Blockchain-Based Privacy-Preserving Driver Monitoring for MaaS in the Vehicular IoT

Qinglei Kong, *Student Member, IEEE*, Rongxing Lu, *Senior Member, IEEE*, Feng Yin, *Member, IEEE*, and Shuguang Cui, *Fellow, IEEE*

***Abstract*—Driving behaviors are highly relevant to automotive statuses and on-board safety, which offer compelling shreds of evidence for mobility as a service (MaaS) providers to develop personalized rental prices and insurance products. However, the direct dissemination of driving behaviors may lead to violations of identity and location privacy. In this paper, our proposed mechanism first achieves the verifiable aggregation and immutable dissemination of performance records by exploiting a blockchain with the proof-of-stake (PoS) consensus. Moreover, to acquire a driver's aggregated performance record from the blockchain, the proposed scheme first realizes quick identification with a Bloom filter and further approaches the target performance record through an oblivious transfer (OT) protocol. A performance evaluation shows that during the acquisition of the records, the computational complexity of our scheme is only related to the scale of the records contained in one transaction. However, the computational complexity of one traditional scheme without a Bloom filter depends on the scale of the records generated during each time slot. Furthermore, the computational complexity of another traditional scheme without aggregation relies on the scale of the records contained in one transaction, as well as the length of a driver's performance history. We also investigate the trade-off between the privacy level and computational complexity, and we determine the optimal number of data records in each transaction.**

***Index Terms*—Mobility as a Service (MaaS), privacy preservation, proof-of-stake (PoS) blockchain, vehicular IoT.**

I. Introduction

MOBILITY as a service (MaaS) may revolutionize vehicle ownership and challenge traffic management in future intelligent transportation systems (ITSs) [1], [2], as it provides public benefits such as traffic congestion control and environmental sustainability [3], [4]. An MaaS operator is eager to recognize an applicant's improper driving habits when leasing a human-controlled vehicle, such as sharp braking, speeding, and fast acceleration/deceleration. Furthermore, to support driving behavior analysis, some performance monitoring systems [5] have been developed with vehicular IoT features, such as steering wheel movements and brake pedal pressure. However, there are a few challenges in the secure dissemination and acquisition of vehicular sensory data among mutually independent MaaS operators.

The first challenge lies in privacy-preserving data dissemination. As a driver may engage in vehicle leasing services provided by multiple MaaS operators, their performance records are collected and archived by different MaaS operators. Thus, it is difficult for an individual MaaS operator to retrieve a user's performance history. On the other hand, a digital ledger can help to automate the record dissemination process. For example, blockchains have recently created a series of use cases for the vehicular IoT sector [6], such as vehicular IoT-based insurance [7], multichain vehicular data storage [8], and even vehicular edge computing [9]. However, if the performance records are published directly on a blockchain, the movement trajectories may be disclosed, leading to location privacy violations [10]. Based on snippets of sensory data collected from a single turn, experimental results show that it is possible to identify one driver among a group of drivers [11], and this indicates the high probability of location and identity leakage. Although the topic of a *privacy-preserving blockchain* was investigated in [12]–[15], these blockchain networks were built and maintained by collaborative entities and treated as immutable ledgers. However, in our performance record sharing scenario, the MaaS operators are mutually competitive and independent. Therefore, a desirable privacy-preserving mechanism for the blockchain is required to enable privacy-preserving data sharing among mutually independent MaaS operators.

Another challenge resides in a privacy-preserving acquisition method for performance records. Due to the large-scale generation of performance records, we first need to propose an efficient mechanism with quick record identification. However, to achieve identity privacy preservation, a blockchain should not display real driver identities. Furthermore, the contents of a record should not be recovered without the permission of the corresponding driver. Although some privacy-preserving data

Manuscript received August 10, 2020; revised November 8, 2020, January 10, 2021, and February 17, 2021; accepted February 28, 2021. Date of publication March 9, 2021; date of current version May 5, 2021. This work was supported in part by the National Key R&D Program of China under Grant 2018YFB1800800, in part by the Key Area R&D Program of Guangdong Province under Grant 2018B030338001, in part by Shenzhen Outstanding Talents Training Fund, by Guangdong Research Project No. 2017ZT07X152, in part by the Natural Science Foundation of China under Grant 92067202, and in part by the China Postdoctoral Science Foundation under Grant 2020M671897. The review of this article was coordinated by Dr. Vuk Marojevic *(Corresponding author: Qinglei Kong.)*

Qinglei Kong is with the Future Network of Intelligence Institute (FNii), the Chinese University of Hong Kong, Shenzhen 518172, China, and also with the University of Science and Technology of China, Hefei 230052, China (e-mail: kql8904@163.com).

Rongxing Lu is with the Faculty of Computer Science, University of New Brunswick, Fredericton, NB E3B 5A3, Canada (e-mail: rlu1@unb.ca).

Feng Yin and Shuguang Cui are with the Future Network of Intelligence Institute (FNii), the Shenzhen Research Institute of Big Data, and the Chinese University of Hong Kong, Shenzhen 518172, China (e-mail: yinfeng@cuhk.edu.cn; sgcui@ucdavis.edu).

Digital Object Identifier 10.1109/TVT.2021.3064834

acquisition schemes were developed in [16], [17], they do not consider the secure localization of a user's performance records on the blockchain. Thus, there is a need for an efficient and privacy-preserving performance record acquisition scheme.

Based on the above analysis, in this paper, we propose a privacy-preserving driver performance monitoring mechanism in the vehicular IoT that achieves blockchain-based performance record dissemination and acquisition among the MaaS operators. Specifically, the contributions of this paper are as follows.

First, we propose a privacy-preserving and verifiable data aggregation scheme for characterizing driver behaviors. Specifically, the proposed scheme exploits the modified Paillier cryptosystem and an identity-based signature scheme for verifiable record aggregation, so it securely aggregates and authenticates a user's performance history. Additionally, our proposed scheme also exploits a permissioned blockchain with the proof-of-stake (PoS) consensus for immutable performance record sharing.

Second, to achieve the quick identification of a transaction containing the target performance record, the proposed scheme exploits a Bloom filter to store pseudonyms. Furthermore, to guarantee forward separation, driver pseudonyms are generated with a one-way hash chain and then authenticated through an identity-based signature scheme. Then, the scheme can further acquire the desired performance history through an oblivious transfer (OT) protocol.

Third, we carry out extensive performance evaluations to demonstrate the benefits of the proposed scheme. To illustrate its computational efficiency, we compare the proposed scheme with two traditional mechanisms, i.e., one that does not include the Bloom filter and the other that does not involve aggregation. Evaluation results show that our scheme's computational complexity is highly superior to those of the compared traditional mechanisms. Moreover, we formulate an objective function to capture the privacy and efficiency trade-off and determine the optimal number of performance records in each transaction.

The rest of this paper is organized as follows. We review some related works in Section II. We describe our system model, identify our security requirements, and present our design goals in Section III. We review the preliminaries in Section IV and describe the blockchain framework in Section V. We propose our performance record collection, identification, and acquisition scheme in Section VI. Then, we illustrate the blockchain-based data dissemination process in Section VII. A security analysis and performance evaluation are carried out in Section III-B and Section IX, respectively. Finally, we draw our conclusion in Section X.

II. Related Works

In this section, we review some research works related to our scheme; these include privacy-preserving data aggregation and querying, as well as blockchain-based mechanisms in vehicular networks.

A. Privacy-Preserving Data Aggregation and Querying

A privacy-preserving data aggregation scheme for the smart grid was proposed in [18], which securely aggregates a group of multidimensional sensory data reports. In [17], a privacy-preserving location-matching scheme, which aggregates a group of vehicular data reports with varying data dimensions, was developed for vehicular networks. However, the above secure data aggregation schemes are from the views of either a predefined sensor type or a divided location grid. In our system, secure data aggregation occurs from the perspective of the driver.

A location-based privacy-preserving data querying scheme was proposed in [16], where a user first determines the point of interest with an OT protocol and then exploits a private information retrieval (PIR) protocol to acquire the target data. In the proposed scheme, a user does not reveal the queried location to the server, while the location server only distributes the data being queried. A privacy-preserving data collection and acquisition scheme was proposed in [19]; it supports secure data collection and querying at the edge of a given network. Specifically, the proposed mechanism achieves the privacy-preserving acquisition of location-based sensory data at the network edge by exploiting an OT protocol.

However, the above querying schemes do not involve identifying a target user's data report. Our proposed scheme collects and aggregates vehicular sensory data from the perspectives of drivers. Furthermore, it requires the identification and acquisition of performance records to be privacy-preserving.

B. Blockchain Technology for Vehicular Networks

A decentralized trust management scheme for vehicular networks that exploits the joint proof-of-work (PoW) and PoS consensus mechanisms was proposed in [20]. With the proposed scheme, vehicles can generate ratings for the messages transmitted among them, and the blockchain stores the trust values of the involved vehicles. To achieve secure data sharing with auditability, a flexible hyperledger based payment structure that realizes anonymous payments was proposed for vehicle-to-grid networks in [14]. The scheme proposed in [13] exploits the consortium blockchain and smart contracts to achieve secure data storage and sharing in vehicular edge computing. Furthermore, the proposed data sharing scheme is reputation-based, thereby also realizing high-quality data sharing among vehicles.

Although some blockchain-based vehicular applications have been developed, they do not consider privacy-preserving data querying from the blockchain. Another problem that must be solved is how to achieve the secure and efficient identification of a user's performance history. In our proposed scheme, we investigate how to realize the privacy-preserving identification and retrieval of the target sensory data using the blockchain.

III. System Model, Security Model, and Design Goals

In this section, we introduce the system model, identify the threat model, and present the design goals.

A. System Model

To achieve driver performance sharing across the MaaS operators, we propose a blockchain-based sensory data dissemination framework in the vehicular IoT. The proposed framework

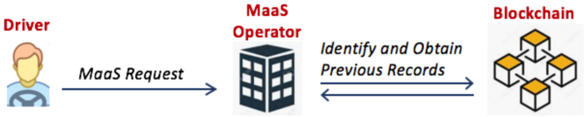


Fig. 1. MaaS querying and the performance record acquisition phase.

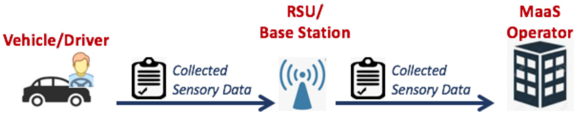


Fig. 2. Performance record collection phase.

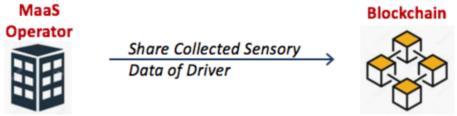


Fig. 3. Driver performance record sharing phase.

consists of three entities: drivers, vehicles, and MaaS operators. Moreover, we make the following assumptions: 1) each driver registers himself/herself with multiple MaaS operators; 2) each vehicle connects to only one MaaS operator.

- *MaaS Querying and Performance Record Acquisition.* When a MaaS operator receives a service request, it queries the blockchain for the performance history, as shown in Fig. 1. With the obtained performance history, the MaaS operator can perform various evaluations, such as personalized vehicle recommendations and pricing.
- *Performance Record Collection.* For each trip, a vehicle gathers vehicular IoT data related to driving habits and vehicle movement. For example, the typical driver drowsiness detection features are steering wheel movement and lateral/lane position standard deviation [5]. Furthermore, each vehicle organizes and uploads the collected sensory data to the connected MaaS operator through the nearest road side unit (RSU) or cellular base station, as shown in Fig. 2.
- *Driver Performance Record Sharing.* Each MaaS operator periodically shares the collected performance records with the rest of the MaaS operators through a permissioned ledger, as shown in Fig. 3.
- *Communication Model.* Vehicle-to-infrastructure (V2I) communication can be either realized through the IEEE 802.11p wireless access in vehicular environment (WAVE) standard (a short- to medium-range communication technology operating at the 5.9 GHz band) [21] or cellular communications. The connections between the RSUs (or cellular base stations) and the MaaS operators can be secure wired links with high bandwidths and low transmission delays.

B. Security Model

In the proposed performance monitoring system, security threats originate from both internal and external adversaries. First, we assume that the MaaS operators are honest-but-curious. They will follow the protocol but try to learn a driver's performance records. Second, we assume that the MaaS operators try to learn the mobility history of each driver. Third, some drivers may try to manipulate the ciphertexts for better service prices. Fourth, we assume there is no collusion between any groups of MaaS operators, vehicles, and drivers, as they are independent participants. Last, we assume that there is an external adversary who eavesdrops on the data transmissions. It may also launch active attacks to forge a driver/vehicle or threaten data integrity. Based on the above analysis, the following security requirements should be satisfied.

Privacy Preservation. Since performance records are highly location-dependent and individual-characterizing, drivers may not be willing to share their sensory data with the MaaS operators. Thus, only the driver and the vehicle can learn the data record. During the data querying phase, one MaaS operator (query operator) queries a driver's aggregated performance record history formulated by another MaaS operator (origin operator). To protect the driver's privacy, the origin MaaS operator should not learn which driver is under query, while the query MaaS operator should only derive the driver's performance record history.

Forward Separation. Since a driver's performance records may expose his/her regular mobility patterns and violate location privacy, the proposed scheme should carefully protect these records. After each personalized service, the MaaS operator should not identify and learn the driver's future data records from the ledger.

Verifiability. We need to guarantee the correctness of the data records published on each transaction since the blockchain can only achieve immutability. As performance records are distributively generated and encrypted, the proposed scheme should authenticate them. Thus, upon the recovery of an aggregated data record, the MaaS operator should verify its origin and correctness.

C. Design Goals

Under the above-defined system model and security model, we develop a privacy-preserving performance monitoring mechanism. Specifically, the proposed scheme should achieve the following design goals.

The proposed scheme should achieve the above-defined security requirements. If it does not satisfy the above security requirements, it may violate the location privacy of drivers. Furthermore, without the security guarantee, fog servers could identify the linkages between different performance records, and vehicles could modify their performance records. Then, drivers may not be willing to share their performance records, and so the system cannot properly function in this case.

The proposed scheme should achieve the goal of high efficiency. Even though the drivers, vehicles, and MaaS operators are assumed to have high computational power, we should also evaluate the computational overhead introduced by the large-scale performance records. Additionally, the proposed scheme should also achieve computationally efficient performance record collection, identification, and retrieval.

IV. Preliminaries

In this section, we briefly review the security techniques exploited: the Bloom filter, the modified Paillier cryptosystem, and bilinear maps. Roughly speaking, the Bloom filter is a probabilistic data structure that checks the existence of an element in a set, the modified Paillier cryptosystem is the basic building block of secure data aggregation and the OT protocol, and verifiable signature generation exploits bilinear maps.

A. Bloom Filter

A Bloom filter is a space-efficient data structure that utilizes hash functions to compactly store a set of data items with false positives [22].

In the insertion phase, the mechanism computes z hash functions $(h_1, h_2, \ldots, h_z)$ for every data item, and each computation returns the location of a bit in a bitmap with length b_0. Furthermore, the chosen z bit locations are set to one, and they act as a summary of the data item. The bitmap could contain the summaries of k data items if we repeat the same process. In addition, the optimal number of hash functions z for minimizing false positives is $z = ln2 * b_0/k$, and the minimal probability of false positives in a Bloom filter is $(1/2)^z$.

In the lookup phase, the bitmap generated above can be exploited to check the existence of an item. Given a query item, the same z hash functions are computed and the corresponding z bits in the new bitmap are probed. If all z bits in the newly generated bitmap are set to one, the query item is said to be present; otherwise, the query item is absent.

B. Modified Paillier Cryptosystem

The modified Paillier cryptosystem [23] consists of three components: key generation, encryption, and decryption.

- *Key Generation:* Given a security parameter κ, two large primes (p_1, p_2) are selected, where $|p_1| = |p_2| = \kappa$. Additionally, (p_1, p_2) satisfies the conditions that $p_1 = 2 \cdot p_1' + 1$ and $p_2 = 2 \cdot p_2' + 1$, where p_1' and p_2' are also large primes. Then, the RSA modulus $n = p_1 \cdot p_2$ and $\lambda = lcm(p_1 - 1, p_2 - 1)$ are computed. We consider $\mathbb{G} = QR_{n^2}$ to be the cyclic group of quadratic residues modulo n^2, whose order is $ord(\mathbb{G}) = \lambda(n^2)/2 = p_1 p_1' p_2 p_2' = n\lambda/2$. A random number $\mu \in \mathbb{Z}_{n^2}^*$ and a random value $x \in [1, ord(\mathbb{G})]$ are chosen, and then $g = \mu^2 \bmod n^2$ and $h = g^x \bmod n^2$ are set. The public key is $pk = (n, g, h)$, and the secret key is x.
- *Encryption:* Given a message $m \in \mathbb{Z}_n^*$, a random number $r \in \mathbb{Z}_{n^2}$ is chosen, and the ciphertext pair (c_1, c_2) is generated as $c_1 = g^r \bmod n^2$ and $c_2 = h^r \cdot (1 + mn) \bmod n^2$.
- *Decryption 1:* Given a ciphertext pair (c_1, c_2), the message can be decrypted as $m = \frac{c_2/(c_1)^x - 1 \bmod n^2}{n}$.
- *Decryption 2:* If $n = p_1 \cdot p_2$ is provided, one can compute $x \bmod n$ and $r \bmod n$. Let $xr \bmod ord(\mathbb{G}) = \gamma_1 + \gamma_2 n$; then, the value $\gamma_1 = xr \bmod n$ is efficiently computable. Furthermore, we compute the value $D = (\frac{c_2}{g^{\gamma_1}})^\lambda = 1 + m\lambda n \bmod n^2$ and further decrypt the message m: $m = \frac{D - 1 \bmod n^2}{n} \lambda^{-1} \bmod n$.

Remark 1: The modified Paillier cryptosystem has the property of additive homomorphism. Additionally, it supports the secure aggregation of the performance records of each driver. In *Decryption 1*, knowing the discrete logarithm x of h with a base g in $\mathbb{Z}_{n^2}^*$ allows for decrypting any ciphertext pair generated using (g, h) as an underlying public key. The system can generate public-key pairs for multiple users with one safe-prime modulus $n = p_1 \cdot p_2$ and base g in $\mathbb{Z}_{n^2}^*$. Thus, during system bootstrapping, it only needs to select two large primes p_1 and p_2.

Under the blockchain framework, since performance records are generated by multiple vehicles and disseminated across the MaaS operators, the exploited cryptosystem should support efficient data sharing among them. Using the same security parameter setup with multiple public-key pairs is suitable for this distributive data sharing situation. However, in the classical Paillier cryptosystem, one safe-prime modulus $n = p_1 \cdot p_2$ (requires system bootstrapping) can only generate one public-private key pair for one user.

C. Bilinear Maps

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two cyclic groups with the same prime order q, i.e., $|\mathbb{G}_1| = |\mathbb{G}_2| = q$. A bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ has the following properties:

- Bilinearity: $\forall P, Q \in \mathbb{G}_1$ and $\forall a, b \in \mathbb{Z}_q^*$, we can derive $e(aP, bQ) = e(P, Q)^{ab}$.
- Non-degeneracy: There exists $P \in \mathbb{G}_1$, which satisfies the condition that $e(P, Q) \neq 1_{\mathbb{G}_2}$.
- Computability: $\forall P, Q \in \mathbb{G}_1$, there is an efficient algorithm for computing $e(P, Q)$.

Definition 1: A bilinear parameter generator $\mathcal{G}en(\cdot)$ denotes a probabilistic algorithm that takes a parameter κ_1 as input and outputs a 5-tuple $(q, P, \mathbb{G}_1, \mathbb{G}_2, e)$, where q is a κ_1-bit prime, $\mathbb{G}_1$ and $\mathbb{G}_2$ are two cyclic groups with order q, $P \in \mathbb{G}_1$ is a generator, and $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ is a non-degenerated and computable bilinear map.

V. Blockchain Framework

Blockchain is a distributed ledger that enables decentralized data sharing among network nodes [13]. Each node maintains a unique ledger replica and synchronizes with other nodes via a consensus algorithm. Since driver performance records are relevant to public safety and commercial benefits, we exploit a permissioned blockchain constructed by MaaS operators for data sharing and impose a few restrictions on its access rights. In this section, we first define the roles and access rights of entities and then describe the design details.

A. Roles and Access Rights

There are two types of entities in our blockchain architecture: vehicles and MaaS operators.

- *Vehicle:* Each vehicle collects driving data and creates a performance record. Then, it sends the performance record to the MaaS operator.
- *MaaS Operator:* Each MaaS operator periodically collects and verifies the received performance records during each data collection interval. Then, it organizes the received performance records into one or a few transactions.

In our proposed blockchain architecture, we specify the following access rights.

- *Read:* All the involved entities own the "read" access rights, i.e., they can view the contents on the blockchain, including all the "transactions" and "ledgers".
- *Write:* All the MaaS operators possess the "transaction write" rights, i.e., formulating transactions with the drivers' performance records. Moreover, the MaaS operators also possess the "ledger write" rights to propose new blocks.
- *Verify:* New transactions need to be verified by the rest of the MaaS operators before being structured into a new block. In addition, a new block also needs to be verified by all the MaaS operators before being attached to a blockchain.

B. Design Details

In the proposed scheme, the driver-characterizing performance records are disseminated through a blockchain, and we describe the blockchain design details, which include transactions, ledgers, and consensus.

- *Transaction:* A transaction generated by each MaaS operator is an elementary component in the blockchain system and each transaction contains a group of drivers' driving performance records. After the formulation of a transaction, the MaaS operator broadcasts the transaction to the entire network and waits for verification.
- *Ledger:* The blockchain ledger consists of a chain of data blocks. The "genesis" block is created during the initialization phase, and each subsequent block links to its previous block. Furthermore, in our proposed scheme, each block includes all the transactions generated during a given time slot. In addition, a selected MaaS operator collects and verifies the received transactions and then proposes a novel block.
- *Consensus:* In our proposed architecture, each MaaS operator owns a virtual stake resource, i.e., proportional to the scale of the driver performance records. Our scheme exploits a PoS consensus protocol [24], which runs a pseudorandom selection process, and the probability of being selected is proportional to the scale of the performance records. When a MaaS operator proposes a novel block, it formulates the block, attaches the block to the blockchain, and notifies the entire network.

VI. Proposed Performance Record Sharing Scheme

In this section, we propose a privacy-preserving performance monitoring mechanism. Specifically, it achieves secure performance record dissemination and acquisition among different MaaS operators. We first introduce the system initialization phase and then describe the service request phase. To clarify the proposed scheme, we alternate the illustration sequence, i.e., we present the record collection phase and then demonstrate the acquisition phase.

A. System Initialization

For the driver performance monitoring system under consideration, we assume that a trusted authority (TA), such as the traffic management authority, bootstraps the entire system.

- Given a security parameter κ, the TA selects two large prime numbers (p_1, p_2) for the modified Paillier cryptosystem with $|p_1| = |p_2| = \kappa$. Then, the TA computes the public parameters $(n = p_1 \cdot p_2, g = \mu^2 \bmod n^2)$, where $\mu \in \mathbb{Z}_{n^2}^*$ is a random number.
- Given another security parameter κ_1, the TA generates the bilinear parameters $(q, P, \mathbb{G}_1, \mathbb{G}_2, e(\cdot,\cdot))$ by running $\mathcal{G}en(\kappa_1)$ and selects a hash function $H(\cdot) : \{0,1\}^* \to \mathbb{Z}_q^*$. Moreover, the TA chooses a secret value $s \in \mathbb{Z}_q^*$, computes the system public key $pk = s \cdot P$, and generates the identity-based private key $sk_t^0 = \frac{1}{s+H(id_t)} \cdot P$, where id_t denotes the TA's identity.
- The TA identifies two public values α and u that satisfy the conditions: $\alpha^u < n$ and $|\alpha| = \kappa_2$. Additionally, the TA also selects a hash function $H_0(\cdot) : \{0,1\}^* \to \{0,1\}^{\kappa_2 - 2}$. Furthermore, the TA chooses a set of k hash functions to build a Bloom filter, $\mathcal{H} = (h_1, h_2, \ldots, h_k)$, and identifies the length of a bitmap b_0 [22].
- The TA defines an initial time TS_0 and the length of each time slot ts such that it can disseminate the records on a ts time basis. Moreover, the TA also identifies a record collection frequency f, which it uses to gather vehicular sensory data on a $1/f$ time basis.
- The TA publishes the public parameters: $params = \{(n, g), (q, P, \mathbb{G}_1, \mathbb{G}_2, e(\cdot,\cdot), pk), \alpha, u, H_0, H, \mathcal{H}, TS_0, ts\}$.

In addition, during the registration of an MaaS operator with identity id_a, the following steps are performed.

- The TA generates an identity-based private key $sk_a^0 = \frac{1}{s+H(id_a)} \cdot P$ and securely transmits sk_a^0 to the MaaS operator id_a.
- The MaaS operator id_a selects a secret key $sk_a^1 = x_a \in [1, ord(\mathbb{G})]$ for the modified Paillier cryptosystem and computes the public key $pk_a^1 = g^{x_a} \bmod n^2$.
- The MaaS operator selects another secret value t_a, publishes pk_a^1, and retains the private values (sk_a^0, sk_a^1, t_a).

Furthermore, during the registration of a vehicle with identity id_v, the TA calculates an identity-based secret key $sk_v^0 = \frac{1}{s+H(id_v)} \cdot P$ and securely forwards sk_v to the vehicle. In addition, during the registration of a driver with identity id_i, the TA performs the following procedures.

- Driver id_i selects a secret value s_i and computes a hash chain $H^m(s_i) = H(H(\ldots(H(s_i))))$ by applying the one-way function $H(\cdot)$ to value s_i m times, where each hash value stands for a pseudonym $pid_{i,j} = H^{m+1-j}(s_i), j \in \{1, 2, \ldots, m\}$. Obviously, given a pseudonym $pid_{i,j}$, it is easy to compute the backward pseudonym $pid_{i,j-1}$ but infeasible to obtain the forward pseudonym $pid_{i,j+1}$.

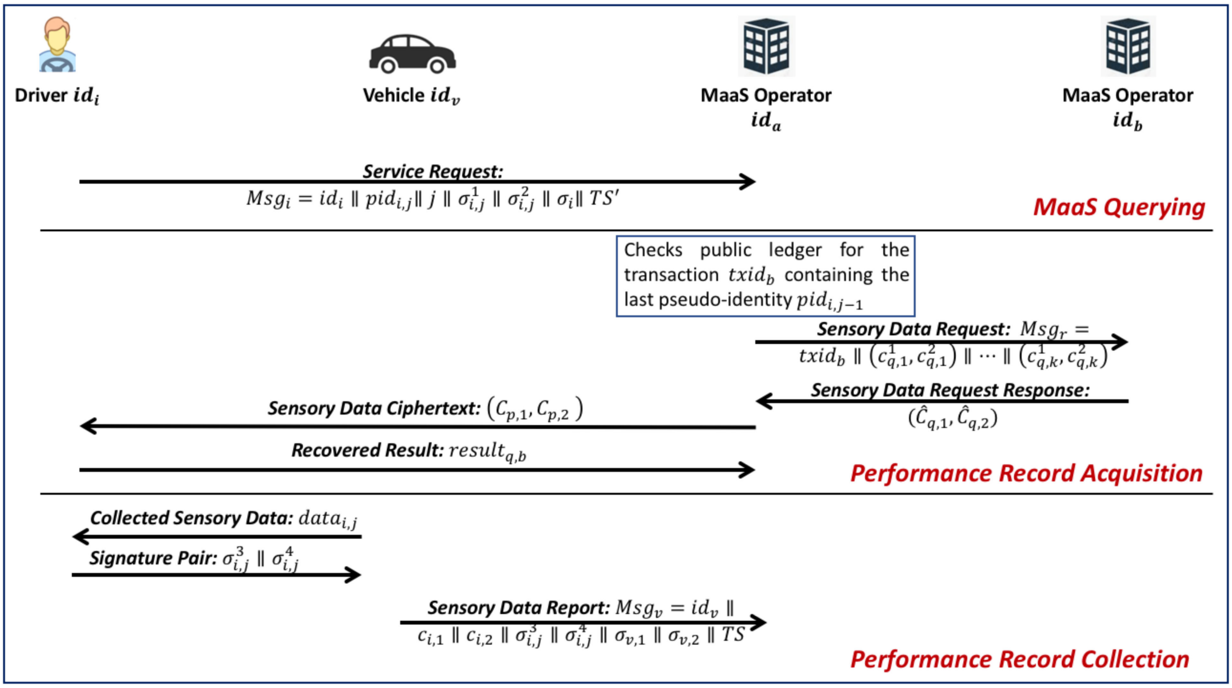

Driver id_i
Vehicle id_v
MaaS Operator id_a
MaaS Operator id_b
Service Request:
$Msg_i = id_i \parallel pid_{i,j} \parallel j \parallel \sigma_{i,j}^1 \parallel \sigma_{i,j}^2 \parallel \sigma_i \parallel TS'$
MaaS Querying
Checks public ledger for the transaction $txid_b$ containing the last pseudo-identity $pid_{i,j-1}$
Sensory Data Request: $Msg_r = txid_b \parallel (c_{q,1}^1, c_{q,1}^2) \parallel \cdots \parallel (c_{q,k}^1, c_{q,k}^2)$
Sensory Data Request Response: $(\hat{C}_{q,1}, \hat{C}_{q,2})$
Sensory Data Ciphertext: $(C_{p,1}, C_{p,2})$
Recovered Result: $result_{q,b}$
Performance Record Acquisition
Collected Sensory Data: $data_{i,j}$
Signature Pair: $\sigma_{i,j}^3 \parallel \sigma_{i,j}^4$
Sensory Data Report: $Msg_v = id_v \parallel c_{i,1} \parallel c_{i,2} \parallel \sigma_{i,j}^3 \parallel \sigma_{i,j}^4 \parallel \sigma_{v,1} \parallel \sigma_{v,2} \parallel TS$
Performance Record Collection

Fig. 4. Message flows in the performance record sharing process.

- Driver id_i sends the initial pseudonym $pid_{i,1} = H^m(s_i)$ and its real identity id_i to the TA, and the TA generates a signature for the pseudonym $\sigma_i = \frac{pid_{i,1}}{s+H(id_t)} \cdot P$. Then, the TA generates an identity-based private key $sk_i^0 = \frac{1}{s+H(id_i)} \cdot P$ for driver id_i and securely transmits (sk_i^0, σ_i) to driver id_i.
- Driver id_i also chooses another secret key $sk_i^1 = x_i \in [1, ord(\mathbb{G})]$ of the modified Paillier cryptosystem and computes the public key $pk_i^1 = g^{x_i} \bmod n^2$.
- Driver id_i publishes the public key pk_i^1 and securely keeps the secret values $(sk_i^0, sk_i^1, s_i, \sigma_i)$.

Note that the role of the TA does not conflict with the blockchain. The TA only performs parameter initialization and registration, and it remains offline after initialization.

B. MaaS Querying

As shown in Fig. 4 (*MaaS querying*), a driver id_i sends a service request to a MaaS operator id_a and performs the following steps.

Step 1: Driver id_i takes a pseudonym $pid_{i,j}$, where the pseudonyms $(pid_{i,1}, pid_{i,2}, \ldots, pid_{i,j-1})$ have been utilized before and the pseudonyms $(pid_{i,j+1}, pid_{i,j+2}, \ldots, pid_{i,m})$ are unused. Driver id_i selects a random number $r_{i,0} \in \mathbb{Z}_q^*$ and generates a signature as follows:

$$\sigma_{i,j}^1 = e(P,P)^{r_{i,0}}, \sigma_{i,j}^2 = \frac{r_{i,0} + pid_{i,j}}{s + H(id_i)} \cdot P. \tag{1}$$

Step 2: Driver id_i formulates a service request $Msg_i = id_i || pid_{i,j} || j || \sigma_{i,j}^1 || \sigma_{i,j}^2 || \sigma_i || TS'$, where TS' is the time slot of its last service, and it securely sends the message Msg_i to a chosen MaaS operator id_a.

After receiving the service request Msg_i, MaaS operator id_a performs the following steps:

Step 1: MaaS operator id_a verifies the correctness of the signature pair $(\sigma_{i,j}^1, \sigma_{i,j}^2)$, which is

$$\sigma_{i,j}^1 \cdot e(P,P)^{pid_{i,j}} \overset{?}{=} e(\sigma_{i,j}^2, pk + H(id_i) \cdot P). \tag{2}$$

Step 2: If Eq. (2) is verified to be correct, the MaaS operator id_a applies the one-way hash function to pseudonym $pid_{i,j}$ $(j - 1)$ times and derives the initial pseudonym $H^{(j-1)}(pid_{i,j}) = H^m(s_i) = pid_{i,1}$. Then, it verifies the correctness of σ_i as follows:

$$e(\sigma_i, pk + H(id_i) \cdot P) \overset{?}{=} e(P,P)^{H^{(j-1)}(pid_{i,j})}. \tag{3}$$

If Eq. (3) is also verified to be correct, MaaS operator id_a accepts the service request and queries the driver's performance history from the permissioned ledger.

To clearly illustrate the proposed scheme, we first describe the performance record collection phase (*performance record collection* in Fig. 4) instead of the performance record acquisition phase (*performance records acquisition* in Fig. 4).

C. Performance Record Collection

We assume that after the *MaaS querying* phase, MaaS operator id_a connects vehicle id_v to driver id_i (with pseudonym $pid_{i,j}$). During the MaaS period, vehicle id_v collects the e-dimensional vehicular IoT data denoted as $(d_{i,1}, d_{i,2}, \ldots, d_{i,e})$, which satisfies the following conditions: $e < u$ and $|d_{i,w} \cdot m| < \kappa_2 - 2, \forall w \in \{1, 2, \ldots, e\}$. Then, it performs the following steps to organize the obtained performance records, as shown in Fig. 4 (*performance record collection*).

Step 1: Vehicle id_v structures the e-dimensional sensory data with α: $data_{i,j} = \sum_{w=1}^{e} \alpha^w \cdot d_{i,w}$.

Step 2: To encrypt $data_{i,j}$, vehicle id_v selects a random number $r_{i,1} \in \mathbb{Z}_{n^2}$ and generates the ciphertext pair:

$$\begin{cases} c_{i,1} = g^{r_{i,1}} \bmod n^2, \\ c_{i,2} = (pk_i)^{r_{i,1}} \cdot (1 + data_{i,j} \cdot n) \bmod n^2. \end{cases} \tag{4}$$

Note that we encrypt $data_{i,j}$ with the public key pk_i, which enables the secure aggregation of multiple performance records collected during different time slots.

Step 3: To generate a signature, driver id_i selects a random number $r_{i,2} \in \mathbb{Z}_q^*$ and performs the following computation:

$$\begin{cases} \sigma_{i,j}^3 = e(P,P)^{r_{i,2}}, \\ \sigma_{i,j}^4 = \frac{r_{i,2} + data_{i,j}}{s + H(id_i)} \cdot P. \end{cases} \tag{5}$$

Step 4: Driver id_i sends the signature pair $(\sigma_{i,j}^3, \sigma_{i,j}^4)$ to vehicle id_v, and vehicle id_v verifies its correctness by checking

$$\sigma_{i,j}^3 \cdot e(P,P)^{data_{i,j}} \overset{?}{=} e(\sigma_{i,j}^4, pk + h(id_i) \cdot P). \tag{6}$$

Step 5: To guarantee the correctness of $c_{i,1}||c_{i,2}||\sigma_{i,j}^3||\sigma_{i,j}^4$, vehicle id_v chooses a random number $r_v \in \mathbb{Z}_q^*$ and generates the signature:

$$\begin{cases} \sigma_{v,1} = e(P,P)^{r_v}, \\ \sigma_{v,2} = \frac{r_v + H(id_i||c_{i,1}||c_{i,2}||\sigma_{i,j}^3||\sigma_{i,j}^4||TS)}{s + H(id_v)} \cdot P, \end{cases} \tag{7}$$

where TS denotes the current time slot.

Step 6: Vehicle id_v formulates a message $Msg_v = id_v||c_{i,1}||c_{i,2}||\sigma_{i,j}^3||\sigma_{i,j}^4||\sigma_{v,1}||\sigma_{v,2}||TS$, and sends Msg_v to MaaS operator id_a.

After receiving Msg_v, MaaS operator id_a performs the following steps.

Step 1: MaaS operator id_a verifies the correctness of $(\sigma_{v,1}, \sigma_{v,2})$ as follows:

$$\begin{aligned} &\sigma_{v,1} \cdot e(P,P)^{H(id_i||c_{i,1}||c_{i,2}||\sigma_{i,1}||\sigma_{i,2}||TS)} \\ &\overset{?}{=} e(\sigma_{v,2}, pk + H(id_v) \cdot P). \end{aligned} \tag{8}$$

Step 2: If Eq. (8) is verified to be correct, MaaS operator id_a aggregates the signature pairs $(\sigma_{i,j}^1, \sigma_{i,j}^2)$ and $(\sigma_{i,j}^3, \sigma_{i,j}^4)$ as follows:

$$\begin{cases} \sigma_{i,1} = \sigma_{i,j}^1 \cdot \sigma_{i,j}^3 = e(P,P)^{r_{i,0} + r_{i,2}}, \\ \sigma_{i,2} = \sigma_{i,j}^2 + \sigma_{i,j}^4 = \frac{r_{i,0} + r_{i,2} + pid_{i,j} + data_{i,j}}{s + H(id_i)} \cdot P. \end{cases} \tag{9}$$

Step 3: We assume that MaaS operator id_a has derived the aggregated sensory data $\sum_{t=1}^{j-1} data_{i,t}$ and the signature of driver id_i ($\hat{\sigma}_{i,1} = e(P,P)^{\hat{r}_i}, \hat{\sigma}_{i,2} = \frac{\hat{r}_i + \sum_{t=1}^{j-1}(data_{i,t} + pid_{i,t})}{s + H(id_i)} \cdot P$) from the permissioned ledger. Note that Section VI-D will illustrate the *performance record acquisition* phase. Furthermore, MaaS operator id_a selects a random number $r_i \in \mathbb{Z}_{n^2}$ and carries out the following computation:

$$\begin{cases} C'_{i,1} = g^{r_i} \cdot c_{i,1} \bmod n^2 = g^{r_i + r_{i,1}} \bmod n^2, \\ C'_{i,2} = g^{x_i \cdot r_i} \cdot c_{i,2} \cdot (1 + \sum_{t=1}^{j-1} data_{i,t} \cdot n) \bmod n^2 \\ \quad = g^{x_i \cdot (r_i + r_{i,1})} \cdot (1 + \sum_{t=1}^{j} data_{i,t} \cdot n) \bmod n^2. \end{cases} \tag{10}$$

MaaS operator id_a also aggregates the signature pair $(\sigma_{i,1}, \sigma_{i,2})$ with $(\hat{\sigma}_{i,1}, \hat{\sigma}_{i,2})$:

$$\begin{cases} \hat{\sigma}'_{i,1} = \hat{\sigma}_{i,1} \cdot \sigma_{i,1} = e(P,P)^{\hat{r}_i + r_{i,0} + r_{i,2}}, \\ \hat{\sigma}'_{i,2} = \hat{\sigma}_{i,2} + \sigma_{i,2} = \frac{\hat{r}_i + r_{i,0} + r_{i,2} + \sum_{t=1}^{j}(data_{i,t} + pid_{i,t})}{s + H(id_i)} \cdot P. \end{cases} \tag{11}$$

MaaS operator id_a also organizes all the received performance records generated during time slot TS. Additionally, MaaS operator id_a divides the performance records into a few groups, where each group contains at most k data records. Then, the following steps are performed for each formulated group with the transaction identifier $f \in \{1, 2, \ldots\}$:

Step 1: MaaS operator id_a inserts the pseudonyms into an empty Bloom filter and obtains a bitmap BF_a.

Step 2: MaaS operator id_a computes a value v_i for each driver $id_i, i \in \{1, 2, \ldots, k\}$, and this value is

$$v_i = \sum_{s=1}^{k} H_0(id_i||s) \cdot H_0(t_a||s||f||TS). \tag{12}$$

MaaS operator id_a also computes the value $V_{i,1} = H_0(v_i||id_a||0), i \in \{1, 2, \ldots, k\}$ and another value $V_{i,2} = \sum_{w=1}^{e} \alpha^w \cdot H_0(v_i||id_a||w), i \in \{1, 2, \ldots, k\}$.

Step 3: MaaS operator id_a updates the ciphertext pair for each driver $id_i, i \in \{1, 2, \ldots, k\}$ as well:

$$\begin{cases} \hat{C}_{i,1} = C'_{i,1}, \\ \hat{C}_{i,2} = C'_{i,2} \cdot (1 + V_{i,2} \cdot n) \bmod n^2. \end{cases} \tag{13}$$

Then MaaS operator id_a formulates a transaction with the steps described in Section VII.

D. Performance Record Acquisition

Now, we describe the *performance record acquisition* phase, which is shown in Fig. 4. Upon receiving a service request from driver id_i in Section VI-B, MaaS operator id_a inserts the backward pseudoidentity $pid_{i,j-1} = H(H^{m+1-j}(s_i)) = H^{m+2-j}(s_i)$ of driver id_i into an empty Bloom filter $\{0\}^{b_0}$ and derives a new bitmap BF_q. Then, MaaS operator id_a performs the following steps.

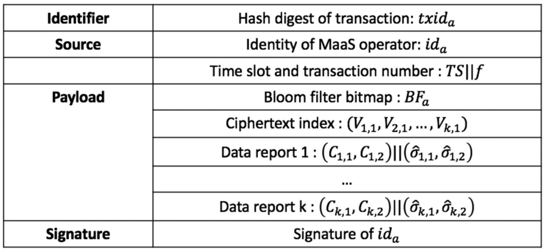
Step 1: MaaS operator id_a selects a sequence of random numbers $(r_{q,1}, r_{q,2}, \ldots, r_{q,k}) \in \mathbb{Z}_{n^2}$ and generates the following ciphertext pairs:

$$\begin{cases} c_{q,s}^1 = g^{r_{q,s}} \bmod n^2, \\ c_{q,s}^2 = pk_a^{r_{q,s}} \cdot (1 + H_0(id_i||s) \cdot n) \bmod n^2. \end{cases} \tag{14}$$

where $s \in \{1, 2, \ldots, k\}$. Then, MaaS operator id_a sends a data request $Msg_r = txid_b||(c_{q,1}^1, c_{q,1}^2)||\ldots||(c_{q,k}^1, c_{q,k}^2)$ to MaaS operator id_b.

Step 2: MaaS operator id_b performs the following computations:

$$\begin{cases} \hat{C}_{q,1} = \prod_{s=1}^{k} (c_{q,s}^1)^{H_0(t_b||s||f||TS')} \bmod n^2 \\ = g^{\sum_{s=1}^{k} r_{q,s} \cdot H_0(t_b||s||f||TS')} \bmod n^2, \\ \hat{C}_{q,2} = \prod_{s=1}^{k} (c_{q,s}^2)^{H_0(t_b||s||f||TS')} \bmod n^2 = \\ \hat{C}_{q,1}^{x_a} (1 + \sum_{s=1}^{k} H_0(id_i||s) H_0(t_b||s||f||TS') n) \bmod n^2, \end{cases} \tag{15}$$



Identifier	Hash digest of transaction: $txid_a$
Source	Identity of MaaS operator: id_a
	Time slot and transaction number : $TS\|\|f$
Payload	Bloom filter bitmap : BF_a
	Ciphertext index : $(V_{1,1}, V_{2,1}, \ldots, V_{k,1})$
	Data report 1 : $(C_{1,1}, C_{1,2})\|\|(\hat{\sigma}_{1,1}, \hat{\sigma}_{1,2})$
	...
	Data report k : $(C_{k,1}, C_{k,2})\|\|(\hat{\sigma}_{k,1}, \hat{\sigma}_{k,2})$
Signature	Signature of id_a

Fig. 5. Transaction format with performance records.

then transmits the ciphertext pair $(\hat{C}_{q,1}, \hat{C}_{q,2})$ to MaaS operator id_a.

Step-3: MaaS operator id_a decrypts $(\hat{C}_{q,1}, \hat{C}_{q,2})$ with the private key x_a and obtains the value

$$\begin{aligned} sum_{q,b} &= \frac{\hat{C}_{q,2}/(\hat{C}_{q,1})^{x_a} \bmod n^2 - 1}{n} \\ &= \sum_{s=1}^{k} H_0(id_i||s) \cdot H_0(t_b||s||f||TS'). \end{aligned} \tag{16}$$

Then, MaaS operator id_a computes the value $\hat{V}_q^1 = H_0(sum_{q,b}||id_b||0)$ and identifies the position p of the ciphertext pair $(C_{p,1}, C_{p,2})$. If $\hat{V}_q^1$ exists in the "ciphertext index" of transaction $txid_b$ (as shown in Fig. 5), MaaS operator id_a transmits the ciphertext pair $(C_{p,1}, C_{p,2})$ to driver id_i for decryption.

Step-4: Driver id_i first checks whether $(C_{p,1}, C_{p,2})$ exists in transaction $txid_b$ and decrypts it with the private key x_i, which is done as follows:

$$\begin{aligned} result_{q,b} &= \frac{C_{p,2}/(C_{p,1})^{x_a} \bmod n^2 - 1}{n} \\ &= \sum_{t=1}^{j-1} data_{i,t} + \sum_{w=1}^{e} \alpha^w \cdot H_0(sum_{q,b}||id_b||w). \end{aligned} \tag{17}$$

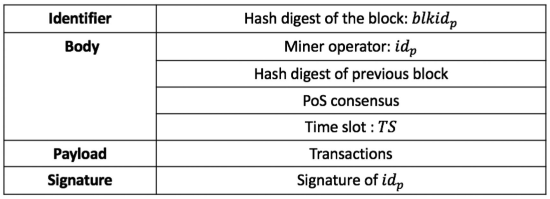
Then, driver id_i sends the $result_{q,b}$ to MaaS operator id_a.

Step 5: MaaS operator id_a recovers the value $\hat{V}_q^2 = \sum_{w=1}^{e} \alpha^w \cdot H_0(sum_{q,b}||id_b||w)$, removes $\hat{V}_q^2$ from $result_{q,b}$ in Eq. (17), and obtains the aggregated sensory data $\sum_{t=1}^{j-1} data_{i,t}$. It also performs the batch authentication process for the aggregated signature pair $(\hat{\sigma}_{i,1}, \hat{\sigma}_{i,2})$:

$$\hat{\sigma}_{i,1} \cdot e(P,P)^{\sum_{t=1}^{j-1} data_{i,t} + pid_{i,t}} \overset{?}{=} e(\hat{\sigma}_{i,2}, pk + h(id_i) \cdot P). \tag{18}$$

If Eq. (18) is also verified to be correct, MaaS operator id_a links vehicle id_v with driver id_i, provides personalized pricing and services, and resumes the *performance record collection* in Section VI-C.

Remark 2: Note that our proposed scheme mainly focuses on the vehicular IoT scenario. The proposed mechanism can also adapt to other IoT scenarios when a user's sensory data needs to be collected from multiple devices and then shared among different entities.



Identifier	Hash digest of the block: $blkid_p$
Body	Miner operator: id_p
	Hash digest of previous block
	PoS consensus
	Time slot : TS
Payload	Transactions
Signature	Signature of id_p

Fig. 6. Format of a block with miner id_p.

VII. Blockchain-Based Data Sharing

To share the obtained performance records with a permissioned ledger, each operator formulates one or a few transactions generated during each time slot. Each transaction contains a group of performance data records collected in Section VI-C. Fig. 5 illustrates the format of a transaction i, where $txid_a = H(id_a||TS||f||payload)$ denotes the transaction identifier of a transaction f.

Each transaction contains a summary of pseudonyms with a Bloom filter BF_a. A transaction also contains a ciphertext index $(V_{1,1}, V_{2,1}, \ldots, V_{k,1})$ for the involved drivers, and this enables the identification of a given driver. Furthermore, transaction $txid_a$ includes the ciphertexts of the aggregated performance data records and signatures. In addition, it generates the signature $(\sigma_{a,1}, \sigma_{a,2})$ for transaction $txid_a$, which is as follows:

$$\begin{cases} \sigma_{a,1} = e(P,P)^{r_a}, \\ \sigma_{a,2} = \frac{r_a + H(txid_a||source||payload)}{s + H(id_a)} \cdot P. \end{cases} \tag{19}$$

Then, MaaS operator id_a broadcasts transaction $txid_a$ to the entire network.

After receiving transaction $txid_a$, all the MaaS operators verify the correctness of $(\sigma_{a,1}, \sigma_{a,2})$:

$$\sigma_{a,1} \cdot e(P,P)^{H(txid_a||payload)} \overset{?}{=} e(\sigma_{a,2}, pk + h(id_a) \cdot P). \tag{20}$$

If Eq. (20) is verified to be correct, so is the transaction.

During each time slot, the MaaS operators select one MaaS operator to propose a novel block; the probability of a given MaaS operator being selected is proportional to the scale of the stake it owns, i.e., the number of performance records it has generated. Each block contains a block identifier, the miner MaaS operator, the hash digest of the previous block, the stakes of all MaaS operators, the timestamp, all the transactions generated during the given time slot, and the block's signature, as shown in Fig. 6. The miner MaaS operator creates a block, generates a signature for the block, and broadcasts the block to the entire network. In addition, the rest of the MaaS operators authenticate the block and attach it to the blockchain.

VIII. Security Analysis

In this section, we analyze the security properties of the proposed privacy-preserving performance monitoring mechanism. Since the proposed scheme exploits a permissioned block for record dissemination, it achieves the security goal of immutability. For the rest of this section, we show that the proposed scheme can achieve the security goals defined in Section III-B.

The proposed scheme can achieve the security goal of privacy preservation. During the record collection phase, each piece of performance record $data_{i,j}$ is encrypted with the public key pk_i, as shown in Eq. (4). Since the modified Paillier cryptosystem is proven to be semantically secure against the adaptive chosen-ciphertext attack, $data_{i,j}$ is also semantically secure. In addition, as we assume there is no collusion between any two entities in the proposed scheme, vehicle id_v does not disclose $data_{i,j}$ to operator id_a. Thus, each performance record can be protected during the record collection phase.

During the record acquisition phase, to hide the real identity of driver id_i, MaaS operator id_a encrypts the driver's identity id_i with the public key $pk_a = g^{x_a} \bmod n^2$ and derives the ciphertext pairs $(c^1_{q,s}, c^2_{q,s}), s \in \{1, 2, \ldots, k\}$ of the modified Paillier cryptosystem. Furthermore, as the transaction $txid_b$ under query contains a group of performance records, given the ciphertext pairs $(c^1_{q,s}, c^2_{q,s}), s \in \{1, 2, \ldots, k\}$, MaaS operator id_b cannot distinguish which driver is under query. On the other hand, based on $v_i = \sum_{s=1}^{k} H_0(id_i||s) \cdot H_0(t_b||s||TS')$, it is impossible for operator id_a to recover $(H_0(t_b||1||TS'), H_0(t_b||2||TS'), \ldots, H_0(t_b||k||TS'))$. Additionally, the aggregated record $\sum_{t=1}^{j-1} data_{i,t}$ is encrypted with the public key pk_i and further protected by v_i, which can only be collaboratively recovered by driver id_i and operator id_b.

The proposed scheme can achieve the security goal of forward separation. To guarantee the secure identification of a driver's performance records and disrupt the linkages to future data records, we exploit the one-way hash chain to generate pseudoidentities. Given a pseudoidentity $pid_{i,j}$, MaaS operator id_a can derive the previous pseudoidentities $(pid_{i,j-1}, pid_{i,j-2}, \ldots, pid_{i,1})$ and identify the transactions containing the previous pseudoidentities with false positives. Since the hash chain is a one-way function, operator id_a cannot obtain the forward pseudoidentity $pid_{i,j+1}$ and further infer the future performance data records. Thus, the proposed scheme can achieve the security goal of forward separation.

The proposed scheme can achieve the security goal of verifiability. During the data collection phase, to guarantee the correctness of each data record $data_{i,j}$, driver id_i generates a signature pair $(\sigma^3_{i,j}, \sigma^4_{i,j})$ for $data_{i,j}$. By checking the correctness of $(\sigma^3_{i,j}, \sigma^4_{i,j})$, vehicle id_v first verifies the correctness of the performance data record $data_{i,j}$. To securely transmit the ciphertext pair $(c_{i,1}, c_{i,2})$ and the signature pair $(\sigma^3_{i,j}, \sigma^4_{i,j})$ to MaaS operator id_a, vehicle id_v also generates another signature pair $(\sigma_{v,1}, \sigma_{v,2})$. As there is no collusion between driver id_i and vehicle id_v and we exploit an identity-based signature scheme [25] for signature generation (which is proven to be secure against the k-collision attack algorithm (kCAA) complexity assumption), the correctness of the ciphertext pair $(c_{i,1}, c_{i,2})$ and the signature pair $(\sigma^3_{i,j}, \sigma^4_{i,j})$ can be guaranteed by verifying $(\sigma_{v,1}, \sigma_{v,2})$. In addition, MaaS operator id_a also combines the signature pair of the pseudoidentity $(\sigma^1_{i,j}, \sigma^2_{i,j})$ with $(\sigma^3_{i,j}, \sigma^4_{i,j})$, and this guarantees the proper exploitation of the pseudoidentities. To correctly disseminate the previous performance records, MaaS operator id_a integrates the latest signature pair $(\sigma_{i,1}, \sigma_{i,2})$ with the aggregated signature pair $(\hat{\sigma}_{i,1}, \hat{\sigma}_{i,2})$ and then derives $(\hat{\sigma}'_{i,1}, \hat{\sigma}'_{i,2})$. During the data acquisition phase, upon the recovery of $\sum_{t=1}^{j} data_{i,t}$, MaaS operator id_a can verify the correctness of the performance history and the pseudoidentities with $(\hat{\sigma}'_{i,1}, \hat{\sigma}'_{i,2})$. Therefore, the security goal of verifiability can be achieved in the proposed scheme.

IX. Performance Evaluation

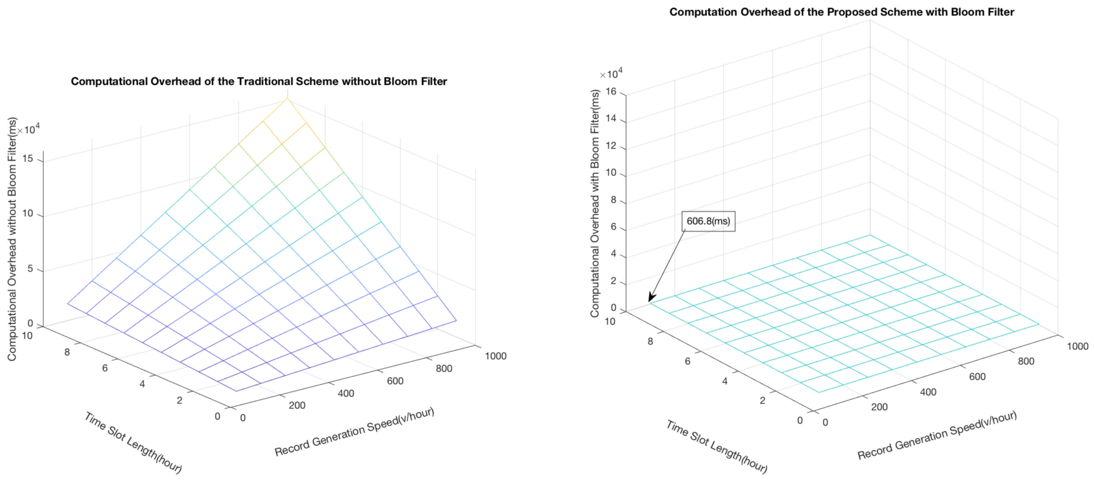
In this section, we first compare the computational complexity of our proposed scheme with those of two traditional schemes during the record acquisition phase. In addition, we analyze the computational efficiency and privacy level trade-off. We conduct experiments using a desktop with a dual-core processor Intel(R) Core(TM) i7 $-$ 8700 CPU @3.20 GHz and 8.00 GB of installed RAM on a Windows 10 Enterprise platform. In [26], the authors implemented the Paillier cryptosystem based on [27] using Java, and we exploit their codes for cryptosystem bootstrapping, as the Paillier and modified Paillier cryptosystems share the same security parameter setup. We set the security parameters mentioned in Section VI-A, i.e., the large prime numbers p_1 and p_2 are $|p_1| = |p_2| = 512$ bits, and the length of the modulus is $|n| = 1024$ bits. Based on the security parameter setup, we examine a single exponentiation operation in $\mathbb{Z}_{n^2}$ 1000 times and derive that the average computational cost of a single exponentiation operation in $\mathbb{Z}_{n^2}$ is $c_e = 7.4$ ms, which is computationally intensive.

A. Computational Complexity

To demonstrate the efficiency of the proposed scheme, we first compare the proposed scheme with a traditional scheme that does not consider the Bloom filter during the *performance record acquisition* phase. In our proposed scheme, we exploit a Bloom filter with a bitmap length $b_0 = 1024$ bits and set the number of items contained in a transaction to be $k = 20$. Thus, according to Section IV-A, the optimal number of hash functions is $z = ln2 * b_0/k = 35$, and the probability of false positives is $p_f = 1.5 * 10^{-11}$.

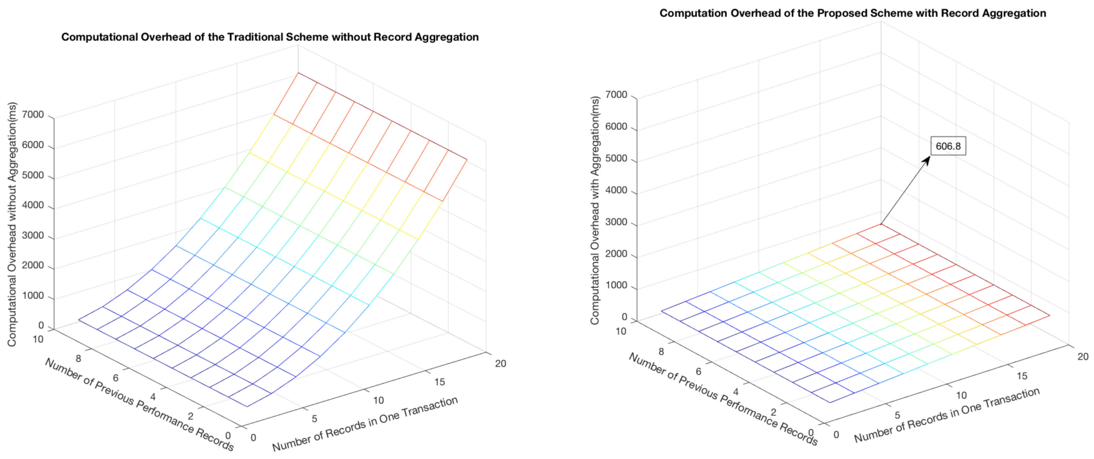
We assume that the performance records are generated at a speed of v, and the number of performance records generated during time slot ts is $v \cdot ts$. For the traditional scheme without the Bloom filter, the query operator needs to check all the ciphertext pairs generated during time slot ts to identify the target driver's ciphertext pair, and the average computational overhead is $(2 * k * \lceil v * ts/k \rceil + 2 * k + 2) \cdot c_e$. In our scheme with the Bloom filter, the computational complexity introduced by the hashing operation is negligible in comparison with that of an exponentiation operation. Then, the query operator only needs to acquire the target performance record within one transaction, and the computational complexity is $(4 * k + 2) \cdot c_e/(1 - p_f) + z \cdot c_h$.

Fig. 7(a) and Fig. 7(b) show the computational overheads incurred during the acquisition phase, where the performance record generation speed v ranges between 100 and 1000 records per hour, and the time slot length ts varies between 1 and 10 hours. As shown in Fig. 7(a), the involved computational overhead in the traditional scheme is proportional to the increases in both the speed v and time slot length ts. Additionally, as shown in Fig. 7(b), the involved computational overhead in the



(a) (b)

Fig. 7. Computational complexity comparison in terms of the bloom filter. (a) Computational complexity of the proposed scheme without a Bloom filter. (b) Computational complexity of the traditional scheme with a Bloom filter.



(a) (b)

Fig. 8. Computation Complexity Comparison in Terms of Report Aggregation. (a) Computational complexity of the proposed scheme without report aggregation. (b) Computational complexity of the traditional scheme with report aggregation.

proposed scheme is only related to the number of drivers in each transaction, and this greatly reduces the computational overhead. For example, in the most extreme case, when the performance record generation speed is 1000 per hour and the time slot length ts is 10 hours, the computational overhead of the traditional scheme without the Bloom filter is $1.48 * 10^5$ ms and that of the proposed scheme is 606.8 ms.

Then, we compare the proposed scheme with another traditional scheme, in which each transaction stores performance records without aggregation. Specifically, each transaction contains the ciphertext pair $(c_{i,1}, c_{i,2} \cdot (1 + V_{i,2} \cdot n) \bmod n^2)$ and the signature pair $(\sigma^1_{i,j} \cdot \sigma^3_{i,j}, \sigma^2_{i,j} \cdot \sigma^4_{i,j})$ for performance record $data_{i,j}$. Fig. 7(a) and Fig. 7(b) show the computational overheads incurred during the *performance record acquisition* phase, where the number of performance records contained in one transaction k ranges between 2 and 20, and the length of a driver's performance history j varies between 1 and 10. As shown in Fig. 8(a), the computational overhead of the traditional scheme without record aggregation is $(4 * k + 2) \cdot j \cdot c_e$, and it increases with both the number of performance records contained in one transaction and the length of the performance record history. As shown in Fig. 8(b), the involved computational overhead of our proposed scheme is $(4 * k + 2) \cdot c_e$, and it is only related to the number of performance records contained in one transaction. For example, in the most extreme case, when the number of performance records contained in one transaction is 20 and the length of the performance record history is 10, the computational overhead of the traditional scheme without record aggregation is 6068 ms and that of the proposed scheme is 606.8 ms.

B. Trade-Off Between Privacy and Efficiency

We consider the trade-off between the privacy level and query efficiency; this defines the privacy level relative to the number of involved indistinguishable users. Specifically, we exploit a logistic regression function to measure the privacy level [28]:

$$g_1(k) = \frac{1}{1+e^{-\beta\cdot k}}, \tag{21}$$

where $\beta > 0$ and the logistic regression function satisfies the conditions that $g_1(1) \to 0.5$ and $g_1(\infty) \to 1$. Moreover, the gradient of function $g_1(k)$ decreases as k increases, i.e., as the user scale increases, the speed of privacy level accumulation decreases. Next, we consider the trade-off between the privacy level and query efficiency. During the querying process, to hide one query among k indistinguishable queries, the involved computational overhead is $g_2(k) = (4\cdot k+2)\cdot c$, where c denotes the computational overhead for each exponentiation operation in $\mathbb{Z}^*_{n^2}$. Thus, we formulate the following objective function:

$$\begin{aligned} &\alpha_1\cdot g_1(k) - \alpha_2\cdot g_2(k) \\ &= \alpha_1\cdot\frac{1}{1+e^{-\beta\cdot k}} - \alpha_2\cdot(4\cdot k+2)\cdot c, \end{aligned} \tag{22}$$

where α_1 and α_2 are both positive. Furthermore, the optimization problem can be formulated as

$$\begin{aligned} &\max_k\ \alpha_1\cdot\frac{1}{1+e^{-\beta\cdot k}} - \alpha_2\cdot(4\cdot k+2)\cdot c \\ &\text{s.t.}\ \ k\geq 1. \end{aligned} \tag{23}$$

To derive the optimal result of k, we first check the existence of k by proving the concaveness of the objective function $f(k)$; this is done by computing its second derivative, which is

$$\frac{\mathrm{d}^2 f(k)}{\mathrm{d}k^2} = \alpha_1\cdot\frac{\beta^2\cdot e^{-\beta\cdot k}\cdot(e^{-\beta\cdot k}-1)}{(1+e^{-\beta\cdot k})^3} < 0. \tag{24}$$

Then, we compute the first derivative of $f(k)$ to obtain the value k^*, and this derivative is

$$\frac{\mathrm{d}f(k)}{\mathrm{d}k} = \frac{\alpha_1\cdot\beta\cdot e^{-\beta\cdot k}}{(1+e^{-\beta\cdot k})^2} - 4\cdot c\cdot\alpha_2 = 0. \tag{25}$$

Then, we obtain the optimal value

$$k^* = max(-\frac{log(\frac{\frac{\alpha_1\cdot\beta}{4\alpha_2\cdot c}-2-\sqrt{(\frac{\alpha_1\cdot\beta}{4\alpha_2\cdot c})^2-4\frac{\alpha_1\cdot\beta}{4\alpha_2\cdot c}}}{2})}{\beta}, 1), \tag{26}$$

when it satisfies the condition that $\alpha_1\cdot\beta > 16\alpha_2\cdot c$.

We set $\alpha_2 = 0.1$ and $c = 7.4$ ms, and we examine the optimal number of drivers k in one transaction. Fig. 9 shows the optimal values of k with respect to α_1 and β when α_1 ranges between 100 and 500 and β ranges between 0.2 and 1.0. As shown in Fig. 9, the optimal value of k increases as α_1 increases and as β decreases.

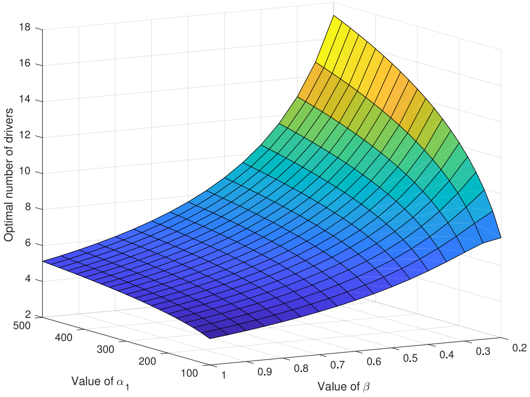


Fig. 9. Optimal values of k with respect to α_1 and β.

X. Conclusion

In this paper, the goal of privacy-preserving performance record sharing across mutually independent MaaS operators has been achieved with a PoS consensus blockchain. To acquire performance records, we exploited the Bloom filter for quick localization on the blockchain, and we obtained an aggregated performance history through an OT protocol. A security analysis demonstrated the security properties of the proposed scheme. The results of a performance evaluation have shown that our scheme achieves high computational efficiency because the computational overhead is only related to the scale of records in one transaction during the data acquisition phase. We also derived the optimal number of performance records in each transaction to balance the trade-off between privacy and efficiency. For future work, we will consider collusion attacks and design new schemes to resist them.

References

[1] Accenture, “Mobility as a service mapping a route towards future success in the new automotive ecosystem,” Accessed: Feb. 15, 2021. [Online]. Available: https://www.accenture.com/us-en/insight-mobility-automotive-ecosystem

[2] Deloitte, “Accelerating technology disruption in the automotive market blockchain in the automotive industry,” Accessed: Feb. 15, 2021. [Online]. Available: https://www2.deloitte.com/content/dam/Deloitte/cn/Documents/consumer-business/deloitte-cn-consumer-blockchain-in-the-automotive-industry-en-180809.pdf

[3] J. Sun, J. Yan, and K. Z. Zhang, “Blockchain-based sharing services: What blockchain technology can contribute to smart cities,” *Financial Innov.*, vol. 2, no. 1, p. 26, 2016.

[4] F. Yin *et al.*, “Fedloc: Federated learning framework for data-driven cooperative localization and location data processing,” *IEEE Open J. Signal Process.*, vol. 1, pp. 187–215, 2020.

[5] S. Kaplan, M. A. Guvensan, A. G. Yavuz, and Y. Karalurt, “Driver behavior analysis for safe driving: A survey,” *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 3017–3032, Dec. 2015.

[6] P. Fraga-Lamas and T. M. Fernandez-Carames, “A review on blockchain technologies for an advanced and cyber-resilient automotive industry,” *IEEE Access*, vol. 7, pp. 17 578–17 598, 2019.

[7] F. Lamberti, V. Gatteschi, C. Demartini, M. Pelissier, A. Gomez, and V. Santamaria, “Blockchains can work for car insurance: Using smart contracts and sensors to provide on-demand coverage,” *IEEE Consum. Electron. Mag.*, vol. 7, no. 4, pp. 72–81, Jul. 2018.

[8] T. Jiang, H. Fang, and H. Wang, “Blockchain-based internet of vehicles: Distributed network architecture and performance analysis,” *IEEE Int. Things J.*, vol. 6, no. 3, pp. 4640–4649, Jun. 2019.

[9] Y. Dai, D. Xu, K. Zhang, S. Maharjan, and Y. Zhang, “Deep reinforcement learning and permissioned blockchain for content caching in vehicular edge computing and networks,” *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4312–4324, Apr. 2020.

[10] Q. Kong, R. Lu, F. Yin, and S. Cui, “Privacy-preserving continuous data collection for predictive maintenance in vehicular fog-cloud,” *IEEE Trans. Intell. Transp. Syst.*, to be published, doi: 10.1109/TITS.2020.3011931.

[11] M. Enev, A. Takakuwa, K. Koscher, and T. Kohno, “Automobile driver fingerprinting,” *Proc. Priv. Enhancing Technol.*, vol. 2016, no. 1, pp. 34–50, 2016.

[12] A. Dorri, M. Steger, S. S. Kanhere, and R. Jurdak, “Blockchain: A distributed solution to automotive security and privacy,” *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 119–125, Dec. 2017.

[13] M. Li, L. Zhu, and X. Lin, “Efficient and privacy-preserving carpooling using blockchain-assisted vehicular fog computing,” *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4573–4584, Jun. 2019.

[14] F. Gao, L. Zhu, M. Shen, K. Sharif, Z. Wan, and K. Ren, “A blockchain-based privacy-preserving payment mechanism for vehicle-to-grid networks,” *IEEE Netw.*, vol. 32, no. 6, pp. 184–192, Nov./Dec. 2018.

[15] M. Li, J. Weng, A. Yang, J. Liu, and X. Lin, “Toward blockchain-based fair and anonymous ad dissemination in vehicular networks,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 11, pp. 11 248–11 259, Nov. 2019.

[16] R. Paulet, M. G. Kaosar, X. Yi, and E. Bertino, “Privacy-preserving and content-protecting location based queries,” *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1200–1210, May 2014.

[17] Q. Kong, R. Lu, M. Ma, and H. Bao, “Achieve location privacy-preserving range query in vehicular sensing,” *Sensors*, vol. 17, no. 8, p. 1829, 2017.

[18] R. Lu, X. Liang, X. Li, X. Lin, and X. Shen, “EPPA: An efficient and privacy-preserving aggregation scheme for secure smart grid communications,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 9, pp. 1621–1631, Sep. 2012.

[19] Q. Kong, R. Lu, M. Ma, and H. Bao, “A privacy-preserving and verifiable querying scheme in vehicular fog data dissemination,” *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1877–1887, Feb. 2019.

[20] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung, “Blockchain-based decentralized trust management in vehicular networks,” *IEEE Int. Things J.*, vol. 6, no. 2, pp. 1495–1505, Apr. 2019.

[21] G. Karagiannis *et al.*, “Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions,” *IEEE Commun. Surv. Tutor.*, vol. 13, no. 4, pp. 584–616, Fourth Quarter 2011.

[22] H. Song, S. Dharmapurikar, J. S. Turner, and J. W. Lockwood, “Fast hash table lookup using extended bloom filter: An aid to network processing,” in *Proc. ACM SIGCOMM*, 2005, pp. 181–192.

[23] E. Bresson, D. Catalano, and D. Pointcheval, “A simple public-key cryptosystem with a double trapdoor decryption mechanism and its applications,” in *Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur., 2003*, pp. 37–54.

[24] W. Li, S. Andreina, J. Bohli, and G. Karame, “Securing proof-of-stake blockchain protocols,” in *Proc. Int. Workshop Data Privacy Manage.*, 2017, pp. 297–315.

[25] S. Cui, P. Duan, C. W. Chan, and X. Cheng, “An efficient identity-based signature scheme and its applications,” *I. J. Netw. Secur.*, vol. 5, no. 1, pp. 89–98, 2007.

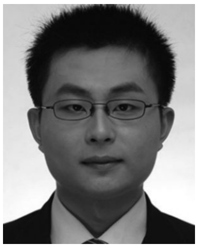
[26] K. Liu, “Paillier Cryptosystem Version 1.0,” Accessed: Feb. 15, 2021. [Online]. Available: https://www.csee.umbc.edu/ kunliu1/research/Paillier.html

[27] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in *Proc. Int. Conf. Theory Appl. Cryptogr. Tech.*, 1999, pp. 223–238.

[28] J. Friedman *et al.* “Additive logistic regression: A statistical view of boosting (with discussion and a rejoinder by the authors),” *Ann. Stat.*, vol. 28, no. 2, pp. 337–407, 2000.

Qinglei Kong (Student Member, IEEE) received the B.Eng. degree in communication engineering from the Harbin Institute of Technology, Harbin, China, in 2012, the M.Eng. degree in electronic and information engineering from the Shenzhen Graduate School, Harbin Institute of Technology (Shenzhen), Shenzhen, China, in 2015, and the Ph.D. degree from the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore, in 2018. She was a Research Scientist used to work with Cyber Security Cluster, Institute for Infocomm Research, Singapore and Tencent Security, Shenzhen, China. She is currently a Postdoc Researcher working with The Chinese University of Hong Kong, Shenzhen, China. Her research interests include applied cryptography, blockchain, VANET, and game theory.

Rongxing Lu (Senior Member, IEEE) received the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, in 2012. He is currently an Associate Professor with the Faculty of Computer Science, University of New Brunswick, Fredericton, Canada. Before that from April 2013 to August 2016, he was an Assistant Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. From May 2012 to April 2013, he was a Postdoctoral Fellow with the University of Waterloo, Waterloo, ON, Canada. He is currently a Senior Member of the IEEE Communications Society and he is currently the Vice-Chair (Conferences) of the IEEE ComSoc CIS-TC. He was the recipient of the most prestigious “Governor General’s Gold Medal,“ when he received the Ph.D. degree and the 8th IEEE Communications Society Asia Pacific Outstanding Young Researcher Award, in 2013, and the 2016-17 Excellence in Teaching Award, FCS, UNB.



Feng Yin (Member, IEEE) received the B.Sc. degree from Shanghai Jiao Tong University, Shanghai, China, in 2008, the M.Sc. and Dr.-Ing. degrees from the Technische Universität Darmstadt, Darmstadt, Germany, in 2011 and 2014, respectively. In 2014, he received MarieCurie Scholarship from European Union. From 2014 to 2016, he was with Ericsson Research, Linköping, Sweden, mainly working on the European Union FP7 Marie Curie Training Programme on Tracking in Complex Sensor Systems. He is currently working with the Chinese University of Hong Kong, Shenzhen, China and Shenzhen Research Institute of Big Data, in June 2016. His research interests include statistical signal processing, machine learning, and sensory data fusion with applications to wireless positioning and tracking. He was the recipient of the Chinese Government Award for Outstanding Selffinanced Students Abroad, in 2013.

Shuguang Cui (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, USA, in 2005. Afterward, he was an Assistant Professor in electrical and computer engineering with the University of Arizona, Tucson, AZ, USA, an Associate Professor in electrical and computer engineering with the Texas A&M University, College Station, TX, USA, a Full Professor in electrical and computer engineering with the University of California, Davis, Davis, CA, USA, and a Chair Professor in electrical and computer engineering with the City University of Hong Kong, Shenzhen, China. He was also the Vice Director with the Shenzhen Research Institute of Big Data. His current research interests include data driven large-scale system control and resource management, large dataset analysis, IoT system design, energy-harvesting-based communication system design, and cognitive network optimization. He was selected as the Thomson Reuters Highly Cited Researcher and listed in the Worlds Most Influential Scientific Minds by ScienceWatch in 2014. He was the General Co-Chair and TPC Co-Chair for many IEEE conferences, an Area Editor for the *IEEE Signal Processing Magazine*, and Associate Editor for the IEEETRANSACTIONS ON BIG DATA, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE JSAC Series on Green Communications and Networking, and the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. He was an Elected Member of the IEEE Signal Processing Society SPCOM Technical Committee (2009C2014) and the Elected Chair of the IEEE ComSoc Wireless Technical Committee (2017C2018). He is a Member of the Steering Committee for the IEEE TRANSACTIONS ON BIG DATA and the Chair of the Steering Committee for the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING. He was also a Member of the IEEE ComSoc Emerging Technology Committee. He was elected as an IEEE Fellow in 2013, an IEEE ComSoc Distinguished Lecturer in 2014, and IEEE VT Society Distinguished Lecturer in 2019. He was the recipient of the IEEE Signal Processing Society 2012 Best Paper Award.