2022 IEEE 95th Vehicular Technology Conference (VTC2022-Spring) | 978-1-6654-8243-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/VTC2022-Spring54318.2022.9860503

Digital Twin Enabled Multi-task Federated Learning in Heterogeneous Vehicular Networks

Yilong Hui†, Gaosheng Zhao†, Zhisheng Yin‡,*, Nan Cheng†, and Tom H. Luan‡
†State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an, 710071, China
‡School of Cyber Engineering, Xidian University, Xi'an, 710071, China
*Corresponding Author: zsyin@xidian.edu.cn

***Abstract*—In the heterogeneous vehicular networks (HetVNets), the base stations (BSs) can exploit the massive amounts of valuable data collected by vehicles to complete federated learning tasks. However, most of the existing studies consider the scenario of one task requester (TR) and ignore the fact that multiple TRs may concurrently generate their model training requests in the HetVNets. In this paper, we consider the scenario of multi-TR and multi-BS and propose a digital twin enabled scheme for multi-task federated learning to address the two-way selection problem between the TRs and the BSs. We first analyze the diversified requirements of the TRs in the HetVNets. Then, we develop a novel model that jointly considers the available training data, the declared price, and the training experience to evaluate the differentiated training capabilities of the BSs. After that, based on the requirements of the TRs and the training capabilities of the BSs, the two-way selection problem between the TRs and the BSs is formulated as a matching game in the digital twin networks, where a matching algorithm is designed to obtain their optimal strategies. The simulation results demonstrate that the proposed scheme can obtain the highest model accuracy and bring the highest utility to the TRs compared with the conventional schemes.**

***Index Terms*—Digital twins, multi-task federated learning, game theory, heterogeneous vehicular networks.**

I. Introduction

With the development of the space-air-ground integrated networks (SAGINs) [1]–[3], the massive amounts of valuable data can be collected by vehicles in the heterogeneous vehicular networks (HetVNets) to train different machine learning models [4]. In the traditional task training process, each base station (BS) in the networks collects the data from the vehicles in its coverage to complete the training task after receiving the request of the task requester (TR). However, this approach has the following drawbacks. For one thing, the traditional centralized learning architecture may leak the private information of vehicles. For another thing, the transmission of massive data between the vehicles and the BS consumes a lot of resources, resulting in high communication overhead.

The federated learning framework is an effective solution to solve the above problems [5]. In the process of federated learning, the vehicles within the coverage of the BS will first obtain the global model of the current round. Then, the vehicles use their local data to train the model and deliver the training results to the BS. After that, the BS performs parameter aggregation to generate a new global model, where the updated global model will be fed back to the vehicles. Repeat the above process until the model meets the accuracy requirement. In this way, the vehicles only need to upload the model parameters to the BS, where the risk of privacy leakage and the overhead of data transmission between the vehicles and the BS can be significantly reduced.

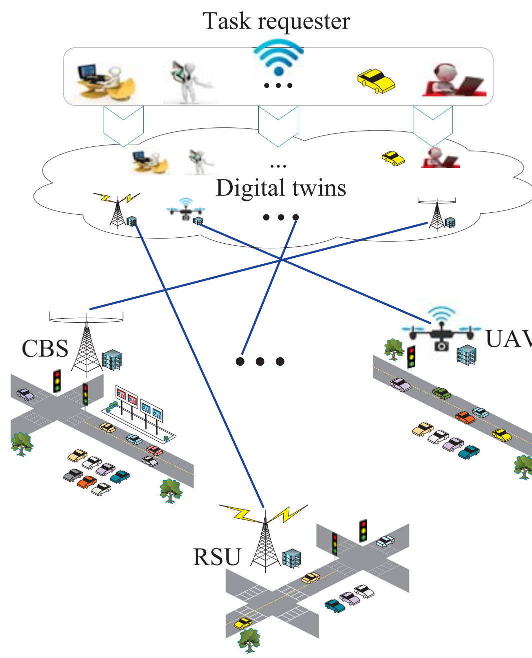

Fig. 1. The system model of the multi-TR and multi-BS scenario.

With the above advantages, the federated learning schemes have been widely studied in HetVNets to facilitate various vehicular applications. However, the existing schemes lack the consideration of the fact that different TRs have diversified requirements for the model accuracy and the price paid for the training tasks. In addition, due to the differences in the number of vehicles in the coverage and the storage of historical information, the BSs in the networks have different training capabilities. However, the existing studies are used to measure the training ability from the amount of data and the non-independent and identically distributed (non-IID) degree, while ignoring the influence of the historical training information on the model. In general, the BS with storage resources usually caches the trained model information. When training a new model, the historical information can be exploited to provide model parameters. Besides, the existing research usually considers the scenario where the appropriate trainers in the networks are selected to complete the training model for one TR [6]–[10]. This ignores the fact that multiple TRs may concurrently generate their model training requests in the HetVNets, as shown in Fig. 1. By jointly considering the diversified requirements of TRs and the differentiated capabilities of BSs, a BS may meet the training requirements of multiple TRs, and the personalized training requirement of a TR may also be satisfied by multiple BSs. With the above challenges, it is necessary to design an efficient scheme to handle the two-way selection problem between the multi-TR and the multi-BS, so as to maximize the utility of each TR while ensuring the accuracy of the training model.

In this paper, we propose a digital twin enabled scheme for multi-task federated learning in the HetVNets to address the above challenges. In our scheme, based on different model training preferences, we first design the diversified requirements of the TRs. Then, the training capabilities of the BSs are analyzed by jointly considering the available training data, the declared price, and the training experience. After that, we design a game-based scheme to achieve efficient matching between the TRs and the BSs in the digital twin networks. By obtaining the optimal strategies of the TRs and the BSs in the game, the TRs and the BSs can effectively be matched to complete the federated learning tasks.

The remainder of this paper is organized as follows. Section II introduces the system model. Section III presents the proposed multi-task federated learning scheme in the HetVNets. Section IV evaluates the performance of the proposed scheme and Section V closes the paper with the conclusion.

II. System Model

A. Task Requesters

The TRs in the networks are the requesters that plan to use the data of vehicles within the coverage of the BS to complete model training tasks. Let $\mathbb{I} = \{1, ..., i, ..., I\}$ and $\mathbb{N} = \{1, ..., n, ..., N\}$ denote the sets of the TRs and the model types in the HetVNets. For TR i that intends to complete learning task n, it can select the optimal BS to train the learning model based on its training requirements and the differentiated capabilities of the BSs. After selecting the optimal BS, the BS will exploit the data owned by the vehicles to undertake the training task requested by TR i. Specifically, the BS will broadcast learning model n to the vehicles within its coverage. Then, the vehicles that have the requested training data train the model and upload the trained model parameters. After the iterative training process is finished, the training result will be delivered from the BS to the TR to obtain profits.

B. Base Stations

As shown in Fig. 1, the BSs in the HetVNets refer to the cellular base stations (CBSs), the roadside units (RSUs), and the unmanned aerial vehicles (UAVs) [11]. Each BS can train the model by using the data cached by the vehicles in its coverage and aggregate the training result to provide model training services. Let $\mathbb{J} = \{1, ..., j, ..., J\}$ denote the set of the BSs in the HetVNets. In general, the BSs usually have different training capabilities. This is because the number of vehicles in the coverage of each BS is different. Accordingly, the available data cached in the vehicles and the non-IID degree of the cached data are different from each other. In addition, the BSs have different model training experiences. We use the times that the BS completes the training models to describe its training experience. For BS j, it maintains a training set $j_{\mathbb{N}} = \{j_1, ..., j_n, ..., j_N\}$, where j_n refers to the times that BS j trains learning model n.

C. Vehicles

The vehicles are the trainers of the learning model in the networks. When a vehicle enters or parks in the coverage of the BS, it can connect to the BS through the wireless link to upload its data information [12]–[15]. Let $\mathbb{K}_j = \{1_j, ..., k_j, ..., K_j\}$ denote the set of vehicles in the coverage of BS j. For vehicle k_j, its data information can be expressed as $\{D_{k_j}, \sigma_{k_j}, C_{k_j}\}$, where D_{k_j} is the amount of data that can be provided by vehicle k_j. σ_{k_j} is the average earth mover's distance (EMD) which is used to measure the non-IID degree of data [7]. C_{k_j} is the price declared by vehicle k_j to train the model, we have

$$C_{k_j} = c_{k_j} + a_{k_j} D_{k_j}, \tag{1}$$

where c_{k_j} is the cost of vehicle k_j for collecting and caching the data. $a_{k_j} > 0$ is the price per unit data.

D. Digital Twins

Digital twin is a technology that can be used to map the states of the entities in the physical world to the digital twin networks [16]. With this technology, the complex decisions can be made in the digital twin networks to reduce the interaction delay of the physical entities. In our scheme, both the TRs and the BSs create their digital twins in the cloud, where the digital twins of TR i and BS j are denoted as i' and j', respectively. For digital twin i', it maps the model training requirements of the TR. Similarly, the training capability of BS j is updated to digital twin j' in real time. If TR i has a learning task needs to be completed, digital twin i' then interacts with the digital twins of the BSs based on the requirement of the learning task and the capabilities of the BSs. After that, the optimal BS can be selected by digital twin i' in the digital twin networks to complete the task.

III. Digital Twin Enabled Scheme for Multi-task Federated Learning

A. Diversified Requirements of TRs

Considering different vehicular applications in the HetVNets, the diversified training requirements of the TRs are reflected by the following aspects: the type of the model, the EMD and the amount of data, the accuracy of the requested model, and the price paid for completing the learning tasks. First, due to the differences in applications, the TRs generally request different training models. Correspondingly, different training models mean that the TRs have different preferences for the

BSs. For TR i that intends to train learning task n, the BS that has trained this model more times is more likely to be selected. Second, in the process of model training, the model accuracy is usually determined by the average EMD degree of the training data [7]. In other words, the TRs can pay a higher price to choose the better data to complete the training task with the target of obtaining a higher model training accuracy. Therefore, the TRs need to put forward different requirements for accuracy and price according to their personal preferences.

In our paper, we define the matching degree between BS j and TR i to balance the amount of training data and the training experience, shown as

$$\epsilon_{j,i} = \lambda_i \frac{j_n}{n_{\max}} + (1 - \lambda_i) \frac{D_{j,i}}{D_{i,\max}}, \tag{2}$$

where $D_{j,i}$ is the amount of data that can be provided by BS j. $n_{\max} = \max\{j_n, \forall j \in \mathbb{J}\}$ and $D_{i,\max} = \max\{D_{j,i}, \forall j \in \mathbb{J}\}$. λ_i is a parameter that reflects the preference of TR i. For example, $\lambda_i = 1$ indicates that TR i only pays attention to the number of historical training times. In addition, TR i can determine the accuracy range of its model and the price P_i that TR i plans to pay for the task. The accuracy range requested by TR i can be expressed as $[\alpha_{i,\max} - \varepsilon_i, \alpha_{i,\max}]$, where ε_i is the maximum error of the model accuracy and $\alpha_{i,\max}$ is the limitation of the model accuracy. Then, based on $\alpha_{i,\max} - \varepsilon_i$, we can obtain the limitation of the data's EMD requested by TR i. It can be expressed as

$$\sigma_{i,\max} = -\kappa_1 + \kappa_2 \sqrt{- \ln \frac{\alpha_{i,\max} - \varepsilon_i}{\kappa_3}}, \tag{3}$$

where κ_1, κ_2 and κ_3 are the parameters to detemine $\sigma_{i,\max}$.

B. Differentiated Training Capabilities of BSs

In the HetVNets, the training capability of each BS is related to a number of factors. On the one hand, the differences in the size of the coverage and the geographic location result in different numbers of vehicles and data in the coverage of each BS. On the other hand, the training experiences of the BSs are different, which means that the model parameters provided by the training experiences of the BSs are also different for different types of learning models. Therefore, based on the above two factors (i.e., vehicle data and training experience), we can define the training ability of BS j for the training task requested by TR i. Specifically, based on the task information of TR i, BS j will analyze the historical training times j_n of the model with type n and the data information of the vehicles that meet the requirements of TR i, including the amount of data $\{D_{1_j}, ..., D_{k_j}, ..., D_{K_j}\}$ and the declared data price $\{C_{1_j}, ..., C_{k_j}, ..., C_{K_j}\}$. After that, a training strategy will be generated by the BS, shown as

$$S_{j,i} = \{j_n, D_{j,i}, C_{j,i}\}, \tag{4}$$

where $D_{j,i}$ is related to the amount of training data in the coverage of the BS, we have

$$D_{j,i} = \sum_{k=1}^{K} D_{k_j}. \tag{5}$$

$C_{j,i}$ is the price declared by BS j for completing the model training task, we have

$$\begin{aligned} C_{j,i} &= \sum_{k=1}^{K} C_{k_j} + b_j D_{j,i} \\ &= \sum_{k=1}^{K} (c_{k_j} + a_{k_j} D_{k_j}) + b_j \sum_{k=1}^{K} D_{k_j}, \end{aligned} \tag{6}$$

where b_j is the price per unit data. $\sum_{k=1}^{K} C_{k_j}$ is the price that the BS pays to the vehicles. In (5) and (6), if vehicle k_j does not have the data to train learning model n, we have $D_{k_j} = 0$ and $C_{k_j} = 0$.

C. Matching Game in Digital Twin Networks

In our scheme, the requirements of TRs and the capabilities of BSs will be updated to the digital twin networks to help the digital twins of the TRs and the BSs make decisions. We formulate the two-way selection process between the digital twins of the TRs and the BSs as a matching game and design a matching algorithm to obtain their optimal strategies. The specific matching process is as follows.

(1) Issue task information: Each digital twin i' delivers the task information of its TR $T_i = \{\sigma_{i,\max} | i_n\}$ to the BSs in the digital twin networks, where $\sigma_{i,\max}$ is the limitation of the EMD of the data.

(2) Determine training strategy: After receiving the task information published by the digital twins of the TRs, each digital twin j' sends the corresponding training strategy to each i'.

(3) Generate BS preference list: Each digital twin i' generates a BS preference list after receiving the training strategy sent by each digital twin j'. Here, we take digital twin i' as an example to introduce the process.

- Matching degree selection: Based on the training strategies, digital twin i' first selects the BSs according to the matching degree. After digital twin i' obtains the matching degree for each digital twin j', it will select those BSs that meet the requirements according to its own matching degree condition $\epsilon_{i,\min}$.
- Price selection: If digital twin j' satisfies the matching degree condition, namely, $\epsilon_{j,i} \geq \epsilon_{i,\min}$, digital twin i' will further check the training price $C_{j,i}$. Specifically, if $P_i - C_{j,i} < 0$, the BS will be removed from the list.
- Accuracy selection: For the BSs that meet the matching degree and the price conditions, digital twin i' will select those BSs that meet the training accuracy requirements. In the learning process, the accuracy of the training model is determined by the amount of data, the average EMD, and the number of historical training times. Therefore, the accuracy of the learning task that is requested by TR i and completed by BS j can be defined as

$$A_{j,i} = \alpha(\sigma_{j,i}) - \kappa_4 e^{-\kappa_5 (\kappa_6 (D_{j,i} + D(j_n)))^{\alpha(\sigma_{j,i})}}, \tag{7}$$

where κ_4, κ_5 and κ_6 are the parameters of the model. According to (3), we have $\alpha(\sigma_{j,i}) = \kappa_3 e^{-(\frac{\sigma_{j,i} + \kappa_1}{\kappa_2})^2}$.

$D(j_n) = \kappa_7 j_n$ is the initial amount of data to reflect the impact of the historical training times on the training accuracy. $-\kappa_4 e^{-\kappa_5(\kappa_6(D_{j,i}+D(j_n)))^{\alpha(\sigma_{j,i})}}$ reflects the diminishing marginal returns when the amount of training data increases [7].

- Generate preference list: After digital twin i' completes the selection of the BSs, we design the utility function to help digital twin i' generate a list of preferences for the BSs. The utility function can be expressed as

$$U_{j,i} = (1 - q_i)(P_i - C_{j,i}) + q_i e^{\kappa_8 A_{j,i}}, \tag{8}$$

where κ_8 is the parameter of the utility function. q_i is the preference of digital twin i'. Based on (8), digital twin i' will generate a BS preference list in a descending order.

(4) Send matching request: After deciding the list, digital twin i' sends the matching requests to the selected BSs.

(5) Generate TR preference list: According to (6), we can see that the profit of digital twin j' increases with the increase of $D_{j,i}$. Therefore, the preference list generated by the BSs is determined by the amount of available training data. Specifically, after receiving the matching requests sent by each digital twin i', each digital twin j' sorts the amount of available training data for each digital twin i' in a descending order and generates a TR preference list.

(6) Match the TRs and the BSs: The digital twins of the TRs and the BSs start to match based on their preference lists, where a set of stable TR-BS matching pairs will be determined to help each TR select the optimal BS.

(7) Train the model: Each digital twin i' delivers the learning model to the matched digital twin j'. Then, j' delivers the model to its physical entity. The entity BS trains the model according to the training strategy and delivers the training result to the TR after the model is completed.

(8) Pay the price: After completing the model training task, each digital twin i' pays the price of model training to the selected digital twin j'. Meanwhile, the BS pays the prices of the vehicles participating in the model training process.

(9) Update the system: After the matching process of this round is over, each BS will update the information (i.e., training set and the data owned by the vehicles in its coverage) cached in its digital twin, waiting to participate in the next round of matching.

IV. Performance Evaluation

A. Simulation Scenario

In the simulations, the number of BSs in the HetVNets is set to be 5, where the declared price and the available data of each BS is randomly selected from (0, 200] and (0, 1000]. The number of TRs in the networks changes from 1 to 5. For TR i, its expected price paid for completing the requested learning task lies in (0, 100]. The parameters used in the simulation are listed in TABLE I. With this scenario, we evaluate the average model accuracy and average utility of the TRs by changing the limitation of the EMD from 0 to 1. The algorithms used in the simulations are summarized as follows.

TABLE I
Simulation Parameters

Parameters	Values
J	5
I	[1, 5]
P_i	(0, 100]
$C_{j,i}$	(0, 200]
$D_{j,i}$	(0, 1000]
λ_i	0.5
q_i	0.15
$\epsilon_{i,\min}$	0.8
$\kappa_1, \kappa_2, \kappa_3, \kappa_4$	0.31, 1.743, 0.993, 0.361
$\kappa_5, \kappa_6, \kappa_7, \kappa_8$	4.438, 5×10^{-4}, 10, 5

- Experience-Based Algorithm: The digital twin of the TR builds the BS preference list based on the training times (i.e., $\lambda_i = 1$) to perform the matching process.
- Data-based Algorithm: The digital twin of the TR builds the BS preference list based on the amount of data (i.e., $\lambda_i = 0$) to perform the matching process.
- Immediate-Acceptance Algorithm: In this algorithm, the digital twin of the TR builds the BS preference list by jointly considering the historical training times and the amount of training data. However, the matching process is performed according to the immediate matching algorithm. In this algorithm, if the digital twin of a TR sends a matching request to the digital twin of a BS, the matching is immediately successful. In addition, if multiple digital twins send matching requests, the digital twin of the BS matches the optimal one based on its preference list.
- The Proposal: The digital twin of the TR builds the BS preference list by considering the historical training times and the amount of data. Furthermore, the matching process is performed based on the designed scheme.

B. Simulation Results

Fig. 2 shows the average model accuracy of the TRs by changing the limitation of the EMD from 0 to 1. It can be seen from this figure that the average model accuracy of the TRs gradually decreases with the increase of the EMD. In addition, we can see that no matter whether the value of the EMD is large or small, the model accuracy of our scheme is higher than the conventional schemes. The reasons for this are as follows. First, in our scheme, the preference list is built by jointly considering the historical training times and the amount of training data. In this way, the ability of each BS can be effectively evaluated. Second, a stable matching algorithm is designed so that the optimal BS for each TR can be selected to train the learning model.

Fig. 3 shows the average utility of the TRs by changing the values of the limitation of the EMD. From this figure, we can see that the average utility of the TRs decreases with the increase of the value of the EMD. This is mainly because the training accuracy decreases with the increase of the EMD. In addition, we can see that the proposed matching scheme can bring the highest utility to the TRs compared to the conventional schemes. This is because the utility of the TRs is

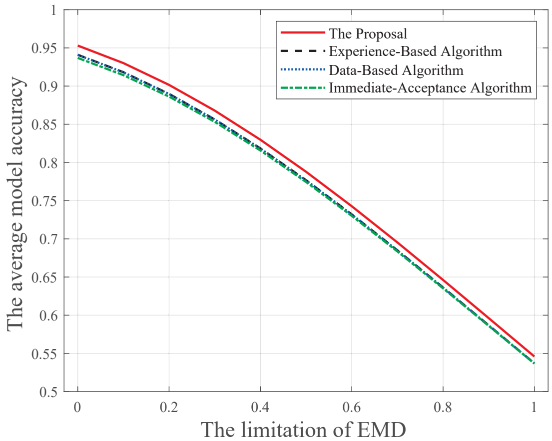

Fig. 2. The average model accuracy by changing the limitation of the EMD.

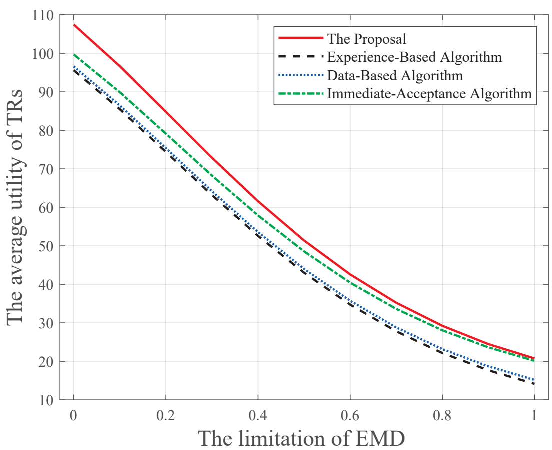

Fig. 3. The average utility of the TR by changing the limitation of the EMD.

determined by the price paid to the BSs and the accuracy of the trained models. Compared with other schemes, the proposed matching scheme can not only improve the model accuracy, but also reduce the price paid to the BSs.

V. Conclusion

In this paper, we have proposed a digital twin enabled matching scheme for multi-task federated learning in the HetVNets. Specifically, we have designed the diversified requirements of the TRs for different models through which the requirement of each TR can be set to select the BS. Then, we have developed a novel model to evaluate the differentiated training capabilities of the BSs, where the available training data, the declared price, and the training experience are jointly considered. Based on the requirements of the TRs and the training capabilities of the BSs, we have designed a digital twin architecture and formulated the two-way selection problem between the digital twins of the TRs and the BSs as a matching game to help them obtain their optimal strategies. Compared with the traditional schemes, we have shown that the proposed scheme can obtain the highest model accuracy and bring the highest utility to the TRs. For future work, based on the various learning requirements of the TRs, the selection of vehicles within the coverage of the BSs will be studied to improve the performance of our scheme.

Acknowledgement

This work was supported in part by the National Key R&D Program of China under Grant 2020YFB1807700, in part by the NSFC under Grant 61901341, in part by the China Postdoctoral Science Foundation under Grant 2021TQ0260 and Grant 2021M700105, and in part by the XAST under Grant 095920201322.

References

[1] Z. Yin, M. Jia, N. Cheng, W. Wang, F. Lyu, Q. Guo, and X. Shen, “Uav-assisted physical layer security in multi-beam satellite-enabled vehicle communications,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2739–2751, Mar. 2022.

[2] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. Zhuang, and X. S. Shen, “Software defined space-air-ground integrated vehicular networks: Challenges and solutions,” *IEEE Communications Magazine*, vol. 55, no. 7, pp. 101–109, July 2017.

[3] Y. Hui, N. Cheng, Y. Huang, R. Chen, X. Xiao, C. Li, and G. Mao, “Personalized vehicular edge computing in 6g,” *IEEE Network*, vol. 35, no. 6, pp. 278–284, Dec. 2021.

[4] Y. Hui, Y. Huang, Z. Su, T. H. Luan, N. Cheng, X. Xiao, and G. Ding, “Bcc: Blockchain-based collaborative crowdsensing in autonomous vehicular networks,” *IEEE Internet of Things Journal*, vol. 9, no. 6, pp. 4518–4532, Mar. 2022.

[5] W. Zhang, D. Yang, W. Wu, H. Peng, N. Zhang, H. Zhang, and X. Shen, “Optimizing federated learning in distributed industrial iot: A multi-agent approach,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3688–3703, Dec. 2021.

[6] N. Ding, Z. Fang, and J. Huang, “Optimal contract design for efficient federated learning with multi-dimensional private information,” *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 186–200, Jan. 2021.

[7] Y. Jiao, P. Wang, D. Niyato, B. Lin, and D. I. Kim, “Toward an automated auction framework for wireless federated learning services market,” *IEEE Transactions on Mobile Computing*, vol. 20, no. 10, pp. 3034–3048, Oct. 2021.

[8] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani, “Reliable federated learning for mobile networks,” *IEEE Wireless Communications*, vol. 27, no. 2, pp. 72–80, Apr. 2020.

[9] Y. Zhan, P. Li, Z. Qu, D. Zeng, and S. Guo, “A learning-based incentive mechanism for federated learning,” *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6360–6368, July 2020.

[10] M. Wu, D. Ye, J. Ding, Y. Guo, R. Yu, and M. Pan, “Incentivizing differentially private federated learning: A multidimensional contract approach,” *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 639–10 651, July 2021.

[11] Y. Hui, N. Cheng, Z. Su, Y. Huang, P. Zhao, T. H. Luan, and C. Li, “Secure and personalized edge computing services in 6g heterogeneous vehicular networks,” *IEEE Internet of Things Journal*, vol. 9, no. 8, pp. 5920–5931, Apr. 2022.

[12] S. Chen, J. Hu, Y. Shi, and L. Zhao, “Lte-v: A td-lte-based v2x solution for future vehicular network,” *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 997–1005, Dec. 2016.

[13] S. Chen, J. Hu, Y. Shi, L. Zhao, and W. Li, “A vision of c-v2x: Technologies, field testing, and challenges with chinese development,” *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 3872–3881, May 2020.

[14] Y. Hui, Z. Su, and T. H. Luan, “Unmanned era: A service response framework in smart city,” *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–15, Feb. 2021.

[15] G. Mao, Y. Hui, X. Ren, C. Li, and Y. Shao, “The internet of things for smart roads: A road map from present to future road infrastructure,” *IEEE Intelligent Transportation Systems Magazine*, vol. PP, no. 99, pp. 1–12, Nov. 2021.

[16] J. Li, W. Shi, Q. Ye, S. Zhang, W. Zhuang, and X. Shen, “Joint virtual network topology design and embedding for cybertwin-enabled 6g core networks,” *IEEE Internet of Things Journal*, vol. 8, no. 22, pp. 16 313–16 325, Nov. 2021.