respectively. In Section III, we introduce our proposed BSIF overview, threat model and design goal. The details of secure and interactive task allocation for BSIF are illustrated in Section IV. In Section V, we design a Stackelberg game-based incentive scheme for our BSIF. The correctness and security for our proposed protocol is analyzed in Section VI. In Section VII, we evaluate the performance of our proposed BSIF in terms of computational/communication overhead and on-chain performance. The fairness of our proposed protocol is discussed in Section VIII. Finally, we conclude this article in Section IX.

II. Related Work

A. Blockchain-Oriented MCS and Federated Learning (FL)

Decentralization, traceability, and immutability are the three most attractive characteristics, which can tackle unreliable central server issues in the traditional MCS. Moreover, the blockchain-based MCS is able to take the identical responsibilities as centralized ones do. For example, Huang et al. [12] proposed a blockchain-based MCS system called BMCS, which drives miners to confirm the validity of collected data and stimulate workers' participation depending on a flexible reward ranking incentive scheme. To protect privacy of task content and participants, Hu et al. [13] built the blockchain on the basis of base station miner, which can efficiently verify the identities of task requesters and mobile nodes. Kadadha et al. [14] developed an integral blockchain-based MCS platform, which includes three smart contracts (i.e., task deployment contract, task manager contract and task detailed contract) for user information maintenance, requester task publish and sensory data upload. Cai et al. [15] utilized zero-knowledge range proofs (ZKRPs) to design a privacy-preserving MCS on blockchain, where nodes can rent out their computation power to conduct MCS tasks. Based on the blockchain and trusted hardware (i.e., Intel SGX), Duan et al. [16] established a decentralized and robust MCS, which can avoid the single point of failure and non-transparent operation. Focusing on fair trade, Liang et al. [17] utilized the Intel SGX and blockchain to prevent the requesters from issuing the spiteful MCS tasks, which aims to tease mobile workers without payment. Although the blockchain-based crowdsensing which applies Intel SGX [16], [17] can achieve less centralized compared with works using smart contracts, the inaccurate threat model and attack-proof cannot prove other potential attacks (e.g., unauthorized communication, unreliable payment and worker location privacy) also can be prevented. The recently emerging distributed machine learning framework–FL has similar mechanisms to MCS, which outsources data processing tasks to workers without data privacy leaks. However, the FL also meets the centralized and single-point-of-failure issues for the data aggregation at the server side, so the blockchain-based FL seems to be a good solution that can provide decentralized and robust data collection for FL training with the aid of immutable block ledgers. For example, Rehman [18] proposed a blockchain-based reputation-aware FL, which can obtain a personalized and fine-grained training model for task requesters and some malicious nodes can be excluded by trustworthy collaboration between edges. Toyoda et al. [19] proposed an incentive mechanism for the FL nodes on a public blockchain network, which can motivate more workers to participate in the training process and audit all the behaviors. Besides, a decentralized AI-market that combines blockchain with FL called SingularityNET[2] also becomes popular. In SingularityNET, users can utilize digital currency to buy AI services, and some developers are available to sell the corresponding services to users. Although MCS and FL share similar mechanisms, MCS pays more attention to outsourcing data collection tasks to participants and the main focus for FL is to outsource data processing tasks.

B. Privacy-Oriented MCS

Privacy preservation is also a dispensable factor that cannot be overlooked in the MCS. For instance, if the location privacy of worker cannot be ensured, the mobile nodes may become unwilling to share sensory data. To prevent the occurrence of this situation, Pournajaf et al. [20] utilized a cloaking strategy to mix up the real locations of participants while keeping the efficiency of management. K-anonymity is also widely-used to conceal worker location. Based on this method, Wang et al. [21] proposed an incentive mechanism with k-anonymity location-privacy preserving for MCS, where not only the location can be protected but also the activeness of worker can be stimulated. To alleviate the data processing load for the participants, Zhang et al. [22] leveraged the proxy re-encryption to guarantee sensitivity of sensing task content and delegate the decryption process to the fog-computing servers. To achieve a balance between data quality assurance and privacy preservation, Zhao et al. [10] combined a novel reward distribution mechanism with zero-knowledge to construct a novel MCS model, which can meet expected data requirement and does not disclose any data. Given the accurate and efficient task allocation, Jiang et al. [23] proposed a reduced-dimensionality enabled Q-learning for MCS task interaction meanwhile the communication privacy is assured by the location-based symmetric key generator. Despite numerous privacy-preserving incentive mechanisms for MCS have been proposed, few works are concentralized on issues of decentralized server, secure data transmission and optimal incentive solution concurrently.

C. Fairness-Oriented MCS

In order to improve the performance of the MCS, fairness is one of the premises to be ensured. Up to now, some attention has been paid to the fairness-oriented MCS system. Nie et al. [24] proposed the socially-aware incentive mechanism to recruit and pay for the MWs, where the multiple SRs are modeled as the leaders in the established Stackelberg game. As for the scenarios under vehicle networks-based MCS, Xiao et al. [25] formulated the participation and reward-based MCS game, where the server chooses the vehicle reward strategy based on the sensing accuracy, and the vehicle takes actions according to the transmission costs and expected costs. Tembine et al. [26] investigated the participation level of the users under three typical kinds of network, public good, information sharing, and resource sharing, where Bayesian game is used to derive the optimal strategy for the user. In [27], Gao et al. aimed to jointly protect the sensing data privacy and enhance the data quality, in which the trade-off problem between the data utility and privacy-preservation is modeled with game theory, and derived the Nash equilibrium solutions by the dynamic programming methods. To solve the problems of users' selfishness, limited resources, and data authenticity, Cao et al. [28] designed different game models to derive the optimal strategies for users and servers in the MCS system, where multi-stage stochastic programming-based approaches are adopted. Under the unstable social network scenario, Nie et al. [29] designed a two-stage Stackelberg game (i.e., Bayesian Stackelberg game) to explore the activeness of the mobile users for the optimal incentive mechanism design. In general, current efforts in fairness-oriented MCS focus less

[2] https://singularitynet.io/

TABLE I
A SUMMARY OF MCS SECURITY, INTERACTION AND FAIRNESS ISSUE

Reference	Security consideration	Security countermeasure	Threat model	Attack-proof	Interaction consideration	Fairness consideration	Incentive mechanism
Huang et al. [12]	Yes	Public blockchain and smart contract	Medium	Feasibility discussion	No	Yes	Dynamic reward ranking payment
Hu et al. [13]	Yes	Customized blockchain and smart contract	Not given	No	No	Yes	Three-stage Stackelberg game payment
Kadadha et al. [14]	Yes	Public blockchain and smart contract	Not given	Feasibility discussion	No	Yes	Data quality-based payment
Cai et al. [15]	Yes	Zero-knowledge range proofs, blockchain and smart contract	Strong	Feasibility discussion and theoretical analysis	No	Yes	Average payment
Duan et al. [16]	Yes	Public blockchain, smart contract TEE, DP and Paillier Cryptosystem	Medium	Feasibility discussion	No	Yes	Data quality-based payment
Liang et al. [17]	Yes	Public blockchain, smart contract, TEE	Medium	Feasibility discussion	No	Yes	Data quality-based payment
Pournajaf et al. [20]	Yes	Cloaking strategy	Not given	No	No	Yes	Two-stage optimization payment
Wang et al. [21]	Yes	K-anonymity location-privacy preservation	Not given	No	No	Yes	Reverse auction winner selection payment
Zhang et al. [22]	Yes	Proxy re-encryption algorithm	Strong	Feasibility discussion and theoretical analysis	No	Yes	Data quality-based payment
Zhao et al. [10]	Yes	Zero-knowledge proof	Strong	Feasibility discussion and theoretical analysis	No	Yes	Data quality-based payment
Jiang et al. [23]	Yes	Proxy re-encryption algorithm and location-based symmetric key generator	Strong	Feasibility discussion and theoretical analysis	No	Yes	Reduced-dimensionality enabled Q-learning payment
Nie et al. [24]	No	No	Not given	No	No	Yes	Multi-leader and multi-follower Stackelberg -based payment
Xiao et al. [25]	No	No	Not given	No	No	Yes	Q-learning-based payment
Tembine et al. [26]	No	No	Not given	No	No	Yes	Bayesian game-based payment
Gao et al. [27]	No	No	Not given	No	No	Yes	Dynamic programmin -based payment
Cao et al. [28]	No	No	Not given	No	No	Yes	Mutil-stage stochastic programming -based game payment
Nie et al. [29]	No	No	Not given	No	No	Yes	Bayesian Stackelberg game-based payment
BSIF	Yes	Yes	Strong	Feasibility discussion and theoretical analysis	Yes	Yes	Bayesian Stackelberg game-based payment

Medium: threat model only describes adversary abilities; **Strong**: threat model describes both adversary abilities and security goals.

on jointly enhancing user security, optimizing the user reward, and ensuring the SR utility simultaneously.

Table. I offers a comprehensive comparison between existing literature and our BSIF. Although many security-oriented MCS techniques adopt blockchain, trusted execution environment (TEE), different privacy, or some techniques for constructing a secure framework, they commonly lack a comprehensive threat model and security proof. Regarding the interaction, existing approaches' data is evaluated at blockchain nodes with limited computation resources. Regarding fairness, although existing literature considers the incentives, they largely overlook essential security aspects. To mitigate the above issues, our proposed BSIF aims to construct a secure, interactive, fair MCS based on blockchain.

III. System Formulation

In this section, we give brief introduction of our proposed BSIF system, and then the threat model and design goal are illustrated.

A. System Overview

Our BSIF MCS system is identical to the traditional MCS system, which owns four roles: mobile workers (MWs), service provider (SP), mining nodes (MNs) and service requester (SR). The comprehensive description of four parts is presented as follows.

1) *MWs*: Smartphones, wearables and vehicles equipped with GPS, camera, IMU and other functional sensors can serve as the MWs, which participate in the sensing task and collect the data from their surroundings.
2) *SP*: SP in our proposed scheme is not only a sole server, which consists of a large number of MNs in the blockchain. In the blockchain-based SP, task publish and allocation rules are programmed into a smart contract, whose interfaces are publicized for easy invocation.
3) *MNs*: The MNs maintain smooth execution of blockchain system, where new transactions are validated and recorded in the blockchain. Besides, MNs not only can negotiate the interaction with all participants but also can contribute to some storage and computation resource. As a return, MNs have the chance to gain transaction fees and rewards.
4) *SR*: Any organization, group, company and individual can serve as the SR, which outsources some data-intensive or location-specific tasks to MWs for information collection and analysis. Moreover, in our proposed BSIF, SR has an obligation to undertake encrypted data evaluation to alleviate the pressure from the blockchain.

The architecture of our BSFI is presented in Fig. 3, which consists of four parts: MWs, SP, MNs and SR. From this figure, we can also see eleven steps in our crowdsensing system. The detailed process of BSIF is presented as follows:

1) **Requester registration:** SR calls *SRRegister* function defined in the smart contract, aka SP, to register its identity information in order to acquire public/private keys for further authenticated communication through a MN.
2) **Worker registration:** The MWs call the *MWRegister* function in the SP for registration.

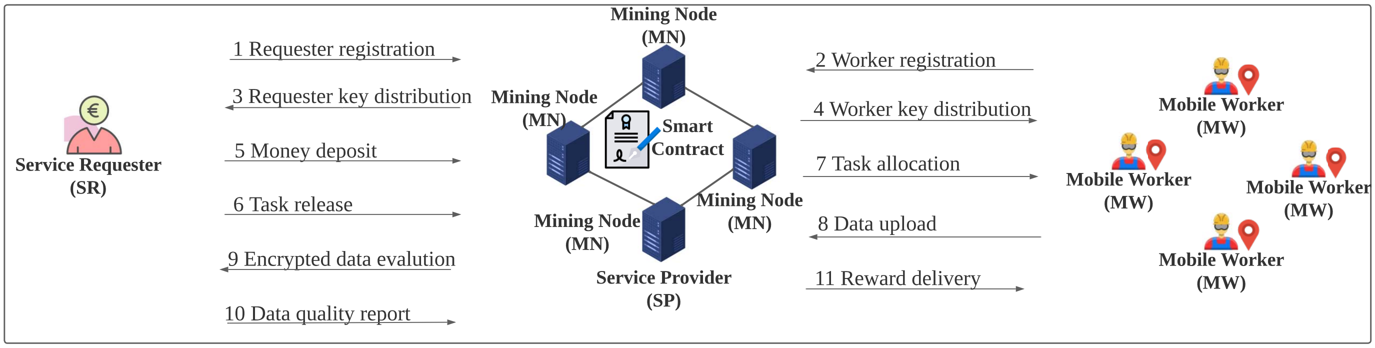


Fig. 2. System framework of BSIF.

3) **Requester key distribution:** According to security parameters provided in the SP, the public/private keys for SR/NWs are also generated in the registration process. Note that since the relevant public/private keys can be referred in the SP directly, a blind factor uploaded by SR is used to perturb the original private keys.
4) **Worker key distribution:** The key distribution for worker is identical to requester key distribution process.
5) **Money deposit:** SR needs to transfer the money to the SP as deposit.
6) **Task release:** SR is permitted to release the task to the SP once deposit is received by the SP.
7) **Task allocation:** MWs which are in the target range can decode the task published by the SP.
8) **Data upload:** MWs collect required data and submit them to the SP.
9) **Encrypted data evaluation:** SP outsources the encrypted sensing data to the SR for data quality evaluation.
10) **Data quality report:** SR assesses the data quality and reports result to the SP.
11) **Reward delivery:** SP distributes the reward to the MWs and then calculates the latest reputation.

B. Threat Model

In this subsection, we consider the possible threats for our BSIF in terms of mobile workers (MWs), service provider (SP), mining nodes (MNs) and service requester (SR). The elaborate discussion is stated as follows.

1) MWs are assumed untrusted since MWs may not follow the directives of the task to upload some contributed-less sensing data or submit erroneous information directly, which intends to deviate real data estimation. Moreover, MWs out of the target range or whose reputation are below the threshold may participate in sensing tasks due to monetary reward temptation. Besides, MWs may change their identities to report sensory data many times.
2) Although SP is considered fully trusted, MWs and SR are curious about the security parameters defined in the contract for key generation. Moreover, the sensitive data in the communication channel faces many unknown attackers who may eavesdrop and tamper with them.
3) MNs is considered honest-but-curious. MNs cannot collude with the MWs and SP, since MNs serve as the intermediary between them. However, MNs are curious about the content in the communication process. Moreover, MNs have the chances to be compromised.
4) SR is semi-honest. On the one hand, SR does not have to release false tasks since this action requires non-trivial cost. On the other hand, SR is likely to get sensing data but declines to pay the money for sensing data.

C. Design Goal

In this subsection, in terms of the above-mentioned potential threats, we define the design goal for our proposed BSIF, whose details are presented as follows:

1) *Authenticated Communication:* The identity for mobile workers (MWs) and service requester (SR) should be verified for each communication process. In other words, unregistered users cannot participate in our MCS system.
2) *Encrypted Task Content:* Regarding task privacy, the task is encrypted by location-based symmetric key, only the SR and MWs in the target range are able to decode task content.
3) *Privacy-Preserving Location:* The real location of mining nodes (MNs) cannot be leaked to anyone.
4) *Reliable Payment:* The SR cannot refuse to pay the reward to the service provider (SP) once the sensing task is finished.
5) *Accurate Task Allocation:* The task is allocated to the MWs whose reputation are over the pre-defined threshold and locate in the target area.
6) *Efficient and Secure Data Evaluation:* The data quality evaluation period should be speeded up with low cost and the sensing data cannot be leaked until sensing task is completed.
7) *Reasonable Reward Distribution:* Each MW is able to get a rational reward which depends on the sensing data quality and participation level.
8) *Real-Time Reputation Update:* Since the eligibility of MWs for sensing task is determined by their reputation, this value should be updated at the first time.
9) *Fair Reward and Incentive Mechanism:* SR and MWs need to achieve a dynamic equilibrium between offered rewards and participation level.

TABLE II
NOTATIONS

Symbol	Description
p, q	Two large primes
sk, pk	The secret key and public key
$Z^*_{n^2}$	The multiplicative group
g	The generator for $Z^*_{n^2}$
r	The random number
m, c	The message and encrypted message
$LSKenGen(\cdot)$	The location-based key generation
LSK	The location-based secret key
σ	The blind factor
$Addr$	The address
Enc/Dec_{AES}	The encryption/decrytion of AES
i, j	The precision of LSK
RT	The reputation threshold
M	The prospective reward
d	The sensing data
D	The sensing data set
q_d	The quality of sensing data
M	The reward

IV. SECURE AND INTERACTIVE TASK ALLOCATION

To achieve the above-mentioned design goals, we propose a blockchain-based secure, interactive and fair MCS called BSIF. In this section, we mainly implement the first goal, which protects the security and privacy of sensing task release and allocation. The notation used in the section is illustrated in Table II. The details for key generation, encryption and decryption for Paillier Cryptosystem can be referred to Appendix A.

A. Location-Based Key Generation

In our proposed BSIF, service requester (SR) and mobile workers (MWs) cannot negotiate a secret key with each other through direct communication. Moreover, the location of MWs should be preserved. Hence, we propose a location-based symmetric key generator based on previous work in [30]. LABE proposed in [30] requires the service provider (SP) to design the access policy including location attributes and privacy of mobile nodes in advance. Once data privacy are leaked to the SP, the task content and node security cannot be guaranteed. The other technique Geohash proposed in [30] can generate a secret key according to the a specified coordinate. However, Geohash cannot generate a uniform value for multiple nodes in a range area. Compared with LABE and Geohash proposed in [30], our protocol do not need the MWs to disclose their location attributes or privacy to the SP for secret key generation. Moreover, using a simple binary search, our proposed location key-based generation can generate a secret key for multiple MWs in a range area. The comprehensive location conversion details for longitude and latitude can be referred to Algorithm 1 and Appendix B, respectively.

Algorithm 1 Location-Based Key Generation for BSIF

```
Input: $long_{max} = 180, long_{min} = -180, lati_{max} = 90, lati_{min} = -90, long_{MW}, lati_{MW}, i, j$.
Output: $sk$.
$long_{med} \leftarrow \frac{(long_{min} + long_{max})}{2}$.
$lati_{med} \leftarrow \frac{(lati_{min} + lati_{max})}{2}$.
for $k = 1$ to $i$ do
    if $long_{MW} \leq long_{med}$ then
        $long_{min} \leftarrow long_{med}$;
        $x \leftarrow x||1$.
    else
        $long_{max} \leftarrow long_{med}$;
        $x \leftarrow x||0$.
    end if
    $long_{med} \leftarrow \frac{(long_{min} + long_{max})}{2}$.
end for
for $k = 1$ to $j$ do
    if $lati_{MW} \leq lati_{med}$ then
        $lati_{min} \leftarrow lati_{med}$;
        $y \leftarrow y||1$.
    else
        $lati_{max} \leftarrow lati_{med}$;
        $y \leftarrow y||0$.
    end if
    $lati_{med} \leftarrow \frac{(lati_{min} + lati_{max})}{2}$.
end for
return $sk = H(x||y)$.
```

B. Detailed Procedures

In this subsection, we present the detailed procedures of the secure and interactive task allocation for our BSIF. In our BSIF, firstly service requester (SR) and mobile workers (MWs) should register in the blockchain network. After successful registration, they will get corresponding pairs of private and public key issued by the service provider (SP). Subsequently, SR releases the task once the money is transferred to the smart contract in the SP. Since the task content is encrypted, only the MWs in the target area and with the expected reputation can conduct the sensing task. MWs collect the surrounding environmental data and submit them to the SP, which then outsources encrypted sensing data to the SR for data quality evaluation. When SP receives the data quality report from SR, the reward will be distributed to the MWs depending on their contribution. The details of BSIF are presented as follows:

1) The SR initiates a transaction with its address $Addr_{SR}$ and a blind factor σ_{SR} to trigger the $SRRegister$ function defined in the smart contract through SP. Subsequently, the *KeyGen* function of Paillier cryptosystem is invoked to generate $pk_{SR} = (n_{SR}, g_{SR})$ and $sk^*_{SR} = (\lambda_{SR}, \mu_{SR}) \oplus \sigma_{SR}$ for SR.
2) The registration process of MW is similar to SR, where the $MWRegister$ function is called with MW's address $Addr_{MW}$ and its blind factor σ_{MW}. Then the $pk_{MW} = (n_{MW}, g_{MW})$ and the perturbed secret key $sk^*_{MW} = (\lambda_{MW}, \mu_{MW}) \oplus \sigma_{MW}$ are provided for MW.

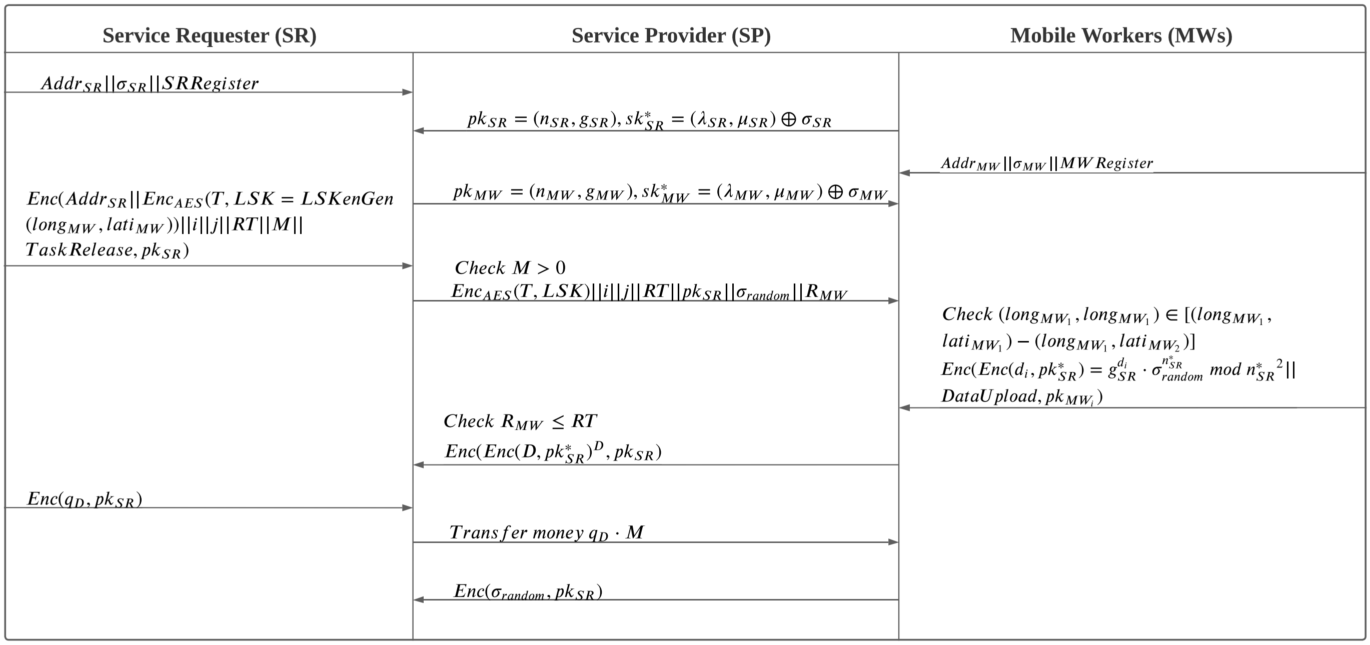

Service Requester (SR)
Service Provider (SP)
Mobile Workers (MWs)
$Addr_{SR}||\sigma_{SR}||SRRegister$
$pk_{SR}=(n_{SR},g_{SR}), sk^*_{SR}=(\lambda_{SR},\mu_{SR})\oplus\sigma_{SR}$
$Addr_{MW}||\sigma_{MW}||MWRegister$
$pk_{MW}=(n_{MW},g_{MW}), sk^*_{MW}=(\lambda_{MW},\mu_{MW})\oplus\sigma_{MW}$
$Enc(Addr_{SR}||Enc_{AES}(T, LSK=LSKenGen(long_{MW},lati_{MW}))||i||j||RT||M||TaskRelease, pk_{SR})$
$Check\ M>0$
$Enc_{AES}(T,LSK)||i||j||RT||pk_{SR}||\sigma_{random}||R_{MW}$
$Check\ (long_{MW_1},long_{MW_1})\in[(long_{MW_1},lati_{MW_1})-(long_{MW_1},lati_{MW_2})]$
$Enc(Enc(d_i,pk^*_{SR})=g^{d_i}_{SR}\cdot\sigma^{n^*_{SR}}_{random}\ mod\ n^*_{SR}{}^2||DataUpload,pk_{MW_i})$
$Check\ R_{MW}\le RT$
$Enc(Enc(D,pk^*_{SR})^D,pk_{SR})$
$Enc(q_D,pk_{SR})$
$Transfer\ money\ q_D\cdot M$
$Enc(\sigma_{random},pk_{SR})$

Fig. 3. Proposed protocol for secure and interactive task allocation in BSIF.

3) Since the pk_{SR} and sk^*_{SR} are accessible in the SP, SR can obtain corresponding pk_{SR} and sk^*_{SR} from $SRRegister$ function in the smart contract. Then using the known σ_{SR}, SR can restore the $sk_{SR}=sk^*_{SR}\oplus\sigma_{SR}$ for further use.
4) MWs can execute the similar process to acquire pk_{MW} and sk^*_{MW} by calling $MWRegister$ function in the smart contract.
5) The SR determines the sensing task content T and selects a random location $(long_{MW}, lati_{MW})$ in the target range. Then the SR encrypts the T as $Enc_{AES}(T, LSK)$ with the symmetric key LSK generated by $LSKenGen(long_{MW}, lati_{MW})$, where the full name for $Enc_{AES}(T, LSK)$ is the advanced encryption standard encryption algorithm.
6) SR launches a transaction which owns $Addr_{SR}$, encrypted task content $Enc_{AES}(T, LSK)$, precision i, j, reputation threshold $RT \in [0, 1]$ and prospective reward M to invoke $TaskRelease$ function in the smart contract. Note that these communication messages are encrypted by pk_{SR}.
7) The SP accepts the task request as long as $M > 0$ and publicizes related information.
8) The MWs who are interested in the sensing task can download the $Enc_{AES}(T, LSK), i, j, RT, pk_{SR}$ and an one-time blind factor σ_{random} from the smart contract. Simultaneously, the MWs can look up their reputation R_{MW} to confirm the eligibility. If $R_{MW} \le RT$, the MWs in the target area (i.e., $(long_{MW_1}, lati_{MW_1})$ – $(long_{MW_1}, lati_{MW_2})$) are qualified to decrypt task content by computing

$$T = Dec_{AES}(Enc_{AES}(T, LSK), LKS'), \qquad (1)$$

where LSK' is derived from $LSKenGen$ $(long_{MW}, lati_{MW})$ and Dec_{AES} is advanced decryption standard decryption algorithm.
9) Each MW collects the surrounding environmental data d_i such as temperature, noise intensity and pressure. d_i is encrypted by modified $pk^*_{SR} = (n^*_{SR} = n_{SR} + \sigma_{random}, g_{SR})$ as

$$Enc(d_i, pk^*_{SR}) = g^{d_i}_{SR} \cdot \sigma^{n^*_{SR}}_{random}\ mod\ {n^*_{SR}}^2. \qquad (2)$$

The encrypted data set $Enc(Enc(D, pk^*_{SR}), pk_{MW}) = Enc(Enc(d_1, pk^*_{SR}), pk_{MW_1}), Enc(Enc(d_2, pk^*_{SR}), pk_{MW_2}) \ldots, Enc(Enc(d_i, pk^*_{SR}), pk_{MW_i})$ is uploaded to the SP by calling $DataUpload$ function defined in the smart contract. Note that only the MW whose reputation is over the threshold can submit the data. Otherwise, the unauthorized operation will be informed.
10) Since the smart contract needs to filter some sensing data submitted by the malicious MWs, a trusted MW_t is set in advance. According to the encrypted data $Enc(d_t, pk^*_{SR})$ from the MW_t, the smart contract can obtain the variance var_i between trusted data and the remaining ones by computing $var_i = d_i^2 - d_t^2$, where d_i and d_t are decrypted by pk^*_{SR}, respectively. Finally, the smart contract will remove some sensing data whose var_i is large. Note that though there is a trusted MW_t can provide some reliable data, the accurate baseline data still needs to be obtained from all the filtered data.
11) The SR obtains $Enc(D, pk^*_{SR})^D$ from the smart contract and then evaluate the quality of sensing data, whose baseline value $Enc(d_o, pk^*_{SR})^{d_o}$ is calculated as

$$\underset{Enc(d_o,pk^*_{SR})^{d_o}}{\arg\min} \sqrt{Enc(d_i, pk^*_{SR})^{d_i} - Enc(d_o, pk^*_{SR})^{d_o}}. \qquad (3)$$

Algorithm 2 Smart Contract Serves as the Service Provider (SP)

```
Function $SRRegister(Addr_{SR}, \sigma_{SR})$
$p_{SR}, q_{SR} \leftarrow PrimeGen(k)$
$n_{SR} \leftarrow p_{SR} \cdot q_{SR}, \lambda_{SR} \leftarrow lcm(p_{SR}-1, q_{SR}-1)$
$g_{SR} \leftarrow {Z_{SR}}^*_{{n_{SR}}^2}$
$\mu_{SR} \leftarrow \left(L\left(g_{SR}^{\lambda_{SR}} \bmod n_{SR}^2\right)\right)^{-1} \bmod n_{SR}$
$pk_{SR} \leftarrow (n_{SR}, g_{SR}), sk_{SR} \leftarrow (\lambda_{SR}, \mu_{SR})$
Publish $pk_{SR}, sk^*_{SR} = sk_{SR} \oplus \sigma_{SR}$
Function $MWRegister(Addr_{MW}, \sigma_{MW})$
$p_{MW}, q_{MW} \leftarrow PrimeGen(k)$
$n_{MW} \leftarrow p_{MW} \cdot q_{MW}, \lambda_{MW} \leftarrow lcm(p_{MW}-1, q_{MW}-1)$
$g_{MW} \leftarrow {Z_{MW}}^*_{{n_{MW}}^2}$
$\mu_{MW} \leftarrow \left(L\left(g_{MW}^{\lambda_{MW}} \bmod n_{MW}^2\right)\right)^{-1} \bmod n_{MW}$
$pk_{MW} \leftarrow (n_{MW}, g_{MW}), sk_{MW} \leftarrow (\lambda_{MW}, \mu_{MW})$
Publish $pk_{MW}, sk^*_{MW} = sk_{MW} \oplus \sigma_{MW}$
Function $TaskRelease(Enc(Addr_{SR}||Enc_{AES}(T, LSK = LSKenGen(long_{MW}, lati_{MW}))||i||j||RT||M||, pk_{SR}))$
if $M \geq 0$ then
    $Dec((Enc(Addr_{SR}||Enc_{AES}(T, LSK = LSKenGen(long_{MW}, lati_{MW}))||i||j||RT||M, pk_{SR})), sk_{SR})$
    Publish $Enc_{AES}(T, LSK)||i||j||RT||pk_{SR}||\sigma_{random}||R_{MW}$
else
    Exit
end if
Function $DataUpload(Enc(Enc(d_i, pk^*_{SR}), pk_{MW_i}))$
if $R_{MW} \geq RT$ then
    $Dec(Dec(Enc(Enc(d_i, pk^*_{SR}), pk_{MW_i}), sk_{MW_i}))$
    $var_i = {d_i}^2 - {d_t}^2$
    if $var_i\ is\ large$ then
        $Drop\ d_i$
    else
        Publish $Enc(Enc(D, pk^*_{SR}), pk_{SR})$ to $SR$
    end if
else
    Exit
end if
Function $DataReport(Enc(Enc(q_D, pk^*_{SR}), pk_{SR}))$
$Dec(Enc(q_D, pk^*_{SR}), sk_{SR})$
TransferMoney($q_{d_i} \cdot M$)
Publish $Enc(\sigma_{random}, pk_{SR})$
```

Note that $d_o \in D$. Then the quality of data can be measured as

$$q_{d_i} = \frac{Enc(d_i, pk^*_{SR})^{d_i} - Enc(d_o, pk^*_{SR})^{d_o}}{\sum_{i=1}^{n}(Enc(d_i, pk^*_{SR})^{d_i} - Enc(d_o, pk^*_{SR})^{d_o})}. \tag{4}$$

The quality of each data also represent the reputation of each MW $R_{MW} = q_{d_i}$ in the next turn [31]. Finally, SR submits the encrypted quality evaluation result set $Enc(q_D, pk_{SR})$ to the SP through $DataReport$ function in the smart contract.

12) SP distributes reward M to the corresponding MWs in terms of their data quality, where the money each MW can receive is determined by stackelberg game analysed in the Section V.

13) SR obtains the one-time blind factor σ_{random} which mixes its public key pk_{SR}. Then SR can infer its new private key as $sk^*_{SR} = (\lambda_{SR}, \mu^*_{SR})$, where μ^*_{SR} is calculated as

$$\begin{aligned}\mu^*_{SR} &= (L(g_{SR}^{\lambda_{SR}} \bmod {n^*_{SR}}^2))^{-1}\\ &= \frac{n^*_{SR}}{(g_{SR}^{\lambda_{SR}} \bmod {n^*_{SR}}^2) - 1} \bmod n^*_{SR}.\end{aligned} \tag{5}$$

The previous sensory data value can be restored as

$$\begin{aligned}d_i &= L\left(Enc(d_i)^{\lambda_{SR}} \bmod {n^*_{SR}}^2\right) \cdot \mu^*_{SR} \bmod n^*_{SR}\\ &= \frac{\left((Enc(d_i)^{\lambda_{SR}} \bmod {n^*_{SR}}^2 - 1) \cdot \mu^*_{SR} \bmod n^*_{SR}\right)}{n^*_{SR}}.\end{aligned} \tag{6}$$

Finally, SR obtains the result of sensing task successfully.

Note that all the communication messages between SR/MWs and SP are encrypted by their public keys pk_{SR} and pk_{MW}, respectively. Fig. 3 presents the comprehensive workflow for the above-mentioned interaction. Furthermore, the execution logic of MCS smart contract is briefly illustrated in Algorithm 2.

V. Stackelberg Game-Based Fair Reward and Incentive Design

A. Notation for Fair Reward and Incentive Design

As shown in Fig. 3, in our considered BSIF-based MCS system, the mobile workers (MWs) set is denoted by $\mathcal{M} = \{1, 2, \ldots, M\}$, where M means the number of the MWs, and the reputation value set is $\mathcal{S} = \{S_1, S_2, \ldots, S_M\}$. When performing the MCS tasks, each MW participates with certain level, which are represented by $\mathcal{P} = \{P_1, P_2, \ldots, P_M\}$, and P_M is the participant level of the m-th MW. After finishing the MCS tasks, the service requester (SR) would give rewards to each MW, and it can be represented by set $\mathcal{R} = \{R_1, R_2, \ldots, R_M\}$. Note that the reputation values of MWs depend on the observation from SR, i.e., the calculation of reputation is based on the degree of deviation from the perception center data. Without loss of generality, the observation error set for each MW is defined as $\mathcal{O} = \{O_1, O_2, \ldots, O_M\}$. For ease of presentation, the notations for the symbols used in this section are listed in Table III.

B. Problem Formulation

To formulate the fair MCS for the MWs and the SR, the participation level-based utility function f_{MW_i} of the i-th MW, and the reward-based utility function f_{SR} of SR need to be defined, respectively. Firstly, as for the i-th MW, following the similar baselines in [29], its utility includes four parts, mathematically,

$$f_{MW_i} = S_i\left(\alpha_i P_i - \beta_i P_i^2 + \sum\nolimits_{j \in \mathcal{M}}^{j \neq i} \gamma_{ij} P_i P_j + R_i P_i - \eta P_i\right). \tag{7}$$

TABLE III
NOTATIONS

Symbol	Description
$\mathcal{M}$	The mobile worker set
$\mathcal{S}$	The reputation value set for MWs
$\mathcal{P}$	The participation level set for the MWs
$\mathcal{R}$	The reward set paid to the MWs by the SR
$\mathcal{O}$	The observation error set for MWs
f_{MW_i}	The utility function of the i-th MW
f_{SR}	The utility function of the SR
η	The joining cost of the MW
γ_{ij}	The mutual effect of the MW
α_i	The coefficients of f_{MW_i}
β_i	The coefficients of f_{MW_i}
ζ	The parameter related with MWs' participation level
s	The coefficients to represent the concavity of the function
t	The coefficients to represent the concavity of the function
P_{i_opt}	The optimal value of the participation level of the i-th MW
R_{i_opt}	The optimal value of the reward paid to the i-th MW

where $\alpha_i P_i - \beta_i P_i^2$ represents the private utility for performing the MCS tasks of i-th MW, and α_i, β_i are both positive coefficients, which can determine the intrinsic value of the participation to different MWs with heterogeneity [29]. $\sum_{j\in\mathcal{M}}^{j\neq i}\gamma_{ij}P_iP_j$ is the mutual effect for the participation utility among the MWs, and γ_{ij} is the influence effect for j-th MW, which results from the participation level of i-th MW. R_iP_i denotes the utility obtained from the SR, and η is the cost for each MW to perform the MCS tasks called joining cost. Besides, the reputation value S_i has the direct and positive influence for the MW's utility. Hence, the total utility of the MW is the sum value of $\alpha_i P_i - \beta_i P_i^2$, $\sum_{j\in\mathcal{M}}^{j\neq i}\gamma_{ij}P_iP_j$, R_iP_i and minus ηP_i.

As far as the SR is concerned, its utility is the total utility which produced minus the cost paid for employing the MWs. Tthe mathematical expression is [29]:

$$f_{SR} = \zeta \sum\nolimits_{i=1}^{M}\left(sP_i - tP_i^2\right) - \sum\nolimits_{i=1}^{M} R_iP_iS_iO_i, \tag{8}$$

where ζ is the parameter which indicates the equivalent monetary worth of MWs' participation level, and s, t are the positive coefficients that represent the concavity of the function [29].

Note that the participation strategy of the MW depends on the reward paid from the SR, i.e., the MWs changes the participation level after the SR adjusts the reward strategies. Hence, the Stackelberg game model is adopted here, where the SR acts as the leader and the MWs play the role of follower. Especially, based on the defined utility function for MWs and SR, the optimization goal can be expressed as follows:

$$\begin{cases} \max\limits_{R_i} f_{SR}\left(P_i, R_i\right) \\ subject\ to:\ 0 \le R_i \le R_{i\_\max} \\ The\ optimal\ solution:\ \left(P_{i\_opt}, R_i\right) \\ \begin{cases} \max\limits_{P_i} f_{MW_i}\left(P_i, R_i\right) \\ subject\ to:\ 0 \le P_i \le 1 \end{cases} \end{cases}. \tag{9}$$

C. Detailed Strategy

In this section, the Backward Induction (BI) analysis and Lagrange dual optimization approaches are adopted. Specifically, we firstly derive the optimal strategy of the follower, MWs, where the participation level-based convex optimization issue is addressed. Then, based on the optimal strategy of the MWs, the optimal value of the reward is also determined by deriving the extreme value of SR's reward function.

1) Follower-Sub Game: Theorem 1: The optimal participation level strategy P_{i_opt} for the i-th MW can be determined as:

$$P_{i\_opt} = \begin{cases} {P_i}^*, & \prod_1 \\ 0, & \prod_2 \end{cases}, \tag{10}$$

where ${P_i}^*$ is defined in Eq. 35, $\prod_1$ and $\prod_2$ are determined in Eq. 11 and Eq. 12, shown at the bottom of the page, respectively.

Proof: Please refer to Appendix C. ■

2) Leader-Sub Game:

Theorem 2: The optimal reward strategy R_{i_opt} for the SR can be determined as:

$$R_{i\_opt} = \begin{cases} {R_i}^*, & \prod_1 \cap \prod_3 \\ 0, & \prod_2 \cup \prod_4 \\ R_{i\_\max}, & \prod_1 \cap \prod_5 \end{cases}, \tag{14}$$

where $\prod_3$ is defined in Eq. 17, shown at the bottom of the next page, $\prod_4$ and $\prod_5$ are defined in Eq. 18, and Eq. 19, shown at the bottom of the next page, respectively.

Proof: Please refer to Appendix D. ■

D. Sub-Gradient Update-Based Fairness Optimization Algorithm

In order to derive the values of the dual variable μ, following the similar sub-gradient update method in [32], we propose the sub-gradient update-based fairness optimization algorithm for the considered MCS, where the dual variable and optimal strategy for MWs and SR are determined through iterations.

As shown in Algorithm 3, during the iteration process, we firstly update the iteration index in step 1. Then,

$$\prod_1 : {P_i}^* \in (0,1) \Leftrightarrow \eta - \alpha_i - \mu/S_i - \sum\nolimits_{j\in\mathcal{M}}^{j\neq i}\gamma_{ij}P_j < R_i < 2\beta_i + \eta - \alpha_i - \mu/S_i - \sum\nolimits_{j\in\mathcal{M}}^{j\neq i}\gamma_{ij}P_j, \tag{11}$$

$$\prod_2 : {P_i}^* \notin (0,1) \Leftrightarrow \left\{\eta - \alpha_i - \mu/S_i - \sum\nolimits_{j\in\mathcal{M}}^{j\neq i}\gamma_{ij}P_j \ge R_i\right\} \cup \left\{R_i \ge 2\beta_i + \eta - \alpha_i - \mu/S_i - \sum\nolimits_{j\in\mathcal{M}}^{j\neq i}\gamma_{ij}P_j\right\}, \tag{12}$$

$$L_{MW_i}\left(P_i, R_i, \mu\right) = S_i\left(\alpha_i P_i - \beta_i P_i^2 + \sum\nolimits_{j\in\mathcal{M}}^{j\neq i}\gamma_{ij}P_iP_j + R_iP_i - \eta P_i\right) - \mu\left(P_{i\_\max} - P_i\right). \tag{13}$$

Algorithm 3 Sub-Gradient Update-Based Fairness Optimization algorithm

```
Input: $\mathcal{K} = \{0, 1, \ldots, k_{max}\}$, $\mathcal{M}$, $\mathcal{P}$, $\mathcal{R}$, $\mathcal{S}$, $\mathcal{O}$, $\alpha_i$, $\beta_i$, $\gamma_{ij}$,
    $\eta$, $\zeta$, $s$ and $t$.
Output: $\mu$, $P_{i\_opt}$, and $R_{i\_opt}$.
    Step 1) Update $k = k + 1$.
    Step 2) Determine $P_{i\_opt}(k+1)$ based on Eq. (10).
    Step 3) Determine $R_{i\_opt}(k+1)$ based on Eq. (14).
    Step 4) Determine $\mu$ based on
        $\mu(k+1) = \left[\mu(k) - g_{\mu}^{k}\left(P_{i\_\max} - \mu(k+1)\right)\right]^{+}$. (20)
    Step 5) When $k \geq k_{\max}$, iterations end. Moreover, the
        value of $P_{i\_opt}$, and $R_{i\_opt}$ are returned.
```

in step 2–3, we determine the values for P_{i_opt} and R_{i_opt} with Eq. 10 and Eq. 14 respectively. Next, the dual variable μ is updated with Eq. 20. Finally, after the iterations, the optimal strategies for participation level and reward are derived.

E. Existence and Uniqueness of SE

Theorem 3: Stackelberg Equilibrium (SE) exists for the formulated fairness optimization problem of the MCS.

Proof: According to proof analysis for Theorem 1 and Theorem 2, the utility functions for the MW f_{MW_i} and f_{SR} all are concave functions with respect to participation degree P_i and reward R_i. Based on the conclusions in [33], there exists SE for the fairness-oriented participation-reward incentive design problem. ■

Theorem 4: For the existing SE of the fairness-oriented participation-reward incentive design problem, it is unique.

Proof: Based on the defined utility function f_{MW_i} of MW, it is a concave function for variable participation degree P_i. According to the duality optimization theory, once upon the other parameters are determined, the optimal participation degree strategy R_{i_opt} is unique in the established Stackelberg model. Besides, following the similar above analysis, the utility function of the reward function can also prove to be concave, so the SE R_{i_opt} is also unique. Therefore, for the existing SE of the fairness-oriented participation-reward incentive design problem, it is unique. ■

VI. Correctness and Security Analysis

In this section, we firstly prove the correctness of our BSIF regarding encrypted communication and data quality evaluation processes. Then we analyze the security of task release and allocation.

Theorem 5: SP can restore the encrypted task content $c = Enc(Addr_{SR}||Enc_{AES}(T, LSK)||i||j||RT||M, pk_{SR})$ from the SR in the task release process successfully:

Proof: Since the sk_{SR} is recorded in the smart contract, SP can decrypt the ciphertext as:

$$\begin{aligned} m_{SR}^{TaskRelease} &= L\left(g^{\lambda m_{SR}^{TaskRelease}} \cdot r_{SR}^{\lambda_{SR} n_{SR}} \bmod n_{SR}^{2}\right) \\ &\quad \cdot \mu_{SR} \bmod n_{SR} \\ &= \lambda_{SR} m_{SR}^{TaskRelease} \cdot \mu_{SR} \bmod n_{SR} \\ &= \lambda_{SR} m_{SR}^{TaskRelease} \cdot \frac{1}{\lambda_{SR}} \bmod n_{SR} \\ &= Addr_{SR}||Enc_{AES} \\ &\quad (T, LSK||i||j||RT||M, pk_{SR}). \end{aligned} \tag{21}$$

■

Theorem 6: Only MWs in the target area can obtain the task content correctly:

Proof: Assume the coordinate of MW_i is $\{long_{MW_i}, lati_{MW_i}\}$, so the location-based symmetric key can be generated as

$$LKS' = LSKenGen(long_{MW}, lati_{MW}). \tag{22}$$

Then the task content T can be decrypted if MW_i in the target area $[long_{MW_1}, lati_{MW_1}]$ – $[long_{MW_1}, lati_{MW_2}]$ as:

$$T = Dec_{AES}(Enc_{AES}(T, LSK), LKS'). \tag{23}$$

■

Theorem 7: SR can decode encrypted sensing data set $Enc(Enc(D, pk_{SR}^{*}), pk_{SR})$ and an one-time blind factor

$$f_{SR}\left(P_{i\_opt}, R_i\right) = \begin{cases} -\sum_{i=1}^{M}\left(\frac{\xi t}{4\beta_i^2} + \frac{S_i O_i}{2\beta_i}\right) \cdot R_i^2 + \sum_{i=1}^{M}\left(\frac{s\xi - 2t\xi\Delta}{2\beta_i} - S_i O_i \Delta\right) \cdot R_i \\ \qquad + \sum_{i=1}^{M} \xi\left(s\Delta - t\Delta^2\right), \quad \prod_{1} \\ 0, \quad \prod_{2}, \end{cases} \tag{15}$$

$$\frac{\partial f_{SR}\left(P_{i\_opt}, R_i\right)}{\partial R_i} = -2\sum_{i=1}^{M}\left(\frac{\xi t}{4\beta_i^2} + \frac{S_i O_i}{2\beta_i}\right) \cdot R_i + \sum_{i=1}^{M}\left(\frac{s\xi - 2t\xi\Delta}{2\beta_i} - S_i O_i \Delta\right) = 0, \tag{16}$$

$$\prod_{3} : {R_i}^{*} \in \left(0, R_{i\_\max}\right) \Leftrightarrow 0 \leq \frac{\sum_{i=1}^{M}\left(\frac{s\xi - 2t\xi\Delta}{2\beta_i} - S_i O_i \Delta\right)}{2\sum_{i=1}^{M}\left(\frac{\xi t}{4\beta_i^2} + \frac{S_i O_i}{2\beta_i}\right)} \leq R_{i\_\max}, \tag{17}$$

$$\prod_{4} : {R_i}^{*} \in (-\infty, 0) \Leftrightarrow \sum_{i=1}^{M}\left(\frac{s\xi - 2t\xi\Delta}{2\beta_i} - S_i O_i \Delta\right) < 0, \tag{18}$$

$$\prod_{5} : {R_i}^{*} \in \left(R_{i\_\max}, +\infty\right) \Leftrightarrow \sum_{i=1}^{M}\left(\frac{s\xi - 2t\xi\Delta}{2\beta_i} - S_i O_i \Delta\right) > 2R_{i\_\max}\sum_{i=1}^{M}\left(\frac{\xi t}{4\beta_i^2} + \frac{S_i O_i}{2\beta_i}\right). \tag{19}$$

$Enc(\sigma_{random}, pk_{SR})$ from the SP in the encrypted data evaluation and quality report processes, respectively.

Proof: Since the proof of **Theorem 7** is similar to **Theorem 5**, we omit the detailed process here. ■

Theorem 8: The baseline value $Enc(D, pk_{SR*})$ can well reflect the quality of sensory data set D.

Proof: Due to the homomorphic properties of the Paillier cryptosystem, the d_i and d_o can be decrypted as:

$$Dec(Enc(d_i, pk^*_{SR})^{d_i}) = d_i^2. \tag{24}$$

$$Dec(Enc(d_o, pk^*_{SR})^{d_i}) = d_o^2. \tag{25}$$

Hence, the calculation of baseline value $Enc(d_o, pk^*_{SR})$ is

$$\mathop{\arg\min}_{Enc(d_o, pk^*_{SR})} \sqrt{Enc(d_i, pk^*_{SR})^{d_i} - Enc(d_o, pk^*_{SR})^{d_o}} \tag{26}$$

can be seen as

$$d_o = \mathop{\arg\min}_{d_o} \sqrt{d_i^2 - d_o^2}, \tag{27}$$

which can well represent the variance between each sensing data d_i and the baseline value d_o. ■

Theorem 9: Based on the decisional composite residuosity assumption (DCRA), our proposed BSIF is proved secure against chosen-plaintext attacks (IND-CPA)

Proof: We model the interaction between an adversary $\mathcal{A}$ and challenger $\mathcal{C}$ as a game to prove **Theorem 9**.

Attacker's ability: $\mathcal{A}$ can conduct the following queries to the $\mathcal{C}$:

- *Public key query Q_{pk}:* $\mathcal{C}$ holds a list $\mathcal{L}_{Q_{pk}}$ that consists of tuples (pk_i, sk_i), where i is the index of the tuple. When Q_{pk} is called by $\mathcal{A}$ with input i, $\mathcal{C}$ firstly checks whether the tuple (pk_i, sk_i) exist in $\mathcal{L}_{Q_{key}}$. If (pk_i, sk_i) already exist, $\mathcal{C}$ returns corresponding pk_i to $\mathcal{A}$. Otherwise, $\mathcal{C}$ generates a new pair of $\{pk_i, sk_i\}$ and adds this tuple into the $\mathcal{L}_{Q_{key}}$, where pk_i is returned to $\mathcal{A}$.
- *Private key query Q_{sk}:* $\mathcal{A}$ inputs pk_i to invoke this query. If $\mathcal{L}_{Q_{key}}$ contains pk_i, the corresponding sk_i will be returned to $\mathcal{A}$. Otherwise, the Q_{pk} will be called firstly.
- *Encryption query Q_{Enc}:* $\mathcal{A}$ inputs a plaintext m and pk_i to the Q_{Enc}. If $pk_i \in \mathcal{L}_{Q_{key}}$, $\mathcal{C}$ will generate a ciphertext $c = g^m \cdot r^n \bmod n^2$ back to $\mathcal{A}$. Otherwise, the Q_{pk} and Q_{sk} will be invoked firstly.

Challenge: After numerous queries, $\mathcal{A}$ decides to challenge the game. $\mathcal{A}$ picks up two messages m_1, m_2 with the same length and two public keys pk_1, pk_2 to the $\mathcal{C}$. Note that $m_1 \neq m_2$ and $pk_1 \neq pk_2$. Then according to a random bit $b \in \{0, 1\}$, $\mathcal{C}$ generates a ciphertext $c_b = g^{m_b} \cdot r^n$, which will be sent back to the $\mathcal{A}$. Furthermore, $\mathcal{A}$ should also comply with the following rules.

- $\mathcal{A}$ cannot conduct Q_{sk} query with input pk_1 or pk_2.
- $\mathcal{A}$ cannot conduct Q_{Enc} query with input m_1 or m_2.

Guess: $\mathcal{A}$ guesses the result of b' based on the received c_b. If $b' = b$, $\mathcal{C}$ outputs 1 which means $\mathcal{A}$ wins the game. Otherwise, $\mathcal{C}$ outputs 0 and $\mathcal{A}$ loses the game. The advantage that $\mathcal{A}$ can win the game is defined as $\varepsilon(n) = \Pr\left[\mathrm{PubK}^{\mathrm{eav}}_{\mathcal{A},\Pi}(n) = 1\right]$, where eav means the existence of an eavesdropper. Since the ciphertext is calcualted as $c = g^m \cdot r^n \bmod n^2$, we can get

$$\begin{aligned} \varepsilon(n) &= \Pr\left[\mathrm{PubK}^{\mathrm{eav}}_{\mathcal{A}.\Pi}(n) = 1\right] \\ &= \Pr\left[Dec\left(g, \left[r^g \bmod g^2\right]\right) = 1\right] \\ &= Adv_{\mathcal{C}}^{IND-CPA}, \end{aligned} \tag{28}$$

due to the hardness of DCRA, $Adv_{\mathcal{C}}^{IND_CPA}$ is negligible. Hence, our proposed BSIF is secure against IND-CPA. ■

Theorem 10: Our proposed BSIF can prevent single-point-of-failure attacks

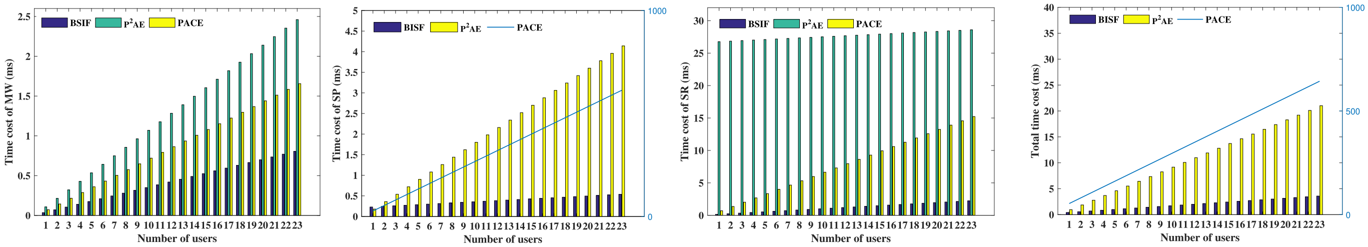
Proof: In our BSIF protocol, the used blockchain technique helps us convert a centralized SP into a smart contract which is maintained by many decentralized mining nodes MNs. If an adversary $\mathcal{A}$ would like to intrude or compromise the whole SP network, which requires him to seize more than one half computation power in the blockchain network. Hence, the successful attack threshold can be computed as:

$$\frac{H_A}{H_t} \geq 51\%. \tag{29}$$

Note that H_A, H_t means $\mathcal{A}$'s computation power and total computation power, respectively. Since our protocol is planned to be established in some open blockchain platforms such as Ethereum or EOS, A is tough to compromise these systems at present due to the unreachable computation power. Hence, our proposed BSIF can mitigate single-point-of-failure attacks. ■

Theorem 11: Our proposed BSIF can achieve the correctness and trustness of off-blockchain computation on the SR side.

Proof: The SR takes the responsibilities of data evaluation, which finally figure out a encrypted baseline value $Enc(d_o, pk^*_{SR}) = \mathop{\arg\min}_{Enc(d_o, pk^*_{SR})} \sqrt{Enc(d_i, pk^*_{SR})^{d_i} - Enc(d_o, pk^*_{SR})^{d_o}}$ within the sensing dataset. Moreover, this optimal data is returned back to the SP for data filtration, which means the bad sensing data far away from the optimal data will be excluded. Hence, the SR owns the common interest with the SP. Once the SR provides a falsified baseline, then the SP will give back the inaccurate sensing data which may lead to the ultimate failure for mobile crowdsensing task. Starting from the self-interest, the SP will provide the correct baseline data in possible. The computation correctness of this off-blockchian computation is ensured by the homomorphic multiplication provided by the Paillier Cryptosystem. Since the decryption of $Enc(d_i, pk^*_{SR})^{d_i}$ and $Enc(d_o, pk^*_{SR})^{d_o}$ are $Dec(Enc(d_i, pk^*_{SR})^{d_i}) = d_i^2$ and $Dec(Enc(d_o, pk^*_{SR})^{d_i}) = d_o^2$, the encrypted baseline value $Enc(d_o, pk^*_{SR})$ can also represent the centroid of the original sensing data $d_o = \mathop{\arg\min}_{d_o} \sqrt{d_i^2 - d_o^2}$. According to this baseline, the SP can evaluate the data quality and MW reputation as $q_{d_i} = \frac{Enc(d_i, pk^*_{SR} - Enc(d_o, pk^*_{SR})}{\sum_{i=1}^{n}(Enc(d_i, pk^*_{SR})^{d_i} - Enc(d_o, pk^*_{SR})^{d_o})}$ and $R_{MW} = q_{d_i}$, respectively. In conclusion, the correctness and trustness of this off-blockchain computation can be guaranteed by the common interest and homomorphic multiplication, respectively. ■



(a) Computational overhead for SR (b) Computational overhead for SP (c) Computational overhead for MWs (d) Total computational overhead

Fig. 4. Comparison about computational overhead.

VII. Security Performance Evaluation

In this section, we evaluate the performance of our proposed BSIF in terms of computational/communication overhead and security features. We implement a prototype of BSIF based on Ganache,[3] a personal Ethereum blockchain platform for smart contract test and development. The operations of SR, SP and MW are executed on a laptop with Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz, 8.0 GB memory and Windows 10 home operation system.

A. Computational Overhead

Regarding the computational overhead evaluation, the total number of the time-consuming cryptographic operations is a common metrics to measure the computational cost. We choose two random prime numbers p, q of 256 bits and a random 160 bit integer g of multiplicative group $Z^*_{n^2}$ as our system initial parameters. The cryptographic operations that are composed of our protocol include modular exponentiation T_{Exp}, modular multiplication T_M, modular inverse T_{Inv}, homomorphic multiplication T_{HM}, and AES encryption/decryption $T_{AES_{Enc}}/T_{AES_{Dec}}$. Note that compared with the above mentioned cryptographic operations, the computation cost of point addition and integer multiplication can be omitted. The cryptographic operations T_{Exp}, T_M, T_{Inv}, T_{HM}, $T_{AES_{Enc}}$, and $T_{AES_{Dec}}$ measured in the above-mentioned desktop are 0.010ms, 0.004ms, 0.066ms, 0.095ms, 0.017ms and 0.007 ms at the service requester (SR) and mobile worker (MW) side. Since SR executes three modular exponentiation operations T_{Exp}, three modular multiplication operations T_M, n homomorphic multiplication operations T_{HM} (e.g., n depends on the number of MWs), and one AES encryption operation $T_{AES_{Enc}}$, the total running time for SR is $3T_{Exp} + 3T_M + nT_{HM} + T_{AES_{Enc}} = (0.059 + 0.095n)ms$. The number of modular exponentiation operations T_{Exp}, modular multiplication operations T_M and modular inverse T_{Inv} executed by SP are six plus n, six plus n and two, respectively. Hence, the computational cost for SP is $(6 + n)T_{Exp} + (6 + n)T_M + 2T_{Inv} = (0.216 + 0.014n)ms$. The computational overhead for MW is $2T_{Exp} + 2T_M + T_{AES_{Dec}} = (0.035n)ms$ due to the two modular exponentiation operations T_{Exp}, two modular multiplication operations T_M and one AES decryption operation $T_{AES_{Enc}}$. We compare BSIF with several privacy-preserving MCS protocols such as P^2AE [23] and $PACE$ [10], which are simulated under the identical setting. Furthermore, the number of the crytographic operations and execution time for the above-mentioned schemes are calcualted in Table IV. To evaluate the performance of three schemes, we assume the number of MW is dynamically changed. Fig. 4(a) presents the trend of computational overhead for SR with the increasing number of users. It is clear that the time cost of our proposed BSIF is much low even if the number of users reaches 23. In Fig. 4(b) and Fig. 4(c), the computational cost of BSIF is also lower than other two schemes. Hence, BSIF achieves the least computational overhead compared with P^2AE and $PACE$ as shown in Fig. 4(d).

B. Communication Overhead

In our proposed BSIF, since the key generation process requires the participant to transmit its personal information firstly and then computes $pk = (n = p \cdot q, g), sk = (\lambda = lcm(p-1, q-1), \mu = \left(L\left(g^{\lambda} \bmod n^2\right)\right)^{-1} \bmod n)$, we conclude the communication cost for registration as $(n+1)\cdot(||m||+||g||)$ bits, where $||m||$ means the length of plaintext. Besides, service provider (SP) exchanges the ciphertext with service requester (SR) and mobile worker (MW) six times, which can be calculated as $(4 + 2n) \cdot ||c||$. Note that $||c||$ denotes the length of ciphertext. Hence, we can conclude the total communication cost for our scheme is $2 \cdot (n+1)(||m|| + ||g||) + (4 + 2n) \cdot ||c||$.

C. Feature Comparison

As can be seen from the Table. V, though P^2AE [23] and $PACE$ [10] both adopt the cryptographic primitives (i.e., zero-knowledge proof and proxy re-encryption) to achieve the identity authentication, encrypted task content and worker location privacy preservation, the single-point-of-failure (SPOF) resulted from the centralized server has not been well addressed. In our proposed BSIF, we introduce the blockchain-based MCS framework that combines the Paillier Cryptosystem and location-based key generation algorithm can achieve all the security and decentralized features mentioned above. Regarding task allocation, BSIF and P^2AE [23] utilize the reputation threshold and reduced-dimensionality enabled Q-Learning to select the qualified MWs for accurate task allocation. Moreover, the reward deposited in the SP in advance can guarantee the reliable payment in BSIF and

[3] https://trufflesuite.com/ganache/

TABLE IV

COMPARISON FOR COMPUTATIONAL OVERHEAD

Scheme	**SR**	**SP**	**MW**	**Total**
$BSIF$	$3T_{Exp} + 3T_M + nT_{HM} + T_{AES_{Enc}}$	$(6+n)T_{Exp} + (6+n)T_M + 2T_{Inv}$	$2nT_{Exp} + 2nT_M + nT_{AES_{Dec}}$	$(9+3n)T_{Exp} + (9+3n)T_M + nT_{HM} + 2T_{Inv} + T_{AES_{Enc}} + nT_{AES_{Dec}}$
P^2AE [23]	$(2+n)T_{Exp} + T_M + nT_{Inv} + T_{Bp} + 2T_{AES_{Enc}} + nT_{AES_{Dec}}$	nT_{Bp}	$nT_{Exp} + nT_{Inv} + 2nT_{AES_{Dec}} + nT_{AES_{Enc}}$	$(2+2n)T_{Exp} + T_M + 2nT_{Inv} + (n+1)T_{Bp} + (2+n)T_{AES_{Enc}} + 3nT_{AES_{Dec}}$
$PACE$ [10]	$4T_{Exp} + 2T_M + nT_{Inv}$	$4nT_{Exp} + 2nT_M + 2nT_{Inv}$	$6nT_{Exp} + 3nT_M$	$(4+10n)T_{Exp} + (2+5n)T_M + 3nT_{Inv}$

TABLE V

FEATURE COMPARISON

Scheme	BSIF	P^2AE [23]	$PACE$ [10]
Decentralization	Yes	No	No
SPOF defence	Yes	No	No
Authenticated Communication	Yes	Yes	Yes
Encrypted task content	Yes	Yes	Yes
Worker location privacy preservation	Yes	Yes	Yes
Accurate task allocation	Yes	Yes	No
Offloading data evaluation	Yes	No	No
Reliable payment	Yes	No	Yes
Fair reward and incentive mechanism	Yes	No	No

TABLE VI

EXPENSE FOR DIFFERENT BLOCKCHAIN EXECUTION PHASES

Scheme	**BSIF**	**Cai et al. [15]**
Contract deployment	4.37 USD	1.74 USD
Data collection	0.18 USD	2.60 USD
Data evaluation	0.15 USD	0.47 USD
Total cost	4.70 USD	4.81 USD

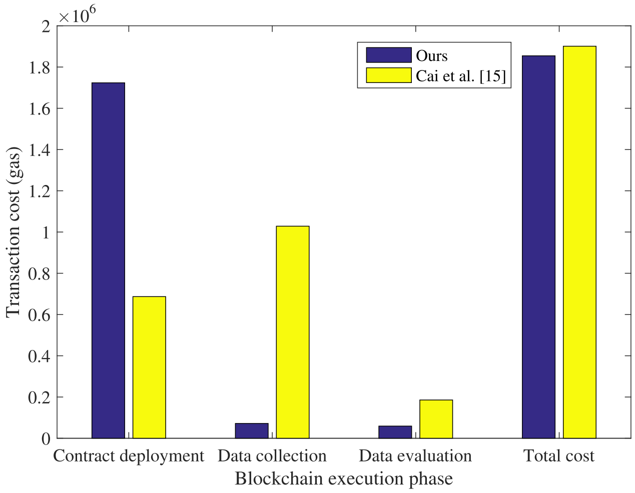


Fig. 5. Transaction cost for on-chain performance evaluation.

$PACE$ [10]. However, P^2AE [23] and $PACE$ [10] figure out the reward just depending on the data variance compared with the baseline value, which cannot motivate the activeness of SP and SR at the utmost. BSIF designs a fair reward and incentive mechanism by using multi-leader and multi-follower Stackelberg. Finally, one striking feature is that BSIF constructs an interative MCS, which offloads the data evaluation to the SR with the aid of homomorphic multiplication feature provided by Paillier Cryptosystem. In conclusion, our proposed BSIF achieves security, interaction and fairness requirements compared with two recent representative MCS work.

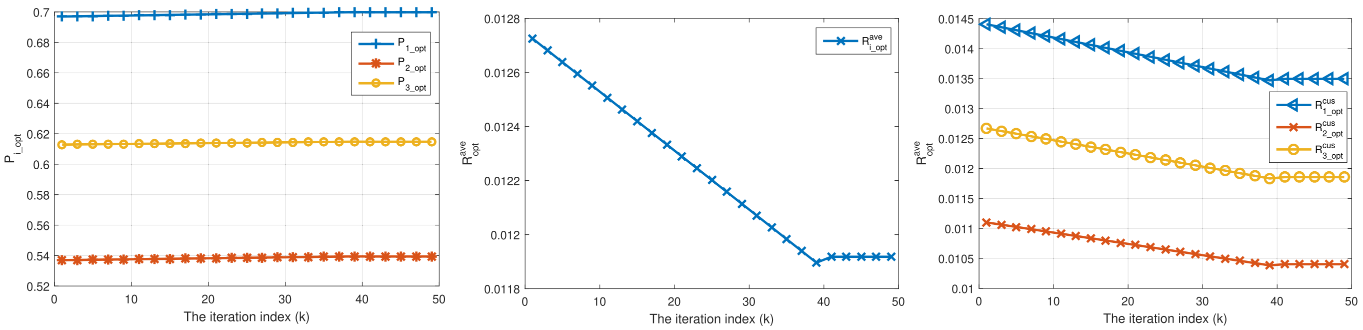
D. BSIF On-Chain Performance

We implement our BSIF MCS framework on the Ethereum smart contract to evaluate the on-chain performance. Our blockchain execution phases mainly consider three operations (i.e., contract deployment, data collection and data evaluation), which corresponds to the settings in previous work [15] for comparison. Note that the gas cost in “Select” and “Submit” phases in [15] are added as gas cost for data collection phase, the “Commit” and “Harvest” phases in [15] are concluded as data evaluation phase in our BSIF. As can been from Fig. 5, though the gas cost for our contract deployment (i.e., 1723560 gas) is nearly 2.5 times larger than the same phase in Cai et al.’s work [15] (i.e., 686882 gas), our deployed smart contract can be applied to various MCS tasks without re-deployment. For the data collection and data evaluation processes, our BSIF has remarkable advantages that the gas cost is 71881 and 59098 gas, respectively. Due to the zero-knowledge range proofs and off-chain data validity check adopted in [15], the complicated interaction and computation process lead to higher gas cost (i.e., 1028324 and 185678 gas) in the corresponding phases for [15]. Hence, our BSIF can achieve the similar functions and security guarantees with a little lower total cost (i.e., 1854539 gas) compared with cost (i.e., 1900884 gas) in [15]. Moreover, the gas cost also can be converted into the USD for more direct economic effects. As indicated by the ethereumprice,[4] the gas price equals to $1.88 * 10^{-9}$ ether and 1 ether is USD. Subsequently, we can figure out expenses for each operation in our BSIF: contract deployment≈ 4.37 USD, data collection ≈ 0.18 USD and data evaluation ≈ 0.15 USD. As we have mentioned before, our deployed MCS contract can handle multiple tasks rather than once time. For example, there are 100 MCS tasks need to be deployed, the contract deployment takes 437 USD and 174 USD in BSIF and [15], respectively. Finally, we can see from Table. VI that the total expenses for BSIF (i.e., 4.70 USD) is also lower than (i.e., 4.81 USD) in [15].

VIII. FAIRNESS PERFORMANCE EVALUATION

In this section, to evaluate the performance of the proposed fairness-oriented algorithm, various experiments are conducted under different parameters settings. At first, the simulation settings and the compared baselines for the MCS are introduced. Then, the correctness of the proposal is verified by the convergence behavior of the service requester (SR) and mobile workers (MWs). Finally, the participation level and reward payoff are presented under different joining costs, reputation values, and observation errors, which evaluate the effectiveness of the proposal.

[4] https://ethereumprice.org/gas/



(a) Convergence behavior of P_{i_opt}.

(b) Convergence behavior of $R^{ave}_{i\_opt}$.

(c) Convergence behavior of $R^{cus}_{i\_opt}$.

Fig. 6. Convergence behavior evaluation.

TABLE VII
SIMULATION PARAMETER SETTINGS

Symbol	Value	Symbol	Value
M	3	$\mathcal{P}$	$\{0.95, 0.85, 0.9\}$
$\mathcal{R}$	$\{10, 5, 8\}$	$\mathcal{S}$	$\{0.6, 0.7, 0.9\}$
$\mathcal{O}$	$\{0.15, 0.2, 0.05\}$	α	$\alpha \sim N(15, 2)$
β	$\beta \sim N(15, 2)$	γ_{ij}	$\gamma_{ij} \sim N(15, 2)$
η	11	ζ	0.0001
s	1.5	t	0.004
k_{max}	50	g^k_μ	0.015

TABLE VIII
REPUTATION VALUES SETTINGS

Reputation cases	Value
U_1	$\mathcal{S} = \{0.6, 0.7, 0.9\}$
U_2	$\mathcal{S} = \{0.5, 0.6, 0.8\}$
U_3	$\mathcal{S} = \{0.4, 0.5, 0.7\}$
U_4	$\mathcal{S} = \{0.35, 0.45, 0.65\}$
U_5	$\mathcal{S} = \{0.3, 0.4, 0.6\}$
U_6	$\mathcal{S} = \{0.25, 0.35, 0.55\}$

A. Parameter Settings and Baselines

In this subsection, we firstly set the simulation parameter settings, which are presented in Table VII. Then, to evaluate the performance of the proposal, following the benchmark in [29], the baseline for MW's reward is introduced as well.

1) Parameter Settings: The detailed parameter settings are shown in Table VII. For ease of presentation, we set the number of the MWs is 3, and α, β, γ_{ij} all are random variables with the normal distributions.

2) Compared Baselines: Based on [29], there are two kinds of reward payment schemes for SR, i.e., the discriminatory incentive mechanism and uniform incentive mechanism. To be specific, under the discriminatory incentive mechanism, the reward paid by the SR to different MWs are different, while for uniform incentive mechanism, all MWs gain the same reward after participating the MCS tasks, mathematically,

$$R^{ave}_{i\_opt} = \frac{1}{M}\sum\nolimits_{i=1}^{M} R_{i\_opt}, \tag{30}$$

$$R^{cus}_{i\_opt} = \frac{P_{i\_opt}}{\sum\nolimits_{i=1}^{M} P_{i\_opt}} \cdot \sum\nolimits_{i=1}^{M} R_{i\_opt}, \tag{31}$$

where $R^{ave}_{i\_opt}$ is the reward under discriminatory incentive mechanism, and we call it the average reward. For $R^{cus}_{i\_opt}$, it is defined as the customized reward, which is calculated under the uniform incentive mechanism.

B. Convergence Behavior Evaluation

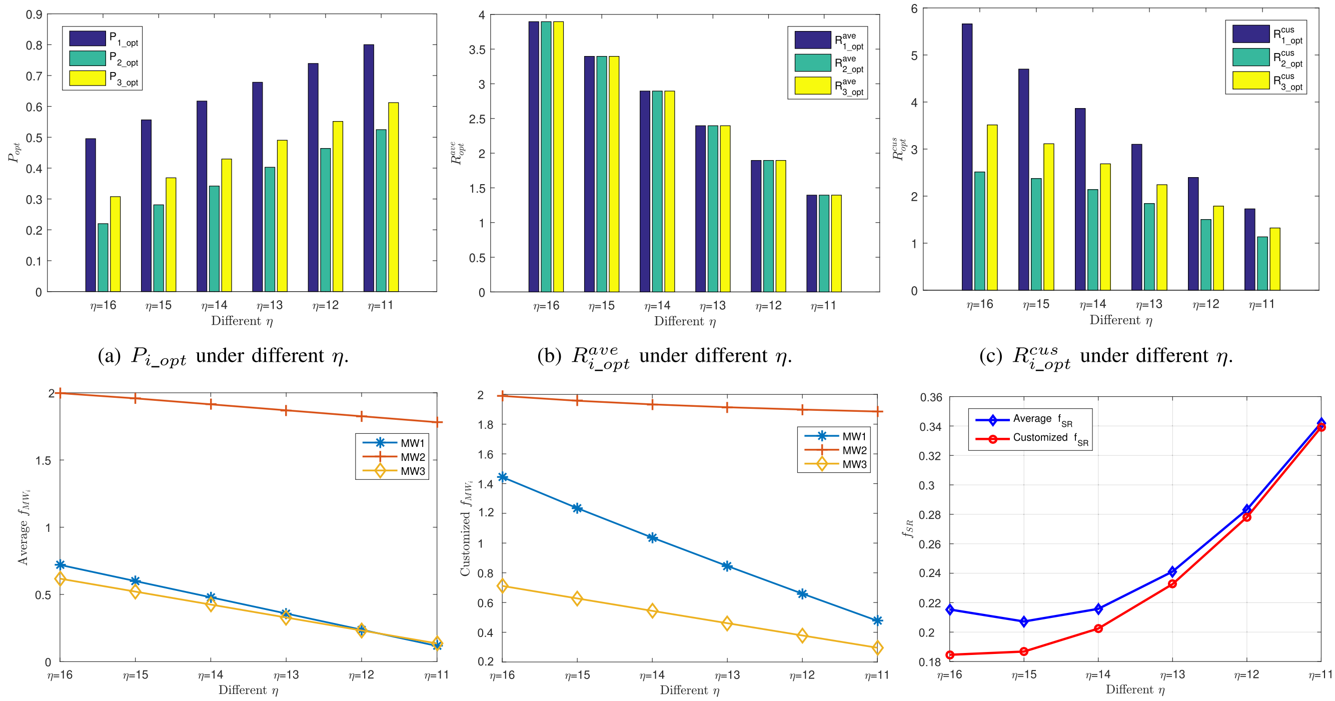
Fig. 6 is the convergence behavior of P_{i_opt}, $R^{ave}_{i\_opt}$ and $R^{cus}_{i\_opt}$. As shown in Fig. 6, in order to increase the utility of the MW, the participation level of the MW increases. At this time, the SR chooses to decrease the reward paid to the MWs to maximize its utility. On the contrary, if the reward obtained by the MW is always decreasing, the MW would refuse to perform the MCS tasks, and the SR needs to adjust the reward value to ensure the achievement of the tasks. In general, when the number of iteration increases, the value of P_{i_opt}, $R^{ave}_{i\_opt}$ and $R^{cus}_{i\_opt}$ are converged gradually. Hence, the correctness of the proposed algorithm is verified, and the proposed scheme is feasible.

C. Performance Under Different Joining Costs

Fig. 7 is the proposal's performance under different joining costs. As shown in Fig. 7(a), when the values of the joining cost decrease, the MW would choose to participate in the MCS tasks more actively, which aims to maintain its own utility f_{MW_i}. In Fig. 7(b) and Fig. 7(c), the SR also decreases the reward paid to the MWs with the value of joining cost decreasing. Further speaking, the detailed reasons are shown in Fig. 7(d)–Fig. 7(e), if the value of the joining cost decreases, the utility for the MW will decrease. According to Eq. 13, in order to maximize the MW's own utility, the MW should enhance its participation level. At this time, based on Eq. 15, the SR can decrease the reward which is paid to the MWs to maximize its own utility. In addition, Fig. 7(f) also indicates that the proposed approach, which is equal to the average incentive mechanism, is more effective than the customized incentive mechanism.

D. Performance Under Different Reputation Values

In this subsection, 6 different kinds of reputation values are considered, which are defined in Table VIII.



(a) P_{i_opt} under different η.

(b) $R^{ave}_{i\_opt}$ under different η.

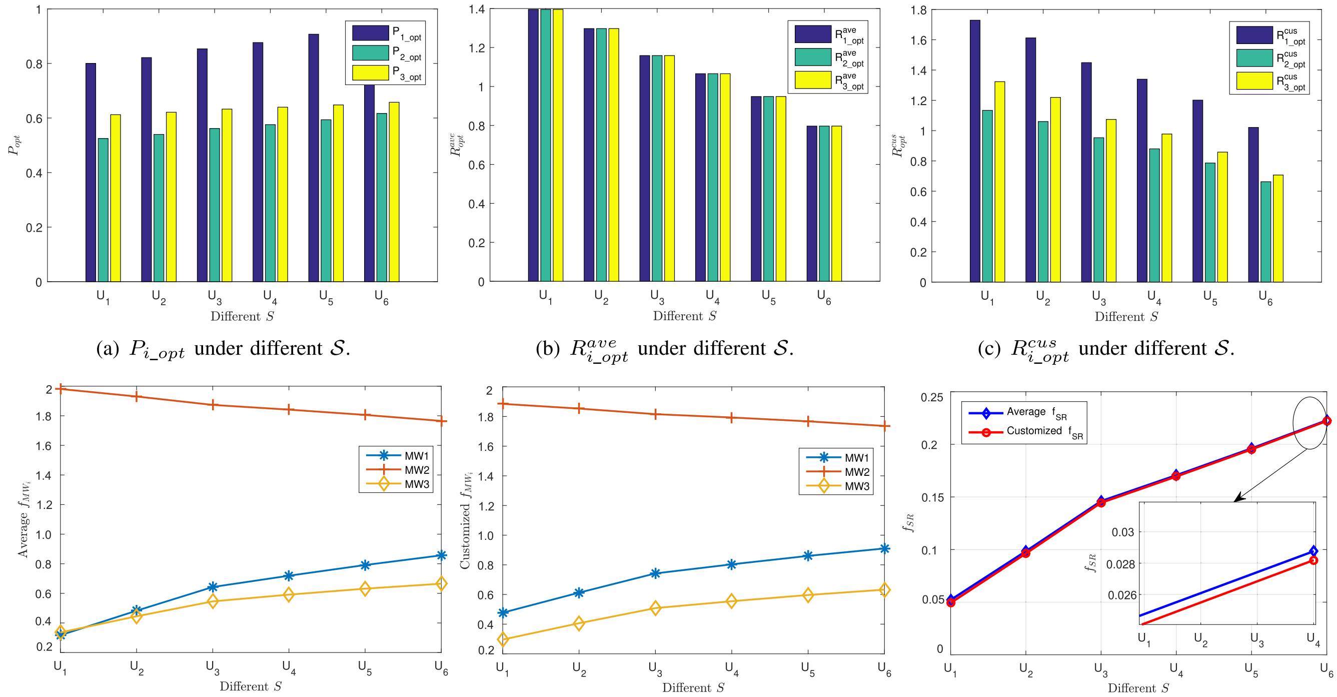
(c) $R^{cus}_{i\_opt}$ under different η.

(d) Average f_{MW_i} under different η.

(e) Customized f_{MW_i} under different η.

(f) f_{SR} under different η.

Fig. 7. Performance under different joining costs.



(a) P_{i_opt} under different $\mathcal{S}$.

(b) $R^{ave}_{i\_opt}$ under different $\mathcal{S}$.

(c) $R^{cus}_{i\_opt}$ under different $\mathcal{S}$.

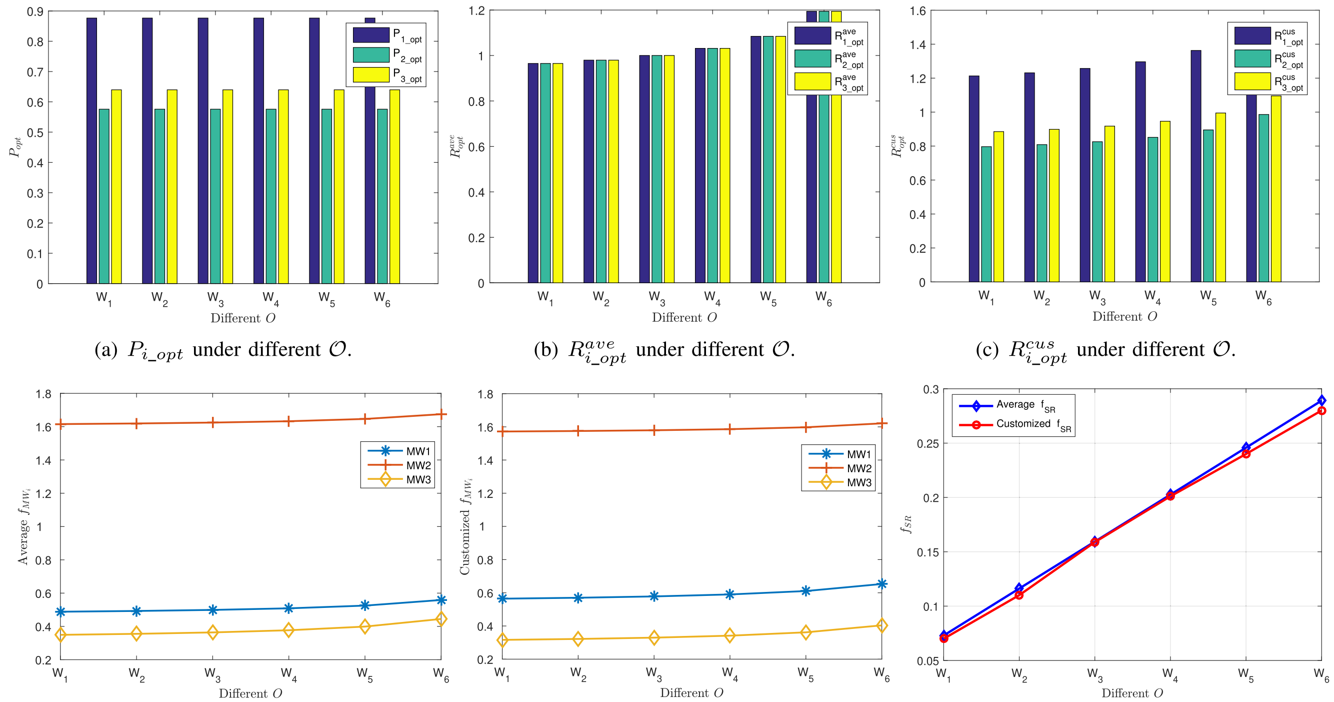
(d) Average f_{MW_i} under different $\mathcal{S}$.

(e) Customized f_{MW_i} under different $\mathcal{S}$.

(f) f_{SR} under different $\mathcal{S}$.

Fig. 8. Performance under different reputation values.

Fig. 8 is the proposal's performance under different reputation values. As can be seen from Fig. 8(a), when the reputation values decrease, the participant level increases. For Fig. 8(b) and Fig. 8(c), the SR also decreases the reward when the value of reputation decreases. The reasons are presented in Fig. 8(d)–Fig. 8(f), when the value of the reputation decreases, the utility for the MW will decrease. Based on Eq. 13, to maximize the utility f_{MW_i}, the MW should increase the participation level. Besides, according to Eq. 15, the SR needs to decrease the reward for maximizing its own utility. At last, under different reputation values, Fig. 8(f) evaluates that the



(a) P_{i_opt} under different $\mathcal{O}$.

(b) $R^{ave}_{i\_opt}$ under different $\mathcal{O}$.

(c) $R^{cus}_{i\_opt}$ under different $\mathcal{O}$.

(d) Average f_{MW_i} under different $\mathcal{O}$.

(e) Customized f_{MW_i} under different $\mathcal{O}$.

(f) f_{SR} under different $\mathcal{O}$.

Fig. 9. Performance under different observation errors.

TABLE IX
OBSERVATION ERROR CASES AND VALUES

Observation error cases	Value
W_1	$\mathcal{O} = \{0.35, 0.4, 0.3\}$
W_2	$\mathcal{O} = \{0.3, 0.35, 0.25\}$
W_3	$\mathcal{O} = \{0.25, 0.3, 0.2\}$
W_4	$\mathcal{O} = \{0.2, 0.25, 0.15\}$
W_5	$\mathcal{O} = \{0.15, 0.2, 0.1\}$
W_6	$\mathcal{O} = \{0.1, 0.15, 0.05\}$

proposed approach can achieve better performance and higher utility than the customized incentive mechanism.

E. Performance Under Different Observation Errors

In this subsection, 6 different kinds of observation errors are considered, which are defined in Table IX.

Fig. 9 is the proposal's performance under different observation errors. As shown in Fig. 9(a), if the values of the observation errors decreases, the MW cannot adjust the participation level, which focuses on maintaining its own utility f_{MW_i}. For Fig. 9(b) and Fig. 9(c), the SR would increase the reward when the value of observation errors decreases. To be specific, the reasons are shown in Fig. 9(d)–Fig. 9(f), when the value of the observation errors decreases, the utility for the MW will increase. By Eq. 13, to maximize the utility f_{MW_i}, the MW does not need to change the participation level. Hence, according to Eq. 15, the SR would increase the reward for maximizing its own utility. Besides, under different observation errors, Fig. 9(f) also evaluates that the proposed approach is more effective than the customized incentive mechanism.

IX. CONCLUSION

In this paper, we propose a blockchain-based secure, interactive and fair MCS framework called BSIF, which is able to solve the over-centralized server issue and privacy-preserve the task allocation process with the aid of decentralized ledge technique and Paillier Cryptosystem. Moreover, the location-based key generator is utilized to select mobile workers (MWs) in the target area without coordinate privacy leakage. Unlike other blockchain-based MCS frameworks, BSIF transfers data evaluation process to the service requester (SR)'s side, which means to alleviate the pressure and excessive cost from blockchain platform. The economic effective cost is also considered here, so we introduce the Stackelberg game to find the balance between participant level of MWs and fair reward for the SR. Finally, we also illustrate the security and efficiency of BSIF through theoretical analysis and experiment evaluation.

APPENDIX A

The comprehensive procedures for Location-based Key Generation are presented as follows:

1) **Key Generation** (KeyGen): Let $p,\ q$ be two large primes which are selected from a security parameter k. The modulus of the cryptosystem n and the first part of secret key sk are calculated as $n = p \cdot q$ and $\lambda = lcm(p-1, q-1)$, respectively. Note that lcm stands for least common multiple. Subsequently, this algorithm selects a random integer g as the generator for multiplicative group $Z^*_{n^2}$. Finally, the last part of sk can be computed as $\mu = \left(L\left(g^{\lambda} \bmod n^2\right)\right)^{-1} \bmod n$, where $L(x) = \frac{x-1}{n}$. The generated public key pk is (n, g) and sk is (λ, μ).

TABLE X

EXAMPLE FOR LONGITUDE CONVERSION

Sequence	Min longitude	Mid longitude	Max longitude	Bit result	
				$long_{MW}$=32.0000000	$long_{MW}$=32.0900000
1	-180	0	180	1	1
2	0	90	180	0	0
3	0	45	90	0	0
4	0	22.5	45	1	1
5	22.5	33.75	45	0	0
6	22.5	28.125	33.75	1	1
7	28.125	30.9375	33.75	1	1
8	30.9375	32.34375	33.75	0	0
9	30.9375	31.640625	32.34375	1	1
10	31.640625	31.9921875	32.34375	1	1
11	31.9921875	32.1679688	32.34375	0	0
12	31.9921875	32.0800781	32.1679688	0	1
...	...	...	...	...	...

2) **Encryption** (Enc): Input a message m, $pk = (n, g)$ and a random number r, where $0 \leq m < n$ and $0 < r < n$. Then, the encrypted message c can be calculated $c = g^m \cdot r^n \bmod n^2$.
3) **Decryption** (Dec): Given the c and $sk = (\lambda, \mu)$, the message can be decoded as $m = L\left(c^\lambda \bmod n^2\right) \cdot \mu \bmod n$.
4) **Homomorphic Multiplication** (HM): Due to the homomorphic properties of Paillier Cryptosystem, a ciphertext raised to the power of a plaintext can be decrypted to the sum of their corresponding plaintexts: $Dec(Enc(m_1, pk)^{m_2} \bmod n) = m_1 m_2 \bmod n$.

APPENDIX B

The computation process for Location-based Key Generation are illustrated as follows:

1) **Longitude Conversion:** (LongCon): Firstly, input the maximum and minimum longitude (i.e., $long_{max}$ and $long_{min}$) for the MW such as 180 and -180. Then calculate the median latitude $long_{med}$ for the MW. Note that the $long_{med}$ for the MW is 0 at the initial iteration. Assume the target longitude for the MW is $long_{MW}$. If $long_{MW} > long_{mid}$, the bit result is set as 1. $long_{min}$ and $long_{mid}$ are replaced by $long_{mid}$ and $\frac{(long_{min}+long_{max})}{2}$ in sequence for the next turn. Otherwise, the bit result x is set as 0. $long_{max}$ and $long_{mid}$ are updated to $long_{mid}$ and $\frac{(long_{min}+long_{max})}{2}$ in order. By repeating this process, the latitude can be converted to a series of bit sequence with i precision. For example, assume the longitude area for the sensing task is [32.0000000, 32.0900000], we can obtain $x = 100101101100$ as shown in Table X, whose i is 11. Note that SR can set the required precision at the beginning.
2) **Latitude Conversion:** (LatiCon): The latitude conversion is similar to the above-mentioned LongCon. If the input longitude area for the sensing task is [52.5100000, 52.5180000], we can obtain $y = 11001010101$, whose precision j is 11.
3) **Symmetric Key Generation:** (SKeyGen): Calculate the hash value of x and y as a location-based secret key $LSK = H(x||y)$, where $H(\cdot)$ is a hash function with 256 bits.

APPENDIX C

For the defined participation level-based utility function f_{MW_i}, its concavity or convexity are determined by the positive or negative of the second derivation. Mathematically,

$$\frac{\partial^2 f_{MW_i}}{\partial {P_i}^2} = -2 S_i \beta_i < 0. \tag{32}$$

Since the value of second derivation of f_{SR} is less than 0, the utility function f_{MW_i} is concave. At this time, the convex optimization approach, Lagrange dual optimization, is adopted. To be specific, based on the dual variable μ, the Lagrange function L_{MW_i} is denoted as Eq. 13.

At this time, the optimization problem can be rewritten as follows:

$$D(\mu) = \max_{P_i} L_{MW_i}(P_i, R_i, \mu), \tag{33}$$

where the dual problem is $d^* = \min_{\mu > 0} D(\mu)$.

Based on the Karush-Kuhn-Tucker (KKT) conditions [34], we let $\frac{\partial L_{MW_i}}{\partial P_i} = 0$, i.e.,

$$\begin{aligned}\frac{\partial L_{MW_i}}{\partial P_i} &= S_i \left(\alpha_i - 2\beta_i P_i + \sum\nolimits_{j \in \mathcal{M}}^{j \neq i} \gamma_{ij} P_j + R_i - \eta\right) \\ &\quad + \mu = 0. \end{aligned} \tag{34}$$

Then we can get the participation level strategy ${P_i}^*$ of the i-th MW, which is expressed as:

$${P_i}^* = \frac{\alpha_i + \sum\nolimits_{j \in \mathcal{M}}^{j \neq i} \gamma_{ij} P_j + R_i - \eta + \mu / S_i}{2\beta_i} = \frac{\alpha_i}{2\beta_i} + \Delta, \tag{35}$$

where Δ is denoted as Eq. 36 for ease of presentation.

$$\Delta = \frac{\sum\nolimits_{j \in \mathcal{M}}^{j \neq i} \gamma_{ij} P_j + R_i - \eta + \mu / S_i}{2\beta_i}. \tag{36}$$

Based on the above analysis, we can obtain the optimal value of the participation level strategy P_{i_opt}. To be specific, in case $\prod_1$, ${P_i}^* \in (0, 1)$ is ensured, and $P_{i\_opt} = {P_i}^*$, otherwise, $P_{i\_opt} = 0$ in case $\prod_2$.

APPENDIX D

As shown in Eq. 9, when determining the value of R_{i_opt}, the value of the P_{i_opt} should be brought at first, and the expression of $f_{SR}(P_{i\_opt}, R_i)$ is presented in Eq. 15.

Note that the value of the second derivation for f_{SR} is less than 0, i.e.,

$$\frac{\partial^2 f_{SR}(P_{i\_opt}, R_i)}{\partial {R_i}^2} = -2 \sum\nolimits_{i=1}^{M} \left(\frac{\xi t}{4\beta_i^2} + \frac{S_i O_i}{2\beta_i}\right) < 0. \tag{37}$$

At this time, based on [29], the reward paid by the SR can be solved by letting $\frac{\partial f_{SR}(P_{i\_opt}, R_i)}{\partial R_i} = 0$, which is expressed in Eq. 16.

Hence, the mathematical expression of reward ${R_i}^*$ is

$${R_i}^* = \frac{\sum\nolimits_{i=1}^{M} \left(\frac{s\xi - 2t\xi\Delta}{2\beta_i} - S_i O_i \Delta\right)}{2 \sum\nolimits_{i=1}^{M} \left(\frac{\xi t}{4\beta_i^2} + \frac{S_i O_i}{2\beta_i}\right)}. \tag{38}$$

Based on the above derivation, the optimal value R_{i_opt} is determined under three following cases:

1) case 1: $\prod_1 \cap \prod_3$. Under this circumstance, the value of P_{i_opt} is $P_i{}^*$. Besides, owing to the range value of $R_i{}^*$ is within $(0, R_{i\_\max})$. Hence, the optimal value R_{i_opt} can be obtained at $R_i{}^*$.
2) case 2: $\prod_2 \cup \prod_4$. In this case, the value of P_{i_opt} is 0. At this time, taking $P_{i\_opt} = 0$ into Eq. 15, the optimal value R_{i_opt} is 0.
3) case 3: $\prod_1 \cap \prod_5$. In this situation, on the one hand, the value of P_{i_opt} is $P_i{}^*$. On the other hand, the range value of $R_i{}^*$ is within $(R_{i\_\max}, +\infty)$. And the optimal value R_{i_opt} can be obtained at $R_{i\_\max}$. Therefore, according to the above analysis, the optimal reward strategy R_{i_opt} can be determined with Eq. 14.

References

[1] R. K. Ganti, F. Ye, and H. Lei, “Mobile crowdsensing: Current state and future challenges,” *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 32–39, Nov. 2011.

[2] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y. A. Zhang, “The roadmap to 6G: AI empowered wireless networks,” *IEEE Commun. Mag.*, vol. 57, no. 8, pp. 84–90, Aug. 2019.

[3] M. Li, L. Zhu, and X. Lin, “Privacy-preserving traffic monitoring with false report filtering via fog-assisted vehicular crowdsensing,” *IEEE Trans. Serv. Comput.*, vol. 14, no. 6, pp. 1902–1913, Nov. 2021.

[4] E. Wang, Y. Yang, J. Wu, K. Lou, D. Luan, and H. Wang, “User recruitment system for efficient photo collection in mobile crowdsensing,” *IEEE Trans. Human-Mach. Syst.*, vol. 50, no. 1, pp. 1–12, Feb. 2020.

[5] X. Yan et al., “Verifiable, reliable, and privacy-preserving data aggregation in fog-assisted mobile crowdsensing,” *IEEE Internet Things J.*, vol. 8, no. 18, pp. 14127–14140, Sep. 2021.

[6] Z. Ning et al., “Blockchain-enabled intelligent transportation systems: A distributed crowdsensing framework,” *IEEE Trans. Mobile Comput.*, early access, May 13, 2021, doi: 10.1109/TMC.2021.3079984.

[7] Y. Hui et al., “BCC: Blockchain-based collaborative crowdsensing in autonomous vehicular networks,” *IEEE Internet Things J.*, vol. 9, no. 6, pp. 4518–4532, Mar. 2022.

[8] Q. Hu, Z. Wang, M. Xu, and X. Cheng, “Blockchain and federated edge learning for privacy-preserving mobile crowdsensing,” *IEEE Internet Things J.*, early access, Nov. 16, 2021, doi: 10.1109/JIOT.2021.3128155.

[9] F. Khan, A. U. Rehman, J. Zheng, M. A. Jan, and M. Alam, “Mobile crowdsensing: A survey on privacy-preservation, task management, assignment models, and incentives mechanisms,” *Future Gener. Comput. Syst.*, vol. 100, pp. 456–472, Nov. 2019.

[10] B. Zhao, S. Tang, X. Liu, and X. Zhang, “PACE: Privacy-preserving and quality-aware incentive mechanism for mobile crowdsensing,” *IEEE Trans. Mobile Comput.*, vol. 20, no. 5, pp. 1924–1939, May 2021.

[11] A. Singla and A. Krause, “Truthful incentives in crowdsourcing tasks using regret minimization mechanisms,” in *Proc. 22nd Int. Conf. World Wide Web*, 2013, pp. 1167–1178.

[12] 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.

[13] J. Hu, K. Yang, K. Wang, and K. Zhang, “A blockchain-based reward mechanism for mobile crowdsensing,” *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 1, pp. 178–191, Feb. 2020.

[14] M. Kadadha, H. Otrok, R. Mizouni, S. Singh, and A. Ouali, “Sensechain: A blockchain-based crowdsensing framework for multiple requesters and multiple workers,” *Future Gener. Comput. Syst.*, vol. 105, pp. 650–664, 2020.

[15] C. Cai, Y. Zheng, Y. Du, Z. Qin, and C. Wang, “Towards private, robust, and verifiable crowdsensing systems via public blockchains,” *IEEE Trans. Depend. Sec. Comput.*, vol. 18, no. 4, pp. 1893–1907, Jul./Aug. 2019.

[16] H. Duan, Y. Zheng, Y. Du, A. Zhou, C. Wang, and M. H. Au, “Aggregating crowd wisdom via blockchain: A private, correct, and robust realization,” in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom*, Mar. 2019, pp. 1–10.

[17] Y. Liang, Y. Li, and B.-S. Shin, “FairCs—Blockchain-based fair crowdsensing scheme using trusted execution environment,” *Sensors*, vol. 20, no. 11, p. 3172, Jun. 2020.

[18] M. H. U. Rehman, K. Salah, E. Damiani, and D. Svetinovic, “Towards blockchain-based reputation-aware federated learning,” in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Jul. 2020, pp. 183–188.

[19] K. Toyoda, J. Zhao, A. N. S. Zhang, and P. T. Mathiopoulos, “Blockchain-enabled federated learning with mechanism design,” *IEEE Access*, vol. 8, pp. 219744–219756, 2020.

[20] L. Pournajaf, L. Xiong, V. Sunderam, and S. Goryczka, “Spatial task assignment for crowd sensing with cloaked locations,” in *Proc. IEEE 15th Int. Conf. Mobile Data Manage.*, vol. 1, Jul. 2014, pp. 73–82.

[21] X. Wang, Z. Liu, X. Tian, X. Gan, Y. Guan, and X. Wang, “Incentivizing crowdsensing with location-privacy preserving,” *IEEE Trans. Wireless Commun.*, vol. 16, no. 10, pp. 6940–6952, Oct. 2017.

[22] Y. Zhang and C. L. P. Chen, “Secure heterogeneous data deduplication via fog-assisted mobile crowdsensing in 5G-enabled IIoT,” *IEEE Trans. Ind. Informat.*, vol. 18, no. 4, pp. 2849–2857, Apr. 2022.

[23] Y. Jiang, K. Zhang, Y. Qian, and L. Zhou, “P2AE: Preserving privacy, accuracy, and efficiency in location-dependent mobile crowdsensing,” *IEEE Trans. Mobile Comput.*, early access, Sep. 14, 2021, doi: 10.1109/TMC.2021.3112394.

[24] J. Nie, J. Luo, Z. Xiong, D. Niyato, P. Wang, and H. V. Poor, “A multi-leader multi-follower game-based analysis for incentive mechanisms in socially-aware mobile crowdsensing,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 1457–1471, Mar. 2021.

[25] L. Xiao, T. Chen, C. Xie, H. Dai, and V. Poor, “Mobile crowdsensing games in vehicular networks,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1535–1545, Feb. 2017.

[26] H. Tembine, “CrowdSensing games,” in *Proc. 29th Chin. Control Decis. Conf. (CCDC)*, May 2017, pp. 4660–4665.

[27] H. Gao, H. Xu, L. Zhang, and X. Zhou, “A differential game model for data utility and privacy-preserving in mobile crowdsensing,” *IEEE Access*, vol. 7, pp. 128526–128533, 2019.

[28] B. Cao, S. Xia, J. Han, and Y. Li, “A distributed game methodology for crowdsensing in uncertain wireless scenario,” *IEEE Trans. Mobile Comput.*, vol. 19, no. 1, pp. 15–28, Jan. 2020.

[29] J. Nie, J. Luo, Z. Xiong, D. Niyato, and P. Wang, “A Stackelberg game approach toward socially-aware incentive mechanisms for mobile crowdsensing,” *IEEE Trans. Wireless Commun.*, vol. 18, no. 1, pp. 724–738, Jan. 2018.

[30] I. Denisow, S. Zickau, F. Beierle, and A. Küpper, “Dynamic location information in attribute-based encryption schemes,” in *Proc. 9th Int. Conf. Next Gener. Mobile Appl., Services Technol.*, Sep. 2015, pp. 240–247.

[31] Y. Yu, L. Guo, X. Wang, and C. Liu, “Routing security scheme based on reputation evaluation in hierarchical ad hoc networks,” *Comput. Netw.*, vol. 54, no. 9, pp. 1460–1469, Jun. 2010.

[32] G. Ding, Q. Wu, and J. Wang, “Sensing confidence level-based joint spectrum and power allocation in cognitive radio networks,” *Wireless Pers. Commun.*, vol. 72, no. 1, pp. 283–298, Jan. 2013.

[33] J. B. Rosen, “Existence and uniqueness of equilibrium points for concave n-person games,” *Econometrica: J. Econ. Soc.*, pp. 520–534, 1965.

[34] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.