A Hierarchical Blockchain-Enabled Federated Learning Algorithm for Knowledge Sharing in Internet of Vehicles

Haoye Chai, Supeng Leng, *Member, IEEE*, Yijin Chen, and Ke Zhang

***Abstract*— Internet of Vehicles (IoVs) is highly characterized by collaborative environment data sensing, computing and processing. Emerging big data and Artificial Intelligence (AI) technologies show significant advantages and efficiency for knowledge sharing among intelligent vehicles. However, it is challenging to guarantee the security and privacy of knowledge during the sharing process. Moreover, conventional AI-based algorithms cannot work properly in distributed vehicular networks. In this paper, a hierarchical blockchain framework and a hierarchical federated learning algorithm are proposed for knowledge sharing, by which vehicles learn environmental data through machine learning methods and share the learning knowledge with each others. The proposed hierarchical blockchain framework is feasible for the large scale vehicular networks. The hierarchical federated learning algorithm is designed to meet the distributed pattern and privacy requirement of IoVs. Knowledge sharing is then modeled as a trading market process to stimulate sharing behaviours, and the trading process is formulated as a multi-leader and multi-player game. Simulation results show that the proposed hierarchical algorithm can improve the sharing efficiency and learning quality. Furthermore, the blockchain-enabled framework is able to deal with certain malicious attacks effectively.**

***Index Terms*— Hierarchical blockchain, federated learning, knowledge sharing.**

I. Introduction

ALONG with the development of intelligent transportation system, artificial intelligence (AI)-based machine learning technologies have been widely utilized in Internet of Vehicles (IoVs) [1], [2]. Equipped with AI-enhanced on board units and sensors, IoVs have a tendency to share data among vehicles and infrastructures. The data being shared are not only limited to the sharing of computation, communication and spectrum resources [3], but also the exchange of knowledge during machine learning process. Knowledge sharing [4] allows vehicles to exchange their learning experiences. It can accelerate learning process and improve the decision capabilities. For example, the shared knowledge of the collaborative sensing of vehicles can be the learned traffic flow models. One vehicle can train an individual model of traffic flow based on its collected data. A comprehensive model can be further obtained by gathering those learning models from all vehicles. In this case, the sharing of knowledge is the reflection of swarm intelligence that is of significance for future Intelligent Transportation System (ITS) applications such as autonomous driving and traffic control [5], [6].

Despite with the aforementioned great benefits, knowledge sharing is still facing with critical challenges. On one hand, existing vehicular systems cannot guarantee the security and reliability of knowledge data during sharing process. Malicious vehicles can disturb the sharing system by sending fake knowledge or tampering received knowledge. On the other hand, most vehicular networks utilize distributed machine learning technology to realize knowledge sharing due to the decentralization feature of IoVs [7], [8]. This may cause serious privacy issues when distributed vehicular knowledge is aggregated in a central server to perform global model training, since vehicular knowledge includes sensitive personal information such as coordinates and driving preference, which closely relates to personal safety and driving condition.

Fortunately, emerging blockchain technology shows defensibility towards the security issues [9]. The interaction among vehicles during knowledge sharing can be encapsulated as a form of *transaction*, and the transaction will be audited and recorded by all peers in the network through a consensus process. The process contributes to a global and tamper-proof ledger that can maintain the security and reliability of knowledge sharing. Nevertheless, the consensus process of conventional blockchains such as Bitcoin and Ethereum rely on exorbitant computing power as well as frequent block information interactions. These systems cannot be directly applied for knowledge sharing in IoVs, since high mobility of vehicle causes unstable connectivity. Moreover, intelligent vehicles have intensive computing tasks in future applications of ITS. Hence, vehicles have no extra computation power to implement mining process [10].

To combat the privacy leakage issue, federated learning (FL) emerges as a new distributed machine learning approach which allows all nodes collaboratively train a global model in a distributed manner [11]. During the FL process, nodes only need to share the training parameters to servers, rather than sharing the total raw dataset so that the privacy risk can be avoided. Nevertheless, the large scale IoVs focus on the road traffic features, network characteristics, and local correlation in different regions [12], [13]. In this case, one

Manuscript received February 4, 2020; revised April 10, 2020; accepted June 3, 2020. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFE0117500, in part by the Science and Technology Program of Sichuan Province, China, under Grant 2019YFH0007, and in part by the EU H2020 Project COSAFE under Grant MSCA-RISE-2018-824019. The Associate Editor for this article was S. Mumtaz. *(Corresponding author: Supeng Leng.)*

The authors are with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: spleng@uestc.edu.cn).

Digital Object Identifier 10.1109/TITS.2020.3002712

common learning model is unable to adapt to multiple regions with different traffic characteristics, and the conventional FL approaches cannot be directly applied in IoVs.

In order to tackle the aforementioned problems, a hierarchical blockchain-enabled federated learning algorithm is proposed for knowledge sharing in IoVs, in which knowledge is shared in the form of learning parameters during the federated learning process. Vehicles serving as FL nodes train their own model by the collected data and share the training parameters with other vehicles. Furthermore, the blockchain is introduced to enhance the security during the sharing process. The hierarchical algorithm divides vehicles as well as infrastructures (Roadside Units and Base Stations) into several groups according to their regional features, and each group maintains an exclusive blockchain ledger to record the FL model. Because the sharing of learning parameters is embedded in the proposed hierarchical blockchain framework, the FL algorithm is also conducted hierarchically. A primary FL process is implemented among vehicles and RSUs in a small region, and a further FL process is operated among RSUs and BSs in the whole IoV. The hierarchical FL algorithm can accommodate to the variety features among different traffic regions. The contributions of the paper are summarized as follows,

- Unlike existing blockchain schemes, we propose a new hierarchical blockchain framework for knowledge sharing and a light-weight Proof-of-Knowledge (PoK) consensus mechanism. The hierarchical framework can effectively reduce the computation consumption compared with conventional blockchain system, and it is suitable for the dynamic vehicular scenarios.
- Combined with this blockchain framework, a hierarchical federated learning algorithm is proposed, which introduces a middle layer to aggregate bottom knowledge and explore the correlation of all data. The proposed learning algorithm shows superiority on the learning accuracy comparing to traditional federated learning algorithms.
- The knowledge sharing process is modeled as a multi-leader and multi-player non-cooperative game in the trading market, in which the learning parameters are traded. To the best of our known, it is the first work to employ game theory for knowledge sharing in a distributed manner.

The remainder of this paper is organized as follows. The related work of blockchain in vehicular networks and federated learning are proposed in Section II. The hierarchical model is proposed in Section III. In Section IV, a multi-leader and multi-player game is formulated for the sharing process. Section V describes the transaction process and security of the proposed PoK protocol, followed by performance evaluation in Section VI. Finally, we conclude this paper in Section VII.

II. Related Work

Federated learning (FL) is a kind of emerging distributed machine learning approach that enables nodes to learn a shared prediction model collaboratively [14]. It is a potentially effective technology for knowledge sharing in a decentralized environment. In the FL process, training nodes are named as *workers* and aggregating nodes are called as *servers*. *Workers* can share their learned knowledge with *servers* to enhance learning accuracy.

The learning process includes local update step and global aggregation step. FL aims to find a minimal global loss function L^R through minimizing the weighted sum of all the local workers' loss function [11]. Existing works about FL mainly focus on the privacy-preserving features for user data. In literature [15], the authors proposed a FL-aided vehicular communication framework, wherein vehicles serving as workers send learning results of context to the nearby RSUs. A blockchain-based FL framework was proposed in [16] to ensure the security of learning parameters. The single-point issue in FL can be resolved by means of blockchain distributed ledger. Nevertheless, most work regarded the workers as irrational nodes that would contribute their knowledge to servers unconditionally, but this is impractical in IoVs scenarios. More importantly, existing FL algorithms aggregate the knowledge to obtain one common global model, which is unsuitable for the complicated vehicular network environment. A multi-model FL algorithm should be carefully designed in IoVs.

In order to deal with the security issues during the sharing process, blockchain technology has been widely studied in IoVs to establish distributes trust. Literature [12] proposed a blockchain-enabled edge computing system, in which a consortium blockchain architecture was introduced to guarantee the security of computation results during computation process. A privacy-preserving carpooling scheme was proposed in [17], wherein a private blockchain was designed to encrypt carpooling data for processing at vehicles. The work [18] proposed a blockchain-based secure energy delivery system and [19] proposed a new debt-based data trading system. A lightweight blockchain DAG was proposed in [20], in which the directed acyclic graph can represent the relationship of transactions and the DAG-based system was suitable for the dynamic scenario in IoVs. The work in [21] integrated blockchain and federated learning for data sharing to guarantee the privacy of data. However, most existing works considered one-layer architecture, which means that all peers in blockchain system maintain only one global ledger. This is impractical for knowledge sharing in vehicular network with various local characteristics of multiple regions.

It is necessary to improve existing blockchian and federated learning algorithms towards secure and efficient knowledge sharing in IoVs. New algorithms should be suitable for large scale vehicular networks with different regional features. In this case, a hierarchical framework will be a potential solution for knowledge sharing in IoVs.

III. System Model

The hierarchical blockchain framework for knowledge sharing is presented in Fig.1, in which both Roadside Units (RSUs) and vehicles can collect their surrounding environment data to implement federated learning process. In order to encourage nodes to participate in the knowledge sharing and improve the security of the learning process, blockchain is adopted for

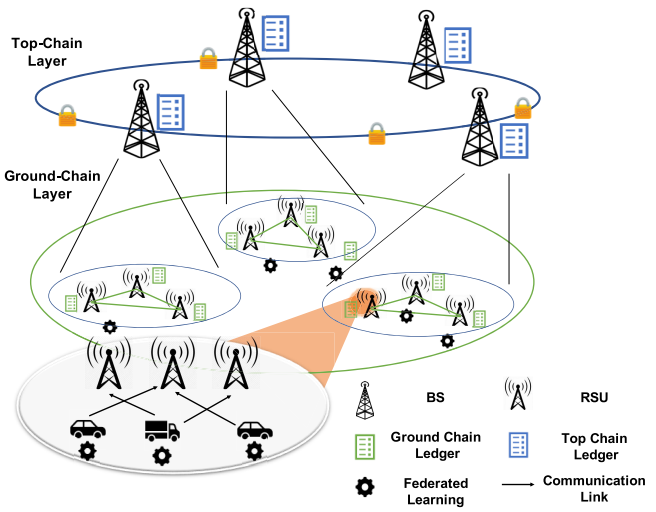


Fig. 1. Hierarchical blockchain framework.

recording the learning results. Vehicles follow diverse driving routes and collect environmental data in different regions. In this case, a hierarchical blockchain ledger is proposed to record all collected data and enable the distributed knowledge sharing process.

A. Hierarchical Blockchain Framework for Knowledge Sharing

The proposed framework consists of one Top Chain (TC) and multiple Ground Chains (GCs), in which different chains are responsible for recording different sharing knowledge of environment data.

1) Ground Chain Layer: In this layer, vehicles are responsible for collecting their surrounding environmental data as federated learning training set. Each vehicle acts as a **worker** in federated learning, which is named as FL Vehicle (FV), and implements the learning process through its own On Board Units. The learning results are sent to nearby RSU in the form of transaction. RSUs serving as **servers** are responsible for collecting the transactions within their communication ranges and package the transactions as candidate blocks. According to the variety of locations and communication ranges of RSUs, the ground chain layer contains multiple GC chains. The consensus process is implemented among RSUs located in one identical GC chain, and the published blocks reveal the federated learning results, which are also the representatives of vehicular knowledge.

2) Top Chain Layer: In addition to being servers in GC layers, RSUs are also workers of federated learning in TC layer. In this layer, the RSUs which implement FL are called FL RSUs (FRs). Considering that the FRs can also sense their surrounding environment to implement the learning process, thereafter, FRs integrate the learning results from both the GC chain ledgers and their local training results into the transactions of TC chain, and then the transactions are collected by BSs and recorded in TC ledger. In TC chain, the ledger assembles the sharing knowledge from both FVs and FRs, which can be used for traffic analysis in a global perspective. It is worth noting that RSUs have two different identities in the proposed blockchain framework. RSUs are both the publishers of blocks in GC chains and the producers of transactions in the TC chain.

B. Computation Consumption Model

For future applications of ITS, intelligent vehicles have a tendency to work collaboratively and have intensive computing tasks [23]. Therefore, it is necessary to investigate the computation consumption of vehicles during the federated learning process.

In the proposed FL process, all the workers collaboratively train their local models to obtain a shared global model. The training accuracy of the global model relies on the iterative methods with multiple communication rounds which is also named as global iterations. During each global iteration, workers send the training results (i.e. local model update) to the servers. Considering the two-layer framework, the global iteration can also be divided into two learning phases, which are named as federates learning of GC (gFL) and Federated learning of TC (tFL). We consider the framework with a set of FVs $\mathcal{N} = \{1,2,\ldots, \mathrm{N}\}$ and a set of FRs $\mathcal{M} = \{1,2,\ldots, \mathrm{M}\}$. The computation ability, i.e. CPU cycle frequency, of FV $n \in \mathcal{N}$ and FR $m \in \mathcal{M}$ can be denoted as $\{f_n^v, f_m^r\}$. Each worker with a local training dataset uses the collected data $\{d_n^v, D\}$ to participant in the federated learning process. Compared with FVs, FRs have no mobility, hence, the collected data of FRs is fixed as D.

In this paper, we consider a scenario that FRs will purchase the results from FVs in order to enhance the accuracy of tFL. Thus, FVs will accordingly split their learning results to multiple portions, and sell the portions to various FRs to make profits. Assume that one bit training data requires U CPU cycles to implement the federated learning process. According to [11], when worker FV_n sell the learning results to FR_m, the computation consumption for one local iteration can be expressed as $C_{n,m}^{cmp} = \zeta U d_{n,m}^v (f_n^v)^2$, where ζ represents the effective capacitance parameter of computing chipset. $d_{n,m}$ represents the size of training set, which is the allocated portion to FR_m. Similar with FVs, the computation consumption of client FR_m for one iteration is $C_m^{cmp} = \zeta U D (f_m^r)^2$.

In addition to the global training accuracy, there is local accuracy $\epsilon \in (0, 1)$ for local learning process which determines the local iteration times, and a higher ϵ means a more accurate learning model. According to [22], the local iteration times are independent of the choice of machine learning optimization algorithm. Given the local accuracy ϵ, the local iteration times can be expressed as

$$L = -log(1-\epsilon). \tag{1}$$

It is reasonable to consider that the local accuracy $\epsilon_{n,m}$ of worker FV_n has a direct relationship with the size of training set, i.e.

$$\epsilon_{n,m} = alog(1 + d_{n,m}), \tag{2}$$

where the logarithmic function can be explained that the accuracy of local learning cannot keep increasing with the amount of training dataset due to the blockage of overfitting [24]. Therefore, the computation consumption for local

training process of worker FV_n is

$$C_{n,m}^{gFL} = -log(1-\epsilon_{n,m})C_{n,m}^{cmp}. \quad (3)$$

After purchased the learning results from FVs, the local training accuracy of FR_m is $\epsilon_m = \alpha log(1+D+\sum_{n=1}^{N}\beta d_{n,m})$, where $\beta d_{n,m}$ represents the purchased training results from FV_n by FR_m. As the FRs cannot directly purchase training data from FVs, parameter β is set as the scale parameter, which realizes the conversion from training data to training results. Therefore, FRs can improve local accuracy through purchasing more training results from FVs in GCs. The computation consumption for local training process of FR_m is

$$C_m^{tFL} = -log(1-\epsilon_m)C_m^{cmp}, \quad \forall m \in M. \quad (4)$$

The transmission rate of learning results from worker FV_n to FR_m can be accordingly expressed as $R_{n,m} = Blog(1+\frac{P_t h_{n,m}}{\sigma^2})$, where B is the transmission bandwidth. P_t is the transmit power and $h_{n,m}$ is the channel gain between worker FV_n and server FR_m, σ denotes the noise power. Therefore, the total computation consumption of FV n for one global iteration is shown as

$$C_n^{tot} = \sum_{m=1}^{M}[C_{n,m}^{gFL} + \frac{P_t d_{n,m}}{R_{n,m}}], \quad \forall n \in N. \quad (5)$$

Similarly, the computation consumption of FR m for one global iteration is $C_m^{tot} = \frac{P_t Y}{R_m} + C_m^{tFL}, \quad \forall m \in M.$

IV. Hierarchical Federated Learning and Stackelberg Game Formulation

A. Hierarchical Federated Learning Algorithm

As the federated learning process is embedded in the proposed hierarchical blockchain framework, the sharing of training model is also hierarchical. In this paper, a hierarchical federated learning algorithm is proposed for knowledge sharing. The basic workflow of the proposed hierarchical federated learning algorithm is shown as follow:

Step 1. Federated Learning in Ground-Chain Layer: Local FVs continuously collect surrounding environment data to implement FL process. After local training, each FV sends the training result to the adjacent RSU in the form of a GC transaction, *i.e.*, TX_{GC}. The training results encapsulated in TX_{GC} can be regarded as the shared knowledge of FVs, which are the reflection of individual intelligence.

Step 2. Block Publishing in Ground-Chain Layer: Each RSU collects TX_{GC}s in the network within a predefined time interval, and then encapsulates the collected transactions into new blocks. After the consensus process, the shared knowledge are gathered at RSUs for further federated learning.

Step 3. Federated Learning in Top-Chain Layer: After receiving the newly published blocks from GC, FRs update their local training models according to the shared knowledge in the blocks. The training results are encapsulated as TC transactions, i.e. TX_{TC} and are sent to BSs for further consensus.

Step 4. Block Publishing in Top-Chain Layer: Similar to the process in GC layer, BSs collect TX_{TC}s and produce new blocks in the network. The final global model of knowledge sharing can be derived after the consensus process in TC. The proposed hierarchical federated learning algorithm is illustrated in **Algorithm 1**.

Algorithm 1 Hierarchical Weighted Updating Federated Learning

Initialization: N FVs, M FRs, learning rate η
FV executes:
$B^v \leftarrow \mathrm{FV}_n$ collects surrounding data as local training batch
for each local epoch **do**
  for batch b $\in B^v$ **do**
    learning model $\omega^v \leftarrow \omega^v - \eta \bigtriangledown l^R(\omega^v, b)$
Return: ω^v to FRs
FR executes:
$I_m \leftarrow$ learning model set of FVs collected by FR_m
for each FV in I_m **do**
  $\omega^r = \sum \frac{s_n}{S^v}\omega^v$ (s_n is the size of dataset for FV_n, S^v is the total size of dataset)
$B^r \leftarrow \mathrm{FR}_m$ collects surrounding data as further training batch
for each local epoch **do**
  for batch b $\in B^r$ **do**
    learning model $\omega^r \leftarrow \omega^r - \eta \bigtriangledown l^R(\omega^r, b)$
Return: ω^r to BSs
BS executes:
$\omega \leftarrow \sum \frac{s_m}{S^r}\omega^r$ (s_m is the size of dataset for FR_m, S^r is the total size of dataset)
Output: global learning model ω

B. Multi-Leader and Multi-Player Stackelberg Game

In the proposed hierarchical framework, most raw data are collected and trained by individual vehicles, and then RSUs purchase the learned knowledge from vehicles for further federated learning. In this case, the market between FVs and FRs is crucial for enhancing global federated learning accuracy. On the one hand, each FR would compete with others and set price to purchase vehicular training results as much as possible to improve the accuracy of further learning. On the other hand, to make more profits, FVs would collect more surrounding data for sale. Therefore, the interaction among FVs and FRs is modeled as a multi-leader and multi-player Stackelberg game, in which FRs act as leaders to set prices for the training results. Based on the prices set by all leaders, FVs then act as players to determine the optimal size of collected data.

1) The Utility of FVs: The utility of FVs includes two parts, one part is the revenue of knowledge which are sold to FRs according to their bid prices. The other part is the computation cost during data collection and local training processes. Consider the situation that FVs are not coordinated with each others, and they make decision in a distributed manner. Given the bid prices $\mathbf{P} = \{p_{1,1}, \ldots, p_{m,n}, \ldots p_{M,N}\}$ and other FVs strategies $\mathbf{D}^{-n} = \{d_{1,1}, \ldots, d_{n_0,m,\ldots}, \ldots, d_{N,M}\}_{n_0 \neq n}$,

the utility of FV_n is

$$U_n(\mathbf{d}_n|\mathbf{P},\mathbf{D}^{-n}) = C_n^{tot} - \sum_{m=1}^{M} \frac{p_{m,n}d_{n,m}}{\sum_m \sum_n d_{n,m}}, \tag{6}$$

where $\mathbf{d}_n = \{d_{n,1}, \ldots d_{n,m}, \ldots, d_{n,M}\}$ denotes the vector of FV_n's learning strategy, *i.e.*, the collected and learned data. The term $\frac{p_{m,n}d_{n,m}}{\sum_m \sum_n d_{n,m}}$ represents the contribution degree of FV_n during the knowledge sharing process, in which FV_n can achieve lower utility if it collects and learns more data. The sub-game problem of FV_n can be written as

$$\begin{aligned} \textbf{P1:}\ & \min U_n(\mathbf{d}_n|\mathbf{P},\mathbf{D}^{-n}) \\ & s.t.\ d_{n,m} \geq d_{min}, \sum_{m=1}^{M} d_{n,m}p_{m,n} \leq D_{max}, \\ & \forall m \in M, \quad \forall n \in N, \end{aligned} \tag{7}$$

where d_{min} is the minimum amount of collected data of FVs during local learning process. This is necessary to guarantee the fairness during the multi-players game. D_{max} is the maximum profits that one FV could make. The upper bound of profits D_{max} makes sure that FV cannot unboundedly increase the size of their training dataset.

2) The Utility of FRs: Considering the pricing profiles of other FRs $\mathbf{P}^{-m} = \{p_{1,1}, \ldots p_{m_0,n}, \ldots, p_{M,N}\}_{m_0 \neq m}$ as well as the FVs' strategy $\mathbf{D}$, the utility of FR m can be expressed as

$$\begin{aligned} & C_m(\mathbf{p}_m|\mathbf{D},\mathbf{P}^{-m}) \\ & = [E_m^{tot} + \sum_{n=1}^{N} p_{m,n}d_{n,m}] \\ & \quad - \frac{\sum_{n=1}^{N} p_{m,n}d_{n,m}}{\sum_N \sum_M p_{m,n}d_{n,m}}[P_0(D + \sum_{n=1}^{N} \beta d_{n,m})], \end{aligned} \tag{8}$$

where the first term of the utility equation represents the computation cost as well as the buying cost for the knowledge of FVs. The second term of the equation represents further learning revenue of FRs, which is the reward given by BSs in TC chain. It is worth noting that the revenue of further learning is also related to the contribution degree of FRs during the process of buying FVs' knowledge. In others words, one FR would acquire more revenue from BSs if it purchases more knowledge from FVs, which can make a good incentive for FRs to participant in the federated learning trading market. The prices of further learning is fixed as P_0 and only focus on the interaction among FRs and FVs. Then, the minimization problem of FR_m is formulated as

$$\begin{aligned} \textbf{P2:}\ & \min U_m(\mathbf{p}_m|\mathbf{D},\mathbf{P}^{-m}) \\ & s.t.\ p_{min} \leq p_{m,n} \leq p_{max}, \sum_{n=1}^{N} p_{m,n}d_{n,m} \geq D_{min}, \\ & \forall m \in M, \quad \forall n \in N, \end{aligned} \tag{9}$$

where p_{min} and p_{max} are the lower and upper bound of bidding prices for one FV in order to guarantee the fairness. D_{min} is the minimal sum bidding balance of FR indicating that the FRs cannot provide too low prices to purchase the learning results.

The two-layered optimization problem contains equilibrium problems at both the player and leader stage, and it is a standard Equilibrium Problem with Equilibrium Constraints (EPEC). Considering the high dimensionality of each worker's strategy, it is difficult to employ traditional backward induction method. In this case, we resort to the Alternating Direction Method of Multipliers (ADMM) algorithm to reach the social optimum point regarding of the distributed manner of the proposed hierarchical framework.

C. Alternating Direction Method of Multipliers-Based Algorithm for Multi-Leader and Multi-Player Game

In this paper, a iterated ADMM-based algorithm is designed which is utilized to resolve the proposed multi-leader and multi-player game.

1) Inner Loop: At iteration t, each FV observes the initial prices $\mathbf{P}^t = \{\mathbf{p}_1^t, \ldots, \mathbf{p}_m^t, \ldots, \mathbf{p}_M^t\}$ of FRs, where $\mathbf{p}_m^t = \{p_{m,1}^t, \ldots, p_{m,n}^t, \ldots, p_{m,N}^t\}$, and maximizes its utility function U_n according to Eq.(7).

To applying ADMM algorithm, the original problem Eq.(6) need to be modified to the standard form of ADMM consensus problem [25]. Let $\mathbf{d}_n = \{d_{n,1}, \ldots, d_{n,m}, \ldots, d_{n,M}\}$ be the variable vector, and $\mathbf{z}_n$ be an auxiliary vector, which consists of the same permutation of $\{z_{n,1}, \ldots, z_{n,m}, \ldots, z_{n,M}\}$ as $\mathbf{d}_n$. In this case, $\mathbf{d}_n$ and $\mathbf{z}_n$ have the same structure, and there is a one-to-one correspondence between the two elements. Defining Φ as the feasible set of problem Eq.(7), and an indicator function $g(\mathbf{z}_n)$ is introduced such that $g(\mathbf{z}_n) = 0$ when $\mathbf{z}_n \in \Phi$. Otherwise, $g(\mathbf{z}_n) = \infty^+$. Therefore, the original problem Eq.(7) is equivalent to

$$\begin{aligned} \textbf{P3:}\ & \min_{\mathbf{d}_n, \mathbf{z}_n} [U_n(\mathbf{d}_n|\mathbf{P},\mathbf{D}^{-n}) + g(\mathbf{z}_n)] \\ & s.t.\ \mathbf{d}_n - \mathbf{z}_n = 0, \end{aligned} \tag{10}$$

which is the general form of ADMM consensus problem. Hence, the augmented Lagrangian form of **P3** can be given as $L(\mathbf{d}_n, \mathbf{z}_n, \boldsymbol{\mu}) = U_n(\mathbf{d}_n) + g(\mathbf{z}_n) + (\rho/2)\|\boldsymbol{\mu}_n\|_2^2 + (\rho/2)\|\mathbf{d}_n - \mathbf{z}_n + \boldsymbol{\mu}_n\|_2^2$, where $\boldsymbol{\mu}_n$ is the scaled dual-variable vector corresponding to $\mathbf{d}_n$. Therefore, the iterations of ADMM for **P3** can be written as

$$\begin{aligned} \mathbf{d}_n^{s+1} = & \arg\min_{\mathbf{d}_n} \{U_n(\mathbf{d}_n) + (\rho/2) \\ & \times [(\sum_{m=1}^{M} (d_{n,m} - z_{n,m}^s + \mu_{n,m}^s)^2]\}, \end{aligned} \tag{11}$$

$$\mathbf{z}_n^{s+1} = \arg\min_{\mathbf{s}_n \in \Phi} [\sum_{m=1}^{M} (d_{n,m}^s - z_{n,m} + \mu_{n,m}^s)^2], \tag{12}$$

$$\boldsymbol{\mu}_n^{s+1} = \mathbf{d}_n^{s+1} - \mathbf{z}_n^{s+1} + \boldsymbol{\mu}_n^s, \tag{13}$$

where s represents the iteration index of inner loop of proposed ADMM algorithm. After the iteration of $\mathbf{d}, \mathbf{z}$ and $\boldsymbol{\mu}$, the optimal learning strategy of FV_n $\mathbf{d}_n$ will be obtained and sent as the output to the Outer Loop process.

2) Outer Loop: Upon receiving the strategy $\mathbf{D}$ of FVs from inner loop process, FRs are aware of the behaviours of FVs and accordingly adjust their bidding prices $\mathbf{P}$. The analysis of FRs strategy is similar to that of FVs, hence,

the elaborate description of the iteration process is omitted here. For the m-th FR, the pricing strategy can be derived through $\mathbf{p}_m^{s+1} = arg\min(\mathbf{U}_m(\mathbf{p}_m^s))$.

Then, the updated pricing strategy is updated as input of inner loop at the next iteration t+1. The whole loop of proposed algorithm breaks when the following condition holds:

$$\| \sum_{m\in M} U_m(\mathbf{p}_m^t, \mathbf{D}^t) - \sum_{m\in M} U_m(\mathbf{p}_m^{t-1}, \mathbf{D}^{t-1}) \| \leq \delta, \quad (14)$$

where δ denotes the iteration break accuracy.

According to [26], the ADMM-based algorithm can converge to the optimal results if optimization problem is convex. The convexity of the utility function of both FVs and FRs can be proved by calculating second-order derivative of the two functions, and we will omit the elaborate description of derivative process considering the space limitation.

V. Proof-of-Knowledge Based Hierarchical Blockchain

After the multi-leader and multi-player game, both FVs and FRs start to implement their local learning processes. To ensure the reliability and immutability of the sharing knowledge, workers encapsulate the knowledge into transactions and send the transactions to the hierarchical blockchain for consensus. In the following, the detail of the interaction during GCs and TC is described and the transactions format in each layer is given.

A. Federated Learning Based Transaction Process

1) Primary Learning: This stage is implemented in GCs, in which FV_n starts to collect the training data $\mathbf{d}_n$ for selling to surrounding FRs according to the optimal strategy derived in the Stackelberg game. After the local learning process, a local model update $\beta d_{n,m}$, *i.e.* the learning knowledge of $\mathbf{d}_n$ for FR_m and the corresponding local accuracy $\epsilon_{n,m}$ can be obtained. In this case, FV_n generates a learning transaction $tx_{n,m}^v$ containing $\{\beta d_{n,m}, \epsilon_{n,m}\}$, together with the signature Sig_n of FV_n. Therefore, the format of the learning transaction of FV_n can be expressed

$$tx_{n,m}^v = \{WA_{FV_n} \| 0 \| \beta d_{n,m}, \epsilon_{n,m} \| WA_{FR_m} \| Sig_n\}, \quad (15)$$

where WA_{FV_n} and WA_{FR_m} represent the wallet addresses, the second term 0 represents that the transaction is responsible for transmitting the learning results. After receiving $tx_{n,m}$, FR m first check the validity of the transaction, and then extracts the purchased learning results for further learning process.

Before the further learning process, FR m will generate a payment transaction $tx_{m,n}$, which can be shown as

$$tx_{m,n}^p = \{WA_{FR_m} \| p_{m,n} d_{n,m} \| 0 \| WA_{FV_n} \| Sig_m\}, \quad (16)$$

where $p_{m,n}d_{n,m}$ is the amount of payment for FV_n, and the third term 0 represents that the transaction is responsible for transferring asset from FR_m to FV_n. At this point, a transaction pair $\{tx_{n,m}^v, tx_{m,n}^p\}$ is gathered at FR_m. The format of GC transaction $TX_{m,n}^{GC}$ is designed as the combination of the learning transaction and payment transaction, which is designed as

$$TX_{m,n}^{GC} = \begin{array}{|c|c|c|c|} \hline ADD_s \| & Value & \| \ ADD_r \| & Sig \\ \hline WA_{FV_n} \| & 0 & \| \beta d_{n,m}, \epsilon_{n,m} \| & WA_{FR_m} \| Sig_n \\ \hline WA_{FR_m} \| p_{m,n} d_{n,m} \| & 0 & \| & WA_{FV_n} \| Sig_m \\ \hline \end{array}.$$

The ADD terms represent the wallet addresses of senders and receivers. The combination of addresses can indicate one complete interaction process between FV and FR. Afterwards, FR_m broadcasts the $TX_{m,n}^{GC}$ to other FRs in the GC chain for consensus and starts to implement the further learning process.

2) Further Learning: This stage is implemented in TC, in which each FR trains their local model according to the sensing data collected by itself as well as the purchased knowledge from FVs. The result ϵ_m is then transmitted to BS in TC chain, and a global model can be obtained. The learning and payment transactions of further learning process are also encapsulated together as the TC transactions TX_m^{TC}, which can be shown as

$$TX_m^{TC} = \begin{array}{|c|c|c|c|} \hline ADD_s \| & Value & \| \ ADD_r \| & Sig \\ \hline WA_{FR_m} \| & 0 & \| (D + \sum_{n=1}^{N} \beta d_{n,m}), \epsilon_m \| & WA_{BS} \| Sig_m \\ \hline WA_{BS} \| P_0(D + \sum_{n=1}^{N} \beta d_{n,m}) \| & 0 & \| & WA_{FR_m} \| Sig_b \\ \hline \end{array}.$$

After the consensus process among all the BSs in TC, the transactions TX_m^{TC}s are recorded in the ledger of TC and the final global model of vehicular knowledge sharing can be obtained. As the global model is trained through two stages, the further learning results reflect not only the environment features around FRs but also the features around FVs in GC chains. Therefore, the final global model achieves a higher accuracy than traditional one-layer federated learning methods. The detail of the federated learning based transaction process is illustrated in Fig.2.

The proposed framework combined both blockchain and federated learning technologies to solve the privacy issues. The blockchain technology utilizes asymmetric cryptography and digital signature to transfer the identity of vehicles into hash values, which guarantees the privacy of the peers. Moreover, the proposed hierarchical federated learning utilized parameters uploading scheme rather than row data updating, this setting can effectively protect the privacy of sharing vehicles.

B. Proof-of-Knowledge Consensus Mechanism

At each global iteration of federated learning process, the local training models of workers are gathered by server node, the global model can be trained then transmitted back to workers for the next round of iteration. In our proposed blockchain framework, the gathering of local models is executed by the consensus process. The learning knowledge are packed into blocks and consented hierarchically by FRs in GCs and BSs in TC.

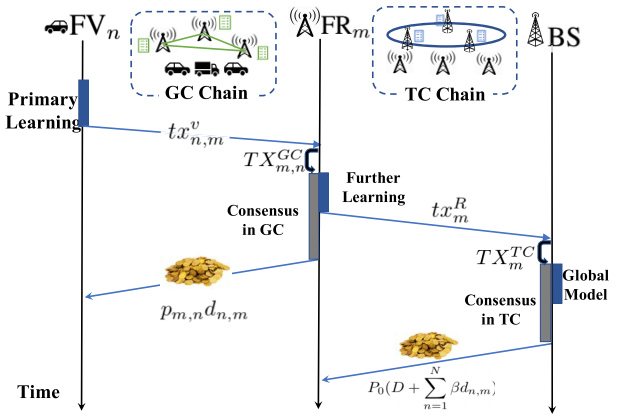


Fig. 2. Federated learning based transaction process.

Directly utilizing existing consensus mechanisms such as Proof-of-Work (PoW) for knowledge sharing either brings high waste of computing power, or introduces additional consensus latency. To encounter the problems, a lightweight consensus mechanism is proposed named as Proof-of-Knowledge (PoK). Considering the federated learning process in the framework, the computation consumption during the knowledge sharing process can also be utilized as the proof of working in blockchain. PoK combines the machine learning with blockchain consensus, which replaces the complicated hash puzzles [27] with learning process. Hence, the power utilization rate can be enhanced. Due to that the framework is decomposed into two layers and the PoK consensus process in GC layer is similar to that of TC layer. We will just give the description about PoK in GC layer. The detailed design is described in the following three steps.

1) Collecting and Verifying Transactions: Each FR continuously collects learning transactions tx^vs sent from FVs, including local learning knowledge βd and accuracy ϵ. In order to prevent FVs from uploading fake knowledge, FRs will first verify the training accuracy of FVs, by utilizing their own test set $\{(x, y)\}$. The verifying results can be obtained by the loss function

$$L^R = \sum Cost[f_{\beta d}(x) - y], \tag{17}$$

where the $Cost()$ is the specific chosen function to calculate loss function such as mean absolute error (MAE) and residual sum of squares (RSS). $f_{\beta d}$ is test model by using the received learning results from FVs. The learning transactions tx^v is considered to be reliable when the gap between L^R and ϵ locates within a certain range, i.e., $|L^R - \epsilon| \leq \theta$. After the verifying process, FRs encapsulate the tx^vs together with the payment transaction tx^ps into TX^{GC}s and broadcast TX^{GC}s to the transaction pool in GC.

2) Generating Blocks and Calculating Accuracy: After the broadcasting of TX^{GC}s, FRs start to implement the further learning process by using their own collected data and purchased knowledge. Meanwhile, FRs constantly collect TX^{GX}s in the transaction pool and pack the transactions into candidate blocks. When the further learning process expires, the further learning accuracy ϵ can be obtained and be packed into

Header	PreHash	Block Number	Timestamp	Signature
Body	Collecting Transactions of FR_m : TX^{GX}s			
	Learning Accuracy of FR_m : ϵ_m, β_m			

Fig. 3. The format of candidate block.

candidate blocks. For the m-th FR, the format of the candidate block in GC is illustrated as

In the candidate block, ϵ_m is the learning accuracy of FR$_m$, and β_m is the learning results which can be expressed as $\beta_m = (D + \sum_{n=1}^{N} \beta d_{n,m})$.

3) Consensus Process: Traditional consensus mechanism such as PoW adopts hash puzzles to decide the publisher of candidate blocks, in which the fastest solver have the right to publish the candidate bock. In this paper, the proposed PoK utilizes the best solver to substitute the fastest solver, which means that node with the most accurate learning result can publish the candidate block.

The consensus in GC is executed after the further learning process, in which the FR with the most accurate ϵ will publish the candidate block for verification. Other FRs first inspect the integrity of the block, including the signature and prehash. Afterwards, FRs calculate the loss function defined in Eq. (17) by using their own testing data, and the block would be appended in the GC ledger if the loss function is within a certain range. At this moment, the consensus process is completed and FRs prepare for a new round of federated learning.

As a proof of learning knowledge, the learning accuracy ϵ can be denoted as the contribution of FRs to the global model. Proposed PoK chooses the most accurate results, which is equivalent to choosing the one working hardest during the knowledge sharing process. Therefore, the proposed consensus mechanism can have a good incentive towards the learning workers, and the fairness can be guaranteed. Though our proposed hierarchical blockchain could enhance the computation utilization rate, it shows weakness when dealing with storage issues. Since we adopt two sub-transactions to represent one complete knowledge sharing process, the ledger of the proposed blockchain is larger than conventional blockchain system. As future work, we will focus on reducing the storage consumption to improve the proposed system.

C. Security Performance

Though the proposed PoK mechanism cancels the hash computing process, it can guarantee the fairness and incentive among blockchain nodes. Meanwhile, the proposed PoK shows defensibility towards double-spending and several malicious attacks, in accordance with mining-based blockchain systems such as Bitcoin. Next, the security performance of the proposed PoK consensus mechanism will be discussed under different malicious attacks.

1) Integrity Attack: The $integrity$ is defined that the transactions recorded in blocks cannot be tampered by adversary.

For the historical blocks in the ledger, the adversary cannot distort the content of blocks because of the $prehas$ designing. For the candidate blocks, the collecting transaction is the combination of both learning transaction and payment transaction, together with the corresponding signatures. If the adversary attempts to tamper the transactions, it cannot forge the signatures of FVs and FRs. More importantly, the address of the receiver does not correspond to the address sent by the sender, and the integrity of proposed blockchain can be guaranteed.

2) Double-Spending Attack: This is the most common attack of all the digital currency systems [28]. Double-Spending is a potential flaw in the digital cash systems in which the same single digital token can be spent more than once. The attack would be implemented by the adversary if it can control the publishing block process. For example in Bitcoin, one node could launch the double-spending attack if it holds more than 50% computing power of the networks. Nevertheless, the proposed PoK mechanism chooses the block publisher according to the learning accuracy. Because the framework is hierarchical, the learning accuracy is not only related to its own computing power, but also related to the purchase underlying data. Considering the limited budget of adversary, it cannot control the publishing of blocks, and the double-spending attack can be resolved.

3) Dishonest Behaviours: The attack is defined as reporting forged knowledge to seek profit. For malicious FV, if it sends tx^v with a forged learning results, the receiver FR would verify the results by utilizing FR' training data. As all the workers in the proposed framework are responsible for training a global model, the forged results can be detected by loss function L^R in the first step of PoK consensus process. For malicious FR, it would build a forged candidate block with wrong learning accuracy. Similarly, the block would be rejected by other FRs during the second step of the consensus. Consequently, the proposed PoK can prevent the dishonest behaviours in both GC layer and TC layer.

VI. Performance Evaluation

A. Parameter Setting

In this section, the simulation results are shown to prove the effectiveness of the proposed ADMM-based algorithm of multi-leader and multi-player game. Furthermore, the hierarchical federated learning process is investigated in terms of loss function and accuracy. Finally the security performance is discussed by compared with two other vehicular frameworks.

Considering the multi-players game with 5 FRs and 6 FVs, and focusing on the utility of four specific nodes, which are FV_1, FV_2, FR_1 and FR_2. Assume that the collecting power of FV 2 is greater than that of FV_1, thus, FV_2 can collect more training data than FV_1, i.e. $d_2 > d_1$. Similarly, FR_2 have more balance than FR_1, thus, FR_2 can provide higher bidding price than FR_1, i.e. $p_2 > p_1$.

For the learning evaluation part, we conduct the evaluation on two real-world datasets, which are widely used for data classification. The first one is MNIST, which consists of 60000 training examples and 10000 testing examples.

TABLE I
Simulation Parameters

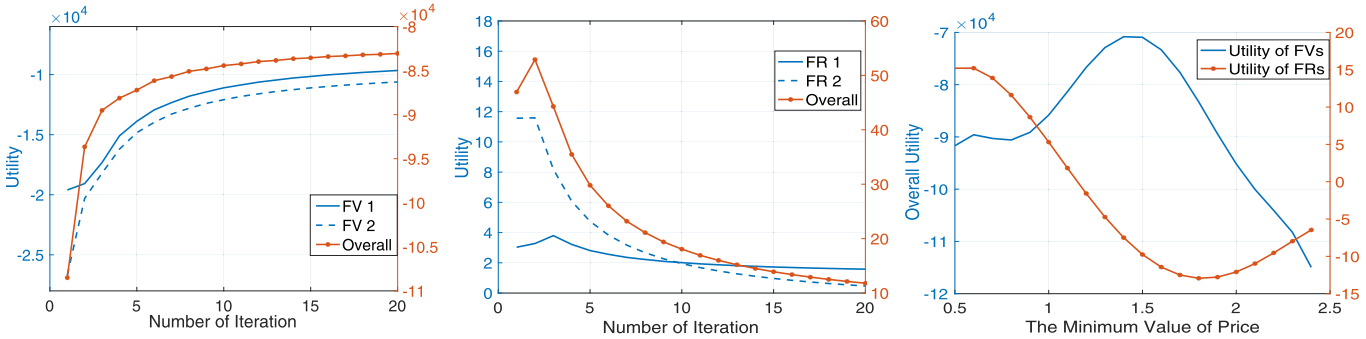
Parameters	Value
training accuracy ϵ	[0, 1]
Conversion parameters α of ϵ	0.2
The minimal amount of collected training data of FVs d_{min}	[0, 2]
The lower bound of bidding price of FRs p_{min}	[0.2, 2.4]
The upper bound of bidding price of FRs p_{max}	3
The maximum profits D_{max}	10
The minimum sum bidding balance D_{min}	5
CPU cycles for training a data sample U	5
transmit power of FVs and FRs Pt	125 mW
The price for further learning of BSs P_0	2
Pre-defined parameters	D=20, ζ =2

The second one is CIFAR10, which consists of 60000 32×32 colour images in 10 classes with 6000 images per class, and there are 50000 training examples and 10000 test examples. More configurations of key parameters are listed in Table I.

B. Numerical Results

Firstly, the convergence of the proposed ADMM-based iteration algorithm is presented. As show in Fig.4 (a) and (b), the utility of both FVs and FRs converge quickly, in which about 14 iterations are required. It can be further discovered that the utility of FV_2 is less than FV_1, this is because that the utility reveals the cost for data training according to Eq. (6), a lower utility means a higher profits which FVs can make. Due to that FV_2 can collect more training data than FV_1, it can sell more training results to FRs. Similarly, FR_2 can provide more reward for training results, it will increase its bidding price to purchase more training results from FVs to reduce its utility.

Fig. 4 (c) investigates the impact of minimum price p_{min} on overall utility of both FVs and FRs. p_{min} indicates the minimum bidding price of FRs for the knowledge of FVs, which should be dynamically adjusted according to environment states. For example, when one area has few FVs, a higher p_{min} should be settled to incentive FVs to participate in the learning process. After observing a higher bidding price, FVs will collect more training data to make profits, however, collecting more training data will contributing a higher local training cost, which results that the utility of FVs firstly increases with price. Nevertheless, as the price continuously increase, the profit of selling local learning results will greater than the local training cost, contributing the rapidly decreasing of FVs' utility. For FRs, enhancing the bidding price enables FRs to purchase more knowledge from FVs and to obtain a high accuracy learning model, hence, the utility of FRs can be decreased. However, the utility of FRs will not continuously decrease due to that the total amount of knowledge to be purchased is limited, and excessive bidding price would enhance the buying cost of FRs.



(a) Utility of FVs vs. number of iteration (b) Utility of FRs vs. number of iteration (c) Overall Utility under different prices

Fig. 4. Utilities of FVs and FRs.

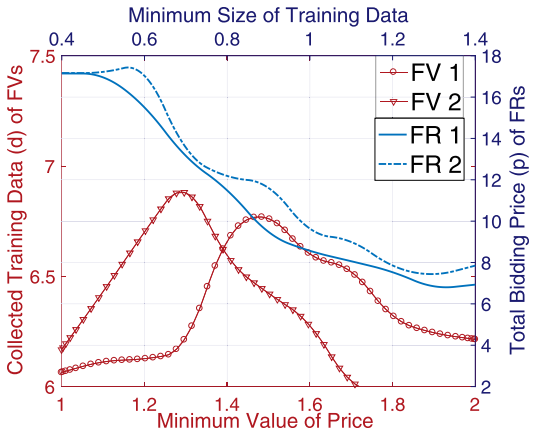


Fig. 5. The relationship between prices and collected data.

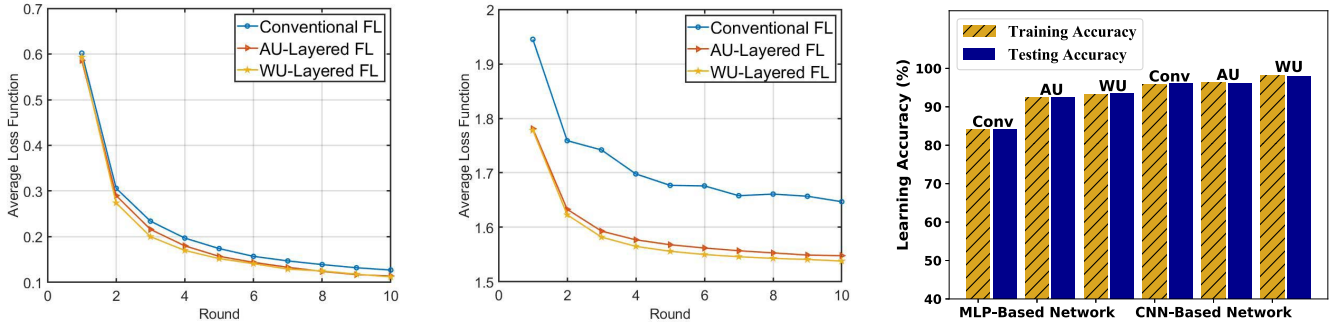
In Fig. 5, the relationship between the bidding price of FRs and the collected training data of FVs is investigated. With a determined bidding price p_{min}, the collected training data d of FVs will first increases due to the enhanced reward. However, according to the overall profit constraint defined in Eq.(7), one FV cannot obtain a unlimited rewards. Therefore, the collected data of FVs is gradually decreasing with the increasing of bidding prices. The simulation results reveal that the strategy in **P1** effectively prevents monopoly attack [29], which can be defined that a FR with excessive balance to monopolize the trading market. With the increasing of training data d_{min}, the total bidding price of FRs firstly decreases and then becomes steady. This can be explained by the constrained in Eq.(8), the provided price cannot be settled to a extremely low value. The simulation results show that the strategy in **P2** can significantly prevent the conspiracy issue [30] between FVs and FRs, in which FRs cannot provide a malicious low price to disturb the trading market.

Next, the performance of the proposed hierarchical weighted updating federated learning algorithm (WU-layered FL) is discussed. We choose two learning algorithms as the comparison group with our proposed algorithm. The comparison algorithms are non-layered conventional federated learning (conventional FL) algorithm [31] and layered average updating federated learning (AU-layered FL) algorithm, respectively. Different from conventional FL, the global loss function $l(\Phi)$ of AU-layered FL is aggregated in an average pattern rather than in a weighted pattern, which can be described as $l(\Phi) = \frac{1}{N}\sum_{n=1}^{N} l_n(\Phi)$. Two different training networks are proposed, which are Convolutional Neural Network (CNN) and Multilayer Perceptron Network (MLP).

Fig.6 illustrates the learning performance of MNIST dataset in terms of loss function and learning accuracy. Observing from Fig.6 (a) and (b), the proposed WU-layered FL achieves the lowest loss function value with respect to both CNN and MLP networks, which can prove the superiority of our proposed algorithm. Furthermore, though conventional FL utilizes weighted aggregating method of loss function, its loss function shows worse performance comparing with AU-layered FL. This effectively proves that the hierarchical learning algorithm is better than the conventional non-layered algorithm. Fig.6 (c) illustrates the performance of learning accuracy in terms of training accuracy and testing accuracy, where the proposed WU-layered FL algorithm achieves the most accurate value in both MLP and CNN networks. Compared with conventional FL, the proposed hierarchical FL algorithms enhance the learning accuracy about 10% for MLP network and 3% for CNN network.

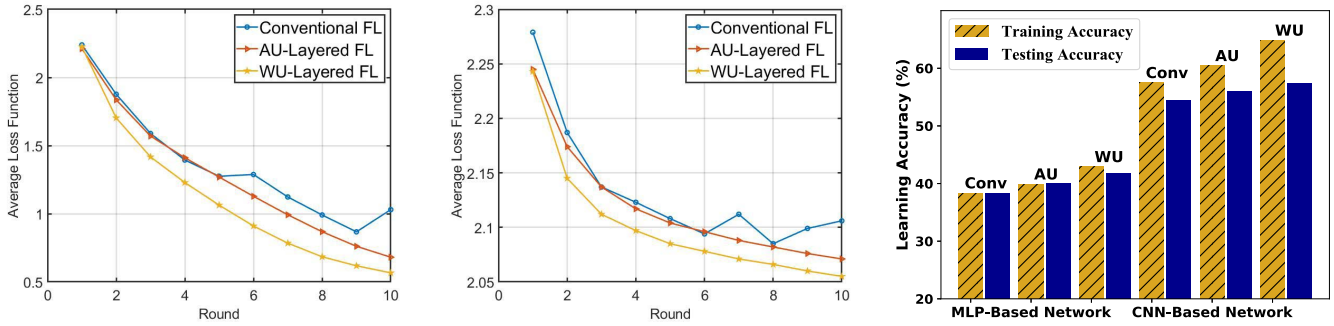
For the training of dataset CIFAR10, the loss function and learning accuracy shows s similar performance with that of dataset MNIST. The final learning accuracy of WU-layered FL achieves 5% and 8% enhancement towards conventional FL algorithm in MLP network and CNN network, respectively. From the perspective of learning performance for two different dataset, it is worth noting that the learning performance of MNIST is better than that of CIFAR10 in terms of both loss function and learning accuracy. This is because that the examples in MNIST are gray scale images with just 1 image channel, while the examples in CIFAR10 collect RGB images with 3 channels. The examples with more channels means more eigenvalues during the training process, contributing to a lower learning accuracy accordingly.

In summary, the proposed hierarchical FL algorithm outperforms conventional FL considering of loss function and accuracy. The reason can be elucidated as that the hierarchical algorithm utilizes a middle layer to collect the knowledge at the bottom of network, and then the middle layer reprocesses the bottom knowledge together with its own data. The process enhances the correlation of all the data and is the reflection of swarm intelligence, consequently increasing the final learning accuracy. This characteristic of hierarchical learning



(a) Loss Function of CNN-based Network (b) Loss Function of MLP-based Network (c) Learning Accuracy

Fig. 6. Learning performance of MNIST dataset.



(a) Loss Function of CNN-based Network (b) Loss Function of MLP-based Network (c) Learning Accuracy

Fig. 7. Learning performance of CIFAR10 dataset.

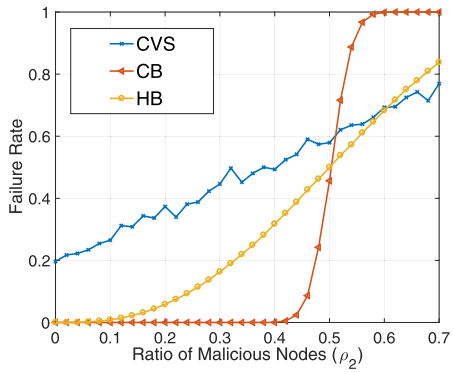


Fig. 8. Failure rate under different frameworks.

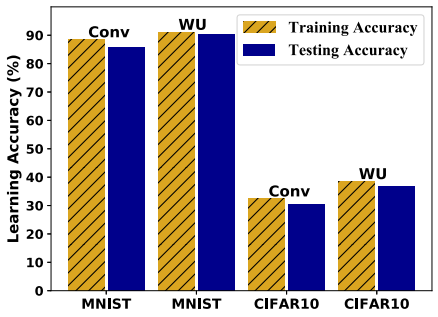


Fig. 9. Learning accuracy under the attacked of Dishonest workers.

is of significance for traffic scenarios, especially for future applications of intelligence transportation systems, where multiple vehicles and RSUs cooperate with each others to realize the prediction or analysis of traffic situation.

Then, the security performance of the proposed hierarchical blockchain (HB) is discussed. As comparison groups, two other frameworks are adopted which are conventional vehicular sensing framework (CVS) and conventional blockchain framework (CB), where CVS regards the sensing data as authentic without the consensus process and CB adopts non-layered framework such as Bitcoin. The Failure Rate is defined as the probability of recording a fake knowledge in the ledger. The probability of fake knowledge sent by vehicles or RSUs is set as $\rho_1 = 0.2$. ρ_2 is the percentage of block publishers under attack which will create false blocks during the consensus process. It can be observed in Fig.8 that HB and CB have a lower failure rate comparing with CVS when $\rho_2 \leq 0.5$. This can be explained that the blockchain frameworks adopt consensus process to audit the data while CVS directly regards the data trustworthy. However, CB collapses when $\rho_2 > 0.5$ due to the well-known 51% attack [32] while the proposed HB shows superiority towards the 51% attack. This is because the layered framework contributes to multiple ledgers, and each ledger is maintained by exclusive block publishers.

At last, the learning accuracy of the proposed hierarchical blockchain under the attack of dishonest learning nodes is investigated, in which dishonest nodes will send wrong learning results β_m or ϵ_m to servers and will degrade the learning accuracy. One benchmark blockchain is chosen as the comparison group, i.e. conventional non-layered consortium blockchain. As shown in Fig.9, the proposed hierarchical blockchain-based FL algorithm achieves a higher learning

accuracy than that of conventional consortium blockchain. This is due to that the proposed Proof-of-Leaning consensus mechanism can audit the learning results sent by FL workers, while conventional consensus mechanism only regards the results as authentic. The proposed PoK introduces a inspection process which utilize the loss function to audit the learning results, and a wrong result rent by malicious dishonest workers will be directly deleted. It can be figured that the proposed algorithm (WU-Layered) has a 5% and 7% enhancement of learning accuracy towards conventional algorithm, under the MNIST dataset and CIFAR10 dataset, respectively.

VII. Conclusion

In this paper, a hierarchical blockchain-enabled federated learning algorithm for knowledge sharing is proposed in IoVs. The hierarchical blockchain framework is able to not only improve the reliability and security of knowledge sharing, but also adapt to the large scale vehicular networks with various regional characteristics. Simulation results demonstrate the effectiveness of the proposed algorithm, by which both vehicles and RSUs can obtain the optimal utility during the sharing process. Our proposed hierarchical federated learning algorithm achieves about 10% more accuracy enhancement over conventional federated learning algorithms. Moreover, the proposed blockchain framework can effectively defend against malicious workers during the sharing process. As future work, we will study the overhead and transaction throughput of the proposed hierarchical blockchain framework in a practical IoV environment.

References

[1] J. Zhang, L. Dai, X. Li, Y. Liu, and L. Hanzo, "On low-resolution ADCs in practical 5G millimeter-wave massive MIMO systems," *IEEE Commun. Mag.*, vol. 56, no. 7, pp. 205–211, Jul. 2018.

[2] K. Xiong, S. Leng, J. Hu, X. Chen, and K. Yang, "Smart network slicing for vehicular fog-RANs," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3075–3085, Apr. 2019.

[3] J. Zhang, L. Dai, S. Sun, and Z. Wang, "On the spectral efficiency of massive MIMO systems with low-resolution ADCs," *IEEE Commun. Lett.*, vol. 20, no. 5, pp. 842–845, May 2016.

[4] X. Zhang, K.-K.-R. Choo, and N. L. Beebe, "How do i share my IoT forensic experience with the broader community? An automated knowledge sharing IoT forensic platform," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6850–6861, Aug. 2019.

[5] B. Fan, S. Leng, and K. Yang, "A dynamic bandwidth allocation algorithm in mobile networks with big data of users and networks," *IEEE Netw.*, vol. 30, no. 1, pp. 6–10, Jan./Feb. 2016.

[6] B. Ai *et al.*, "On indoor millimeter wave massive MIMO channels: Measurement and simulation," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 7, pp. 1678–1690, Jul. 2017.

[7] G. Qiao, S. Leng, S. Maharjan, Y. Zhang, and N. Ansari, "Deep reinforcement learning for cooperative content caching in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 247–257, Jan. 2020.

[8] B. Ai *et al.*, "Challenges toward wireless communications for high-speed railway," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 2143–2158, Oct. 2014.

[9] M. Wu, K. Wang, X. Cai, S. Guo, M. Guo, and C. Rong, "A comprehensive survey of blockchain: From theory to IoT applications and beyond," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8114–8154, Oct. 2019.

[10] W. Chen *et al.*, "Cooperative and distributed computation offloading for blockchain-empowered industrial Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8433–8446, Oct. 2019.

[11] J. Kang, Z. Xiong, D. Niyato, S. Xie, and J. Zhang, "Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10700–10714, Dec. 2019.

[12] L. Lei, Z. Zhong, C. Lin, and X. Shen, "Operator controlled device-to-device communications in LTE-advanced networks," *IEEE Wireless Commun.*, vol. 19, no. 3, pp. 96–104, Jun. 2012.

[13] G. Qiao, S. Leng, K. Zhang, and Y. He, "Collaborative task offloading in vehicular edge multi-access networks," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 48–54, Aug. 2018.

[14] *Federated Learning: Collaborative Machine Learning Without Centralized Training Data*. Accessed: Apr. 2017. [Online]. Available: http://ai.googleblog.com/2017/04/federated-learning-collaborative.html

[15] S. Samarakoon, M. Bennis, W. Saad, and M. Debbah, "Federated learning for ultra-reliable low-latency V2V communications," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Abu Dhabi, United Arab Emirates, Dec. 2018, pp. 1–7.

[16] H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Blockchained on-device federated learning," *IEEE Commun. Lett.*, vol. 24, no. 6, pp. 1279–1283, Jun. 2020.

[17] 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.

[18] Y. Wang, Z. Su, and N. Zhang, "BSIS: Blockchain-based secure incentive scheme for energy delivery in vehicular energy network," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3620–3631, Jun. 2019.

[19] K. Liu, W. Chen, Z. Zheng, Z. Li, and W. Liang, "A novel debt-credit mechanism for blockchain-based data-trading in Internet of vehicles," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 9098–9111, Oct. 2019.

[20] W. Yang, X. Dai, J. Xiao, and H. Jin, "LDV: A lightweight DAG-based blockchain for vehicular social networks," *IEEE Trans. Veh. Technol.*, early access, 2020, doi: 10.1109/TVT.2020.2963906.

[21] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Blockchain and federated learning for privacy-preserved data sharing in industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 6, pp. 4177–4186, Jun. 2020, doi: 10.1109/TII.2019.2942190.

[22] J. Konečný, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," pp. 1–38, Oct. 2016, *arXiv:1610.02527*. [Online]. Available: https://arxiv.org/abs/1610.02527

[23] J. Zhang, L. Dai, Z. He, S. Jin, and X. Li, "Performance analysis of mixed-ADC massive MIMO systems over rician fading channels," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 6, pp. 1327–1338, Jun. 2017.

[24] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. 31st Int. Conf. Mach. Learn. (ICML)*, Beijing, China, Jun. 2014, pp. 387–395.

[25] S. Boyd, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010.

[26] Z. Xiong, J. Kang, D. Niyato, P. Wang, and V. Poor, "Cloud/edge computing service management in blockchain networks: Multi-leader multi-follower game-based ADMM for pricing," *IEEE Trans. Services Comput.*, vol. 13, no. 2, pp. 356–367, Mar./Apr. 2019.

[27] C. H. Liu, Q. Lin, and S. Wen, "Blockchain-enabled data collection and sharing for industrial IoT with deep reinforcement learning," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3516–3526, Jun. 2019.

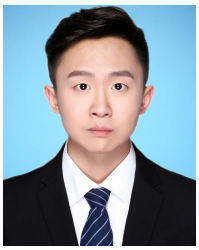
[28] Y. Liu, K. Wang, Y. Lin, and W. Xu, "LightChain: A lightweight blockchain system for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3571–3581, Jun. 2019.

[29] W. U. Mondal and G. Das, "Blocking predation in cellular monopoly through non-linear spectrum pricing," *IEEE Commun. Lett.*, vol. 21, no. 11, pp. 2464–2467, Nov. 2017.

[30] E. K. Wang, Y. Li, Y. Ye, S. M. Yiu, and L. C. K. Hui, "A dynamic trust framework for opportunistic mobile social networks," *IEEE Trans. Netw. Service Manage.*, vol. 15, no. 1, pp. 319–329, Mar. 2018.

[31] H. B. McMahan *et al.*, "Communication-efficient learning of deep networks from decentralized data," pp. 1–11, Feb. 2017, *arXiv:1602.05629*. [Online]. Available: https://arxiv.org/abs/1602.05629

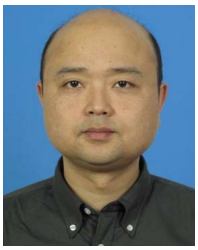
[32] N. Schweitzer, A. Stulman, R. D. Margalit, and A. Shabtai, "Contradiction based gray-hole attack minimization for ad-hoc networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 8, pp. 2174–2183, Aug. 2017.



Haoye Chai received the B.Sc. degree in information and communication engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2016, where he is currently pursuing the Ph.D. degree. His research interests include mobile edge computing, the Internet of Vehicles, and blockchain in wireless networks.

Supeng Leng (Member, IEEE) received the Ph.D. degree from Nanyang Technological University (NTU), Singapore. He is a Full Professor and the Vice Dean of the School of Information and Communication Engineering, University of Electronic Science and Technology of China (UESTC). He is also the Leader of the Research Group of Ubiquitous Wireless Networks. He has been working as a Research Fellow with the Network Technology Research Center, NTU. His research interests include resource, spectrum, energy, routing, and networking in the Internet of Things, vehicular networks, broadband wireless access networks, smart grid, and the next generation mobile networks. He has published over 180 research articles in recent years. He serves as an Organizing Committee Chair and a TPC member for many international conferences, as well as a reviewer for over ten international research journals.

Yijin Chen received the B.E. degree from the University of Electronic Science and Technology of China, in 2015, where she is currently pursuing the Ph.D. degree with the School of Information and Communication Engineering. Her research interests include malware propagation modeling and the security in machine learning.



Ke Zhang received the Ph.D. degree from the University of Electronic Science and Technology of China, in 2017. He is currently a Lecturer with the School of Information and Communication Engineering, University of Electronic Science and Technology of China. His research interests include the scheduling of mobile edge computing, design and optimization of next-generation wireless networks, and the Internet of Things.