Blockchain Empowered Asynchronous Federated Learning for Secure Data Sharing in Internet of Vehicles

Yunlong Lu, *Student Member, IEEE*, Xiaohong Huang, *Member, IEEE*, Ke Zhang, Sabita Maharjan, *Senior Member, IEEE*, and Yan Zhang, *Fellow, IEEE*

***Abstract*—In Internet of Vehicles (IoV), data sharing among vehicles for collaborative analysis can improve the driving experience and service quality. However, the bandwidth, security and privacy issues hinder data providers from participating in the data sharing process. In addition, due to the intermittent and unreliable communications in IoV, the reliability and efficiency of data sharing need to be further enhanced. In this paper, we propose a new architecture based on federated learning to relieve transmission load and address privacy concerns of providers. To enhance the security and reliability of model parameters, we develop a hybrid blockchain architecture which consists of the permissioned blockchain and the local Directed Acyclic Graph (DAG). Moreover, we propose an asynchronous federated learning scheme by adopting Deep Reinforcement Learning (DRL) for node selection to improve the efficiency. The reliability of shared data is also guaranteed by integrating learned models into blockchain and executing a two-stage verification. Numerical results show that the proposed data sharing scheme provides both higher learning accuracy and faster convergence.**

***Index Terms*—Data sharing, Blockchain, Asynchronous federated learning, Deep reinforcement learning, Internet of Vehicles.**

I. Introduction

THE rapid development of new computing and communication technologies in 5G networks and beyond opens up possibilities for advanced vehicular services and applications such as autonomous driving and content delivery, which can yield improved driving experience. In this context, Internet of Vehicles (IoV), a new paradigm that integrates intelligent computing and vehicle's networking into vehicular networks [1], emerges as a crucial area. In the IoV, a large amount of diverse types of data is constantly generated by the moving vehicles, which includes additional data such as trajectories, traffic information and multimedia data. How to efficiently and effectively utilize the massive amount of available data to improve the driving experience and to provide extensive high-quality services in IoV, is a problem of paramount importance.

Data sharing can mitigate the problem by analyzing and mining data collaboratively for improving the quality of IoV applications. However, in IoV, data sharing faces two crucial challenges. First, the vehicles need to share data efficiently despite unreliable inter-vehicle communications. How to improve data sharing efficiency and reliability needs further and thorough investigation. Second, data providers are getting increasingly concerned about data security and privacy issues that can discourage them from providing the data available with them for analysis. How to share data efficiently and securely in IoV, therefore, remains an open research problem.

Multi-access Edge Computing (MEC) enables edge resource sharing by performing the computing and content storage [2] at the edge of mobile networks, through Device-to-Device communication (D2D) [3]. In [4], the authors exploited the DRL-inspired MEC solution to design an optimal edge content caching scheme taking mobility into account in vehicular networks. In [5], the authors addressed the assignment problem of the radio channels of the nodes to promptly construct the dynamic D2D-enabled wireless network by exploiting partially overlapping channels and game theory. Despite these studies focusing on efficiency of MEC, further investigations on how to achieve distributed intelligence in MEC are still needed. In this regard, some recent works have exploited edge intelligence for resource sharing in vehicular networks. For example, in [6], the authors adopted Deep Reinforcement Learning (DRL) for designing a data transmission scheduling scheme to minimize transmission costs in vehicular networks . However, the security problem of resource sharing in distributed scenarios remains unsolved.

Recently, blockchain has emerged as a promising technology to provide distributed secure solutions [7]. With the advanced features such as tamper-proof, anonymity and traceability, blockchain has attracted significant attention for enhancing security in areas such as Internet of Things (IoT) [8], vehicular

Manuscript received September 18, 2019; revised December 9, 2019; accepted January 16, 2020. Date of publication February 13, 2020; date of current version April 16, 2020. This work was supported in part by Joint Funds of National Natural Science Foundation of China and Xinjiang under Project U1603261, in part by the Key R&D Project of Sichuan Province under Grant 2019YFG0520, and in part by the Opening Project of Shanghai Trusted Industrial Control Platform under Grant 2019YY201. The review of this article was coordinated by Prof. J. Ren. *(Corresponding author: Yan Zhang.)*

Yunlong Lu and Xiaohong Huang are with the Institute of Network Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: yunlong.lu@ieee.org; huangxh@bupt.edu.cn).

Ke Zhang is with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 610051, China (e-mail: zhangke@uestc.edu.cn).

Sabita Maharjan is with the Simula Metropolitan Center for Digital Engineering, 0167 Oslo, Norway, and also with the University of Oslo, 0316 Oslo, Norway (e-mail: sabita@simula.no).

Yan Zhang is with University of Oslo, Norway, and also with Simula Metropolitan Center for Digital Engineering, Norway (e-mail: yanzhang@ieee.org).

Digital Object Identifier 10.1109/TVT.2020.2973651

networks [9] and smart grids [10]. A series of works have studied leveraging blockchain for data sharing in vehicular networks. For instance, in [11], the authors exploited consortium blockchain, which is maintained by Road Side Units (RSUs) for achieving secure data sharing in vehicular edge networks. In [12], the authors designed a blockchain empowered secure data sharing architecture for distributed multiple parties. Although use of blockchain offers the possibility to enhance security of data sharing, it may also adversely affect the efficiency aspects due to the need of additional computing and communication for maintaining the blockchain.

To improve the efficiency and intelligence of blockchain, some studies have explored integrating blockchain with Artificial Intelligence (AI). In [13], the authors proposed a secure and intelligent architecture by integrating AI and blockchain into wireless networks for secure resource sharing in 5G beyond. The authors in [14] improved the blockchain framework by offloading the computation-intensive mining tasks to nearby MEC nodes. However, while studying the security and privacy issues of data and the network in these integrated frameworks is a vital research direction, it has witnessed rather limited work. To this end, mitigating the resource cost for integrating blockchain with AI demands closer and further investigation.

Federated learning [15], [16] is a promising approach for privacy preserved edge intelligence in distributed scenarios. While in conventional machine learning, all training data is collected at a centralized curator, federated learning addresses the privacy concerns to a large extent, and also reduces data transmission cost by distributing the training work to users themselves. The local training is executed by users on their own data, which usually adopts the gradient descent optimization algorithm [17]. In a federated learning framework, users keep their data with themselves but send the parameters to the server for aggregation. This provides a parallel scheme for users to learn a global model collaboratively with respect to their data privacy. Thus federated learning achieves edge intelligence by learning from distributed data in a privacy preserved manner, and exploits blockchain to provide a guaranteed collaboration scheme among untrusted participants for efficient sharing.

However, in an IoV, a highly dynamic environment due to the mobility of vehicles and unreliable inter-vehicle communications, give rise to a number of new challenges to be solved. Three aspects are crucial in this context. First, the computing efficiency of blockchain, needs to be improved. Second, the reliability of shared data needs to be guaranteed. The risk that providers share unqualified data such as malicious and redundant data, should be mitigated. Third, the delay due to federated learning should be reduced to deal with the heterogeneous communication and computing capabilities of the vehicles.

In this paper, we address these issues by integrating blockchain and federated learning into IoV for data sharing. We develop a hybrid blockchain - PermiDAG, and improve the federated learning with our node selection algorithm. The contributions of this paper can be summarized as follows.

- We propose a new hybrid blockchain - PermiDAG, which consists of a main permissioned blockchain maintained by the RSUs and the local Directed Acyclic Graph (DAG) run by the vehicles for efficient data sharing in IoV.
- We propose an asynchronous federated learning scheme for learning models from the edge data, and further improve the efficiency of federated learning by selecting the participating nodes to minimize the total cost.
- We enhance the reliability of learned models by integrating the learned parameters into blockchain and verifying the qualities of these parameters through two-stage verification.

The rest of this paper is organized as follows. Related work is discussed in Section II. The architecture of our blockchain empowered federated learning framework is presented in Section III. We further analyze the hybrid blockchain framework in Section IV. In Section V, we present our DRL enabled optimal node selection algorithm for the blockchain empowered federated learning framework in detail. We illustrate numerical results in Section VI. Section VII concludes the paper.

II. Related Work

The last few years have witnessed the rapid developments in 5 G technologies, in particular the D2D communication. D2D communication has been widely applied in wireless networks such as cellular networks [18], [19], and ad hoc networks [20]. In [21], the authors studied the outage probability of D2D communication and analyzed the downlink outage probability in a multichannel environment. Moreover, the concept of multihop D2D communication network systems was proposed in [22], which are applicable to many different wireless technologies with clarified requirements. These works on D2D-enabled wireless networks open up promising possibilities for efficient and reliable resource sharing in IoV, where users can share their resources through Vehicle-to-Vehicle (V2V) communication.

In D2D-enabled wireless networks, MEC plays a crucial role in particular for resource management, data sharing and content caching. In [23], the authors proposed a privacy-preserving data sharing scheme with the assistance of fog node, which leverages MEC for data analysis and encryption. In [24], [25], the authors lowered the delay and raised the scalability of MEC by using the proposed intelligent resource optimization scheme which simultaneously considers communication, computation, and migration in mobile networks. Integrating MEC in vehicular networks, the authors in [26] studied the resource allocation problem by combining load balance with offloading for a multi-user multi-server vehicular edge computing system. The rise of AI brings new possibilities for MEC to achieve edge intelligence. Some works have utilized the advanced DRL for resource allocation. For instance, in [27], the authors adopted a deep Q-learning approach for optimal data transmission scheduling in cognitive vehicular networks to minimize transmission costs. Although these works provide effective MEC schemes to share resources at edge, the security and privacy problem [28] remains a severe threat to edge users.

Blockchain has been widely used to address security issues in distributed scenarios. In terms of resource trading, blockchain plays the role of payment platform, which guarantees the transaction security in Peer-to-Peer (P2P) trading. The authors in [29] proposed a localized P2P electricity trading model for locally buying and selling electricity among electric vehicles.

In [30], the authors studied the resource management and pricing problem between the provider and miners by modeling the interaction between them as a Stackelberg game. Blockchain has also been used for securely providing computing services. A blockchain-based fair service-provisioning scheme was proposed to address the security problems in untrusted and distributed IIoT scenarios [31]. The consensus mechanism, tamper-proof records, and smart contract technologies in blockchain enable secure trading among participants without central authorities. A few works have also utilized blockchain for data sharing in vehicular networks. A data sharing and storage system based on the consortium blockchain (DSSCB) [32] was proposed to address the identity validity and message reliability issues in a vehicular ad-hoc network. In [11], the authors proposed a reputation-based blockchain scheme to ensure security and traceability of data sharing in vehicular networks.

Consensus protocols constitute one of the most important components of a blockchain in terms of both the structure and computation and communication requirements. The Proof-of-Work (PoW) protocol has been widely adopted for blockchain for vehicular networks [11], [14]. The PoW based mechanism can prevent attacks such as the Distributed Denial-of-Service (DDoS) and Byzantine attacks. However, the PoW mechanism costs much computing resource thus introducing scalability and efficiency issues. To improve the efficiency of the consensus process, Delegated Proof of Stake (DPoS) [33] has also been explored for application in vehicular networks. DPoS uses real-time voting combined with a reputation system to achieve consensus, which can perform the consensus process in a more efficient and democratic mechanism. Recently, the DAG enabled blockchain has emerged as a new technology to improve the efficiency and scalability of traditional blockchain. DAG is a directed graph data structure that has a topological order. DAG blockchain uses cumulative PoW instead of the computation-sensitive traditional PoW protocol for achieving efficient consensus. The DAG-based blockchain has been widely used in IoT [34] and 5 G beyond networks [35].

With blockchain to assure the security among untrusted edge users, integration of blockchain and AI has yielded noticeable performance improvement for resource sharing in wireless networks. In [13], the authors proposed a secure and intelligent resource sharing architecture for next-generation wireless networks by integrating AI and blockchain into them. Although the integration enhances the performance of security and efficiency for resource sharing in wireless networks, conventional AI algorithms depend much on the centralized data for training, which are not applicable for distributed scenarios in IoV. Moreover, conventional centralized AI algorithms may incur various security issues such as information leakage [36] and privacy attacks.

In such case, federated learning emerges as a privacy-preserving paradigm for distributed edge intelligence [37]. In [38], a joint transmit power and resource allocation approach for enabling ultra-reliable low-latency communication in vehicular networks was proposed. Some works have improved the federated learning scheme with control algorithms. The authors in [39] formulated federated learning over wireless networks as an optimization problem that captures computation and communication latencies, and energy consumption.

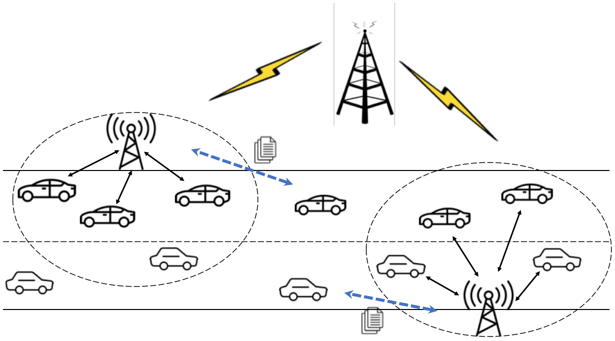

Fig. 1. Data sharing in vehicular networks.

In conventional synchronous federated learning [15], [16], [38], every node fetches the global model from the server and pushes the updates to the server. The server then synchronizes all the updates and aggregates them into the global model. Synchronous learning incurs high communication cost, while also leading to higher idle durations waiting for slower nodes. Several studies have explored the asynchronous learning mechanism to improve learning efficiency. For example, in [40], the authors proposed an asynchronous mini-batch algorithm by exploiting multiple processors to solve regularized stochastic optimization problems. To improve the efficiency of federated learning, we propose an asynchronous federated learning scheme based on our node selection and asynchronous aggregation algorithms.

III. Blockchain Empowered Asynchronous Federated Learning

A. System Model

The vehicular networks consist of vehicles, RSUs, Macro Base Stations (MBSs), as shown in Fig. 1. The MBSs have large capacity of computing and communication resources. Denote the set of vehicles by $V = \{v_i\}$. Vehicles in V have limited computing and communication resources, which may include mobile devices in the vehicles in addition to the on board units. RSUs are equipped with MEC servers, and have certain computing and storage capabilities. RSUs connect to the MBSs through uplink communication and connect to the vehicles within their range through downlink communication. The data is mainly transmitted through V2V and Vehicle-to-RSU (V2R) communication. Suppose that a vehicle v_{req} submits a request for sharing a certain kind of data, with a specific purpose such as traffic prediction or path selection. For vehicle v_{req} with a data sharing request Req, the goal is to leverage the shared data D to obtain the computing results Res. We treat this data sharing process as a computing task. Let $V_I = \{v_1, v_2, .., v_n\}$ be the vehicles with the requested datasets $D = \{D_1, D_2, .., D_n\}$, respectively. The computing task is to learn a data model $\mathcal{M}$ from D to address the data sharing task from v_{req}.

The vehicles are untrusted in our system, which means that a vehicle may turn malicious and aim to get data from others without any cost or obstruct the learning process. The dishonest

TABLE I
SUMMARY OF MAIN NOTATIONS

$F(w)$	The global loss function in federated learning
$w_i(t)$	Local model parameters learned by vehicle i in slot t
$m_i(t)$	Local model learned by vehicle i in slot t
$M(e)$	Global model in episode e
$c_l^t(i)$	The local learning time cost of vehicle i in slot t
$c_c^t(i)$	The communication cost of vehicle i in slot t
c_{te}^t	The time efficiency cost in slot t
c_q^t	The cost of learning quality in slot t
c^t	The total cost in time slot t
W	The weight of transactions in DAG
CW	The cumulative weight of transactions in DAG
P_{xy}	The transition probability of transaction $x \to y$ in DAG
λ_t	The $0-1$ selection state of vehicles in slot t
s_t	The system state of slot t in DRL
R	The reward function in DRL
θ_π	The parameters of actor network
θ_Q	The parameters of critic network

vehicles may provide fake local models to the federated training scheme, which can lead to the failure of the whole learning process. The resources of a vehicle are characterized in the forms of its available computation resource to execute computing tasks in a certain period of time (CPU cycles/s), denoted by ξ_i, its available transmission rate for communication, denoted by τ_i, and its quality of models. We quantify the quality of models with the learning quality denoted by $\sigma_i = \sum_j L(y_j - \hat{y}_j)$, which is related to the quality of data and the honesty of v_i. The data owned by v_i determines the contribution of its learning results to the global computing results. Note that both computing capability and communication capability may vary with time due to the parallel computing tasks and dynamic networks.

B. Federated Learning

We leverage federated learning to fulfill the computing task towards a data sharing request. The vehicles V_I are the training nodes (clients) and the MBS is the aggregator (server).

The main notations in this paper are summarized in Table I.

For vehicle $v_i \in V_I$ with dataset D_i, a loss function quantifies the difference between estimated values and real values of samples in D_i, defined as

$$F_i(w) = \frac{1}{|D_i|} \sum_{j \in D_i} f_j(w, x_j, y_j), \tag{1}$$

where $f_j(w, x_j, y_j)$ is the loss function on data sample (x_j, y_j) with parameter vector w and $|D_i|$ is the size of data samples in D_i. In a specific algorithm, the detailed loss function is usually defined according to the computing tasks, such as the Mean Square Error (MSE) and the Mean Absolute Error (MAE). Thus, the global loss function $F(w)$ is defined as

$$F(w) = \frac{1}{|V_I|} \sum_{i \in I} c_i \cdot F_i(w) = \frac{1}{|V_I|} \sum_{i \in I} \sum_{j \in D_i} c_i \cdot \frac{f_j(w, x_j, y_j)}{|D_i|}, \tag{2}$$

where $I = \{1, 2, \ldots, n\}$ is the number set of vehicles V_I, and C_i is the contribution capability factor denoting the contribution of v_i to the global federated learning, and $\sum_i c_i = 1$.

In our training process, we improve the accuracy of model $\mathcal{M}$ iteratively by minimizing the global loss function $F(w)$ towards the gradient descent direction. The objective is to minimize the loss function $F(w)$

$$Q(w, t) = \underset{i \in I, t \le T}{\arg\min} F(w) \tag{3}$$

$$\text{s.t.} \quad Pr\left(w_i \in \mathbb{R}_d\right) \le exp(\epsilon) Pr\left(w_i^{'} \in \mathbb{R}_d\right), \tag{4}$$

$$\sum_{i=1}^{t} \Delta t(i) \le \min\left(T_1, T_2, \ldots, T_n\right). \tag{5}$$

where $Pr(w_i \in \mathbb{R}_d) \le exp(\epsilon) Pr(w_i^{'} \in \mathbb{R}_d)$ is the ϵ privacy guarantee for parameters w_i [41]. $\{T_1, T_2, \ldots, T_n\}$ is the connection time of each vehicle with the MBS, t is the maximum number of iterations and $\Delta t(i)$ is the execution time of iteration i. Constraint (5) ensures that the vehicles can connect to the MBS during learning phase.

Due to the distributed data sets and privacy concerns of multiple vehicles, we exploit blockchain and federated learning to address Problem (3), which can utilize the limited resources of multiple vehicles and protect their data privacy. Problem (3) is a combinatorial optimization problem that is hard to find a closed-form solution.

C. The Architecture of Blockchain Empowered Asynchronous Federated Learning

Conventional federated learning depends on a synchronous learning scheme to update models between the server and clients. However, there are two main challenges associated with this approach. First, in vehicular networks, the learning time of each vehicle varies due to their heterogeneous computing capacity and dynamic communication condition. Thus, the running time of each learning iteration is decided by the slowest participants, while others have to wait for the slowest one to maintain the synchronous scheme. We propose asynchronous federated learning to address this issue. Our proposed scheme executes asynchronous learning by optimally selecting the participating nodes and splitting the aggregation slot into local aggregation and global aggregation slots. Second, the parameters transmitted between participating nodes raise serious security and privacy issues, and low communication reliability due to dynamic channel conditions exacerbate the reliability of transmission of these parameters. We integrate blockchain to store and verify the model parameters, which can enhance the reliability and security of the proposed scheme. Moreover, we adopt the DRL algorithm, based on Actor-Critic (AC) reinforcement learning framework, to select the participating nodes in our proposed asynchronous federated learning.

The proposed blockchain empowered asynchronous federated learning architecture consists of three phases: node selection, local training, and global aggregation, as shown in Fig. 2. The node selection formulates and solves an optimization problem by using DRL algorithm to select participating vehicles. Then the selected vehicles perform local training and update their trained local models for global aggregation.

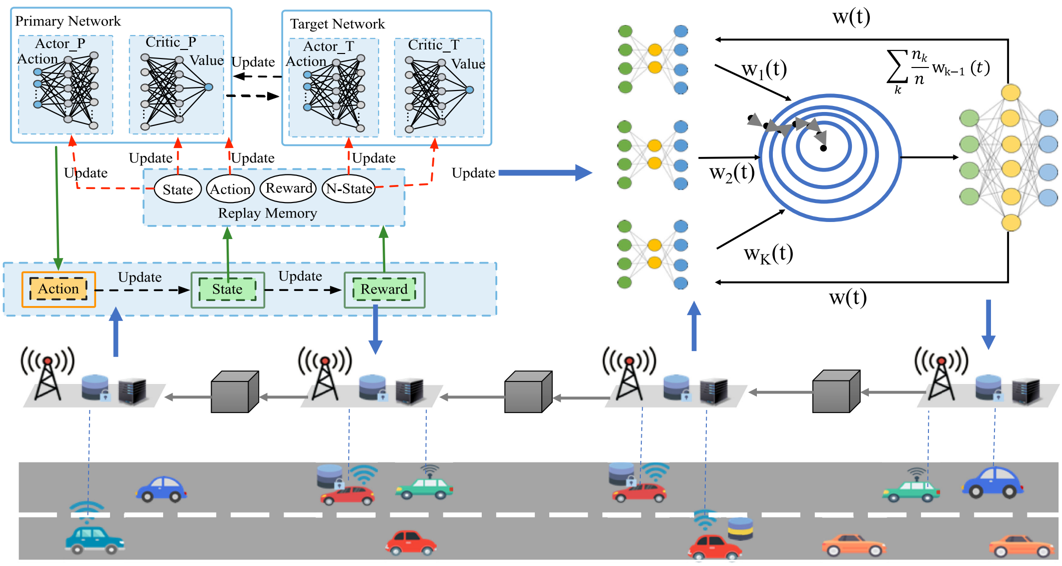

Fig. 2. The architecture of blockchain empowered federated learning for data sharing in IoV.

- *Node selection:* To improve the running efficiency and training accuracy, node selection chooses the nodes with higher amount of resources in the given communication time to participate in federated learning. The selected nodes also play the role of verifier for the permissioned blockchain. At the beginning, the server initializes the federated learning processes by choosing a global machine learning model and initializing parameters $\boldsymbol{w}_{ini}$. Then the server selects the optimal nodes $V_P \subset V_I$ with high computing and communication capacity $\xi_i \cdot \tau_i$ through DRL-based algorithm, and distributes the parameter vector $\boldsymbol{w}$ to each node $v_i \in V_P$.
- *Local training:* The local training is implemented with distributed gradient descent. In iteration t, each participating vehicle $v_i \in V_P$ trains a local model $w_i(t)$ on its data D_i according to $\boldsymbol{w}_{t-1}$, by computing the local gradient descent $\nabla F_i(w_{t-1})$, as shown in (6). v_i then sends the parameters $w_i(t)$ of the trained local model to the nearby RSU and uploads it to the blockchain for further verification and aggregation.

$$w_i(t) = w(t) - \eta \cdot \nabla F_i(w_{t-1}), \tag{6}$$

where η is the learning rate of distributed gradient descent.
- *Aggregation:* The aggregator retrieves the updated local parameters from the permissioned blockchain and executes global aggregation by aggregating the local models $w_i(t)$ from participating nodes to a weighted global model $w(t)$.

$$w(t) = \frac{\sum_{i=1}^{N} C_i w_i(t)}{\sum_{i=1}^{N} C_i}, \tag{7}$$

where N is the number of nodes, and C_i is the contribution of node i to the whole training process in iteration t.

The process of blockchain empowered federated learning scheme is depicted in Fig. 3. The MBS first distributes the global model to the blockchain. Then the participating vehicles download the global model from blockchain and train their local models. A certificate authority performs the identification of participants to access the permissioned blockchain. The trained local model parameters are then uploaded to the blockchain. The MBS retrieves the parameters and executes global aggregation. Note that in the proposed blockchain empowered federated learning scheme, we improve the aggregation efficiency by splitting the aggregation phase into local aggregation and global aggregation phases. For each vehicle v_i, the local aggregation is executed between a local range of vehicles asynchronously, to improve the quality of trained local models. The global aggregation is performed by the RSUs synchronously, which consumes more computing and communication resources. We further propose the hybrid blockchain scheme to perform the combined aggregation, which includes the lightweight local aggregation and the resource-intensive global aggregation. The hybrid blockchain scheme will be described in Section IV.

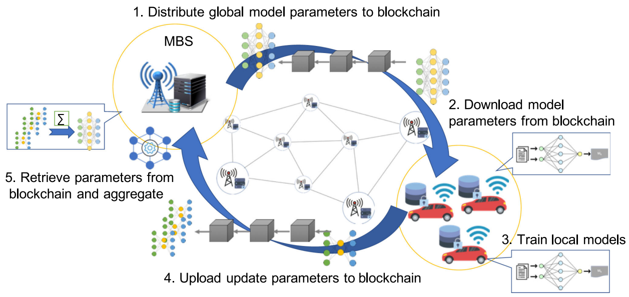
1. Distribute global model parameters to blockchain
MBS
2. Download model parameters from blockchain
Σ
5. Retrieve parameters from blockchain and aggregate
3. Train local models
4. Upload update parameters to blockchain

Fig. 3. The process of blockchain empowered federated learning scheme.

IV. The Hybrid Permissioned Blockchain Scheme: PermiDAG

Due to the time-varying topology and strict delay requirement of vehicular networks, it is hard to maintain a computation efficient blockchain during the V2V data sharing process. Moreover, the process to upload and retrieve parameters is time-consuming due to the synchronization and verification of chain data between participants for achieving consistency in the permissioned blockchain.

To enhance data security, training efficiency, and accuracy, we design a hybrid blockchain mechanism - PermiDAG for our federated learning scheme. The PermiDAG consists of a main permissioned blockchain and the local DAG, which are responsible for the synchronous global aggregation and asynchronous local training in our federated learning scheme, respectively. The PermiDAG is partition-tolerant, which means part of the nodes can also run the blockchain effectively. Moreover, the storage efficiency is improved by letting vehicles only store the local DAG and letting the RSUs store the permissioned blockchain. Based on the hybrid blockchain mechanism, our federated learning scheme is adjusted in the following phases:

- *Node selection:* As before, we select participating nodes $V_P \subset V_I$ through the node selection algorithm. The nodes in V_P then vote for the corresponding delegates (RSUs) to maintain the permissioned blockchain.
- *Local training and aggregation:* The participating vehicles V_P train their local model $m_i(t)$ based on the global model retrieved from the main permissioned blokchain. For vehicle $v_i \in V_P$, it retrieves the verified local models of other participating vehicles from local DAG and executes the local aggregation to improve its local model. Then vehicle i adds its local model $m_i(t)$ to the local DAG as a transaction to be verified. The process is repeated by all vehicles in V_P for several slots.
- *Global aggregation:* After several slots of local training and aggregation, the delegates (RSUs) collect current local models from the local DAG, and perform the global aggregation based on Eq. (7). Then the global model $M(t)$ is broadcasted to all delegates. All the global models $M(t)$ in several episodes will be collected into a block by the rotating leader of delegates, and the candidate block will be appended to the permissioned blockchain after verification.

A. The Hybrid Blockchain Scheme

In PermiDAG, the permissioned blockchain runs in the RSU nodes and acts as the main blockchain, while the local DAG runs on the vehicle nodes. The main permissioned blockchain records all the data sharing events between vehicles, including the providers, consumers, data profiles, and the summary information of transactions in local DAG. Moreover, the permissioned blockchain also records the model parameters in the global aggregation. The vehicles register with the certificate authority (RSUs or MBSs) to become legitimate vehicle nodes of permissioned blockchain, which can obtain the certificate for participating in the V2V data sharing. In the data sharing process, the subset of vehicle nodes $V_P \subseteq V_I$ are selected by our node selection algorithm, as presented in Section V. The selected vehicle nodes are responsible for training local models and running the local DAG. The RSU nodes collect the transactions into blocks and verify the blocks to further add them to the permissioned blockchain.

We leverage local DAG to improve the efficiency of blockchain in delay-sensitive V2V data sharing. The local DAG records the model parameters in our local training process together with the related data sharing events between vehicles as transactions. Thus there are two types of transactions in our scheme: data sharing events and the trained model parameters. The transactions are stored as the nodes of DAG, and the nodes are connected by edges established according to the approval relations between transactions. The DAG is kept by vehicles locally and is updated asynchronously for achieving asynchronous consistence with other vehicle nodes. The asynchronous consistence allows the vehicles to reach an agreement on the historical state, instead of the current state. The update of a local DAG_i from vehicle i is transmitted to its nearby vehicles for synchronization through the gossip scheme. Each vehicle i randomly gossip its latest DAG_i, including transactions and approval relations, to its neighboring vehicles. The gossip scheme disseminates the DAG updates and maintains a relaxed consistency among vehicles, which is less computation-intensive and globally robust.

The data sharing related transactions in the local DAG are synchronized periodically to the main permissioned blockchain. The synchronized transactions record the data requesters, the data providers, the parameters of the global model aggregated from local updates by Eq. (7), and the pointer information between approved transactions (i.e. edges) in DAG. We also use reputation to quantify the performance of a participant in the data sharing process. The participants who provide high-quality data earn high reputation scores. In a data sharing process, we calculate the cumulative reputations of participants based on their verified accuracies and record the reputations in the local DAG.

To ensure the quality of shared data, we develop a two-stage verification mechanism for our proposed hybrid blockchain. Besides the regular validation of transactions and blocks performed by RSUs in the permissioned blockchain, the nodes in V_P also perform the first-stage quality verification in local DAG, which verifies the quality of transactions (trained data models) in our federated learning process.

B. DPoS-Based Consensus in Permissioned Blockchain

We adopt the highly efficient consensus protocol DPoS in our proposed permissioned blockchain. In conventional DPoS, the delegates are voted based on stakes to maintain the blockchain. We formulate the delegates selection problem in permissioned blockchain and participants selection problem in federated learning into a combinational node selection problem in our proposed scheme. In our proposed scheme, the delegates are RSUs voted by the selected participating vehicles $V_P \subset V_I$. The RSUs which support more selected participating vehicles $v \in V_P$ have high probability to be elected as the delegates. The elected delegates govern the running of permissioned blockchain by managing the settings such as the block intervals and block sizes. In our scheme, the delegates also play the role of witness, which are responsible for validating the transactions, generating and verifying the new blocks. In each block verification slot, a leader is selected based on historical performance and random factors from the delegates. The leader broadcasts the candidate blocks

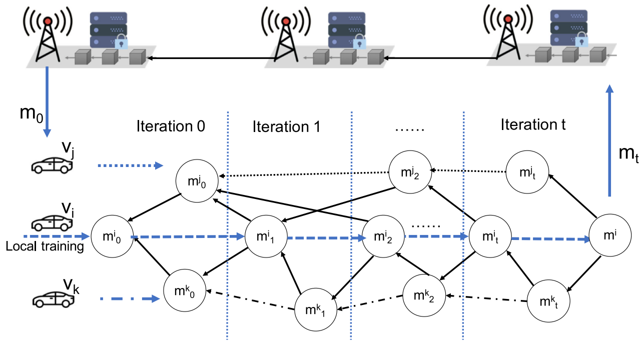

Fig. 4. The hybrid blockchain mechanism.

to the verifiers for validation. The verifiers then validate the transactions in the block and return the audit results to the leader. The leader collects all received audit results to decide whether to commit the candidate block. If the candidate block passes all the verification, the leader will send the block to all participating RSUs for updates and storage in the permissioned blockchain.

C. *Local DAG for Quality Verification Based on Reputation in IoV*

We consider the reputation of vehicles in the IoV for node selection to achieve consensus among vehicles. How to minimize the resource cost led by the additional quality verification is a major concern in our two-stage verification scheme. We combine the quality verification process with the update process of federated learning by using the proposed DAG-based local blockchain in the IoV.

1) The Local DAG: In the proposed local DAG, the transactions are the shared update models in federated learning, which we named as "micro-transactions" to distinguish from the data sharing transactions. The local DAG is maintained locally by each participating vehicle. In iteration t of the federated learning, an aggregator v_a is selected from the participating nodes of federated learning, which are also the verifiers of the permissioned blockchain. To boost the aggregation process, an efficient method is to select the node with long training time (to parallel the aggregation process) and good communication state as the aggregator. In local aggregation, a participating vehicle $v_i \in V_P$ transmits its model updates $m_i(t)$ to the nearby vehicles for aggregation through V2V communication. The local update events are recorded in the local DAG as transactions.

The structure of the local DAG is shown in Fig. 4. Each transaction denotes the parameters of an updated local model $m_i(t)$ in the federated learning process. The weight of a transaction $W(m_i(t))$ is proportional to the model accuracy and the computing resource the issuing vehicle invested into it, which is calculated by the issuing vehicle as

$$W(m_i(t)) = \frac{|d_i| + \rho \cdot \sum_j d_{m_j}}{\sum_{i=1}^{N} |d_i| + \sum_j d_{m_j}} \cdot s_i \cdot Acc\left(m_i(t)\right), \quad (8)$$

where $|d_i|$ is the size of data vehicle i used for local training, $\sum_j d_{m_j}$ is the accumulated data size of the local models used for local aggregation by vehicle i, $\rho \in [0, 1)$ is a small control factor, s_i is the training slots executed by vehicle i in its local training, and $Acc(m_i(t))$ is the accuracy provided by the issuing vehicle i.

To ensure the reliability of the transaction weight, we define the cumulative weight of a transaction $CW(m_i(t))$ based on its own weight and the sum of its reputations estimated by other M transactions based on the verified accuracy, as

$$CW(m_i(t)) = W(m_i(t)) + \frac{1}{M} \sum_{j=1}^{M} \Delta Acc_j \cdot W(j) \quad (9)$$

where $\Delta Acc_j = Acc_j(m_i(t)) - W(m_i(t)), W(j)$ is the weight of transaction j which verifies $m_i(t)$, and Acc_j is the verified accuracy of $m_i(t)$ calculated by the issuing vehicle of transaction j.

2) Adding a Transaction to DAG: To add a model update transaction to the DAG, vehicle v_i first needs to validate two transactions in local DAG by calculating the accuracy of the updated models on its own dataset. Vehicle v_i then attaches the hashes of the two validated transactions to the new transaction. The new transaction is then appended to the DAG, which will be broadcasted to the nodes in the local DAG. Since there are usually more than two unverified transactions, vehicle v_i uses the weighted walk to choose transactions for verification. The Markov-chain Monte Carlo (MCMC) method is employed to simulate the probability of each step towards unverified transactions, and the transaction with high probability will be chosen by the walker. The MCMC simulation is based on (10)

$$E[f(x)] \approx \frac{1}{m} \sum_{i=1}^{m} f\left(x_i\right),$$
$$(x_0, x_1, \ldots, x_m) \sim MC(p) \quad (10)$$

where $MC(p)$ is the Markov Chain generated by Markov Process. The weighted walk decides its steps based on the "high cumulative weight" strategy by walking towards the transactions with high cumulative weight. Through the weight-biased random walk, the transactions with high probability to become confirmed transactions are selected. So that the transaction issued by the verifier can be approved with high probability in the long term.

3) Confirmation and Consensus: The confirmation of transactions are based on their cumulative weights. We leverage the reputation-based weighted walk to select the unverified transactions for validation. When issuing a new transaction, two walkers are placed on the DAG to choose the transactions which have been verified by more transactions and have the higher cumulative weight for validation. The walkers walk towards the unverified transaction y from x with a transition probability P_{xy}, defined as

$$P_{xy} = \frac{e^{CW(y) - CW(x)}}{\sum_{z:z \to x} e^{CW(z) - CW(x)}} \quad (11)$$

where z is the neighboring transaction that points to transaction x, and $y \in \{z : z \to x\}$.

The consensus is based on the rule of "the heaviest branch". Due to the fact that most nodes are honest and want their transactions to be verified in a short term, they are more likely to choose the transactions with high weight to validate. In the DAG,

Algorithm 1: The Hybrid Blockchain Empowered Federated Learning.

```
Input: The registering vehicles as participating nodes
  $V_I = \{v_1, v_2, \ldots, v_N\}$, the dataset of vehicle $i$, $d_i \in D$.
Input: Initialize the permissioned blockchain $B$ and DAG.
  Initialize the initial global model $M_0$
Input: Select the participating vehicles $V_P \subset V_I$ by
  running node selection algorithm. Vote the delegates
  $r_1, r_2, \ldots, r_n$
for each episode $e$ do
  Select a leader $r_0$ from delegates
  for each time slot $t$ do
    for each vehicle $v_i \in V_P$ do
      $v_i$ retrieves global model $M_{t-1}$ from
        permissioned blockchain $B$
      $v_i$ executes the local training on its local data $d_i$
        based on Eq. (6)
      $v_i$ retrieves local model updates from DAG
      $v_i$ executes local aggregation and obtain updated
        local model $m_i(t)$
      $v_i$ add the parameters of model $m_i(t)$ as a
        transaction to the DAG
    end for
  end for
  The leader $r_0$ retrieves the current verified updated
    models from DAG, and aggregates the models into
    $M(e)$ based on Eq. (7)
  $r_0$ broadcasts $M(e)$ to other delegates for
    verification, and collects all transactions into a new
    block
  $r_0$ appends the block including the global model
    $M(e)$ to the permissioned blockchain
end for
return The parameters of the final global model $M$.
```

the "heaviest branches" are more likely to be confirmed, while the branches with low cumulative weight will be isolated at last. The consensus process is secured by using simplified PoW in a distributed manner. Vehicle i obtains the right to add a transaction by computing the simplified PoW locally. The accumulation of the distributed verification work, i.e., the simplified PoW, provides the DAG with protection against malicious attacks such as spamming. The complexity of the simplified PoW is much lower than the traditional PoW, which costs less computing resource and has higher efficiency. The more transactions there are, the more validations are executed, and the faster and safer the DAG will be.

The complete process of our proposed hybrid blockchain empowered federated learning scheme is provided in Algorithm 1.

V. Deep Reinforcement Learning for Node Selection

A. Problem Formulation: Combinational Optimization for Node Selection

The heterogeneous computing resources and the time-varying communication conditions of various vehicles hinder efficient execution of the global aggregation phase. It is effective to select the fast nodes for reducing aggregation time, while the stale nodes are excluded from the process. On the other hand, to improve the quality of the aggregated model, the nodes with accurate learned models should be selected. Therefore, we aim to select a subset of vehicle nodes $V_P \subset V_I$ with the goal of minimizing the execution time and maximizing the accuracy of the aggregated model.

To formulate the node selection problem, we introduce $\boldsymbol{\lambda}^t = [\lambda_i^t]$ in time slot t as indicator vector for the selection state of vehicles, with $\lambda_i^t = 1$ indicating v_i is selected/active and $\lambda_i^t = 0$ otherwise. We first give the cost metrics in our node selection process. The local learning time cost of vehicle i in time slot t is denoted as

$$c_l^t(i) = f_l\left(\xi_i, d_i, t\right) = \frac{d_i \cdot \beta_m}{\xi_i(t)}, \tag{12}$$

where d_i is the training data of vehicle i, and β_m is the number of CPU cycles needed for training model m on a unit data. The communication cost of vehicle i is denoted by

$$c_c^t(i) = f_c\left(\tau_i, w_i, t\right) = \frac{|w_i|}{\tau_i}, \tag{13}$$

where $|w_i|$ is the size of learned local parameters in slot t. Thus, the time cost for slot t is

$$c_{te}^t = \max_{i \in V_P}\left(c_l^t(i) + c_c^t(i)\right). \tag{14}$$

We define the time cost function as

$$c_{te}^t = \frac{1}{|V_P|}\sum_{i=1}^{|V_P|}\left(c_l^t(i) + c_c^t(i)\right). \tag{15}$$

The Quality of Learning (QoL) describes the accuracy of local model parameters learned by vehicle i in slot t. We use the cost of QoL to quantify the learning accuracy loss in slot t

$$c_q^t = \sum_{i \in V_P} \sigma_i^t\left(w^t, d_i\right) = \sum_{i \in V_P}\sum_j L\left(y_j - \hat{w}^t\left(x_j\right)\right), \tag{16}$$

where w^t is the aggregated model in slot t, $L(\cdot)$ is the loss function and $d_i = \{(x_j, y_j)\}$ is the training data of vehicle i. In our scheme, the QoL is measured at the end of each slot. As such, the total cost of federated learning in time slot t can be given by

$$c^t\left(\boldsymbol{\lambda}^t\right) = c_{te}^t + c_q^t. \tag{17}$$

We formulate the combinatorial optimization problem as a Markov Decision Process denoted by $\mathcal{M} = (S, V, P_v, C_v)$, where S is the state space and V is the action space. P_v denotes the state transition probability led by action $v \in V$. C_v denotes the cost for new state caused by action v. The nodes selection problem can be formulated as

$$\min_{\boldsymbol{\lambda}^t} \quad c^t\left(\boldsymbol{\lambda}^t\right) \tag{18a}$$

$$\text{s.t.} \quad \lambda_i^t \in \{0, 1\}, \quad \forall i, \tag{18b}$$

$$|p_{i|\lambda_i=1}(t) - p_c(t)| \le r_0^2, \tag{18c}$$

where constraint (18c) guarantees the distance of the chosen participating vehicles with the calculated center point that cannot exceed the limited distance r_0.

We use DRL to tackle the node selection problem in Eq. (18a). DRL learns the model by interacting with the environment, and needs no prior training data and model assumptions. We leverage the Deep Deterministic Policy Gradient (DDPG), to find the optimal solution for node selection in our asynchronous federated learning. Based on the defined markov decision process $\mathcal{M} = (S, V, P_v, C_v)$, where the parameters are described below:

- *System State:* At each time slot t in the federated learning, the system states consist of wireless data rates between vehicles $\boldsymbol{\tau}(t)$, available computing resource of vehicles $\boldsymbol{\xi}(t)$, locations of vehicles $\boldsymbol{p}(t)$ and the selection state of vehicles $\boldsymbol{\lambda}(t-1)$. The system state $s(t) \in \mathcal{S}$ can be defined as
$$s(t) = \{\boldsymbol{\tau}(t), \boldsymbol{\xi}(t), \boldsymbol{p}(t), \boldsymbol{\lambda}(t-1)\}. \tag{19}$$
- *Action Space:* The action at time slot t is the vehicle selection decision, which can be regarded as a 0–1 problem. The action $\lambda(t) \in \mathcal{A}$ is defined by a vector
$$\boldsymbol{\lambda}^t = \left(\lambda_1^t, \lambda_2^t, \ldots, \lambda_n^t\right), \tag{20}$$
where $\lambda_i^t = 1$ if vehicle i is selected as the participating nodes of federated learning. Otherwise $\lambda_i^t = 0$.
- *Policy:* The policy $\mathcal{P}$ is a mapping from state space to action space $\mathcal{P} : \mathcal{S} \rightarrow \mathcal{A}$. In time slot t, the action to be taken can be calculated by the policy $\boldsymbol{\lambda_t} = \mathcal{P}(s_t)$. The vehicular network states transit according to the node selection actions. For DRL, the actions are generated by a neural network, whose inputs are the system states and the outputs are the actions to be taken.
- *Reward Function:* The system evaluates the effect of an action by using the reward function R. In iteration t, the agent performing the node selection task takes action $\boldsymbol{\lambda}^t$ at state s_t. The action is assessed by the defined reward function as follows:
$$R(s_t, \lambda_t) = -\frac{1}{|\sum_{i=1}^n \lambda_i|} \sum_{i=1}^n c_i^t \cdot \lambda_i^t \tag{21}$$
$$= -\frac{1}{|\sum_{i=1}^n \lambda_i|} \left(\sum_{i=1}^n \lambda_i \left(\frac{d_i \cdot \beta_m}{\xi_i(t)} + \frac{|w_i|}{\tau_i} \right) + \sum_{i=1}^n \lambda_i \sigma_i^t \left(w^t, d_i\right) \right) \tag{22}$$
The reward function $R(s_t, \lambda_t)$ quantifies the performance of taking action λ_t in slot t. The total cumulative reward can be denoted as
$$\mathbb{E}\left[\sum_{t=0}^{T-1} \gamma R\left(s_t, \boldsymbol{\lambda_t}\right)\right] \tag{23}$$
where $\gamma \in (0, 1]$ is the reward discount factor.
- *Next State:* After taking action $\boldsymbol{\lambda_t}$ at state s_t, the system states transit from s_t to s_{t+1}, where $s_{t+1} \Leftarrow s_t + \mathcal{P}(s_t)$. The new updated state includes $\boldsymbol{\tau}(t+1), \boldsymbol{\xi}(t+1), \boldsymbol{p}(t+1), \boldsymbol{\lambda}(t)$. For vehicle i, to simulate its varying communication state, we add random noise to $\tau_i(t)$ as $\tau_i(t+1) = \tau_i(t) + \mathcal{N}_i$, where $\mathcal{N}_i$ is a Gaussian noise which is collected from the Gaussian distribution. The available computation resource is also updated in the same way. The location of vehicle i is estimated and updated based on the trajectories.

The objective of node selection is to minimize the total cost in federated learning. For the DRL model, the objective is to find the $\boldsymbol{\lambda}$ which maximums the cumulative reward (minimize the total cumulative cost), given by
$$\boldsymbol{\lambda} = \arg\max \mathbb{E}\left[\sum_{t=0}^{T-1} \gamma R\left(s_t, \boldsymbol{\lambda_t}\right)\right] \tag{24}$$

B. The Complete Node Selection Algorithm Based on DDPG

We adopt DDPG to solve problem (18a). The basic principle is to use the value function to update the system policy. DDPG consists of three major modules: primary network, target network, replay memory. The primary network includes an actor DNN $\pi(s_t|\theta_\pi)$ and a critic DNN $Q(s_t, \lambda_t|\theta_Q)$, where θ_π and θ_Q represent the parameters of the neural networks. The target network has the same structure as the primary network, and generates the target values which are used to train the primary critic DNN. DDPG further uses a replay memory to store the experience transition information for training the network. The transition information contains the current state s_t, the action λ_t taken on the state, the next state s_{t+1} and the corresponding reward $R(s_t, \lambda_t)$. DDPG improves the training stability by using the target network to provide the objective values and using the replay memory to fetch experience records randomly.

1) Critic Network Training: The critic DNN trains the parameters θ_Q by evaluating the selected action and comparing the results with the objective values obtained from the target networks. The value of a selected function is quantified by the action-value function
$$Q\left(s_t, \lambda_t|\theta_Q\right) = \mathbb{E}\left[\mathcal{R}\left(s_t, \lambda_t\right) + \gamma Q\left(s_{t+1}, \pi\left(s_{t+1}|\theta_Q\right)\right)\right] \tag{25}$$

The critic DNN updates its network parameters with the objective of minimizing the loss function. The loss function is defined as
$$L_Q(\theta_Q) = \mathbb{E}\left[\left(y - Q\left(s_t, \lambda_t|\theta_Q\right)\right)^2\right], \tag{26}$$

where $Q(s_t, \lambda_t|\theta_Q)$ is the estimated value on action λ_t and y is the target value obtained from the target network by
$$y = \mathcal{R}\left(s_t, \lambda_t\right) + \gamma Q'\left(s_{t+1}, \pi'\left(s_{t+1}|\theta_{\pi'}\right)|\theta_{Q'}\right), \tag{27}$$

where $\theta_{\pi'}$ and $\theta_{Q'}$ are parameters of the target actor DNN and critic DNN, respectively.

Then the gradient of the loss function is defined by
$$\nabla L_Q\left(\theta_Q\right) = \mathbb{E}\left[2\left(y - Q\left(s_t, \lambda_t|\theta_Q\right)\right) \cdot \nabla Q\left(s_t, \lambda_t\right)\right]. \tag{28}$$

The critic DNN is trained by stochastic gradient descent. The stochastic process is executed by randomly fetch a mini-batch of experiences from the replay memory. The network parameters

Algorithm 2: The DDPG Based Node Selection Algorithm.

```
Input: Randomly initialize the primary actor DNN parameters $\theta_\pi$, the primary critic DNN parameters $\theta_Q$. Set the target actor DNN parameters $\theta_\pi^T = \theta_\pi$, and set the target critic DNN parameters $\theta_Q^T = \theta_Q$. Initialize the replay memories.
Input: Initialize the $\alpha$, $\gamma$ and $\lambda^0 = [1, \ldots, 1]$
for each episode do
  Initialize the parameters in environment setup.
  for each time slot $t$ do
    Generate and execute action $\lambda_t$
    Calculate immediate reward $\mathcal{R}(s_t, \lambda_t)$ with Eq. (21) and update the system state to $s_{t+1}$.
    Sample a mini-batch of experiences from replay memory.
    Update the primary critic DNN $Q(s, \lambda|\theta_Q)$ based on Eq. (29).
    Update the primary actor DNN $\pi(s|\theta_\pi)$ based on Eq. (32).
    Update the target network parameters
    Store the experience tuples to relay memory.
  end for
end for
return The parameters of the trained deep neural networks.
```

θ_Q are updated by

$$\theta_Q = \theta_Q + \alpha \cdot \mathbb{E}\left[2\left(y^i - Q\left(\boldsymbol{s_i}, \boldsymbol{\lambda^i}|\theta_Q\right)\right) \cdot \nabla Q\left(\boldsymbol{s^i}, \boldsymbol{\lambda^i}\right)\right], \tag{29}$$

where α is the learning rate of the critic DNN.

2) Actor Network Training: The goal of actor DNN is to provide the best node selection action. The input of actor DNN is the current state s_t and the output is the action to be taken $\boldsymbol{\lambda}$. The action is generated by the mapping from current state s_t to the action $\boldsymbol{\lambda} = \pi(s_t|\theta_\pi)$, where θ_π is the parameters of actor DNN which represent the explored policy. The gradient of stochastic policy gradient $\nabla\pi(\lambda|s, \theta_\pi)$ is proved to be equivalent to the deterministic policy gradient [42]. The complete gradient is defined as

$$\nabla Q = \nabla_\lambda Q\left(s^i, \lambda|\theta_Q\right)|_{\lambda=\pi(s^i|\theta_\pi)} \cdot \nabla_{\theta_\pi}\pi\left(s^i\right), \tag{30}$$

Thus, the deterministic policy gradient is given by

$$\nabla\pi\left(\lambda|s, \theta_\pi\right) \approx \mathbb{E}_\pi\left[\nabla_\lambda Q\left(s, \lambda|\theta_Q\right)|_{\lambda=\pi(s|\theta_\pi)} \cdot \nabla_{\theta_\pi}\pi(s)\right]. \tag{31}$$

In each training step, we update the network parameters θ_π with a mini-batch of experiences $(s_t, \lambda_t, \mathcal{R}(s_t, \lambda_t), s_{t+1})$, which are randomly sampled from the replay memory.

$$\theta_\pi = \theta_\pi + \alpha_\pi \cdot \mathbb{E}\left[\nabla_\lambda Q\left(\boldsymbol{s^i}, \boldsymbol{\lambda}|\theta_Q\right)|_{\boldsymbol{\lambda}=\pi\left(\boldsymbol{s^i}|\theta_\pi\right)} \cdot \nabla_{\theta_\pi}\pi\left(\boldsymbol{s^i}\right)\right], \tag{32}$$

where α_π is the learning rate of the actor DNN.

The complete node selection algorithm for our blockchain empowered federated learning is presented in Algorithm 2.

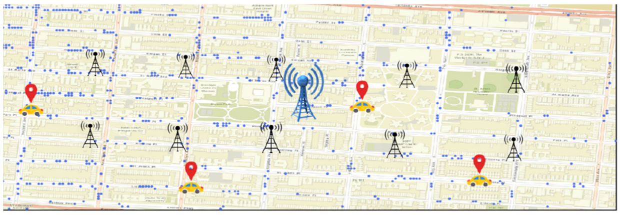

Fig. 5. The simulation scenario of vehicular network.

VI. Numerical Results

In this section, we evaluate the performance of our proposed hybrid blockchain empowered asynchronous federated learning scheme for edge data sharing. We first investigate the performance of hybrid blockchain empowered asynchronous federated learning. Then, we test the proposed DRL-based node selection algorithm.

A. Setup

- *Network Initialization:* We consider a vehicular network with one MBS in the network center and 10 RSUs within the coverage of the MBS. We define a 1500 m × 1000 m tangle area based on the Brooklyn of New York City on the map as our simulation scenario. The tracepoints of vehicles are derived from the dataset of Uber pickups in New York City [43]. The overall simulation scenario plotted by matplotlib basemap toolkit, is shown in Fig. 5. The size of dataset d, the computing capability ξ, and the communication capability τ are derived from the Truncated Gaussian Distribution (1000, 0.5), (900, 0.1), (100, 0.1), respectively. The coverage range of the MBS and RSU are 1000 m and 300 m, respectively.
- *Models and Datasets:* We evaluate the proposed asynchronous federated learning on the MNIST dataset. The dataset is split into 100 shards, and the shards are assigned to 100 providers. The edge data sharing task is to share the computing results on the local data of each data provider. The Convolutional Neural Network (CNN) model is adopted as our local training model. In each iteration, there are one global aggregation and 10 slots for local training. Furthermore, we adopt the local CNN model and the centralized CNN model as the benchmark algorithms on the same dataset. The local CNN trains the model on the dataset of a local provider and the centralized CNN model is trained on the whole centralized dataset. Then, we validate the performance of our proposed node selection algorithm based on DDPG.

B. Results

We first evaluate the accuracy and loss of the proposed scheme on the MNIST dataset with a various number of data providers. In addition, to test the effect of the proposed node selection algorithm, we set 3 data providers as low-quality participants, named bad nodes. The 3 data providers have low communication and computing capabilities and provide low-quality training parameters for aggregation. The low-quality parameters are

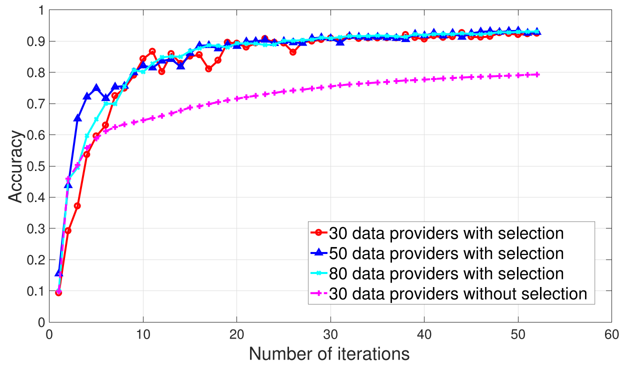

Fig. 6. The accuracy with various numbers of data providers.

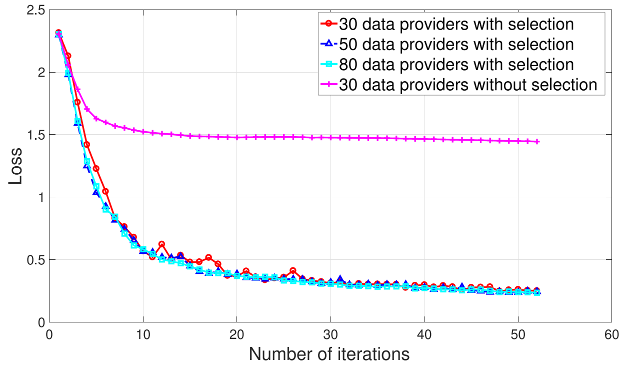

Fig. 7. The loss with various numbers of data providers.

derived by disturbing the original parameters with the random noise. The accuracy and loss results are shown in Fig. 6 and Fig. 7, respectively. The results show that the proposed scheme achieves good accuracy and convergence. The accuracy has a small reduction when the number of participating data providers changes from 30, 50 to 80. Yet the small change illustrates the good scalability of our proposed scheme. The compared results of whether adopting node selection show that the proposed node selection algorithm can prevent the low-quality nodes from affecting the learning results.

To analyze the effect of the bad nodes on the whole federated learning, we compare the performance of the proposed scheme without node selection in various number of bad nodes. From Fig. 8 and Fig. 9 we can see that, the increase of bad nodes degrades the performance dramatically. The results indicate that it is of vital importance to optimize the selection of participating nodes, which can improve the performance significantly.

We compare the proposed scheme with two baseline methods, local CNN and centralized CNN. The local CNN model is trained on a local dataset and evaluated on the whole dataset from 100 providers. The centralized CNN model is trained and evaluated on the whole dataset of 100 providers. Fig. 10 shows that the performance of our proposed scheme is highly close to the centralized CNN. However, the centralized methods bring high data security and privacy risks for providers. The accuracy results of local CNN lag far behind those of the other two methods. The reason is that the objective of local training in local CNN is to minimize the loss on the local dataset. The local

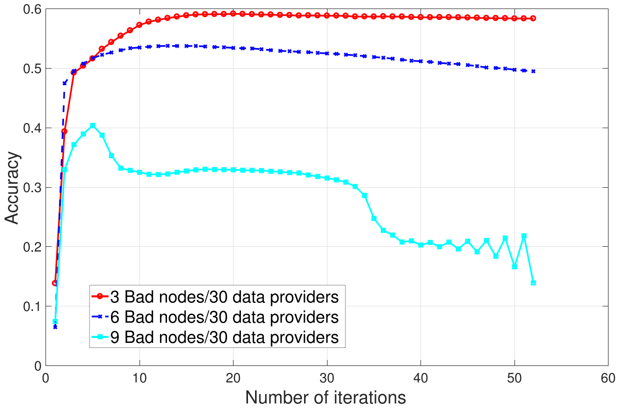

Fig. 8. The accuracy in various numbers of bad nodes.

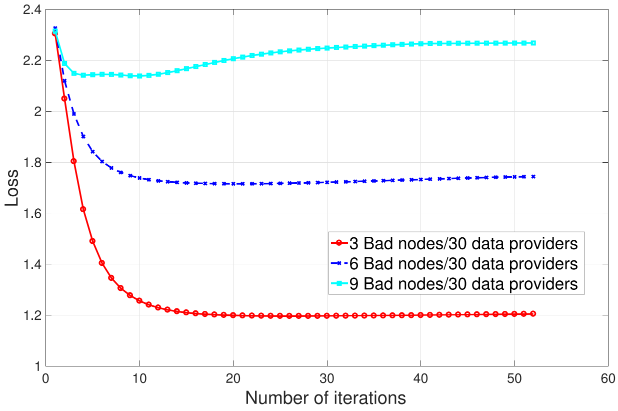

Fig. 9. The loss in various numbers of bad nodes.

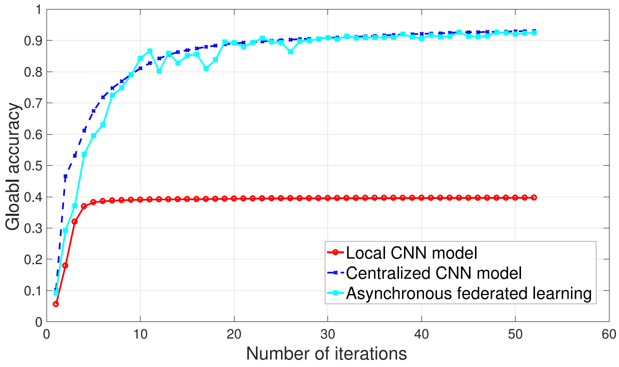

Fig. 10. The global accuracy results of benchmark methods.

CNN can obtain a local optimum, but it may be far from the global optimal solution.

Moreover, we evaluate the time cost of our proposed asynchronous federated learning scheme, and compare it with the synchronous federated learning approach and centralized CNN in Fig. 11. From Fig. 11 we can see that our proposed scheme outperforms other approaches with the minimum time cost, which shows the high efficiency of our proposed scheme. The additional cost values for running our hybrid blockchain scheme with different numbers of vehicles and RSUs are shown in Fig. 12. The cost of running the blockchain scheme increases with the number of participants and the number of transactions,

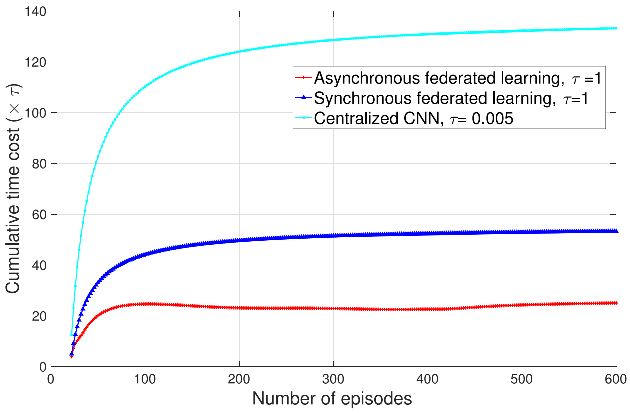

Fig. 11. Performance comparison in terms of cumulative time cost.

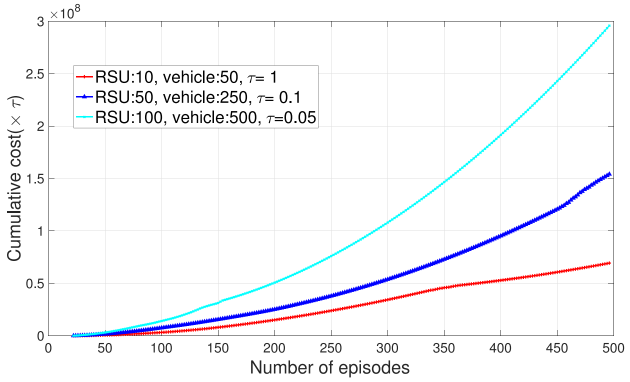

Fig. 12. Cumulative cost of the proposed blockchain scheme.

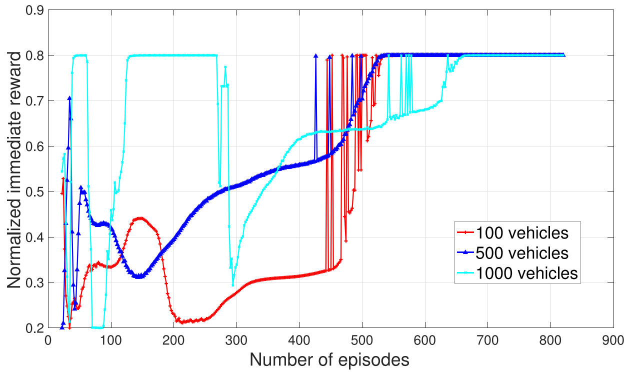

Fig. 13. Immediate reward of DDPG with varying number of vehicles.

which requires increasing communication load and computing overhead.

We further study the performance of the proposed DDPG-based node selection algorithm. The learning rate is 0.001, the replay buffer size is 5000, and the mini-batch size is 32. Fig. 13 shows the iterative exploration process of selecting the optimal participating nodes with various number of vehicles. The proposed algorithm can achieve good convergence after 500 to 600 episodes. The stable immediate reward at last means that the optimal selection solution is found. The more vehicles there are, the more episodes it needed to learn the optimal solution. Fig. 14 shows the cumulative reward obtained in the learning phase with various number of vehicles. The convergence trend is roughly the same with a varying number of vehicles. The high values at the beginning are due to the high initial immediate reward start obtained from the default selection state, as shown in Fig. 13. The high initial immediate reward means that the initial default selection state is close to the final optimal solution.

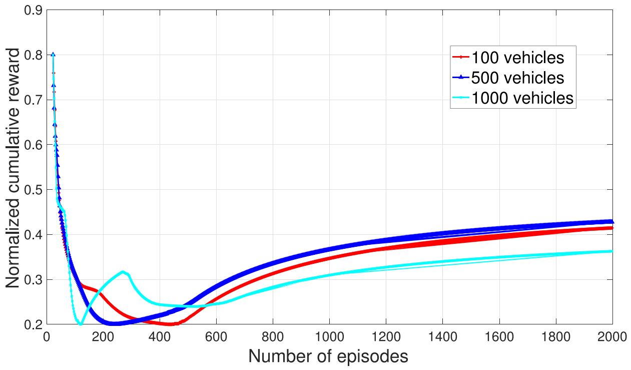

Fig. 14. Cumulative reward of DDPG with varying number of vehicles.

VII. Conclusion

In this article, we have addressed the problem of edge data sharing among vehicles in an IoV framework. We first proposed a hybrid blockchain mechanism that includes the permissioned blockchain and the local DAG in IoV. Based on the hybrid blockchain mechanism, we proposed the asynchronous federated learning scheme and further improved the learning efficiency by using DRL to select the optimized participating nodes. By integrating learning parameters into the blockchain, the qualities of learned models can be further verified through the two-stage verification. Extensive numerical results confirm the effectiveness of our proposed scheme in terms of efficiency and accuracy.

References

[1] J. C.-Castillo, S. Zeadally, and J. A. G.-Ibaez, "Internet of Vehicles: Architecture, protocols, and security," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3701–3709, Oct. 2018.

[2] K. Zhang, S. Leng, Y. He, S. Maharjan, and Y. Zhang, "Cooperative content caching in 5G networks with mobile edge computing," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 80–87, Jun. 2018.

[3] J. Liu, N. Kato, J. Ma, and N. Kadowaki, "Device-to-device communication in LTE-advanced networks: A survey," *IEEE Commun. Surv. Tut.*, vol. 17, no. 4, pp. 1923–1940, Oct.-Dec. 2015.

[4] 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.

[5] F. Tang, Z. M. Fadlullah, N. Kato, F. Ono, and R. Miura, "AC-POCA: Anticoordination game based partially overlapping channels assignment in combined UAV and D2D-based networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1672–1683, Feb. 2018.

[6] K. Zhang, S. Leng, X. Peng, L. Pan, S. Maharjan, and Y. Zhang, "Artificial intelligence inspired transmission scheduling in cognitive vehicular communications and networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1987–1997, Apr. 2019.

[7] H. Dai, Z. Zheng, and Y. Zhang, "Blockchain for Internet of Things: A survey," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8076–8094, Oct. 2019, doi: 10.1109/JIOT.2019.2920987.

[8] K. Zhang, Y. Zhu, S. Maharjan, and Y. Zhang, "Edge intelligence and blockchain empowered 5G beyond for the industrial Internet of Things," *IEEE Netw.*, vol. 33, no. 5, pp. 12–19, Sep. 2019.

[9] L. Jiang, S. Xie, S. Maharjan, and Y. Zhang, "Joint transaction relaying and block verification optimization for blockchain empowered D2D communication," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 828–841, Jan. 2020.

[10] K. Anoh, S. Maharjan, A. Ikpehai, Y. Zhang, and B. Adebisi, "Energy peer-to-peer trading in virtual microgrids in smart grids: A game-theoretic approach," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1264–1275, Mar. 2020, doi: 10.1109/TSG.2019.2934830.

[11] J. Kang *et al.*, "Blockchain for secure and efficient data sharing in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4660–4670, Jun. 2019.

[12] 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.

[13] Y. Dai, D. Xu, S. Maharjan, Z. Chen, Q. He, and Y. Zhang, "Blockchain and deep reinforcement learning empowered intelligent 5G beyond," *IEEE Netw.*, vol. 33, no. 3, pp. 10–17, May 2019.

[14] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Computation offloading and content caching in wireless blockchain networks with mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 11 008–11 021, Nov. 2018.

[15] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," 2016. [Online]. Available: https://arxiv.org/abs/1610.05492

[16] H. B. McMahan *et al.*, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artif. Intell. Stat.*, Fort Lauderdale, FL, USA, Apr. 2017, vol. 54, pp. 1273–1282.

[17] S. Ruder, "An overview of gradient descent optimization algorithms," 2016. [Online]. Available: https://arxiv.org/abs/1609.04747

[18] J. Liu, S. Zhang, N. Kato, H. Ujikawa, and K. Suzuki, "Device-to-device communications for enhancing quality of experience in software defined multi-tier LTE-A networks," *IEEE Netw.*, vol. 29, no. 4, pp. 46–52, Jul. 2015.

[19] Z. Uykan and R. Jntti, "Transmission-order optimization for bidirectional device-to-device (D2D) communications underlaying cellular TDD networksa graph theoretic approach," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 1–14, Jan. 2016.

[20] M. Chen, M. Mozaffari, W. Saad, C. Yin, M. Debbah, and C. S. Hong, "Caching in the sky: Proactive deployment of cache-enabled unmanned aerial vehicles for optimized quality-of-experience," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1046–1061, May 2017.

[21] J. Liu, H. Nishiyama, N. Kato, and J. Guo, "On the outage probability of device-to-device-communication-enabled multichannel cellular networks: An RSS-threshold-based perspective," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 163–175, Jan. 2016.

[22] H. Nishiyama, M. Ito, and N. Kato, "Relay-by-smartphone: Realizing multihop device-to-device communications," *IEEE Commun. Mag.*, vol. 52, no. 4, pp. 56–65, Apr. 2014.

[23] W. Tang, J. Ren, K. Zhang, D. Zhang, Y. Zhang, and X. S. Shen, "Efficient and privacy-preserving fog-assisted health data sharing scheme," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 6, pp. 68:1–68:23, Oct. 2019. [Online]. Available: http://doi.acm.org/10.1145/3341104

[24] T. G. Rodrigues, K. Suto, H. Nishiyama, N. Kato, and K. Temma, "Cloudlets activation scheme for scalable mobile edge computing with transmission power control and virtual machine migration," *IEEE Trans. Comput.*, vol. 67, no. 9, pp. 1287–1300, Sep. 2018.

[25] T. G. Rodrigues, K. Suto, H. Nishiyama, and N. Kato, "Hybrid method for minimizing service delay in edge cloud computing through vm migration and transmission power control," *IEEE Trans. Comput.*, vol. 66, no. 5, pp. 810–819, May 2017.

[26] Y. Dai, D. Xu, S. Maharjan, and Y. Zhang, "Joint load balancing and offloading in vehicular edge computing and networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4377–4387, Jun. 2019.

[27] K. Zhang, Y. Zhu, S. Leng, Y. He, S. Maharjan, and Y. Zhang, "Deep learning empowered task offloading for mobile edge computing in urban informatics," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7635–7647, Oct. 2019.

[28] W. Tang, J. Ren, and Y. Zhang, "Enabling trusted and privacy-preserving healthcare services in social media health networks," *IEEE Trans. Multimedia*, vol. 21, no. 3, pp. 579–590, Mar. 2019.

[29] J. Kang, R. Yu, X. Huang, S. Maharjan, Y. Zhang, and E. Hossain, "Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains," *IEEE Trans. Ind. Informat.*, vol. 13, no. 6, pp. 3154–3164, Dec. 2017.

[30] H. Yao, T. Mai, J. Wang, Z. Ji, C. Jiang, and Y. Qian, "Resource trading in blockchain-based industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3602–3609, Jun. 2019.

[31] Y. Xu, J. Ren, G. Wang, C. Zhang, J. Yang, and Y. Zhang, "A blockchain-based nonrepudiation network computing service scheme for industrial iot," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3632–3641, Jun. 2019.

[32] X. Zhang and X. Chen, "Data security sharing and storage based on a consortium blockchain in a vehicular ad-hoc network," *IEEE Access*, vol. 7, pp. 58 241–58 254, 2019.

[33] L. Jiang, S. Xie, S. Maharjan, and Y. Zhang, "Blockchain empowered wireless power transfer for green and secure Internet of Things," *IEEE Netw.*, vol. 33, no. 6, pp. 164–171, Nov./Dec. 2019.

[34] S. Popov, "The tangle," 2018. [Online]. Available: https://iota.org/IOTA_Whitepaper.pdf

[35] K. Karlsson *et al.*, "Vegvisir: A partition-tolerant blockchain for the Internet-of-Things," in *Proc. IEEE 38th Int. Conf. Distrib. Comput. Syst.)*, Jul. 2018, pp. 1150–1158.

[36] C. Xu, J. Ren, D. Zhang, Y. Zhang, Z. Qin, and K. Ren, "GANobfuscator: Mitigating information leakage under GAN via differential privacy," *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 9, pp. 2358–2371, Sep. 2019.

[37] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Federated learning for data privacy preservation in vehicular cyber-physical systems," *IEEE Netw. Mag., accepted*, p. 1, 2019.

[38] 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)*, 2018, pp. 1–7.

[39] N. H. Tran, W. Bao, A. Zomaya, M. N. H. Nguyen, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *Proc. IEEE INFOCOM 2019–IEEE Conf. Comput. Commun.*, Apr. 2019, pp. 1387–1395.

[40] H. R. Feyzmahdavian, A. Aytekin, and M. Johansson, "An asynchronous mini-batch algorithm for regularized stochastic optimization," in *Proc. 54th IEEE Conf. Decis. Control*, Dec. 2015, pp. 1384–1389.

[41] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, "Differentially private asynchronous federated learning for mobile edge computing in urban informatics," *IEEE Trans. Ind. Informat.*, vol. 16, no. 3, pp. 2134–2143, Mar. 2020.

[42] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *Proc. Int. Conf. Mach. Learning*, pp. 387–395, 2014. [Online]. Available: http://proceedings.mlr.press/v32/silver14.html

[43] "Uber TLC FOIL response," 2015. [Online]. Available: https://github.com/fivethirtyeight/uber-tlc-foil-response

Yunlong Lu received the B.S. degree in electronic information science and technology from Beijing Forestry University, Beijing, China, in 2012, and the M.S degree from the School of Computer Science at Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2015. He is currently working toward the Ph.D. degree in computer science and technology with the Institute of Network Technology, BUPT. He is a Visiting Ph.D. Student with the University of Oslo, Oslo, Norway. His current research interests include blockchain, wireless networks, and privacy-preserving machine learning.

Xiaohong Huang received the B.E. degree from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2000, and the Ph.D. degree from the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, in 2005. Since 2005, she has been with BUPT, where she is currently a Professor and the Director of the Network and Information Center, Institute of Network Technology. She has authored pr coauthored more than 50 academic papers in the area of wavelength division multiplexing optical networks, IP networks, and other related fields. Her current interests include the performance analysis of computer networks and service classification.

Ke Zhang received the Ph.D. degree from the University of Electronic Science and Technology of China, Chengdu, China, 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 scheduling of mobile edge computing, design and optimization of nextgeneration wireless networks, smart grid, and the Internet of Things.

Sabita Maharjan (Senior Member, IEEE) received the Ph.D. degree in networks and distributed systems from the University of Oslo, Oslo, Norway, and Simula Research Laboratory, Norway, in 2013. She is currently a Senior Research Scientist with Simula Metropolitan Center for Digital Engineering, Norway, and Associate Professor (adjunct position) with the University of Oslo. She worked as a Research Engineer with the Institute for Infocomm Research (I2R), Singapore, in 2010. She was a Visiting Scholar with Zhejiang Univeristy (ZU), Hangzhou, China in 2011, and a Visiting Research Collaborator with University of Illinois at Urbana Champaign (UIUC), Champaign, IL, USA, in 2012. She was a Postdoctoral Fellow with the Simula Research laboratory, Norway, from 2014 to 2016. She authors and coauthors prestigious journals in her field such as IEEE TRANSACTIONS ON SMART GRID, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, *IEEE Communications Magazine, IEEE Network Magazine, IEEE Wireless Communications Magazine*, and IEEE INTERNET OF THINGS JOURNAL. Her current research interests include vehicular networks and 5 G, network security and resilience, smart grid communications, Internet of Things, machine-to-machine communication, software defined wireless networking, and advanced vehicle safety. Dr. Maharjan serves/has served in the technical program committee of conferences including top conferences like IEEE International Conference on Computer Communications and IEEE International Symposium on Quality of Service.

Yan Zhang (Fellow, IEEE) received the Ph.D. degree from the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore. He is currently a Full Professor with the Department of Informatics, University of Oslo, Oslo, Norway. His research interests include next-generation wireless networks leading to 5 G beyond/6 G, green and secure cyber-physical systems (e.g., smart grid and transport). Dr. Zhang is a member of CCF Technical Committee of Blockchain. He is the Chair of IEEE Communications Society Technical Committee on Green Communications and Computing (TCGCC). He is an Editor for IEEE publications, including *IEEE Communications Magazine, IEEE Network Magazine*, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, IEEE COMMUNICATIONS SURVEY AND TUTORIALS, IEEE INTERNET OF THINGS JOURNAL, IEEE SYSTEMS JOURNAL, IEEE VEHICULAR TECHNOLOGY MAGAZINE, and IEEE BLOCKCHAIN TECHNICAL BRIEFS. He is a Symposium/Track Chair for a number of conferences, including IEEE International Conference on Communications 2021, IEEE Global Communications Conference 2017, IEEE International Symposium on Personal, Indoor and Mobile Radio Communications 2016, IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids 2015. He is an IEEE Vehicular Technology Society Distinguished Lecturer during 2016-2020 and he is named as CCF 2019 Distinguished Speaker. Since 2018, he was the recipient of the Highly Cited Researcher Award (Web of Science top 1% most cited) by Clarivate Analytics.