Trends for Mobile IoT Crowdsourcing Privacy and Security in the Big Data Era

Shabnam Sodagari, *Senior Member, IEEE*

***Abstract*—From tracking pandemics to applications, such as Google Maps, Uber, environmental monitoring, journalism, healthcare, crisis/disaster response, air quality control, noise and traffic monitoring, urban planning, *etc.*, mobile crowdsourcing systems are interweaved with the society and daily lives. This survey outlines major security and privacy challenges in MCS systems along with solutions and approaches. Comprehensive countermeasures, leveraging the capabilities of blockchains, smart contracts, machine learning, games, incentives, spatio-temporal cloaking, etc., are presented to preserve privacy and security of mobile workers, task requestors, and other aspects of crowdsourcing systems. Security recommendations for use cases, such as Industrial IoT, Internet of Vehicles, wireless crowdsensed systems, social crowdsourcing, edge-computing, personalized and privacy-preserving recommendation, and mobile worker recruitment are further elaborated.**

I. Introduction

Crowdsourcing systems collect large data volumes through mass sensing by pervasive smartphone users, cyber physical systems, social media, vehicles, human intelligence, etc. to solve complex tasks. Mobile crowdsourcing (MCS) finds applications in smart cities, pandemic monitoring, environment monitoring, health care, industrial IoT, smart homes, wearable devices, smart furniture, internet of vehicles, etc. [1]. It is an economical solution to circumvent the need for large core networks, especially in time-sensitive services. Crowdsourcing can even contribute to labor efficiency and mental health by asking users to report sensed noise pollution (loud music, vehicles, constructions, etc.) tagged with their locations to construct urban noise maps [2]. Patients' data collected by wearable devices are sent to remote medical servers. A MCS central server (trusted third-party) is the medium between requestors and workers to recruit workers to collect data for requestors. In other words, the server publishes spatio-temporal tasks of requesters to mobile workers, and mobile workers upload the collected data to the server once they complete the published tasks. For example, a requestor may need images of a particular scene. Mobile workers, who are closer to the scene, take pictures with their devices and send them to the requestor. Rewards managed by MCS server incentivize mobile workers to provide requestors with desired quality of experience (QoE). Upwork, freelancer, Amazon Mechanical Turk, marketing surveys to guide decision making [3], etc. are a few examples of commercial MCS platforms. Mobile crowdsourcing is enabled by the ubiquity of built-in sensors in mobile devices, such as cameras, microphones, accelerometers, GPS, etc. [4]. Workers have some degree of freedom to select their spatio-temporal tasks according to their resources, social interests, schedule, etc.

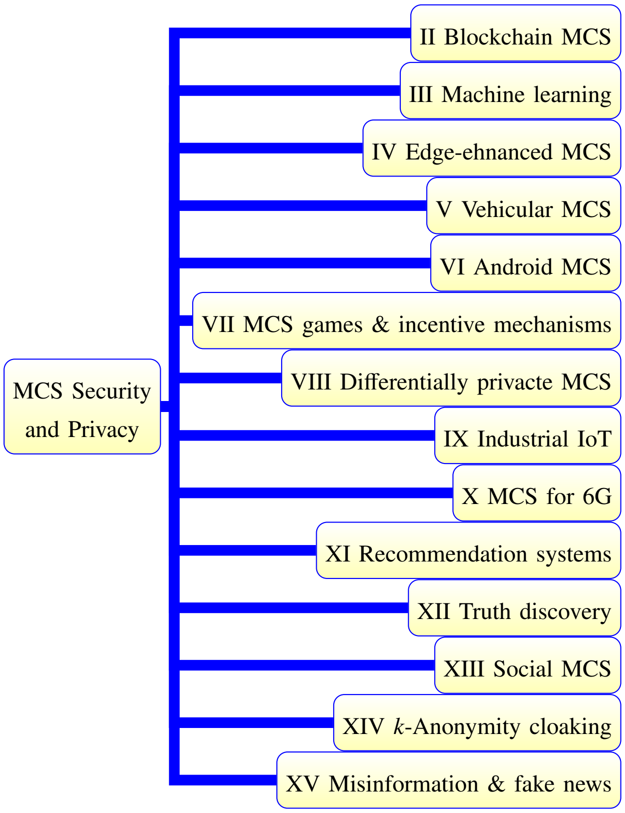

Fig. 1. Taxonomy chart for MCS Security and Privacy

A. Security and Privacy Risks Affecting MCS

Although crowdsourced data produce rich societal knowledge, they also cause unprecedented privacy and security threats to participants [5]. For example, single point of failure is a threat to the centralized MCS architecture [6]. Moreover, the data and private information are prone to malicious task requestors/workers [6] or servers (internal threat), and hackers (external threat). Security and privacy vulnerabilities can cause detrimental financial, emotional, and psychological damages to the citizens of smart cities. The resulting consumer mistrust may hinder social innovation, which degrades social welfare and motivation, thereby imposing social cost. Therefore, it is of utmost importance that the aggregated crowdsourced data

Shabnam Sodagari is with the Computer Engineering and Computer Science Department, California State University, Long Beach, CA 90840, USA, e-mail: shabnam@csulb.edu.

do not expose sensitive information, such as, daily routines, personal health records, visited locations, political views, etc.

1) Untrustworthy MCS Participants: Some types of security and privacy threats in mobile crowdsourcing systems can emerge from malicious or untrusted participants [7]–[10]. Major attacks of this type are [6,7]

- identity or reputation forging by some workers;
- gaining undesired access to data;
- publishing a task by a requestor without a reward;
- task requestor curiosity about the private information of MCS workers, during their communication;
- conflicting behavior attack; and
- collusion attack

In the conflicting behavior attack malicious participants provide partially correct and partially false information to mislead the crowdsourcing system. In the collusion attack malicious users collude to provide completely false information [7]. False reporting, free riding, and data non-trustworthiness are among the threats in large-scale MCS networks [11]. More specifically, false reporting means the task requester pays the worker after the execution of the task and lies about the quality of the work done by the MCS worker to reduce the payment of the worker. On the other hand, free-riding involves a case when the task requester pays the worker before the execution of the task, but the worker does not complete the task [11].

2) Unreliable MCS Server: Many crowd-sourced systems rely on a trusted server with access to raw data. However, the privacy of the user is at risk when the server cannot be trusted (internal attack), or is vulnerable to cyber-attacks and hacking, especially in real-time crowdsourcing [12]. In such cases, the server should neither gain access to raw private data of participants, nor used for data aggregation [12]. Instead, differential privacy mechanisms can be used before publishing the aggregated data.

To distinguish a qualified user for a certain task, the MCS server may take advantage of users' context and history, such as location, time, profile, and activity [13] in both offline and online task selection crowdsourced systems. Nevertheless, disclosure of participants data can jeopardize their privacy [8,11,13]. For example, the spatio-temporal data of MCS participants can be disclosed by having sensing data on mobile devices being tagged with personal information. The revealed spatio-temporal private information can cause leakage of identity, personal activities, political views, health status, etc. [8].

The organization of this survey is depicted in the taxonomy chart in Fig. 1 for Sections II to XIV. Finally, Section XVI concludes the survey.

Fig. 2 outlines some performance tradeoffs and metrics for MCS security and privacy [14]–[18]. Some of these metrics have negative correlations with others. Therefore, it is critical to strike a balance among various parameters, depending on the crowdsensing application. Moreover, protection of workers' privacy is not only to the benefit of workers, but also to the benefit of requestors since it incentivizes the workers to accomplish the tasks.

Next, we discuss how trustworthy MCS services can be offered via blockchains.

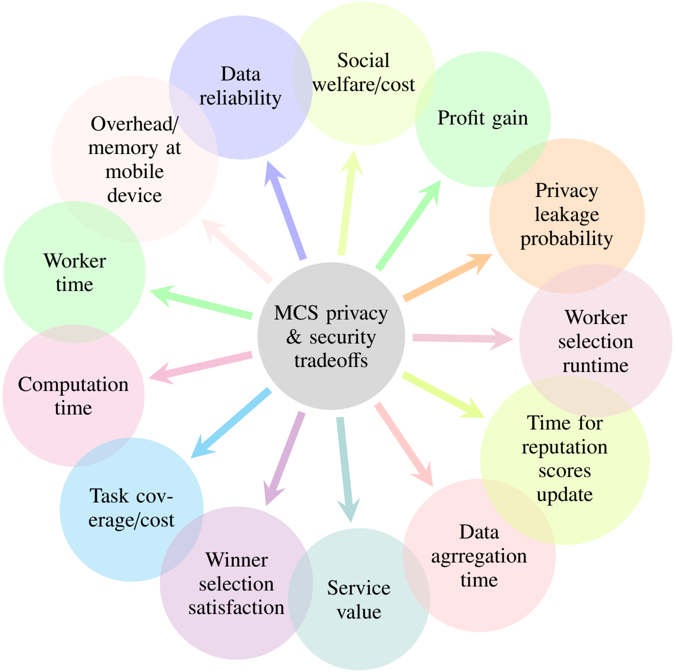
MCS privacy & security tradeoffs
Data reliability
Social welfare/cost
Profit gain
Privacy leakage probability
Worker selection runtime
Time for reputation scores update
Data agrregation time
Service value
Winner selection satisfaction
Task coverage/cost
Computation time
Worker time
Overhead/ memory at mobile device

Fig. 2. Performance tradeoffs and metrics for MCS security and privacy

II. Blockchain-Based Solutions for Security and Privacy of MCS Systems

Using the decentralized nature of blockchains and distributed ledger to impartially record MCS transactions removes the dependency on a centralized server. Blockchain nodes can rent out their computing resources to crowdsensing applications for information integrity against misbehaving participants and data aggregation verification. Compared with a traditional MySQL database, integration of MCS with blockchains for data storage and sharing provides higher security and reduces 1) the cost of a server authority to manage the communications between the requestors and workers [6], 2) vulnerability to a single point of failure (caused by a centralized server), and 3) vulnerability to external attacks, DDoS [19], DoS, eclipse attack, majority attack, device failure [20], etc.

MCS participants except the requester and the worker may record the services on the blockchain if they have enough power to generate the blocks. A recording MCS participant transmits the generated block to other participants to ensure data authenticity [21]. Blockchains detect data tampering, since any changes in the blocks are noticed by participants. Data in one block of a blockchain cannot be changed without causing all subsequent blocks to change. The blocks, which encapsulate the data are linked in order and the information is encrypted using the hash value of the former block, the requestor information, the worker information, etc. [21]. Blockchain offers transparency and integrity and is used for irrefutable identity verification, identity authentication, and secure tracking of digital identities. Ethereum is an open-source tamper-proof public ledger blockchain, shared by the participating IoT nodes. Some studies report [20] that the Ethereum python interface is more accessible than that of Hyperledger. This is due to Hyperledger being designed

TABLE I
MAJOR PERFORMANCE METRICS OF BLOCKCHAIN-BASED MCS

Ethereum gas cost [16]	Worker time [16]	Block generation run time [17]	Worker selection run time [17]	Task coverage [17]	Task cost [17]	Overhead and memory usage at the mobile device [14]	Privacy leakage probability

for storing confidential data (private consortium blockchain), while Ethereum is a public blockchain.

Nevertheless, as outlined in Table II, some attack types that can target the crowdsourcing blockchains are

- Sybil attack in which the attacker creates many accounts to gain majority in the blockchain consensus. As a countermeasure, the MCS consensus validators need to be chosen based on their trust/reputation scores [22].
- Collusion among requestors and miners, or among workers and miners. Reward and punishment mechanism can mitigate collusion by rewarding faithful nodes with service coins as a transaction fee [22]. Colluding nodes may be punished by lower trust or reputation values.
- Distributed Denial of Service (DDoS): Some nodes may flood the blockchain communication with sending too many transactions. To prevent DDoS messages, every message needs to be signed by its transmitter. Traffic cleansing devices, mandating a fee for each transaction [22], can defend against DDoS.

A. Miners in MCS

Miners can be additional participants in blockchain based MCS (not necessarily eliminating the server), but replacing the server to evaluate the quality of workers' task submissions. The role of miners is to verify transactions between workers and requestors and to compile them into blocks. As a security solution against an untrusted server, data is inspected by the miners, instead of being verified by the centralized controller [21]. Miners use an evaluation function provided by the requester to evaluate and confirm the quality of submitted work for task solution performed by the worker. This leads to measuring and updating reputation of workers, and more rewards are provided to better workers [19]. The expectation maximization algorithm and mutual information are used among miners. Some nodes in the blockchain are nominators, i.e., only responsible for receiving and broadcasting data sharing transaction requests. Compared to a miner, a nominator requires less computation resources [20].

The server publishes a task along with its quality evaluation criteria and deposits the reward on the blockchain. The deposit is to secure the reward for workers who finish the task according to the server's quality criteria. However, to eliminate server's free riding, the mobile workers do not submit their task directly to the server, but first to the peer-to-peer network for evaluation. Through a set of predefined smart contracts, the miners verify the identities of MCS participants and validate the sensing procedure before the reward allocation [6]. After validation, the task's data are given to the server while the corresponding hash digests are saved in the peer-to-peer network to remove any doubts about the miners' validation [25]. Workers whose work has met the criteria will receive their rewards from the server through extended bitcoin transaction syntax pricing in accordance with the reward transfer conditions in the transaction script. The transaction script further contains information about the workers who submit a task and their quality. Since this can jeopardize workers' privacy, the miners are assigned to evaluate tasks from a group of workers so that miners cannot distinguish the work of any individual in that group. To defend against *impersonation attacks by miners*, who may direct the payment to themselves, the transaction verification uses commutative encryptions. To avoid the need for hiring extra miners, workers can choose to become either miners or workers, but only one of these roles at a time [25].

B. Smart Contracts in MCS

Smart contracts are a mechanism for secure automation of the main functionalities of MCS. For example, task posting, worker selection, task allocation, task receiving, task execution, reward assignment, and reward payment [6] can be implemented by smart contracts (e.g., on Ethereum public test network [19]).

Three example smart contract types for MCS are: user register smart contract, user summary smart contract, and requester-worker relationship contract. Specifically, the evaluation of tasks can be performed via smart contract, as a countermeasure against free riding requestors and unreliable task solutions provided by unreliable workers.

Worker selection is automated by a smart contract to find workers for tasks published by requestors. This type of smart contract should reliably meet the interests of workers, the requirements of the tasks, data confidentiality, and identity anonymity. When a task has multiple requestors, session key for each task can preserve confidentiality [16]. Inverted index for encrypted keyword search in the smart contract is a computationally efficient method to protect anonymity and the privacy of task requirements and workers' preferences [16]. In this scheme:

- The task requester uploads the task through a one-task-only blockchain address to prevent revealing the identity.
- The worker retrieves the task by querying the read-only function of the blockchain without revealing the query or identity.
- The query cannot be linked to the worker identity by searchable encryption in the smart contract.

C. Encryption in Blockchain-based MCS

The transparent public blockchains lack privacy. Moreover, malicious MCS participants may misbehave by providing in-

TABLE II
Overview of Blockchain-Based Crowdsourcing Privacy and Security

Problem	Solution
Maintain anonymity of blockchain-based MCS mobile users	Elliptic curve algorithm to protect the user identity [14]; User registration without true identity and storing encrypted work in the distributed storage [19]
Impersonation, data tampering, repudiation, and denial	Twice consensus on the blockchain for node selection using fuzzy theories by calculating the reputation degree and matching degree of each node [14]
Validation and block creation of blockchain by the same set of nodes	Separation of transaction validation and block recording between two different groups [22]
Protection of user attribute on the public blockchain from association attack/background knowledge attack	Lightweight homomorphic encryption [14]
Aggregation of sensing results in the blockchain without exposing original sensing value/sensitive information, e.g., daily routines, personal health, locations, etc.	Additive secret sharing by breaking the requester's secret [23]
Lost shares in additive secret sharing due to mobile workers leaving or becoming offline before finishing the task	Delegation to other workers based on local and global reputation scores [23]; Breaking the secret share of the abandoned worker among delegate workers
Data quality and reliability evaluation	Reputation evaluation based on data distortion, data consistency, local rating, and contextual factors [23]
Disclosure of a requester's feedback about a worker on the public blockchain	Two-stage local/global reputation by updating and publishing the reputation scores only after a group of requestors have provided their feedback [23]
DDoS, sybil, collusion, false reporting, and free riding attacks (malicious workers)	Payment of deposit before participation [19]; Allowing the consensus amongst a trusted subgroup of nodes to represent the consensus of the nodes in the whole network [22] (Proposed consensus implemented on Windows 8 with an i7 Intel Core CPU and 32GB RAM)
Subjective third party task evaluation prevention; Transparent monetization	Smart contracts [19]; Automatic reward delivery on the blockchain for crowdsourced knowledge monetization (implemented on Amazon AWS EC2 cloud and Ethereum [24])
Trust in off-chain data	Data quality estimation based on EM algorithm to learn the actual task data from the data submitted by MCS workers [15]
Miners violating participants' privacy by impersonation attacks	Node grouping for *k*-anonymity; Secure incentive mechanism [25]
Eclipse attack, majority attack, terminal device failure, and transaction forgery	Certificate authority to review authenticity and data ownership and manage the public keys of mobile terminals based on the assumption of a private chain owning honest mining nodes [20] (implemented on Geth 1.7.2 (Go Ethereum), on the Ubuntu 16.04 LTS with Intel Core 3.40 GHz i7-6700 CPU and 16 GB RAM)
Privacy during task matching including: workers' preferences, identity anonymity of workers/requesters/task requirements, and reliable task matching according to workers' preferences	Blockchain to store the encrypted index; Combining smart contracts with searchable encryption on Ethereum (with metrics of gas and worker time costs) [16]

valid data or invalid data aggregation. Asymmetric encryption in the digital signature allows the blockchain to verify if a transaction was signed by the correct private key [26], thereby protecting MCS participants against external attacks [19]. The task requester provides the public key to workers to encrypt their aggregated data to prevent plagiarism on the open transparent blockchain. MCS participants use their private key for their signature. To broadcast a task, the requester posts its digital signature and public key and generates a hash digest of all task requirements. However, this requires the requestor to be trusted [19].

Users in a decentralized blockchain-based MCS can register without their true identity (to preserve their privacy) and store encrypted solutions in the distributed storage [19]. To thwart DDoS, Sybil and false-reporting security attacks, each identity is required to deposit a fee before participation, which could be negligible compared to high service fees demanded by a centralized server.

Another application of a cryptocurrency built on blockchains is to securely motivate MCS workers. High quality skilled workers are rewarded with their payments that are recorded in transaction blocks. *k*-anonymity privacy is achieved through a node cooperation verification [25]. The node cooperation-based privacy mechanism hides private information in a group to deal with the impersonation attacks in the open and transparent blockchain. Workers and requestors can use signcryption as a privacy countermeasure against disclosure to the miners. Signcryption is faster than the sequential signature and encryption.

The smart contract will automatically select qualified workers among the pre-registered workers and sends back the deposit to non-selected workers. After completing the task, the workers store their signed encrypted data in the distributed database (along with their public key), waiting for the evaluation of the task requester. After the evaluation stage, the qualified workers receive their payment. The challenge is the design of a fair evaluation mechanism, which prevents free-riding attack by requestors. To measure the run times of block generation, worker selection, task coverage, and task cost, this NP-hard optimization problem has been implemented on Ethereum using Python 3.5 on 2.60 GHz Core(TM) i7-6700HQ CPU, 20.00-GB, Windows10 64bit [17].

In the above schemes, recording public keys on the blockchain and encrypting the data under each recorded public key incurs on-chain storage overhead and monetary costs, e.g., on Ethereum. Moreover, the size of the uploaded ciphertexts increases with the number of MCS participants [24]. Since the storage space on the blockchain is limited, an off-chain distributed database may be used as an accessory [17]. Due to the cost of on-chain processing, it is uneconomical for all the workload to be on-chain. Hence, a balance must be maintained between on-chain and off-chain deployments [27]. Turing-complete programming language depicts complex crowdsourcing logics and increases crowdsourcing flexibility.

Workers may obtain their public and private keys by registering by a certificate authority (CA), for data privacy in the public Ethereum-based mobile crowdsourcing. The public key can identify the user in blockchain, and the signature cannot be forged without the user's private key [23]. Workers encrypt data with their private key before sending it to the CA. The CA checks whether the received data is from a previously registered worker. The role of the trusted key manager [16] (or CA [20]) is protection against data authenticity and ownership forgery, caused by the open nature of Ethereum. After verification, the signature of CA is sent back to worker, before the worker stores its encrypted data and CA's signature verification on the blockchain [20]. The network ID, system mining difficulty and gas limit are among the parameters that need to be setup in the genesis block. Operations such as block query, transactions, and mining can be simulated via JavaScript Console launched by Ethereum nodes. The smart contract may be implemented in the Solidity language, which includes initialization of the MCS workers' accounts, their storage function, etc. [20].

Since crowdsensed data on the blockchain is encrypted for privacy, verifying the truthfulness of this data in the cloud can be done by independent servers, through lightweight cryptographic additive secret sharing [24]. More specifically, workers encrypt executed tasks by splitting the data into secret shares. Each cloud server receives one share without knowing the shares of other servers. Then, the cloud servers process the data using the weight of each worker. MCS data confidentiality implies that the data is not revealed to MCS participants and cloud servers, and it is revealed to the requestor only after the completion of monetization.

After the cloud servers verify the reliability against falsified data, the blockchain comes into operation to automate a tamper-proof MCS monetization. The blockchain nodes can form a secret-managing committee of trustees. Specifically, the smart contract records the identities of requesters who have paid for the crowdsourced data. Then, a group of Ethereum blockchain nodes securely deliver the crowdsourced data to authorized requestors. This scheme is implemented on Amazon AWS EC2 cloud and Ethereum blockchain [24]. Nevertheless, the premise of additive secret sharing is that servers do not collude to share secrets with each other. Otherwise, this method fails to preserve privacy in the cloud. In a fair marketplace for selling the data produced by MCS workers to requestors, the workers are rewarded once the cloud servers verify the truth about data against falsified information.

D. Consensus Protocols in Blockchain MCS

A major challenge in adopting blockchains in MCS is the design of a suitable consensus protocol. The Bitcoin's Proof-of-Work (PoW) [28] mining suffers low throughput and demands resources [22]. Paxos-based and Byzantine Fault Tolerance family of algorithms suffer scalability issues with larger numbers of IoT crowdsourcing participants and might not handle all types of unfaithful behaviors. The Proof-of-Stake (PoS) lacks fairness by giving the easier mining puzzles to richer participants [22]. Another drawback of PoS is that the block generators have nothing to lose by voting for multiple blockchain histories, which may cause the consensus to never resolve. Charging a cost for working on multiple chains can be a remedy.

On the contrary, the Proof-of-Trust (PoT) consensus [22] balances between security and fairness by assigning the transaction validation and block recording to two different sets of blockchain nodes. This contrasts with a traditional blockchain, where the validation and the block creation of the consensus are conducted by the same set of nodes. Its architecture includes a private consortium, the members of which are the crowdsourcing site operator, regulators, notaries, etc. Each consortium member has a consortium ledger management node and a gateway node. The ledger management nodes are responsible for selecting validators from the open crowdsourcing platform. The gateway nodes isolate private information of the private consortium from the Internet while communicating to validators.

The validators in the public crowdsourcing platform (selected by the ledger management nodes residing in the private consortium) send their majority consensus on the transactions to the gateway. The gateway, then, passes the consensus to the ledger management nodes who vote on the transactions chosen by the majority of the validators as another layer of trust. The leader of the ledger management nodes (selected by raft leader selection method) creates the transaction block and broadcasts it to the consortium blockchain [22].

The PoT consensus [22] selects transaction validators based on their trust values to mitigate the sybil attack, collusion, or DDoS. Dual ledgers and dual consensus protocols integrate a public chain, running the delegated proof of stake consensus, and subchains, running the practical Byzantine fault tolerance consensus. As such, they outperform Casper consensus in Ethereum by achieving higher transaction throughput and less execution time compared with traditional PoW/PoS-based blockchain [29].

E. MCS Workers' Location Privacy on the Public Blockchain

At the worker selection stage, the location of MCS workers may be revealed. The worker's account may contain the travel limitations for location-based crowdsourcing tasks [17]. A spatial anonymous area around the true location (called cloaked area) can preserve the location privacy in MCS [17].

A task requester and a worker anonymously pre-register by uploading their information and submitting a deposit to create their accounts on the blockchain to post tasks. The deposit to sign a smart contract with the task requester prevents sybil attacks.

F. MCS Reputation Management on the Blockchain

Reputation management against MCS participants who provide false data can also be left to the blockchain that replaces a malicious centralized server.

To protect private information of MCS participants (sensing data, aggregation result, requester's feedback, etc.) the blockchain and edge computing are integrated on the Hyperledger Sawtooth and Android client [23]. To eliminate inaccurate ratings by malicious requestors, the two main steps of reputation management consist of:

- local reputation evaluation in which the data reliability is evaluated by the requesters' positive or negative rating of the worker's data; and
- global reputation update in which the global reputation scores are updated by the smart contract, based on the average of the ratings from requesters.

Here, the role of the blockchain public ledger is to transparently store the global reputation of MCS workers. However, any new rating of a worker's reputation submitted by a requestor leaks the variation in reputation in a public manner, which violates anonymity. One solution is to refrain from updating the global reputation on the blockchain until after at least a batch of requestors submit their reputation scores. Furthermore, additive secret sharing for global reputation update on the blockchain protects the reputation scores from being revealed [23]. Nevertheless, additive secret sharing is challenging when some mobile workers leave or become off-line before they complete the sensing task. Due to the lost shares, the secret key cannot be reconstructed. As a solution, the sensing task of an off-line worker is delegated to a set of on-line workers, which are selected according to their local or global reputation scores.

G. Future Research Directions

Blockchain technology, still in its early stages, is suffering from limitations, such as

- computational costs of mining;
- recursive calling vulnerability, timestamp dependence, arithmetic problem, and return value problem [20];
- computationally intensive consensus mechanisms [6];
- collusions involving miners, e.g., between an anonymity group and miners, miners and the server, and MCS participants and miners;
- distributed denial-of-service attacks and theft of content [26]; and
- increased latency with the number of IoT nodes in blockchain MCS.

The liveness guarantee of the consensus protocol (due to Fischer, Lynch, and Paterson or FLP impossibility) [22], consensus deadlock, and applications of game theory still need more investigation. Evaluation functions provided by the requester need to be revisited, since, the requester may not know about the solution characteristics beforehand.

III. Machine Learning Based Privacy/Security Solutions for MCS

Table III outlines contemporary approaches in security of machine learning based mobile crowdsourcing. As a defense against fake sensing, provided to the MCS server, by selfish workers' smartphones, the server needs to encourage high quality sensing and discourage fake sensing attacks. In this regard, the interactions between the MCS server and MCS workers are modeled as a Stackelberg game. Here, the server is the lead player that determines and broadcasts its payment policy for each sensing accuracy [30]. Each worker is a follower that chooses the sensing effort/accuracy to receive the payment (based on the payment policy) and the sensing accuracy estimated by the server. The conditions to motivate accurate sensing affect the Stackelberg equilibria. Moreover,

TABLE III
Machine Learning for MCS Security and Privacy

Problem	Solution	Evaluation metrics
Fake sensing results provided to MCS server	Deep Q-network reinforcement learning using convolutional neural networks to learn optimal payment policy [30]	Sensing quality, attack rate, and server utility
Incentivizing MCS workers despite lack of system model	Deep reinforcement learning to derive optimal strategies in a payment-privacy protection game [31]	Utility of MCS workers, utility of MCS server, and data aggregation accuracy
• Privacy of machine learning training data provided by MCS workers • Attacker reversely inferring training data from the classification model	Differential privacy combined with deep neural networks [32]; Injection of noise to the affine transformation of the input data features	Predictive and classification accuracies tested on US census data
Location privacy in crowdsourcing intelligent transportation systems	Obfuscation of spatiotemporal data with obfuscation coefficients [33]; Requestors use EM algorithm to estimate the task results from the obfuscated uncertain worker locations	Location entropy, crowdsourcing results accuracy
Privacy during training for feature learning using crowdsourced big data on the cloud	BGV encryption (homomorphic) to encrypt the private training data on the cloud [34]	Classification accuracy and training efficiency tested on the Animal-20 and NUS-WIDE0-14 datasets [34]

when the worker sensing models are not known in a dynamic MCS game, deep Q-network reinforcement learning can be used to derive the optimal MCS policy against fake sensing results. To achieve the optimal payment policy, the deep Q-network reinforcement learning is implemented using a convolutional neural network with a high-dimensional state space and action set.

Lack of enough payment and data privacy leakage are two obstacles discouraging workers from participating in crowdsourcing tasks. To motivate workers, each MCS worker may be allowed to submit its sensing data with a specified payment-privacy protection level before the MCS server can select a corresponding payment to the worker [31]. This is a game for which the Nash equilibrium needs to be derived. Nevertheless, to derive the optimal strategies in this game, it is combined with deep reinforcement learning (Q-learning or deep Q network) to overcome the lack of knowledge about the workers' dynamic payment protection level [31].

When the crowdsourced data, collected from MCS workers, are fed to machine learning models for classification or prediction purposes, the workers' private data may be exposed in this process. In the mobile crowdsourcing edge computing IoT, i.e., mobile crowdsourcing combined with edge computing, k-anonymity algorithm can protect the users' privacy in the random forest classification of crowdsourced data [35]. To achieve differential privacy in deep neural networks, intentional noise is injected to the affine transformation of the input data features [32]. The importance of data features related to target categories is estimated so that less noise is injected to the more important features. To accommodate the heterogeneous feature values, coefficients of injected noise are adaptively selected [32]. BGV, which is a fully homomorphic encryption, is another method for privacy protection of crowdsourced training data on the cloud [34]. Four core operations of BGV include encryption, decryption, secure addition, and secure product. BGV supports the addition and multiplication operations on ciphertexts without bootstrapping.

Spatiotemporal crowdsourced data in intelligent transportation systems exposes locations of workers. Hiding a worker's location negatively affects the quality of spatial crowdsourcing tasks. However, obfuscation arithmetic with appropriate obfuscation coefficients for workers' space and time provides both worker privacy and MCS task quality [33]. Since the location of the worker is uncertain from the viewpoint of the requester, the requester needs to use the machine learning expectation maximization (EM) algorithm to derive the maximum likelihood estimate of the task results from the obfuscated uncertain locations [33].

A. Future Research Directions

An area of further exploration is the optimal deep reinforcement learning-based payment policies for different applications of crowdsensing, e.g., traffic data aggregation with malicious workers [30]. In addition, noise injection into data features needs to be implemented and tested on various neural network architectures intended for different types of crowdsourced data aggregations [32]. Moreover, novel location obfuscation mechanisms need to be designed against adversaries that launch their attacks based on the probability of workers' locations [33].

IV. The Edge of MCS Security!

Edge/fog computing in MCS relieves the time delay and high bandwidth costs of centralized cloud computing by shifting the computational resources closer to the edge of the network, resulting in location awareness and mobility support [36]. Since fog nodes are closer to the network edge than the cloud server, they can more accurately allocate MCS tasks to suitable workers. Fog-assisted confidential data deduplication eliminates extra communication overhead between fog nodes and the cloud. Moreover, blockchain-based solutions for integrity can be implemented in an edge computing environment.

Registration of participants by a trusted third party to receive the keys may alleviate external attacks on MCS systems, since external attackers have not received the keys [37]. Even if external attackers intercept and replay messages of legitimate entities, they will not be able to pass the time-slot based hash message. The stages of a privacy-preserving fog-assisted spatial crowdsensing are as follows [37]: 1) Each task requestor anonymously (via a pseudonym) sends the encrypted task (using bilinear pairing and homomorphic encryption) and task requirements, such as aggregation types and obfuscated sensing area to the MCS server. 2) The MCS server assigns tasks to the fog nodes based on sensing area. 3) Fog nodes decrypt the task and announce it to the local mobile workers. 4) A worker interested in a task requests the credentials of task authorization from the MCS server (e.g., task secrets). 5) After task completion, each worker hides the sensed data with a random number and encrypts data with its public key and sends it to fog nodes. 6) Fog nodes authenticate and verify the integrity of received encrypted data and compute the encrypted aggregation with the MCS server using the additive homomorphic property. 7) MCS server attaches its digital signature to the encrypted result (integrity) and sends it to the task requestor who then decrypts it.

Table IV provides an overview of the roles of fog nodes in private and secure MCS. The MCS server should not be able to link the content and the identity of each task requestor. For privacy-preserving data aggregation, secure multi-party aggregation using statistics, such as sum, mean, variance, and minimum on encrypted data are adopted [37].

Edge-enhanced context-aware task allocation enables real-time crowdsensing. The role of the cloud in task allocation may include evaluating the workers reputations (based on background information, task context, historical feedbacks, rewards, etc.) and sending a subset of most eligible workers to the edge. Then, the edge optimally allocates the tasks based on task requirements (such as maximizing the sensing coverage under the constraint of the task budget) and workers real-time information [40]. Nevertheless, in the reputation update process, the privacy of both requesters and workers need to be protected. The pool of MCS workers needs to be large enough, to allow for differential privacy to be effective.

Data encryption to protect sensing data may make it more challenging to calculate the reputation of malicious workers. As a countermeasure, the reputation value may be updated based on the deviations of the encrypted sensing data from the final collected truth. Nevertheless, this method reveals the exact deviation value of each worker to the reputation manager and may also expose the final collected truth through collusion with some MCS participants [8]. Instead, the reputation may be updated by considering the rank of deviations.

Fog nodes can detect and remove replicate data (without accessing the content of crowdsensed data) using BLS-oblivious pseudo-random functions, thereby protecting the privacy of MCS workers [39]. To hide identities of workers during data collection (anonymity) Chameleon hash function enables MCS participants to anonymously claim their contributions for reward and identify greedy workers that want to receive their reward more than once. Fog nodes detect identical sensing data while not learning about the data itself by using BLS signature to generate the encryption key of sensing data. The key-homomorphic signature to sign the sensing data enables fog nodes to aggregate the signatures of MCS participants, to let the MCS server know about the contributions of the workers who generate replicate crowdsensed data [39].

The collusion scenarios in a fog assisted MCS may include [38]: 1) Some workers and the cloud MCS server sharing their keys and collected crowdsourced data with each other; 2) collusion among workers and fog nodes; 3) requesters and cloud collusion; and 4) requesters and fog nodes collusion. Methods involving one-way hash chains, marked mix-nets and grouping-based secure searchable encryption are used to thwart the above collusions as follows [38]: • The worker sends data containing a pseudo-identity, a multiencryption of the message, and a hash to the fog node. • The fog node partially decrypts and shuffles the encrypted message to make ciphertexts unlinkable before sending to the cloud server. • A task requester sends a request, containing a pseudo-identity, a group of encrypted indices, and a signature to a fog node. • The fog node sends the received above request to the cloud.

Next section discusses fog-assisted trustworthy vehicular MCS.

V. Trustworthy Vehicular MCS

An important element of smart transportation, autonomous vehicles, and vehicular social networks is the Internet of Vehicles (IoV) [15]. Some applications of vehicular crowdsensing include [41]:

- road surface monitoring, location and route quality, requiring continuous data collection and repeated sensing;
- real-time map updates for autonomous vehicles;
- road events, e.g., accidents and traffic congestion;
- social applications like BikeNet; and
- safety-related and emergency applications (delay sensitive).

Due to delay sensitivity of IoV, the vehicular crowdsensing is interwoven with edge/fog-enhanced crowdsourcing. The fog nodes to connect the cloud to end users can be installed on road side units (RSUs) to communicate with vehicles, e.g., through 5/6G systems. The RSU fog nodes extend the cloud services to the edge, whereas the cloud servers store the history to be utilized later [42]. Another choice for fog nodes includes buses, since they are distributed over urban areas, they are closer to vehicles than RSUs, but they move slower than

TABLE IV
Secure Edge/Fog Assisted MCS

Goal	Approach	Challenges/Evaluation metrics
Privacy-preserving and collusion-resistant aggregation of MCS workers' data and responding to requests with certificate authority	One-way hash chains, marked mix-nets, and grouping-based secure searchable encryption [38]	Mutual information for privacy quantification; computational cost and communication overhead
Elimination of duplicate sensed data collected in MCS without exposing data content	BLS-oblivious pseudorandom function to enable fog nodes to detect and remove replicate data [39]	Communication overhead
Privacy of MCS workers in task allocation (fog to workers) and confidentiality of task content in data aggregation	Bilinear pairing and homomorphic encryption by supporting statistics, such as sum, mean, variance, minimum, etc. [37]	Computation/encryption cost at each MCS participant
Differentially private real time task allocation with the help of edge nodes	Binary tree noise aggregation to reduce the distortions induced by Laplace mechanism for differential privacy [40]	Regret comparison with online learning algorithms; balance between privacy and accuracy

vehicles [43]. MCS privacy serves as a motivation to vehicles that are usually reluctant to participate [41].

Table V provides a summary of Vehicular MCS security and privacy.

A. Secure Fog-Based Vehicular MCS

The importance of fog/edge-enhanced MCS in highly mobile vehicular IoV with a need for location awareness is undeniable. Certificateless aggregate signcryption is a technique to enhance information confidentiality, mutual authenticity, and anonymity in fog-based road monitoring [42,47].

In case of compromised RSUs (fog nodes) and/or vehicles, it is important that the key generation center does not have the MCS participants full private keys (key escrow resilience) [42].

Filtering out false crowdsensed traffic monitoring data can be handled by data fusion strategies based upon worker vehicles' reputation values [43]. Nevertheless, filtering out such false data becomes particularly challenging when the data is encrypted. One countermeasure is to construct a weighted proximity graph at each fog node through range query in Wi-Fi handshaking [47].

Anonymous privacy-preserving messages hinder authentication protocols, which is especially dangerous to driver safety in vehicular MCS. Using fog nodes (RSUs) to establish mutual authentication with vehicles prevents the man in the middle attacks [36].

Although incentive mechanisms comprise an important part of crowdsourcing systems, they may lead to privacy leakage [43]. For example, in reverse auctions, when a MCS worker selects a specific task, its preferences will be revealed. Therefore, the identity of MCS workers should not be linked to their task preferences. To this end, in the token reward generation the unique token identification of a worker is processed using partially blind signature and the zero-knowledge authentication techniques before validation by the fog node. In addition, the worker encrypts the identification code into the crowdsensed data using the public key, so that the fog node cannot track any vehicle [43].

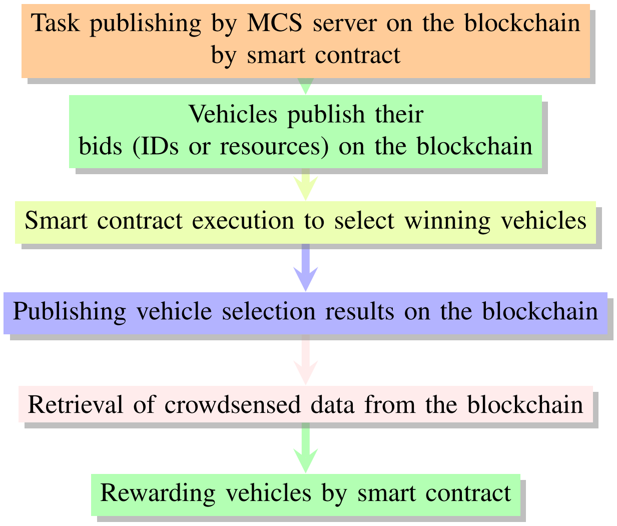
Task publishing by MCS server on the blockchain by smart contract
Vehicles publish their bids (IDs or resources) on the blockchain
Smart contract execution to select winning vehicles
Publishing vehicle selection results on the blockchain
Retrieval of crowdsensed data from the blockchain
Rewarding vehicles by smart contract

Fig. 3. Vehicular crowdsensing on the blockchain

Malicious MCS workers endanger drivers' safety by providing fraudulent and fake data. As a defense, vehicles need to evaluate the credibility of the senders of crowdsourced data, e.g., through a collaborative crowdsourcing-based reputation scheme [45]. Homomorphic encryption and secure multiparty computation help task requestor vehicles hide their reputation rating/feedback along with the list of MCS workers that provided service to them.

To ensure fairness in computing the aggregate reputation of MCS workers, each task requestor vehicle's provided feedback is given a weight proportional to the trust value of that requestor [45]. To preserve anonymity, the individual reputation ranking of each requestor should not be revealed, but rather

TABLE V
Vehicular MCS Security and Privacy

Goal	Approach	Challenges/Evaluation metrics
Vehicles motivation to provide real-time map updates	Reverse auction incentive mechanism [18]	Limited service platform budget; limited vehicles' resources
Security of incentive mechanism and privacy of vehicles	Partially blind signature for secure pseudonym management [18]	Data reliability; task quality of vehicle MCS workers
Privacy of incentive mechanism in vehicular MCS	Using buses as fog nodes along with partially blind signature authentication, homomorphic encryption, zero-knowledge verification, and one-way hashing [43]	Feedback delay; reward exchange efficiency
Secure distribution of rewards among MCS worker vehicles	blockchain-based payment; Blockchain credit management in which vehicles attach importance to the tasks [18]	value of services; profit gain
Secure information exchange between vehicles and IoT center for emergency (e.g., ambulance carrying a patient) vehicles collaboration	Bidding and incentive mechanisms [41]	Time; success rate
Trust in off-chain data in blockchain-based vehicular MCS [41]	Truthful and individually rational reverse auction; data quality evaluation	Social welfare; algorithm scaling with increased number of tasks/mobile nodes; social cost; computational complexity
Vehicles' data confidentiality, integrity, mutual authentication, anonymity, and key escrow resilience	Certificateless aggregate signcryption with fog computing [42]; Identity-based proxy re-encryption (using random oracle) to ensure security of shared content [44]	Computational cost and communication overhead
Secure vehicular MCS in-network cache content sharing & distribution with no private information leakage	Use of a trusted content-centric network controller [44]; Name function combined with encryption	Varying wireless channel conditions; Dynamic network topology
No trusted third party against malicious task requestors	Decentralized aggregation in which reputation scores given to MCS workers by the requestors is weighted by the reputation of the requestors [45]	Communication and computation overheads
Correlation attack against MCS vehicular social networks	Analyzing the correlation function to determine the sensitive data for suppression [46]	Average processing time; identity/location privacy; information loss

the aggregate reputation of vehicles must be provided.

The wireless channel variations and the high speeds of vehicles make the identity and location privacy protection of vehicles a challenging task. Correlation attacks, such as temporal correlation, spatial correlation, and data correlation disclose the privacy of participants in crowdsensing-based vehicular social networks [46]. To thwart inference of private information through correlation attacks, mobile management layer and core control layer are added to vehicular MCS systems with three stages including group generation (for *k*-anonymity), identity management, and location suppression [46]. The defense includes a tradeoff between correlation suppression strategies and data loss. This requires analyzing the correlation function and then determining which information are more sensitive to be suppressed.

B. Blockchain-Based Vehicular MCS

Blockchains and smart contracts are used in MCS IoV for providing trust and incentives in data sharing among vehicles [41] in addition to secure bidding. Fig. 3 shows the flow of IoV MCS operation using smart contracts and the blockchain. The traffic congestion MCS task can be automated by the blockchain, e.g., the communications between the vehicles and the IoT center for bidding, payment, and scheduling [41]. All vehicles need to be registered on the blockchain. Each vehicle will offer a bidding price and its data quality, according to task quality metrics published by the IoT center on the blockchain.

Blockchain is a viable solution for trust in on-chain data, whereas data quality-driven reverse auctions incentives can ensure trust in off-chain data for the blockchain-based MCS reverse auctions. In a reverse auction [18], the IoT center is an auctioneer that purchases data from MCS workers. The auction-based mechanism encourages high-quality sensing data. Another aspect of trust in off-chain data is the data quality evaluation, e.g., via expectation maximization.

To filter out unqualified vehicles from the MCS task and high rewards, a credit threshold can be managed by the blockchain. This motivates vehicles to improve their credibility by providing high quality crowdsensing data. Moreover, secure pseudonym management based on the partially blind signature can protect the privacy of vehicle MCS workers [18]. The steps of a reputation based privacy preserving vehicular MCS [44] may include: 1) encryption of the data of each vehicle MCS worker with the ID and parameters of a trusted content-centric networking controller (key generator); 2) collection of encrypted data of vehicles by the task requester; 3) sending the collected crowdsensed data and the desired name function (to calculate the reputation of MCS workers) to the network controller; 4) decrypting the data by the trusted controller who calculates the reputation of each vehicle worker based on the named function; and 5) sending the calculated reputation of vehicles by the controller back to the task requester.

C. Sybil Attacks on Crowdsensed IoV

Anonymity in crowdsensing can make grounds for sybil attacks. Crowdsourcing in IoV enables real-time updates about the road map, traffic, etc. (e.g., Waze). Sybil attacks on such networks can emerge from location anonymity. Even a resource-limited device can divert other cars by creating virtual software-based vehicles (ghost riders [48]) that report false congestion and accidents, without being detected. As a mitigation, proximity graphs form a web of trust by combining co-location edges and authentication to attest to the one-time physical interaction of a pair of devices. Nevertheless, the formation of such graphs requires a certain amount of time.

Pseudonyms, which are based upon public key infrastructure (PKI) or group signatures, do not scale well with the growing number of IoT devices [49], since the CA cannot handle large numbers of devices. As a remedy, users must be enabled to self-generate an unlimited number of unlinkable pseudonyms, to perform multiple crowdsourcing tasks, without triggering periodic pseudonyms or key renewal from the CA. Nevertheless, to prevent sybil attacks, users should not be allowed to use these pseudonyms for the same task. To achieve this type of pseudonymity, sybil attacks on any MCS system (not only IoV) can be detected during task subscription by identity based cryptography, i.e., identity-based signature authentication [49,50].

D. Future Research Directions

Due to the varying wireless channel conditions and high mobility in the IoV, machine learning methods can be used to predict optimal incentives and fair reward distribution strategies in vehicular MCS.

Another challenge that requires further investigations is when RSU fog nodes are compromised. Since RSU fog nodes send their computational results (collected from potentially false reporting vehicles) to the cloud, the cloud needs mechanisms against untrusted computation results. In general, since fog nodes are closer to MCS workers compared to remote cloud servers, it is more challenging to preserve location privacy of workers against untrusted fog nodes.

When speed and location of vehicles need to remain private, the vehicles' count may be used to acquire traffic conditions. However, this approach may expose the trajectory of vehicles [47], and thus, requires further analysis.

VI. Crowdsourced Security for Android Users

Crowdsourcing can be used as a security solution for Android users with untrusted apps that require permission to access users' data [51]. In an Android user-help-user crowdsourced system, a user with more expertise can help others by suggesting whether to accept or deny a permission to their data when an Android application requests access to user information. However, it is challenging to find and expand the network of reliable experts with sufficient level of expertise that will not provide malicious guidance to other users [51]. In this regard, a rating system (called DroidNet) can identify reliable experts in the network. The reliable experts provide accept/deny permission suggestion to other android application users. The solution starts by forming a small group of internal seed experts. The seed experts define a minimum set of permissions that are necessary for the application functionality. DroidNet expands the network of reliable experts by finding similarities among a common set of permissions by seed experts and other users. These similarities determine the expertise level of each user. The rating of a regular user is iteratively updated by comparing permission suggestions of the user with that of seed experts or other users with a known level of expertise.

VII. Game Theory and Incentive Mechanisms for MCS Security

Table VI summarizes major usage of game theory and incentive mechanisms for the protection of MCS systems.

A. MCS Games

Game theory can trigger cooperation among selfish users. The interactions between the requestor and a worker can be modeled as an iterative two-player prisoners dilemma game. Rewarding/penalizing a worker's cooperation/non-cooperation incentivizes cooperative behavior. In this regard, the requestor changes the expected payoff of the worker based on zero-determinant strategies, i.e., offering the worker more short-term payoff without sacrificing the long-term interest of the requestor. However, a free-riding requestor may penalize a cooperative worker to gain profit. In such a case, the worker needs to adopt an evolutionary strategy to enforce fair cooperation in an iterative prisoners dilemma game [54]. Solutions

TABLE VI
Game Theory and Truthful Incentive Mechanisms for MCS Privacy and Security

Problem	Solution
Profit maximization among MCS players, i.e., monthly-paid workers, task requestors, and instant-paid workers	Design of three two-player Stackelberg subgames and using backward induction to find the equilibria of Stackelberg games [6]
Maximizing the social welfare under lack of cooperation and myopic equilibrium	Two-sided rating with differential punishments of MCS users with different ratings [52]
Balancing individual privacy and data utility under time-varying privacy demands	Noncooperative differential game and dynamic programming to obtain Nash equilibrium [53]
• Reducing privacy leakage of MCS data uploading and trading • Balancing data accuracy and privacy sensitivity, i.e., cumulative difference between the prior and the posterior probabilities	• Three party games among 1) mobile workers, 2) edge nodes, and 3) cloud MCS server; • Game tree to observe the behavioral strategies of players and to model payoff functions [35]
Free-riding and false reporting attacks	Evolutionary game theory [54] for reputation update [11]
Mobile worker privacy in individually rational and truthful incentive mechanisms for crowdsourced indoor localization using continuous time-varying Wi-Fi signals (as opposed to one-shot sensing)	• Two-stage Stackelberg game to maximize mobile worker utility and MCS server profit • Differential privacy with joint optimization of the variable reward for mobile workers (when MCS server knows each mobile user's sensitivity level of data privacy); A demand function model for the relationships among MCS server, mobile workers, and service customers [55] • *Age of data* freshness metric [56] (the time passed since the last generation of the data) to determine the rewards in the reverse auction incentive mechanism
Sybil attacks in auction-based MCS under the assumption of no monopoly mobile worker	Breaking the bid from each mobile user into atomic bids for one task only and paying $(n+1)-$th smallest value to the n winning bids [57];
Simultaneous MCS data trustworthiness, user privacy, and incentive fairness without a trusted third party	Anonymous trust/reputation and cryptographic protocols to enable benign MCS users to request tasks, contribute data, and earn rewards anonymously without data linkability [58]
Bid privacy against inference attack in VCG (Vickrey Clarke Groves) auction-based MCS for worker recruitment and optimal task assignment, while minimizing social cost	• Secure group bidding for disguising the bids within the groups • Lagrange polynomial interpolation to perturb workers' bids within groups [9] • Differentially private exponential mechanism to limit the impact of a worker's bid change on the auction outcome [59]
Reputation update in ciphertext domain	Adding a separate reputation manager to the crowdsensing system; somewhat homomorphic encryption (SHE) and DGK comparison protocol to rank the reputations [8]

based on evolutionary game theory for reputation update can be used against free-riding and false reporting [11]. In this framework, the interaction between worker and requester is modeled as an asymmetric gift-giving game, with different strategy sets for task requestors and workers.

The three subgame Stackelberg game is an incentive mechanism, through which the MCS participants can calculate the strategies that will maximize their profit (profit evaluation smart contract) [6]. The Stackelberg game is based on a leader and a follower. The first movers in the Stackelberg game crowdsourcing market are monthly-paid workers who dominate the market. When the subgame has Nash equilibrium, the follower's strategy is optimal, given the leader's strategy. Followers play the game with strategies that are informed by the strategy and preferences of the leader, until reaching an optimum. In the first subgame, the leader is the monthly-paid MCS worker, while the follower is the task requestor. In the second subgame, the leader is the task requester, and the followers are the instant-paid workers. In the third noncooperative subgame, the players are instant-paid MCS workers. Backward induction finds the equilibria of the subgames [6].

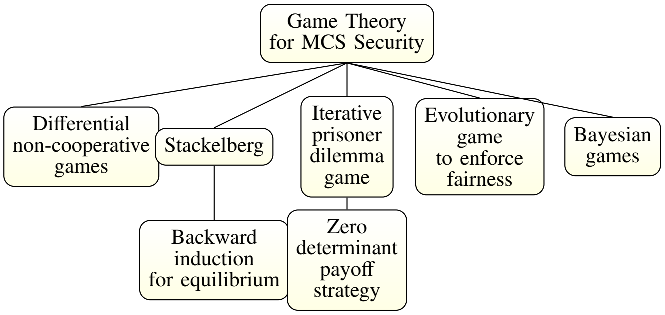
Game Theory for MCS Security
Differential non-cooperative games
Stackelberg
Iterative prisoner dilemma game
Evolutionary game to enforce fairness
Bayesian games
Backward induction for equilibrium
Zero determinant payoff strategy

Fig. 4. Game theory methods for MCS Security

A noncooperative differential game model can balance the tradeoff between contradictory goals of individual privacy and ensuring data utility under time-varying users' privacy demands. The feedback Nash equilibrium, wherein MCS participants and the server achieve maximum privacy and data utility may be obtained by dynamic programming [53]. However, lack of cooperation among self-interested users causes a service exchange dilemma with zero social welfare obtained at myopic equilibrium [52]. Two-stage game model for two-sided rating can overcome the inefficiency of the socially undesirable equilibrium and maximize the social welfare by the optimal choice of parameters that affect users' behaviors and users' valuation of their long-term utilities [52]. To address anonymous MCS requestors with asymmetric service requirements and workers' different service capabilities, the game theoretic two-sided rating with differential punishment applies different punishments for MCS users who have different ratings. To this end, strategy recommendation selects a desirable behavior from predefined plans, while participants update their ratings according to a rating update rule.

B. Robust Incentive Mechanisms for MCS

Mobile IoT crowdsourcing systems are a type of sensory data market [6]. Similar to any market, they need to be fair and provide incentive mechanisms. Incentive mechanism motivate the participation of mobile workers in crowdsourcing tasks. The three main elements in Fig. 5, on major incentive mechanisms for establishing trust in MCS, can be combined to increase reliability.

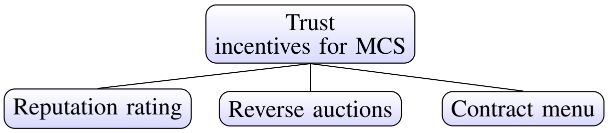

Fig. 5. Major incentive mechanisms toward trustworthy MCS

1) Reputation Schemes: Malicious participants can be detected by a reputation mechanism, by grouping participants, and constantly updating the trust value. The reputation of each participant is updated based on the deviation of the participant's result from the aggregated result or an evaluation function provided by the task requestor [8,11]. This deviation calculation can be performed in the encrypted domain by considering the difference between participant's encrypted result and the encrypted aggregated result [8]. To overcome free-riding and false-reporting attacks, truthful evolutionary games can inform the design of intelligent reputation update incentive mechanisms by predicting the evolution of MCS systems [11]. As a motivation for workers to provide truthful data, various reputation thresholds are dynamically adjusted for various tasks, to avoid excessive punishment of potential truthful workers [7].

To enable the simultaneous goals of incentive fairness, data trustworthiness, and MCS user privacy, anonymous trust or reputation models allow legitimate users to join tasks anonymously and obtain rewards while preventing malicious users. To hinder the abuse of anonymity, the number of issued pseudonyms should be limited [58].

Updating the reputation of a mobile worker, based on the deviations of its sensed data from the aggregated results, exposes the deviation to the MCS server. To conceal this deviation a separate reputation manager is added to the crowdsensing system and somewhat homomorphic encryption (SHE) is used, which allows processing of ciphertext without having to decrypt it [8]. To protect the actual deviations from colluding nodes: 1) The task requestor asks the certificate authority (CA) to generate public and private keys. 2) The task request and the public key are sent to the cloud MCS server to be distributed to mobile workers (via edge nodes). 3) Mobile workers send their encrypted sensing results to edge nodes. 4) The edge nodes calculate the encrypted deviations and apply the DGK comparison protocol for the reputation manager to obtain the rank of the deviations. 5) The reputation manager uses the rank of deviations to update the reputations of workers.

2) Auctions for Truthfulness in Mobile Crowdsourcing: Auctions and payment determination are tools to motivate truthful behavior among MCS participants by minimizing the cost incurred upon task requestors and maximizing the benefit of workers. Auctions need to be individually rational, budget-balanced, and computationally efficient. Threats against auction based MCS systems include [57,58]:

- false-name or sybil attacks on auctions to increase the utilities of malicious users;
- collusion attacks;
- bid privacy leakage of smartphone users by chosen plaintext attack; and
- location privacy leakage of bidding mobile workers.

As a countermeasure against sybil attacks, auctions need to be designed in a way that each user is better off not generating any false names [57]. Core selecting-based incentive mechanisms integrate auctions into MCS, to incentivize truthfulness [7]. Vickrey Clarke Groves (VCG) auctions are used to provide the maximum gain for users who reveal the real value of their bid based on their location and time [9]. MCS participants are assumed to be only aware of their own bid and not of the other bidders (no collusion). To incentivize truthful bids, the reward is the second lowest bid in VCG auctions.

3) Reverse Auction for MCS Task Assignment: Since crowdsourcing tasks demand consumption of workers' device

This article has been accepted for publication in IEEE Transactions on Technology and Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TTS.2022.3191515

resources, selfish MCS workers prefer to save their limited resources (e.g., energy, storage, and computation capacity) rather than exhausting them on the assigned task. Incentive methods based on reverse auction game motivate MCS workers to complete a crowdsourced task [10]. The reverse auction task assignment needs to incorporate truthfulness, individual rationality, efficiency in computation/communication, and privacy protection for workers. However, the winning bid selection in the reverse auction task assignment can be a NP-hard problem, e.g., when it is modeled as a n-to-one weighted bipartite graph matching with multiple binary knapsack constraints. As a remedy, approximation algorithms are used to select winning bids and determine payments [60].

Task assignment for crowdsourcing needs to address the conflicting goals of maximizing the utility of the MCS platform, minimizing the cost of requesters, satisfaction of requestors and workers, truthfulness, individual rationality, and stability. Game theory deals with the satisfaction of both workers and requestor, whereas intelligent matching results in stability [61]. The steps toward a stable task assignment and matching include [61]: 1) The MCS server asks the requesters and workers, separately, to describe their individual requirements and preferences using a multi-attribute information structure. 2) The MCS server collects the multi-attribute information and reward for the tasks from requestors and workers, respectively. 3) Tasks are assigned to mobile workers based on matching the multi-attribute structures of both parties. 4) The integrated satisfaction degrees of both requestors and workers with respect to their published multiple attributes are calculated. 5) The satisfaction degree is used for pricing and payment through a second-price reverse auction mechanism.

4) Bid Privacy in Auction-Based MCS: The bids of MCS participants may include private information, e.g., spatial locations and routes [9], leading to inference attacks. Preserving the bid privacy of smartphone MCS workers while minimizing the social cost in auction-based incentive crowdsourcing can be achieved by semantic disguise of each user bidding for a set of desired tasks followed by selecting users based on score functions (e.g., linear and log functions) [59]. The bid disguise guarantees cipher-text indistinguishably against chosen-plain-text attack, since no statistical information can be gained from the bid. To this end, the key generator distributes a series of polynomial outcomes and IDs to mobile workers, which they use to disguise their original bid value. A third party data aggregator, then, calculates the minimum bid among all participants' bids without knowing each participant's actual bid value [9].

5) Incentives via Contract Menu: When the MCS server is not trusted, MCS workers prefer to randomly add calibrated noise to their sensing data before reporting [62], to preserve their privacy. This, in turn, degrades the quality of task completion. In such cases, contracts can characterize the workers' equilibrium behavior toward maximizing the accuracy of data aggregation under budget constraints. A contract menu offers different contracts to MCS workers to accommodate their varying privacy levels [62], without each worker's precise privacy preference being revealed to the MCS sever. The contract menu, designed with incomplete information, determines the payment to MCS workers to incentivize them to give up some of their differential privacy. The incomplete information contract leverages the probability distribution of worker' privacy preferences to quantify the impact of each worker's privacy on task accuracy.

C. Future Research Directions

Although monetary rewards are encouraging incentives for workers, they cannot prevent denial of payment (free riding) attacks by malicious task requestors [63]. Even worse, malicious workers may launch data pollution, sybil, and replacement attacks. As a countermeasure, data reliability must be measured based on the deviation of the data from ground truth, while preserving the anonymity and location privacy of mobile workers.

Some areas of further exploration include [11,35,52]–[54,59,61,63]–[65]:

- Taking into account the capability variance of task participants into the design of the privacy-preserving incentive mechanisms;
- Device to device or peer to peer communication to save the cost and energy in the spatial coverage of crowdsourced fingerprint collection;
- Investigating the tradeoff between privacy and task quality to incentivize mobile workers;
- Exploring various score functions (other than linear and log) to achieve differential privacy and proximate social cost minimization;
- Utility-maximizing incentive mechanisms combined with novel data aggregation and data perturbation against linking attacks on privacy;
- Extension of two-sided rating protocols from two-level rating to multiple levels, as well as optimal selection of the size of the rating labels and the threshold value for users' punishment or reward;
- Reinforcement learning for the optimal continuous rating labels/continuous MCS server actions;
- Study of the game models on the blockchain, for privacy-preserving intelligent matching between task requesters and mobile workers;
- Repeated game behaviors to capture multiple rounds of interaction among MCS entities in the design of the payoff function, and measuring the risk of data privacy leakage;
- Extension of multi-attribute task assignment to the case when each task can be completed by more than one worker, and each requestor has more than one task at a given time;
- Mitigation of the impacts of collusion among mobile workers on the cooperation probabilities of both the task requestor and the workers in zero-determinant-based incentive mechanisms for multiple-player games.

VIII. Differential Privacy

Table VII refers to differential privacy approaches for MCS.

TABLE VII
MCS Systems Based on Differential Privacy

Goal	Approach	Evaluation metrics
MCS worker selection with differentially-private worker location data	Probabilistic mechanism to minimize the travel distance by selecting workers with the largest probability of being closest to tasks [4]; Vickrey payment mechanism by considering movement cost & privacy level	Ttruthfulness; profitability; probabilistic individual rationality
Maximizing differentially private MCS revenue under a limited budget, while achieving truthfulness	Multi-armed bandit dynamic pricing [66]	Regret; revenue; privacy leakage; individual rationality
Privacy-preserving multi-dimensional joint distribution estimation for data mining from high-dimensional crowdsourced data	Finding correlations among multiple attributes to reduce the dimensionality of crowdsourced data; expectation maximization & Lasso regression for faster distribution estimation; randomized response applied to Bloom filter data features [5]	Computation & communication overhead; data utility; estimation speed; accuracy benchmarked with support vector machines & random forest classification; Mean square error & Jensen-Shannon divergence in crowdsensed data mining [67]; Mutual information and L_2-norm error evaluated on crowdsensed radiation levels in 23 Tokyo wards with 100,000 participants [3]
Reducing the loss of MCS task performance due to obfuscated locations and inference attack without a trusted third party	Distortion privacy combined with differential privacy to bound the expected location inference error against inference attack; learning the function to fit the sensing data to obfuscated locations [68];	Data quality loss; computation time; approximation ratio on traffic monitoring data
Original data feature preservation in differentially-private randomized crowdsourced data collection	Supervised machine learning for predicting randomized data	Accuracy & true positives on MHEALTH dataset [69]
Leakage of prior knowledge to adversaries through correlation in crowdsourced data	Adding noise to aggregated data for differential privacy and applying randomized perturbation to remove data correlation [70]	Privacy leakage evaluated on National Long-Term Case Study (NLTCS) dataset from StatLib, and search logs of Google Trends data

A. Differentially Private Location Obfuscation of MCS Participants

Crowdsourced data, such as road monitoring, indoor floor plan reconstruction, and smart transportation are tagged with locations [71,72]. Even sensing data can be linked to users' locations. Location-privacy incentivizes users to sense data. Differential privacy is particularly useful for location privacy of MCS participants in the presence of an untrusted server. Differential privacy removes the need for any trusted third-party [73]. On one side, knowledge of the precise locations of MCS participants helps the centralized server with the optimal task allocation to minimize MCS workers' travel distances to the task location [73,74]. On the other side, MCS workers risk their location privacy when uploading sensed data tagged with their actual positions, especially in sparse MCS systems. Differential privacy provides an upper bound on the information of third parties regardless of their prior knowledge. Under the assumption of prior knowledge, distortion privacy ensures that the expected inference error is larger than a threshold [68]. To this end, MCS workers are partitioned on the basis of worker density and non-uniform worker distribution [74]. In geo-obfuscation, users obfuscate their reported locations under the guarantee of differential and distortion privacy [73]. In the next step, tasks are allocated based on geocast region selection methods that balance workers' travel distances, system overhead, the number of workers that are notified of available tasks, and the success rate of task assignment [74]. This involves a mixed-integer non-linear optimization, which can be solved by Benders decomposition and genetic algorithms [73].

In a personalized approach, each worker uploads the personal privacy level/budget to the server along with obfuscated distances to tasks [4]. When MCS workers use obfuscated information for differential privacy, the worker selection mechanism for task allocation becomes probabilistic to minimize the cost of the total travel distance. As such, each task is allocated to the worker who has the largest probability of being closest to it [4]. After the worker selection, selected workers are paid through the Vickrey payment mechanism satisfying truthfulness, profitability, and probabilistic individual rationality, based on movement cost and privacy leakage.

Another countermeasure against an untrusted MCS server includes deploying multiple distributed agents to replace the untrusted server. These distributed agents aggregate and perturb the crowdsensed results (e.g., by adding Laplace perturbation) to ensure data privacy [12]. The agents then upload the results to the MCS server. This scheme relies on the assumption that the server and distributed agents cannot collude with each other. A MCS participant can randomly select an agent at each time and upload its information to that agent through onion routing for anonymous connection [12].

Since location obfuscation may degrade the task performance quality in MCS, an optimization solution needs to be devised to find the trade-off between efficiency of assigning tasks and privacy of MCS users. For example, linear programming can learn the optimal location obfuscation function by fitting the original sensing data to the obfuscated location [68]. The linear program constraints include differential-privacy, distortion-privacy, and evenly-distributed obfuscation. MCS participants can determine the extent to which their information and context can be shared with the MCS server. Based on the shared information, the MCS server decides if the user should be recruited [13].

B. Integration of Differential Privacy with Mechanism Design for Mobile Crowdsourcing

While auctions aim to maximize the revenue and guarantee truthfulness, differential privacy protects the privacy in the data collection phase. Integration of auctions with differentially private data pricing/collection prevents the leakage of identity of requestors to workers and vice versa [75]. Moreover, differentially-private mechanism design enables each MCS participant to hide its cost [66], as the MCS participants and the server each aim to maximize their benefits with constrained device resources. When workers send differentially private obfuscated costs (distances to tasks), the server takes a probabilistic approach for cost-efficient worker selection and for individually-rational payment mechanisms [76]. For example, in Wi-Fi-based indoor localization to construct radio maps, incentive mechanisms are combined with differential privacy [55]. The interaction between MCS workers and the server is a two-stage Stackelberg game to maximize the utility of both workers and the server.

Mobile workers are able to hide their exact locations if they form groups and select a group master to collect and report their differentially private and randomized sensing reports. The selection process of the group master leans itself to a non-cooperative Bayesian game model and the Bayes Nash equilibrium. The utility of the players (i.e., MCS workers) is their privacy gain [72]. Due to the inherent noise in differentially-private MCS, the notion of auction truthfulness is replaced with expected truthfulness [66].

C. Differential Privacy for Data Mining by Statistical Inference in Crowdsourced Data Aggregation

Both differential privacy and cryptography fail to protect the aggregated statistics over crowdsourced data. For example, the true number of crowdsourcing participants in an area can still be accurately exposed to an untrusted server. Thus, the two data types that need to be protected are [67] 1) qualitative or categorical data (e.g., screen deployment), and 2) quantitative data (e.g., location data, discrete meter readings, ordinal preference options, etc.). Local differential privacy sanitizes each mobile user's data on the mobile device to preserve privacy in distribution estimation over crowdsourced data [67]. For discrete distribution estimation from categorical crowdsourced data, a k-subset mechanism with mutual information metric is used, whereas for discrete quantitative data (with any distance metric, e.g., L_2 norm), an extension of k-subset mechanism, as a variant of the exponential mechanism, is used to tackle the asymmetry in data. Both these cases aim to minimize the mean distribution estimation error [67].

Randomized response, which involves sending the disguised value of a sensed category, preserves privacy even when the MCS server and all but one MCS participants collude [3]. In randomized response-based MCS, each MCS participant generates a Bloom filter, using multiple hash functions to encode data into a unique bit string. Then, each bit is perturbed, before being sent to MCS server, according to the outcomes of a certain number of coin flips [69]. For example, a MCS worker sends either the true data with probability p or the disguised data with probability $1-p$. The MCS server analyzes the statistics of crowdsensed data in lieu of precise information from sensing results. Randomized response anonymization can be extended to generate multiple disguised values (instead of one) from a single sensing result [3]. The MCS server generates an estimated contingency or cross tabulation table (multi-dimensional histogram) for distribution estimation. Under ϵ-differential privacy constraint, the MCS server minimizes the expected values of utility metrics, such as mean squared error or Jensen-Shannon divergence that measure the difference between the contingency table and the estimated distribution.

Nonetheless, randomized response methods require a lot of samples to converge to an estimation [3]. Another privacy-preserving method in data mining over crowdsourced data is the generation of a synthetic dataset with similar statistical distributions as the original crowdsourced data. To this end, marginal and multi-dimensional distributions are learned from the data after dimensionality and sparsity reduction [5]. Particularly, expectation maximization and Lasso regression estimate the joint distributions and correlations among attributes while MCS participants cloak their original data by applying randomized response on Bloom filter data records. Once the probability distribution is estimated from crowdsensed data, 1) correlated attributes are identified by measuring the mutual information and split into low-dimensional heterogeneous clusters to form an undirected dependency graph; 2) the attributes are split according to pruning the junction tree built from the dependency graph; and 3) the low-dimensional datasets are sampled according to the connectivity of attribute clusters and the estimated distributions on each attribute cluster to synthesize an approximate dataset that replaces the original crowdsourced data to preserve privacy [5].

D. Future Research Directions

Extension of the existing differentially-private MCS models for homogeneous task values to heterogeneous crowdsourcing tasks [66]; shortening the run-time of current methods;

designing online schemes without the assumption of prior knowledge about the number of users; continuous-pricing instead of discrete pricing; evaluation of the existing methods on other datasets; and extensions of differentially-private location information to databases of trajectories of MCS participants positions [3] are among areas that need to be further explored.

IX. Reliable Industrial IoT Using Crowdsourcing

MCS enables industrial and e-healthcare services through Industrial IoT (IIoT) [77,78]. Fig. 6 and Table VIII summarize the elements and solutions for the protection of crowdsourced industrial IoT. A major goal in battery limited MCS-based IIoT is to avoid computationally expensive schemes, such as elliptic curve point multiplication or modular exponentiation operation [78]. To this end, lightweight chaotic-map-based multifactor user authentication (i.e., smart card, password, biometrics, etc.) allows remote access or storage of medical data to authorized users (e.g., healthcare personnel or patients in E-healthcare crowdsourced IoT).

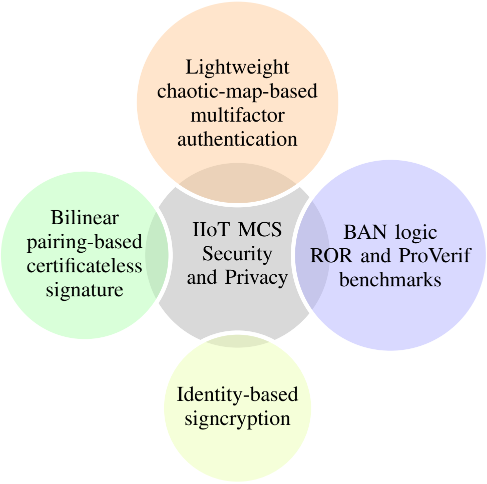
Lightweight chaotic-map-based multifactor authentication
Bilinear pairing-based certificateless signature
IIoT MCS Security and Privacy
BAN logic ROR and ProVerif benchmarks
Identity-based signcryption

Fig. 6. Elements of crowdsourced IIoT protection

Another security issue in MCS based IIoT arises from outsourcing the data to the cloud to reduce the costs of data management, sharing, and computation [79,80]. For data authentication in the cloud, without the need for a key-escrow, certificateless signature (CLS) can be used as an identity-based signature technique. Nevertheless, the computational cost of cryptographic operations along with the probabilistic nature of map-to-point hash function and random oracle model render the CLS scheme impractical for IIoT due to device storage and bandwidth constraints. Instead, bilinear pairing identity based signcryption circumvents the need for map-to-point hash function and random oracle model [79] and enables both authentication and confidentiality of crowdsourced data under limited storage and low-bandwidth IIoT conditions [80]. The computation cost of authenticated IIoT data creation and verification can be further reduced by discarding the pairing computation. Common benchmarks to verify the security of remote authentication schemes in MCS IIoT include: real-or-random (ROR) model (for the verification of key security) [78] and Burrows-Abadi-Needham (BAN) logic (for the verification of the security of mutual authentication between a user and the MCS server).

A. Future Research Directions

Research directions for lightweight CLS methods on low-power MCS IIoT devices include: 1) Further reduction of the pairing overhead of bilinear pairing for signcrypted IIoT [80], 2) Design of identity-based CLS schemes with revocation capability, and 3) Extension of existing authentication schemes for cloud-assisted MCS IIoT to multi-server environments [1].

X. Security in 6G and Wi-Fi Communications Leveraging Mobile Crowdsensing

Mobile crowdsensing measurements can be used to update real-time databases of spectrum white spaces for dynamic spectrum sharing (DSS). This gives rise to several issues, such as

- spectrum misuse by some secondary users [82];
- location privacy of crowdsensing mobile devices [82]; and
- outlier sensing results [86].

The use of mobile crowdsensing in wireless systems is not limited to spectrum sensing in search of white spaces, but also the numerous cellular user equipments (UEs), e.g., mobile phones, can measure various communication parameters, e.g., signal to interference plus noise ratio (SINR), throughput, delay, etc. Nevertheless, there is little knowledge and control on the timing, locations (e.g., indoor, outdoor) or the cell load conditions under which the ubiquitous mobile UEs perform their measurements [86].

A. Untrusted Spectrum Service Provider

The spectrum service provider needs to know the location of mobile users to assign them a spectrum sensing task. Nevertheless, locations of mobile nodes that participate in the spectrum sensing task need to be protected [87] from an untrusted spectrum service provider. To this end, the dynamic spectrum sharing crowdsensing needs a trusted Cellular Service Provider (CSP) that is a reliable third party between sensing mobile nodes and the Spectrum Service Provider [87]. CSP uses differential privacy by spatial decomposition of collected mobile locations. CSP partitions the area into regions and adds noise to the number of users' location data points in each region. Only after adding noise, the CSP shares the results with the spectrum service provider.

k-anonymity grouping enables each UE in MCS spectrum sensing to hide its private information (e.g., location) from an untrusted spectrum database administrator. Here, the sensing results of the whole group of k UEs are collectively reported, as opposed to individual reporting.

B. Untrusted Spectrum Sensing UEs

Even with a fully trusted spectrum database administrator, the participating UEs in the sensing task may be curious to infer the private information of each other [85]. In the k-anonymity grouping mentioned above, there may be some malicious UEs among the group. As a countermeasure, each

This article has been accepted for publication in IEEE Transactions on Technology and Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TTS.2022.3191515

TABLE VIII
MCS Industrial IoT Privacy and Security

Problem	Approach	Challenges/Evaluation metrics
Data authenticity and untrustworthiness of third parties in cloud-assisted IIoT [81]	Bilinear pairing-based certificateless signature without map-to-point hash function and random oracle model [79,81]	Communication bandwidth and IoT device storage constraints
Signature forgery attacks, public key replacement attacks, and malicious passive third parties	Elliptic curve partial private key generation into the short certificateless signature scheme and key exchange in the partial private key generation to enable crowdsourced IIoT communication over public channels [81]	Robustness against chosen-message attacks; computation and storage
Personalized privacy measurement in MCS IIoT with utility maximization for MCS participants	Combination of game theoretical rational uploading strategies and encryption [77]	Balance between task quality and privacy; real-time data confidentiality and integrity
Impersonation attack by obtaining the server's master key	Direct authentication of the user by the server, using the authentication factor instead of the server authenticating the secret key stored at the user's mobile device; all authentication factors (e.g., biometrics, password, etc.) acting as part of the secret key and participating in the authentication and key agreement [1]	Time consumption on 5/6G IoT devices; smaller key size
Lightweight remote user authentication in e-healthcare IoT	Multifactor authentication based on extended chaotic maps verified on ProVerif verification tool [78]; verification of key security, using the real-or-random (ROR) model and verification of the security of mutual authentication between a user and the medical server using Burrows-Abadi-Needham (BAN) logic	Low communication and computational overhead for battery limited healthcare devices

UE needs a grouping rule to ensure joining a reliable group of UEs. For example, a sensing UE can form groups within its social network or location proximity [84]. Moreover, the desire for gaining more payment may prevent some UEs from revealing their true cost of spectrum sensing to the database administrator. Reverse auction is a truthful incentive mechanism to ensure that no UE in the MCS spectrum sensing obtains profit by reporting a higher cost.

After UEs register with the database administrator, each UE will be given a unique pseudonym, shared with other UEs. To show unbiased selection, the spectrum service provider announces the spectrum-sensing auction results by using the unique pseudonym of winning UEs [85]. If the cost of task performance is revealed, curious UEs may be able to infer private information of other UEs by observing the changes in the auction results, e.g., if the cost is related to parameters, such as the spatial distance between UEs locations and the sensing task. This is specially important when task collaboration among spectrum sensing UEs in a crowdsensed database is enabled by device to device (D2D) communication. D2D pairing needs to be established in a secure, autonomous, transparent, and fast manner. To this end, the trustworthiness of participating devices need to be estimated in real time to enable each device to establish a secure connection with the most trustworthy neighboring device [83].

Once UEs register with the base station (database administrator or spectrum service provider), to obtain a verifiable pseudonym, their trust value will be updated after each D2D pairing, based on their performance history.

There are several ways for attackers to compromise D2D pairing [83]:

1) Attacks targeting the connection phase;
   - Eavesdropping of the D2D communication between two legitimate UEs by a third party UE;
   - Impersonation of a legitimate UE's credentials to fool other UEs toward establishing connection;
   - An adversary may come in the middle of two legitimate UEs, without being noticed. In the man-in-the-middle attack the two legitimate UEs mistakenly think they are connecting to each other, whereas their connection is actually via the adversary;
   - A UE forges its trust value to mislead other UEs about its trustworthiness;
   - A malicious UE always or sometimes rates its connection experience with trustworthy UEs as a negative one, to undermine other UEs' trust value;
   - Collusive attack in which a subset of malicious UEs rate each other to be trusted, while they rate other UEs outside their collusion to be untrusted.
2) Attacks on the traffic (e.g., false data injection), after the connection is established.

Some countermeasures require any two of registered D2D UEs to negotiate a shared key. The process involves certificate-

TABLE IX
Security and Privacy for MCS-Based 6G Communications

Problem	Approach
Spectrum misuse detection	Mobile crowdsensing workers authenticate secondary user by verifying and decoding spectrum permits embedded in physical-layer signals [82].
Trust and transparency in collaborative D2D MCS spectrum sensing against passive eavesdropping, impersonating, man-in-the-middle, trust forging, collusion, and independent negative attacks [83]	• Shared secret key for device pairing in the initial encounter • Updating the trustworthiness of a device using Gompertz function reputation mapping based on history of transmission delay, data rate, packet loss, etc. [83]
Location privacy of mobile workers in crowdsourced spectrum sensing	• Location-based *k*-anonymity mobile grouping [84] • Differential privacy with the objective of minimum social cost [85]
Incentives to encourage accuracy in crowdsourced spectrum sensing for white spaces (or received signal strength fingerprinting), while minimizing requestors' payment	Monetary and social motivation via truthful reverse auction-based [64,85] winning group selection [84], considering individual rationality and energy consumption by associating each sensor's true spectrum sensing cost to its current location

less public key cryptography for a unique private-public key pair for each registered UE [83]. The base station (spectrum database administrator or service provider) will not be able to restore such shared keys, owing to a private share exclusively being held by the UE only.

The spectrum service provider can use reverse auctions for sensor selection with the goal of running truthful crowdsourcing to minimize its payment. Even if the claimed cost (tied to distance) of each participant is hidden, still some location proximity information of devices may be inferred when winners for various published tasks at different rounds of reverse auction are revealed [85]. In MCS applications where each sensor's true cost for spectrum sensing is related to its location, differential location privacy is a viable solution.

C. Spectrum Permit Verification by Crowdsensing

To prevent spectrum misuse in DSS through secondary user authentication, the transmitter is required to embed a spectrum permit into its physical-layer signals, which can be decoded and verified by ubiquitous crowdsensing mobile users [82]. The spectrum permit could be embedded in various ways, including the modification of original constellation points to higher and lower power levels, or a higher-order constellation than the original one at the same transmission-power level.

D. Trust in MCS-Based Wi-Fi Sharing

Crowdsourcing is an economic way to enlarge Wi-Fi coverage area by allowing individual owners of private home Wi-Fi access points to share their Wi-Fi access points with each other. For example, Fon is an example of a shared Wi-Fi network with millions of Wi-Fi spots around the world. To resolve the issue of trust among unfamiliar access point owners, contracts that are designed by network operators incentivize Wi-Fi crowdsourcing. However, contracts face the challenge of incomplete information emerging from private mobility patterns and private Wi-Fi access quality of owners [65]. Thus, the contract needs to elicit truthful information from users, via pricing and revenue incentives, to maximize the network operator's profit. More specifically, each contract item has a Wi-Fi access price and a subscription fee. The access price is the amount that each user can charge other users who use its access point. The subscription fee is the amount that a user needs to pay the operator for joining the network of crowdsourced Wi-Fi [65]. However, the choices of each user are tightly coupled to its privacy and the choices of other users. Game theory models help the network operator with pricing and revenue contract design by finding the equilibrium choices of all users.

Received signal strength (RSS) fingerprinting can be done by crowdsensed data collection [64]. Truthful auction-based incentive mechanisms improve the quality of the collected fingerprints, based on a quality metric that characterizes the joint impact of privacy and spatial coverage. The design of the individually rational auction-based incentive must account for the limited budget of the MCS server and the fact that workers may misreport their costs to gain profit. Table IX summarizes approaches toward security and privacy in MCS-based wireless systems.

E. Future Research Directions

- Privacy provisioning for mobile participants in the MCS-based wireless sensing may negatively affect the accuracy of results. Addressing this tradeoff is a future research direction.
- The spectrum white space sensing needs to be done in real-time, due to rapidly changing wireless radio environments. Therefore, the roles of temporal factors in any secure and private crowdsensed spectrum searching task allocation and integration of this application of MCS with edge computing and/or the cloud need further investigation [84,85].

This article has been accepted for publication in IEEE Transactions on Technology and Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TTS.2022.3191515

- Extension of the crowdsourced Wi-Fi communities with a single operator to multiple competing network operators, to serve heterogeneous Wi-Fi users with varying traffic demands and mobility patterns, is an area of further exploration.

XI. Private and Personalized MCS Recommendation Systems

A. Privacy-Preserving MCS Recommendation

Personalized task recommendation mechanisms ensure pushing the matching tasks to the potentially interested workers, instead of simply publishing tasks without targeting any audience of suitable mobile workers. Recommendations are based on MCS workers' time-varying interests and expertise [88]. Nevertheless, the plaintext exposure of requesters' tasks and workers' interests causes privacy concerns in the presence of an honest but curious MCS platform. Although encryption of tasks and interests preserves privacy, for existing MCS recommendation systems to be effective for ciphertexts, task recommendation needs to be transformed into task access control and keyword-based search [89]. With requesters and workers owning their secret keys, multikeyword matching can be performed on task requirements and workers' interests represented by multiple keywords [90].

Keyword search in the ciphertext domain can be conducted using a predefined keyword dictionary. However, to reduce the computation cost of the search through the keyword dictionary, the vector space model is replaced with a polynomial function to represent task requirements and worker interests as smaller vectors. Multiple keywords, related to task requirements and worker interests, can be expressed using a polynomial function [90]. The steps to perform the multiple-keywords matching between multiple requesters and multiple workers include 1) derive keys based on matrix decomposition and distribute different secret keys to requesters and workers; 2) requestors encrypt the task requirements and workers encrypt their interests (with different secret keys by requesters and workers); and 3) securely compute the inner products between the encrypted task requirements and workers interests using asymmetric scalar-product-preserving encryption together with proxy re-encryption. This scheme allows for user revocation [90].

Combining multi-authority attribute-based encryption and searchable encryption constructs a privacy-preserving personalized recommendation scheme [89]. More specifically, instead of relying on a single authority to issue public and private keys, every requester becomes an authority to encrypt their own tasks' attributes and issue keys for authorized workers according to the workers' interest or expertise attributes.

To prevent astroturfing, which negatively affects task quality and wastes resources, a topic is assigned to each task by a fine-grained recommendation mechanism through interest-expertise collaborative awareness [88]. After estimating the topic-specific expertise level and interest degree of workers, by using historical task records, the recommendation system comes up with a list of suitable workers for topic-specific tasks.

B. Crowdsourcing for Recommending Smartphone App Privacy Settings

Crowdsourced solicitation of smartphone users' privacy permission settings for various mobile apps leads to understanding users' privacy expectations and thereby, making app-specific unobtrusive privacy recommendations [91]. For example, PriWe, is a crowdsourcing-driven system that collects privacy permission settings of the apps installed on smartphones to make proper recommendations to users on how to minimize privacy disclosure upon installing smartphone apps [92]. To encourage smartphone users' participation to share their app permission settings, this task was published and tested on the Amazon Mechanical Turk crowdsourcing platform.

C. Privacy-Preserving Mobile Worker Recruitment

During worker recruitment, the reward requested by a mobile worker reveals private information, such as the tasks that the worker can perform, the sensing quality, etc. The tasks that a user can perform may expose users' visited sites, while the sensing quality exposes the visits' frequency, time, geographical distance, etc. [94]. Moreover, the workers' sensed data must be kept private from other competing workers. Randomness methods, such as differential privacy, are not accurate for the crowdsensing tasks that require precision. Despite being a countermeasure, homomorphic encryption and garbled circuit protocols impose expensive computation/communication overhead on mobile devices. To tackle these issues, secret sharing is combined with greedy prioritization of workers with the best total sensing qualities of all tasks (compared to a quality threshold) to recruit optimal workers [94].

To identify the matching mobile workers, cosine similarity score calculates the similarity between the vector of task attributes and the worker's capability/interest vector [93]. To ensure both fairness and privacy in worker recruitment, the fuzzy comprehensive evaluation and the fuzzy closeness are used along with secure multiparty worker sorting. More specifically, homomorphism of semantic security encryption is used to calculate the fuzzy closeness, to sort the matching workers with secret inputs.

D. Future Research Directions

It is essential to design lightweight secure protocols, which do not depend on cryptography or a trusted third-party. Improving the computational efficiency of worker recruitment to adapt to dynamically varying task requirements [93]; coming up with more measurable privacy leakage metrics for worker recruitment [9]; handling task recommendations that involve a group of MCS participants rather than a single user, i.e., larger-scale MCS systems; and deploying big datasets and numerous app store applications [91] are areas for further research.

XII. Truth Discovery Against Data Falsification in MCS

Some MCS platforms pursue monetization of knowledge and hence, need truth discovery to mine the knowledge, from unreliable sensing data [24]. Truth discovery ensures fair

TABLE X
Recommendation Systems for MCS

Goal	Approach	Challenges/Evaluation
Avoiding underperforming mobile workers who cause high error rates	Recommendation and ranking mechanisms to generate a list of suitable workers by assigning a topic model to each task and using logistic regression with Bayesian posterior inference to predict topic-specific interest degree/expertise of each MCS worker [88]; Twitter latent Dirichlet allocation method	Time-varying interests of MCS workers; Evaluated on multi-tag labelling tasks
Private and personalized task recommendation in the ciphertext domain despite a honest but curious MCS cloud server	Requester uses descriptive attributes to specify conditions for the task to be pushed to a worker with matching attributes [89]; Multi-authority attribute-based searchable encryption	Computation and time costs of task encryption/decryption
Simultaneous task privacy and worker privacy empowered by user accountability and user revocation	Representing task requirements and worker interests as small vectors; Key derivation based on matrix decomposition for multi-keyword matching between requesters and workers with proxy re-encryption and asymmetric scalar-product-preserving encryption [90]	Reducing the computation cost associated with predefined keyword dictionary
Recommendations to smartphone Apps users about privacy permission settings	Collaborative filtering to find similar apps and people with similar privacy preferences based on crowdsourced data collection of mobile users' privacy permissions settings [91,92]; Seeking users' feedback to improve the accuracy of recommendations	Evaluated on hundreds of participants on Amazon Technical Turks for several days and on Android SDK (PriWe App [92])
Fairness in mobile worker recruitment by inducing uniform probability of selection	Determining the membership function of each worker attribute and using fuzzy comprehensive evaluation to calculate the value of each attribute; Secure multiparty sorting based on fuzzy closeness with mobile participants using homomorphism of semantic security encryption [93]	Data quality variations caused by dynamically varying tasks requirements; computation/communication overhead; cosine similarity index calculation time; task budget consumption ratio
Collusion among MCS server and participants to reveal private information during recruitment	Secret sharing during mobile worker recruitment [94] using an increasing submodular function as the computation function of the sensing quality	NP-hardness; Achieving logarithmic approximation ratio (through submodular sensing quality function)

monetization by estimating user reliability degrees and by inferring truthful information via reliability-aware data aggregation [96]. However, private information of MCS participants (identity, workers' locations, investment of the task requestor, etc.) may be exposed during truth discovery process. One solution is to conduct truth discovery in the encrypted domain in the cloud [95] using lightweight additively homomorphic encryption and garbled circuits. The encrypted extracted truth is then sent to the requester for decryption. Nevertheless, the iterative transmission of large homomorphic ciphertexts for dynamic users imposes extra costs of computation, bandwidth, fault tolerance, and group management [96]. The process of truth discovery involves a weighted collection of crowdsensed reports to give more weights to more reliable workers. Random masking applied to weighted data aggregation prevents the leakage of individual sensory data or users' weights in iterative truth discovery. Table XI highlights MCS truth discovery schemes to hinder data falsification.

Due to proximity of fog nodes to mobile MCS users, they can perform real-time truth discovery with low latency and lower bandwidth. Closeness to mobile users allows fog nodes to authenticate MCS participants to prevent false data injection attack. Nevertheless, if fog nodes are not trusted, countermeasures include perturbation, homomorphic Paillier encryption, one-way hash chain, and super-increasing sequence techniques to outsource truth estimation to the cloud [98].

Since privacy-preserving methods may sometimes protect the identities of malicious users, they can contradict truth discovery and data quality. Enriching truth discovery with game theory and algorithmic mechanism design, via incentive and penalty policies, motivates truthful sensing results while preserving privacy [97].

A. Future Research Directions

Exploring various types of weight functions in weighted data aggregation for truth discovery [96] (e.g., weight functions that are based on linear operations between a user's distance and the summation of distances across all users) has the potential to improve MCS truth discovery.

TABLE XI
Overview of Truth Discovery Against Data Falsification in MCS

Goal	Approach	Implementation
Privacy of the sensing values and weights (reliability degrees) of MCS users, as well as the requester's inferred truth;	• Two non-colluding independent cloud servers (one bridging the workers and requesters and the other discovering truth in the encrypted domain) perform truth discovery in encrypted domain before sending to requestor for decryption • Lightweight additively homomorphic encryption and garbled circuits [95]	Crowdsourced indoor floorplan reconstruction on Microsoft Azure (MCS requester and cloud server) using D12 instance (4 cores, 28GB RAM, Ubuntu Server 16.04 LTS system), and Samsung Galaxy S4 Android phone (MCS worker), four-core 1.6 GHz processor, four-core 1.2 GHz processor and 2.0 GB RAM; ObliVM-lang2 for garbled circuits
Reducing communication bandwidth and computation in privacy-preserving large-scale iterative MCS truth discovery	Splitting-based encryption combined with homomorphic encryption in a two-server model with random masking applied on weighted data aggregation	Execution time and convergence measured on Amazon EC2 c4.4xlarge instance (as the server) with Xeon E5-2666 processor with 16x 2.9 GHz vCPU, 30 GB memory and Samsung Galaxy S6 Android phone (as the user) [96]
Discouraging untruthful reports while the identities/locations of MCS workers are cloaked for privacy	• Penalization mechanism design to enforce truthfulness as the optimal strategy • Rewarding higher profits to encourage calibration of intrinsic sensing biases	Accuracy of crowdsensed outdoor temperature data reported by taxis in Rome on OMNeT++ 4.6 [97]
Privacy for highly mobile (e.g., vehicular) crowdsensing despite untrustworthy fog nodes involved in MCS truth discovery	• Outsourcing truth estimation to both the cloud and fog nodes • Super-increasing sequence to integrate multidimensional weighted data, instead of direct encryption and upload • Perturbation, hash chain, and homomorphic Paillier encryption to shift all user workload to the MCS server [98]	Computational costs and communication, runtime, and accuracy of truth discovery on a testbed with Android phones (mobile users) with 1.5 GHz and 2 GB RAM, and a laptop with 2.5 GHz Intel Core i7 and 16 GB RAM (as fog and cloud)

XIII. Social MCS Privacy

The popularity and openness of crowdsourced data from mobile users in social networks and social IoT may jeopardize mobile users' privacy. It provides individuals with an opportunity to exhibit antisocial behavior (e.g., free-riding) to decrease the social welfare [99]. Game theoretical mechanisms, such as pricing/reward and reputation/rating, to tag the mobile participants' social status, can incentivize compliance with social norms [99].

Online aggregate monitoring over infinite streams combined with differential privacy enables real-time spatio-temporal data publishing in social networks, where population statistics are continuously published. The privacy of statistics published on infinite time stamps are provided through adaptive sampling, adaptive budget allocation, dynamic grouping, perturbation, and filtering. To improve the utility of released data, i.e., to minimize the total group error, deep learning may be used to predict the statistics of spatial regions [100].

Reporting local histograms, instead of raw data can prevent the privacy leakage caused by participants' social coupling, measured through their data correlation. Gaussian Markov random fields model the correlations underlying the participants' data. The interaction of the MCS sever and participants leans itself to a Stackelberg game, where the server chooses its reward policy and participants choose their noise levels relevant to their social correlation [101]. The policies depend on the relationship between MCS participants' Nash equilibria, the payment mechanism, budget, and the required task accuracy. Malicious users may report falsified noise levels to gain higher payment. Payment mechanisms that assert truthfulness as a dominant strategy can thwart false reporting [101].

In social IoT, the couplings among MCS workers are useful for worker recruitment to perform collaborative multihop tasks. Incentive mechanisms protect the requestors' privacy in multihop routing tasks [102], taking into account the utility budget constraints. For example, consider a spatial MCS in which workers compete to perform multiple tasks by taking detours from their original travel paths. The task assignment must maximize the social welfare of MCS participants and protect workers' privacy [60]. To evaluate trustworthiness of participants, their social trust degrees are updated based on their social ties and the importance of allocated tasks. The reputation update is based on social norm. Social norms consist of a social strategy and a reputation scheme to regulate the behavior of participants. Social norms are designed to punish deviations from the selected social strategy [104].

A. Future Research Directions

Areas of further research include [99,102,103] 1) generalizing the schemes to maximize a participant's average revenue from all tasks instead of just the expected one-period utility, 2) designing competition games for temporary social IoT users

TABLE XII
Trustworthy Social MCS Systems

Problem	Approach	Challenges
Antisocial behavior (e.g., free-riding) to decrease the MCS social welfare	Socially optimal rating protocol based on game theory through an incentive mechanism by integrating the pricing and social status reputation [99]	Quantifying social norm and compliance
Real-time spatiotemporal data publishing in crowdsourced social networks with privacy preservation	Integration of adaptive sampling, adaptive budget allocation, dynamic grouping, perturbation and filtering for privacy-preserving statistics publishing [100]	The utility of data publishing
Privacy leakage through correlation attacks caused by social relationships of MCS participants	Socially private correlated local histograms with MCS participants adding noise to protect their privacy; Stackelberg game modeling of interactions between the MCS server and participants [101]	Falsified noise levels chosen by MCS participants to achieve higher payoffs
Privacy of task requesters in social IoT revealed by friendship/collaboration among MCS workers in routing of tasks among multihop friends	Multihop routing incentive mechanism to motivates workers to forward tasks to their expert friends [102]	Utility maximization under privacy and budget feasibility constraints; tradeoff between privacy and task accuracy; approximation ratio of task assignment algorithm
Sybil-proof social MCS	Reverse auction incentives to recruit workers based on their social neighbors and not fictitious identities [103]	Social cost; running time; MCS users' utilities

instead of permanent users, 3) studying security vulnerabilities of MCS social IoT to various types of tasks (other than labeling), and 4) lightweight edge-based design against sybil attacks in large-scale social IoT MCS networks

XIV. *k*-Anonymity Cloaking for MCS Privacy

Sensitive attributes of published tasks on the crowdsourcing platform may be used by malicious workers that link them to other public databases to reveal private information of requestors. *k*-anonymity is a spatio-temporal cloaking method for privacy protection, specially when the spatio-temporal information of MCS participants need to be tagged to the collected data [11].

k-anonymity may be combined with differential privacy [106] to achieve location privacy in location-based crowdsourcing. One example application is New York City's CityNoise app in which people submit the sensed noise pollution data tagged with locations to generate urban noise maps [2]. *k* anonymity may degrade the accuracy of task performance by causing information loss. The tradeoff between *k*-anonymity privacy and service quality degradation can be modeled by Stackelberg games [11,106] or by optimization formulations [2]. For example, one can maximize the number of protected users, subject to a location quality degradation constraint or simply minimize the MCS task quality degradation, while guaranteeing the location privacy for all users [2]. Furthermore, probability models for the tradeoff between privacy and accuracy in *k*-anonymity can give probabilistic lower and upper bounds for task accuracy [105]. Searching for the optimal anonymity approach is a NP-Hard problem, which needs to be solved by heuristics. Progressive feedback histogram by repeatedly submitting the crowdsourced tasks to collect the humans' opinions and then adaptively adjusting the anonymity approach may enhance the accuracy in the anonymization process [105].

A. Integration of k-anonymity with Incentive Mechanisms

Once MCS users are clustered into groups for *k*-anonymity, reverse auction incentives may be applied to mitigate the information loss. As such, users are recruited based on group values, compensations, and sensing costs for clustered groups [71]. Since *k*-anonymity (e.g., in location aggregation) implies approximate values, the algorithm approximation ratio is a determining factor in the reverse auction design.

B. Future Research Directions

Some research directions on applying *k*-anonymity in MCS include: studying the effects of social relationships among MCS workers on *k* anonymity [106]; extensions to personalized *k*-anonymity models, where each user requires a different anonymity level [2]; and *k*-anonymity models tailored for crowdsensing that perturb spatio/temporal information for time sensitive crowdsensed data.

This article has been accepted for publication in IEEE Transactions on Technology and Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TTS.2022.3191515

TABLE XIII
SOLUTIONS FOR *k*-ANONYMITY CLOAKING FOR MCS PRIVACY

Goal	Approach	Challenges/Evaluation
Tradeoff between *k*-anonymity and accuracy	Probability-based matrix to estimate the lower and upper bounds of the crowdsourcing accuracy for the anonymized data [105]; feedback-based *k*-anonymity by synthetic samples published to human workers; adaptively cutting the dimensions based on feedback results on the synthetic samples	Applied on Mondrian algorithm and evaluated on U.S. census dataset
Location privacy in location-based MCS services while maintaining the quality of service	• Combining *k*-anonymity, differential privacy, and Stackelberg game to solve the tradeoff between privacy and service quality [11,106] • Optimization formulations [2]	Benchmarked with Clique cloaking and differential perturbation [106]

XV. MISINFORMATION THWARTED BY CROWDSOURCING

Rumors, fake news, or misinformation can cause serious problems, especially following disasters, when users do not have enough time for verifying the credibility of online posts [107]. Methods for truth discovery and social MCS, discussed in Sections XII and XIII, respectively, can be used to detect misinformation. Other major fake news detection strategies include content analysis, social context (e.g., diffusion patterns), and machine learning classification/prediction [108].

Seeking feedback from MCS mobile workers about the classification of news (fake or true) is less expensive, compared with professional journalists or subject experts [109]. However, it could be less reliable than the opinion of experts. To strike a balance for this tradeoff, a crowd knowledge graph is constructed by combining the feedback from experts and non-expert MCS workers about misinformation [110]. The crowdsourced judgements can then be combined with machine learning to predict their accuracy. For example, to detect COVID-19 misinformation, natural language processing (NLP) was used to extract key information in the news content [110]. MCS workers were tasked with reading the content to extract the key knowledge related to the NLP outputs.

Moreover, the immutability and transparency of crowdsourced news (e.g., Twitter feeds) stored on the blockchain enable fake news detection. Specifically, machine learning continuously learns from the output of the PoS smart contract, while validators from the public (Ethereum) blockchain are incentivized with rewards for contributing reliable information [111]. The machine learning classifier acts as one of the validators in the PoS consensus. The final verdict from the smart contract, about the truth of the news, enriches the training dataset to dynamically improve the machine learning classification outputs.

Leveraging the differences of diffusion patterns between misinformation and true information, Bayesian logistic regression can infer the credibility of a message by observing the latent attributes of the message, the users interacting with it, and their reactions to the message [107]. To this end, example available datasets include [109]

- CREDBANK (set of Twitter conversations about events and corresponding crowdsourced accuracy assessments for each event);
- PHEME (curated dataset of conversation threads about Twitter rumors and journalists' annotations assessments of their truthfulness); and
- BuzzFeed dataset of highly shared true and false political stories.

Fig. 7 summarizes MCS-based approach to fake news and misinformation discovery.

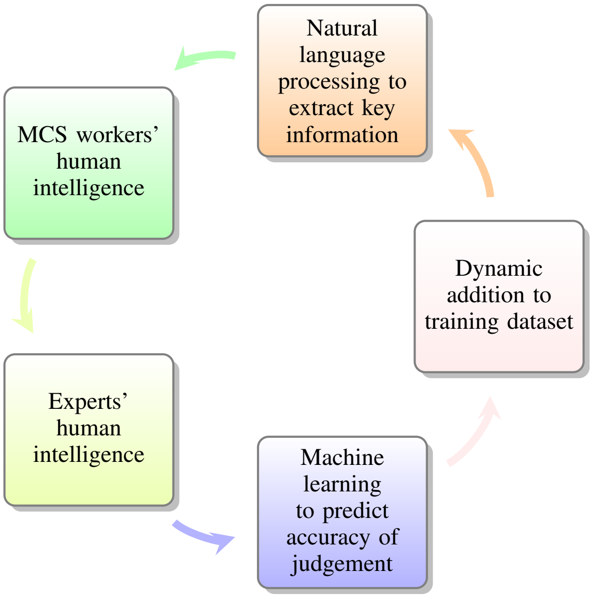
Natural language processing to extract key information
MCS workers' human intelligence
Dynamic addition to training dataset
Experts' human intelligence
Machine learning to predict accuracy of judgement

Fig. 7. MCS-based approach against misinformation and fake news

A. Future Research Directions

Some future investigation areas include:

- Extension of the existing MCS models to news topics with mixed content;
- Estimation of the partial truthfulness of information;
- Classification algorithms for news with true content but misleading headlines;
- Advancing the integration of artificial intelligence with crowdsourced detection and estimation tasks; and
- Faster news interpretation and discovery methods.

XVI. Conclusion

MCS systems owe their ever-increasing ubiquity to social networks, IoT, remote health, widespread mobile Apps, etc. Nevertheless, MCS systems still need to be protected to operate in complex real-world environments, to prevent the overwhelming societal, psychological, and monetary costs resulted from the exposure of personal information. This survey presented the requirements for a multi-faceted design to safeguard MCS systems against various types of vulnerabilities. Different attacks and defenses for MCS requestors, mobile workers, and the MCS platform were explained. Specially, the protection of MCS systems was discussed within various contexts, such as vehicular MCS, Industrial IoT, 6G and Wi-Fi hotspots, social networks, Android systems, etc. A plethora of countermeasures were put forward, based on blockchains, truth discovery, machine learning, edge-computing, recommendation systems, differential privacy, *k*-anonymity models, games, incentive mechanism design, etc. Moreover, future research directions and extensions in each of these areas were highlighted. The combinations of the techniques have great potential to inform future solutions in this field. The solutions need to address design metrics, such as service value, social welfare, profit gain, mobile worker time, memory, overhead, data reliability, mobile worker satisfaction, reputation scores, etc.

References

[1] W. Liu, X. Wang, and W. Peng, "Secure remote multi-factor authentication scheme based on chaotic map zero-knowledge proof for crowdsourcing internet of things," *IEEE Access*, vol. 8, pp. 8754–8767, 2020.

[2] Y. Zhang, M. Li, D. Yang, J. Tang, G. Xue, and J. Xu, "Tradeoff between location quality and privacy in crowdsensing: An optimization perspective," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3535–3544, 2020.

[3] Y. Sei and A. Ohsuga, "Differential private data collection and analysis based on randomized multiple dummies for untrusted mobile crowdsensing," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 926–939, 2017.

[4] Z. Wang, J. Hu, R. Lv, J. Wei, Q. Wang, D. Yang, and H. Qi, "Personalized privacy-preserving task allocation for mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 18, no. 6, pp. 1330–1341, 2019.

[5] X. Ren, C. Yu, W. Yu, S. Yang, X. Yang, J. A. McCann, and P. S. Yu, "LoPub : High-dimensional crowdsourced data publication with local differential privacy," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 9, pp. 2151–2166, 2018.

[6] J. Hu, K. Yang, K. Wang, and K. Zhang, "A blockchain-based reward mechanism for mobile crowdsensing," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 1, pp. 178–191, 2020.

[7] J. Hu, H. Lin, X. Guo, and J. Yang, "Dtcs: An integrated strategy for enhancing data trustworthiness in mobile crowdsourcing," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4663–4671, 2018.

[8] L. Ma, X. Liu, Q. Pei, and Y. Xiang, "Privacy-preserving reputation management for edge computing enhanced mobile crowdsensing," *IEEE Transactions on Services Computing*, vol. 12, no. 5, pp. 786–799, 2019.

[9] T. Li, T. Jung, Z. Qiu, H. Li, L. Cao, and Y. Wang, "Scalable privacy-preserving participant selection for mobile crowdsensing systems: Participant grouping and secure group bidding," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 855–868, 2020.

[10] Q. Xu, Z. Su, S. Yu, and Y. Wang, "Trust based incentive scheme to allocate big data tasks with mobile social cloud," *IEEE Transactions on Big Data*, vol. 8, no. 1, pp. 113–124, 2022.

[11] Y. Wang, Y. Li, Z. Chi, and X. Tong, "The truthful evolution and incentive for large-scale mobile crowd sensing networks," *IEEE Access*, vol. 6, pp. 51 187–51 199, 2018.

[12] Z. Wang, X. Pang, Y. Chen, H. Shao, Q. Wang, L. Wu, H. Chen, and H. Qi, "Privacy-preserving crowd-sourced statistical data publishing with an untrusted server," *IEEE Transactions on Mobile Computing*, vol. 18, no. 6, pp. 1356–1367, 2019.

[13] Y. Gong, L. Wei, Y. Guo, C. Zhang, and Y. Fang, "Optimal task recommendation for mobile crowdsourcing with privacy control," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 745–756, 2016.

[14] J. An, H. Yang, X. Gui, W. Zhang, R. Gui, and J. Kang, "Tcns: Node selection with privacy protection in crowdsensing based on twice consensuses of blockchain," *IEEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 1255–1267, 2019.

[15] W. Chen, Y. Chen, X. Chen, and Z. Zheng, "Toward secure data sharing for the iov: A quality-driven incentive mechanism with on-chain and off-chain guarantees," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 1625–1640, 2020.

[16] Y. Wu, S. Tang, B. Zhao, and Z. Peng, "Bptm: Blockchain-based privacy-preserving task matching in crowdsourcing," *IEEE Access*, vol. 7, pp. 45 605–45 617, 2019.

[17] S. Zou, J. Xi, H. Wang, and G. Xu, "Crowdblps: A blockchain-based location-privacy-preserving mobile crowdsensing system," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4206–4218, 2020.

[18] C. Lai, M. Zhang, J. Cao, and D. Zheng, "SPIR: A secure and privacy-preserving incentive scheme for reliable real-time map updates," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 416–428, 2020.

[19] M. Li, J. Weng, A. Yang, W. Lu, Y. Zhang, L. Hou, J. Liu, Y. Xiang, and R. H. Deng, "Crowdbc: A blockchain-based decentralized framework for crowdsourcing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 6, pp. 1251–1266, 2019.

[20] C. H. Liu, Q. Lin, and S. Wen, "Blockchain-enabled data collection and sharing for industrial IoT with deep reinforcement learning," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3516–3526, 2019.

[21] X. Xu, Q. Liu, X. Zhang, J. Zhang, L. Qi, and W. Dou, "A blockchain-powered crowdsourcing method with privacy preservation in mobile environment," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 6, pp. 1407–1419, 2019.

[22] J. Zou, B. Ye, L. Qu, Y. Wang, M. A. Orgun, and L. Li, "A proof-of-trust consensus protocol for enhancing accountability in crowdsourcing services," *IEEE Transactions on Services Computing*, vol. 12, no. 3, pp. 429–445, 2019.

[23] K. Zhao, S. Tang, B. Zhao, and Y. Wu, "Dynamic and privacy-preserving reputation management for blockchain-based mobile crowdsensing," *IEEE Access*, vol. 7, pp. 74 694–74 710, 2019.

[24] C. Cai, Y. Zheng, A. Zhou, and C. Wang, "Building a secure knowledge marketplace over crowdsensed data streams," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 6, pp. 2601–2616, 2021.

[25] J. Wang, M. Li, Y. He, H. Li, K. Xiao, and C. Wang, "A blockchain based privacy-preserving incentive mechanism in crowdsensing applications," *IEEE Access*, vol. 6, pp. 17 545–17 556, 2018.

[26] "Editorial: Blockchain in industrial iot applications: Security and privacy advances, challenges, and opportunities," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4119–4121, 2020.

[27] C. Cai, Y. Zheng, Y. Du, Z. Qin, and C. Wang, "Towards private, robust, and verifiable crowdsensing systems via public blockchains," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 4, pp. 1893–1907, 2021.

[28] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, "On the security and performance of proof of work blockchains," in *ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 3–16.

[29] S. Zhu, Z. Cai, H. Hu, Y. Li, and W. Li, "zkcrowd: A hybrid blockchain-based crowdsourcing platform," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4196–4205, 2020.

[30] L. Xiao, Y. Li, G. Han, H. Dai, and H. V. Poor, "A secure mobile crowdsensing game with deep reinforcement learning," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 1, pp. 35–47, 2018.

[31] Y. Liu, H. Wang, M. Peng, J. Guan, J. Xu, and Y. Wang, "Deepga: A privacy-preserving data aggregation game in crowdsensing via deep reinforcement learning," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4113–4127, 2020.

[32] Y. Wang, M. Gu, J. Ma, and Q. Jin, "Dnn-dp: Differential privacy enabled deep neural network learning framework for sensitive crowdsourcing data," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 1, pp. 215–224, 2020.

[33] X. Chu, J. Liu, D. Gong, and R. Wang, "Preserving location privacy in spatial crowdsourcing under quality control," *IEEE Access*, vol. 7, pp. 155 851–155 859, 2019.

[34] Q. Zhang, L. T. Yang, Z. Chen, P. Li, and M. J. Deen, "Privacy-preserving double-projection deep computation model with crowdsourcing on cloud for big data feature learning," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2896–2903, 2018.

[35] J. Xiong, M. Zhao, M. Z. A. Bhuiyan, L. Chen, and Y. Tian, "An AI-enabled three-party game framework for guaranteed data privacy in mobile edge crowdsensing of IoT," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 2, pp. 922–933, 2021.

[36] J. Wei, X. Wang, N. Li, G. Yang, and Y. Mu, "A privacy-preserving fog computing framework for vehicular crowdsensing networks," *IEEE Access*, vol. 6, pp. 43 776–43 784, 2018.

[37] H. Wu, L. Wang, and G. Xue, "Privacy-aware task allocation and data aggregation in fog-assisted spatial crowdsourcing," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 1, pp. 589–602, 2020.

[38] L. Zhu, M. Li, and Z. Zhang, "Secure fog-assisted crowdsensing with collusion resistance: From data reporting to data requesting," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5473–5484, 2019.

[39] J. Ni, K. Zhang, Y. Yu, X. Lin, and X. S. Shen, "Providing task allocation and secure deduplication for mobile crowdsensing via fog computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 3, pp. 581–594, 2020.

[40] P. Zhou, W. Chen, S. Ji, H. Jiang, L. Yu, and D. Wu, "Privacy-preserving online task allocation in edge-computing-enabled massive crowdsensing," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 7773–7787, 2019.

[41] B. Yin, Y. Wu, T. Hu, J. Dong, and Z. Jiang, "An efficient collaboration and incentive mechanism for internet of vehicles (iov) with secured information exchange based on blockchains," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 1582–1593, 2020.

[42] S. Basudan, X. Lin, and K. Sankaranarayanan, "A privacy-preserving vehicular crowdsensing-based road surface condition monitoring system using fog computing," *IEEE Internet of Things Journal*, vol. 4, no. 3, pp. 772–782, 2017.

[43] G. Sun, S. Sun, H. Yu, and M. Guizani, "Toward incentivizing fog-based privacy-preserving mobile crowdsensing in the internet of vehicles," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4128–4142, 2020.

[44] C. Li, S. Gong, X. Wang, L. Wang, Q. Jiang, and K. Okamura, "Secure and efficient content distribution in crowdsourced vehicular content-centric networking," *IEEE Access*, vol. 6, pp. 5727–5739, 2018.

[45] M. A. Azad, S. Bag, S. Parkinson, and F. Hao, "Trustvote: Privacy-preserving node ranking in vehicular networks," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 5878–5891, 2019.

[46] H. Li, D. Liao, G. Sun, M. Zhang, D. Xu, and Z. Han, "Two-stage privacy-preserving mechanism for a crowdsensing-based vsn," *IEEE Access*, vol. 6, pp. 40 682–40 695, 2018.

[47] M. Li, L. Zhu, and X. Lin, "Privacy-preserving traffic monitoring with false report filtering via fog-assisted vehicular crowdsensing," *IEEE Transactions on Services Computing*, pp. 1–1, 2019.

[48] G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, and B. Y. Zhao, "Ghost riders: Sybil attacks on crowdsourced mobile mapping services," *IEEE/ACM Transactions on Networking*, vol. 26, no. 3, pp. 1123–1136, 2018.

[49] V. Sucasas, G. Mantas, J. Bastos, F. Damio, and J. Rodriguez, "A signature scheme with unlinkable-yet-accountable pseudonymity for privacy-preserving crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 19, no. 4, pp. 752–768, 2020.

[50] J. Shu, X. Liu, K. Yang, Y. Zhang, X. Jia, and R. H. Deng, "Sybsub: Privacy-preserving expressive task subscription with sybil detection in crowdsourcing," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3003–3013, 2019.

[51] B. Rashidi, C. Fung, A. Nguyen, T. Vu, and E. Bertino, "Android user privacy preserving through crowdsourcing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 3, pp. 773–787, 2018.

[52] J. Lu, Y. Xin, Z. Zhang, X. Liu, and K. Li, "Game-theoretic design of optimal two-sided rating protocols for service exchange dilemma in crowdsourcing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 11, pp. 2801–2815, 2018.

[53] 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. 128 526–128 533, 2019.

[54] Q. Hu, S. Wang, X. Cheng, L. Ma, and R. Bie, "Solving the crowdsourcing dilemma using the zero-determinant strategies," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1778–1789, 2020.

[55] W. Li, C. Zhang, Z. Liu, and Y. Tanaka, "Incentive mechanism design for crowdsourcing-based indoor localization," *IEEE Access*, vol. 6, pp. 54 042–54 051, 2018.

[56] X. Ma, W. Deng, F. Wang, M. Hu, F. Chen, and M. M. Hassan, "Timcc: On data freshness in privacy-preserving incentive mechanism design for continuous crowdsensing using reverse auction," *IEEE Access*, vol. 8, pp. 1777–1789, 2020.

[57] X. Zhang, G. Xue, R. Yu, D. Yang, and J. Tang, "Countermeasures against false-name attacks on truthful incentive mechanisms for crowdsourcing," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 2, pp. 478–485, 2017.

[58] H. Wu, L. Wang, G. Xue, J. Tang, and D. Yang, "Enabling data trustworthiness and user privacy in mobile crowdsensing," *IEEE/ACM Transactions on Networking*, vol. 27, no. 6, pp. 2294–2307, 2019.

[59] J. Lin, D. Yang, M. Li, J. Xu, and G. Xue, "Frameworks for privacy-preserving mobile crowdsensing incentive mechanisms," *IEEE Transactions on Mobile Computing*, vol. 17, no. 8, pp. 1851–1864, 2018.

[60] M. Xiao, K. Ma, A. Liu, H. Zhao, Z. Li, K. Zheng, and X. Zhou, "Sra: Secure reverse auction for task assignment in spatial crowdsourcing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 4, pp. 782–796, 2020.

[61] Y. Xing, L. Wang, Z. Li, and Y. Zhan, "Multi-attribute crowdsourcing task assignment with stability and satisfactory," *IEEE Access*, vol. 7, pp. 133 351–133 361, 2019.

[62] Z. Zhang, S. He, J. Chen, and J. Zhang, "Reap: An efficient incentive mechanism for reconciling aggregation accuracy and individual privacy in crowdsensing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 12, pp. 2995–3007, 2018.

[63] B. Zhao, S. Tang, X. Liu, and X. Zhang, "Pace: Privacy-preserving and quality-aware incentive mechanism for mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 20, no. 5, pp. 1924–1939, 2021.

[64] W. Li, C. Zhang, and Y. Tanaka, "Privacy-aware sensing-quality-based budget feasible incentive mechanism for crowdsourcing fingerprint collection," *IEEE Access*, vol. 8, pp. 49 775–49 784, 2020.

[65] Q. Ma, L. Gao, Y. Liu, and J. Huang, "Incentivizing wi-fi network crowdsourcing: A contract theoretic approach," *IEEE/ACM Transactions on Networking*, vol. 26, no. 3, pp. 1035–1048, 2018.

[66] K. Han, H. Liu, S. Tang, M. Xiao, and J. Luo, "Differentially private mechanisms for budget limited mobile crowdsourcing," *IEEE Transactions on Mobile Computing*, vol. 18, no. 4, pp. 934–946, 2019.

[67] S. Wang, L. Huang, Y. Nie, X. Zhang, P. Wang, H. Xu, and W. Yang, "Local differential private data aggregation for discrete distribution estimation," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 9, pp. 2046–2059, 2019.

[68] L. Wang, D. Zhang, D. Yang, B. Y. Lim, X. Han, and X. Ma, "Sparse mobile crowdsensing with differential and distortion location privacy," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2735–2749, 2020.

[69] Y. Tsou and B. Lin, "PPDCA: Privacy-preserving crowdsourcing data collection and analysis with randomized response," *IEEE Access*, vol. 6, pp. 76 970–76 983, 2018.

[70] J. Chen, H. Ma, D. Zhao, and L. Liu, "Correlated differential privacy protection for mobile crowdsensing," *IEEE Transactions on Big Data*, vol. 7, no. 4, pp. 784–795, 2021.

[71] X. Wang, Z. Liu, X. Tian, X. Gan, Y. Guan, and X. Wang, "Incentivizing crowdsensing with location-privacy preserving," *IEEE Transactions on Wireless Communications*, vol. 16, no. 10, pp. 6940–6952, 2017.

[72] P. Huang, X. Zhang, L. Guo, and M. Li, "Incentivizing crowdsensing-based noise monitoring with differentially-private locations," *IEEE Transactions on Mobile Computing*, vol. 20, no. 2, pp. 519–532, 2021.

[73] L. Wang, D. Yang, X. Han, D. Zhang, and X. Ma, "Mobile crowdsourcing task allocation with differential-and-distortion geo-obfuscation," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 2, pp. 967–981, 2021.

[74] M. Yang, T. Zhu, Y. Xiang, and W. Zhou, "Density-based location preservation for mobile crowdsensing with differential privacy," *IEEE Access*, vol. 6, pp. 14 779–14 789, 2018.

[75] G. Gao, M. Xiao, J. Wu, S. Zhang, L. Huang, and G. Xiao, "Dpdt: A differentially private crowd-sensed data trading mechanism," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 751–762, 2020.

[76] K. Yan, G. Lu, G. Luo, X. Zheng, L. Tian, and A. Maradapu Vera Venkata Sai, "Location privacy-aware task bidding and assignment for mobile crowd-sensing," *IEEE Access*, vol. 7, pp. 131 929–131 943, 2019.

This article has been accepted for publication in IEEE Transactions on Technology and Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TTS.2022.3191515

[77] J. Xiong, R. Ma, L. Chen, Y. Tian, Q. Li, X. Liu, and Z. Yao, “A personalized privacy protection framework for mobile crowdsensing in IIoT,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4231–4241, 2020.

[78] S. Roy, S. Chatterjee, A. K. Das, S. Chattopadhyay, S. Kumari, and M. Jo, “Chaotic map-based anonymous user authentication scheme with user biometrics and fuzzy extractor for crowdsourcing internet of things,” *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2884–2895, 2018.

[79] A. Karati, S. H. Islam, and M. Karuppiah, “Provably secure and lightweight certificateless signature scheme for IIoT environments,” *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, pp. 3701–3711, 2018.

[80] A. Karati, S. H. Islam, G. P. Biswas, M. Z. A. Bhuiyan, P. Vijayakumar, and M. Karuppiah, “Provably secure identity-based signcryption scheme for crowdsourced industrial internet of things environments,” *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2904–2914, 2018.

[81] Y. Zhang, R. H. Deng, D. Zheng, J. Li, P. Wu, and J. Cao, “Efficient and robust certificateless signature for data crowdsensing in cloud-assisted industrial IoT,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 5099–5108, 2019.

[82] X. Jin, J. Sun, R. Zhang, Y. Zhang, and C. Zhang, “Specguard: Spectrum misuse detection in dynamic spectrum access systems,” *IEEE Transactions on Mobile Computing*, vol. 17, no. 12, pp. 2925–2938, 2018.

[83] C. Zhao, S. Yang, X. Yang, and J. A. McCann, “Rapid, user-transparent, and trustworthy device pairing for d2d-enabled mobile crowdsourcing,” *IEEE Transactions on Mobile Computing*, vol. 16, no. 7, pp. 2008–2022, 2017.

[84] X. Li, Q. Zhu, and X. Wang, “Privacy-aware crowdsourced spectrum sensing and multi-user sharing mechanism in dynamic spectrum access networks,” *IEEE Access*, vol. 7, pp. 32 971–32 988, 2019.

[85] X. Jin and Y. Zhang, “Privacy-preserving crowdsourced spectrum sensing,” *IEEE/ACM Transactions on Networking*, vol. 26, no. 3, pp. 1236–1249, 2018.

[86] V. Raida, P. Svoboda, M. Lerch, and M. Rupp, “Crowdsensed performance benchmarking of mobile networks,” *IEEE Access*, vol. 7, pp. 154 899–154 911, 2019.

[87] Z. Huang and Y. Gong, “Differential location privacy for crowdsourced spectrum sensing,” in *2017 IEEE Conference on Communications and Network Security (CNS)*, 2017, pp. 1–9.

[88] Z. Guo, C. Tang, W. Niu, Y. Fu, T. Wu, H. Xia, and H. Tang, “Fine-grained recommendation mechanism to curb astroturfing in crowdsourcing systems,” *IEEE Access*, vol. 5, pp. 15 529–15 541, 2017.

[89] H. Yin, Y. Xiong, T. Deng, H. Deng, and P. Zhu, “A privacy-preserving and identity-based personalized recommendation scheme for encrypted tasks in crowdsourcing,” *IEEE Access*, vol. 7, pp. 138 857–138 871, 2019.

[90] J. Shu, X. Jia, K. Yang, and H. Wang, “Privacy-preserving task recommendation services for crowdsourcing,” *IEEE Transactions on Services Computing*, vol. 14, no. 1, pp. 235–247, 2021.

[91] R. Liu, J. Cao, K. Zhang, W. Gao, J. Liang, and L. Yang, “When privacy meets usability: Unobtrusive privacy permission recommendation system for mobile apps based on crowdsourcing,” *IEEE Transactions on Services Computing*, vol. 11, no. 5, pp. 864–878, 2018.

[92] R. Liu, J. Liang, J. Cao, K. Zhang, W. Gao, L. Yang, and R. Yu, “Understanding mobile users privacy expectations: A recommendation-based method through crowdsourcing,” *IEEE Transactions on Services Computing*, vol. 12, no. 2, pp. 304–318, 2019.

[93] J. Xiong, X. Chen, Q. Yang, L. Chen, and Z. Yao, “A task-oriented user selection incentive mechanism in edge-aided mobile crowdsensing,” *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2347–2360, 2020.

[94] M. Xiao, G. Gao, J. Wu, S. Zhang, and L. Huang, “Privacy-preserving user recruitment protocol for mobile crowdsensing,” *IEEE/ACM Transactions on Networking*, vol. 28, no. 2, pp. 519–532, 2020.

[95] Y. Zheng, H. Duan, and C. Wang, “Learning the truth privately and confidently: Encrypted confidence-aware truth discovery in mobile crowdsensing,” *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 10, pp. 2475–2489, 2018.

[96] Y. Zheng, H. Duan, X. Yuan, and C. Wang, “Privacy-aware and efficient mobile crowdsensing with truth discovery,” *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 1, pp. 121–133, 2020.

[97] C. Zhao, S. Yang, and J. A. McCann, “On the data quality in privacy-preserving mobile crowdsensing systems with untruthful reporting,” *IEEE Transactions on Mobile Computing*, vol. 20, no. 2, pp. 647–661, 2021.

[98] C. Zhang, L. Zhu, C. Xu, X. Liu, and K. Sharif, “Reliable and privacy-preserving truth discovery for mobile crowdsensing systems,” *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1245–1260, 2021.

[99] J. Lu, C. Tang, X. Li, and Q. Wu, “Designing socially-optimal rating protocols for crowdsourcing contest dilemma,” *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 6, pp. 1330–1344, 2017.

[100] Q. Wang, Y. Zhang, X. Lu, Z. Wang, Z. Qin, and K. Ren, “Real-time and spatio-temporal crowd-sourced social network data publishing with differential privacy,” *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 591–606, 2018.

[101] G. Yang, Z. Shi, S. He, and J. Zhang, “Socially privacy-preserving data collection for crowdsensing,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 851–861, 2020.

[102] X. Gan, Y. Li, Y. Huang, L. Fu, and X. Wang, “When crowdsourcing meets social iot: An efficient privacy-preserving incentive mechanism,” *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9707–9721, 2019.

[103] L. Jiang, X. Niu, J. Xu, Y. Wang, Y. Wu, and L. Xu, “Time-sensitive and sybil-proof incentive mechanisms for mobile crowdsensing via social network,” *IEEE Access*, vol. 6, pp. 48 156–48 168, 2018.

[104] Y. Zhang and M. van der Schaar, “Reputation-based incentive protocols in crowdsourcing applications,” in *2012 Proceedings IEEE INFOCOM*, 2012, pp. 2140–2148.

[105] S. Wu, X. Wang, S. Wang, Z. Zhang, and A. K. H. Tung, “K-anonymity for crowdsourcing database,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 9, pp. 2207–2221, 2014.

[106] Z. Chi, Y. Wang, Y. Huang, and X. Tong, “The novel location privacy-preserving ckd for mobile crowdsourcing systems,” *IEEE Access*, vol. 6, pp. 5678–5687, 2018.

[107] A. Osho, C. Waters, and G. Amariucai, “An implicit crowdsourcing approach to rumor identification in online social networks,” in *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2020, pp. 174–182.

[108] M. L. Della Vedova, E. Tacchini, S. Moret, G. Ballarin, M. DiPierro, and L. de Alfaro, “Automatic online fake news detection combining content and social signals,” in *22nd Conference of Open Innovations Association (FRUCT)*, 2018, pp. 272–279.

[109] C. Buntain and J. Golbeck, “Automatically identifying fake news in popular Twitter threads,” in *IEEE International Conference on Smart Cloud (SmartCloud)*, 2017, pp. 208–215.

[110] Z. Kou, L. Shang, Y. Zhang, C. Youn, and D. Wang, “Fakesens: A social sensing approach to covid-19 misinformation detection on social media,” in *17th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, 2021, pp. 140–147.

[111] T. H. Yang Zen, C. B. Hong, P. M. Mohan, and V. Balachandran, “ABC-Verify: AI-Blockchain integrated framework for tweet misinformation detection,” in *IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*, 2021, pp. 1–5.