FL in Motion: Accelerating FL via Mobility-Aware Vehicle Selection and Sparse Training

Haoyu Tu, *Graduate Student Member, IEEE*, Wen Wu, *Senior Member, IEEE*, Lin Chen, *Member, IEEE*, Liang Li, *Member, IEEE*, Xu Chen, *Senior Member, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Although federated learning (FL) can enable advanced autonomous driving via leveraging massive distributed data in vehicular networks, vehicle mobility causes frequent connection interruptions, hindering the FL process. In this paper, we propose a novel Mobility-Aware Vehicular FL (MAVFL) scheme, which can accelerate the training process in dynamic vehicular networks via adaptive vehicle selection and sparse training. Specifically, the MAVFL dynamically selects participating vehicles based on their locations and training loss. By incorporating adaptive model sparsification, the proposed scheme dynamically proceeds with sparse masks during vehicle local training, thereby reducing communication overhead while preserving model accuracy. We conduct a rigorous convergence analysis to uncover how vehicle mobility and model sparsification affect convergence rate. Furthermore, we formulate an optimization problem to accelerate the training process, which jointly optimizes vehicle selection, sparsification ratio, and bandwidth allocation to minimize training delay. To solve the problem, we employ the Lyapunov optimization method to decouple the long-term problem into a series of instantaneous subproblems. Next, a generalized Benders decomposition method structures the original problem into a master subproblem for vehicle selection and a primal subproblem for bandwidth allocation and sparsification ratio selection. The optimal solutions are derived via alternating iterations between these problems. Extensive simulation results based on the SUMO simulator demonstrate that the MAVFL accelerates model convergence by up to 14% and reduces communication overhead by up to 26% while preserving model accuracy, as compared to the state-of-the-art benchmarks.**

***Index Terms*—Vehicular federated learning, mobility-aware, sparse training, vehicle selection.**

Received 21 August 2025; revised 1 December 2025; accepted 13 January 2026. Date of publication 16 January 2026; date of current version 5 June 2026. This work was supported in part by Pengcheng Laboratory Major Key Project under Grant 2025QYA002 and Grant PCL2025A08, in part by the Basic and Frontier Research Project of PCL under Grant 2025QYB014, in part by the National Natural Science Foundation of China under Grant 62172455, and in part by the Guangdong Science and Technology Department Pearl River Talent Program under Grant 2019QN01X140. Recommended for acceptance by Y. Hu. *(Corresponding author: Wen Wu.)*

Haoyu Tu is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China, and also with Frontier Research Center, Pengcheng Laboratory, Shenzhen 518055, China (e-mail: tuhy3@mail2.sysu.edu.cn).

Wen Wu and Liang Li are with Frontier Research Center, Pengcheng Laboratory, Shenzhen 518055, China (e-mail: wuw02@pcl.ac.cn; lil03@pcl.ac.cn).

Lin Chen is with the Engineering Research Centre of Applied Technology on Machine Translation and Artificial Intelligence, Macao Polytechnic University, Macao (e-mail: lchen@mpu.edu.mo).

Xu Chen is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China (e-mail: chenxu35@mail.sysu.edu.cn).

Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

This article has supplementary downloadable material available at https://doi.org/10.1109/TMC.2026.3654937, provided by the authors.

Digital Object Identifier 10.1109/TMC.2026.3654937

I. Introduction

AUTONOMOUS driving is expected to play a pivotal role in modern intelligent transportation systems, requiring extensive data-driven artificial intelligence (AI) tasks to enhance road safety and mitigate urban congestion [1], [2], [3]. These capabilities primarily rely on data collected from on-board sensors of vehicles, including radar [4], [5], LiDAR [6], [7], and cameras [8], [9]. Autonomous driving tasks, such as object detection, object tracking, and trajectory prediction, require a well-trained AI model based on the local data of numerous vehicles [10], [11].

Vehicular federated learning (VFL) has emerged as a promising solution for enabling autonomous driving tasks. This development is significantly enhanced by recent advances in on-board computational units, such as NVIDIA Drive Origin SoC chips, which now can provide up to 500 TOPS of processing power [12]. VFL enables vehicles to train models locally and periodically aggregate them with the assistance of roadside edge servers. The VFL paradigm helps mitigate the privacy leakage risk of local data and reduces significant communication overhead inherent in the traditional centralized training paradigm, offering a scalable and efficient solution for collaborative edge intelligence in vehicular networks [13], [14].

Existing research on VFL mainly focuses on two aspects: model aggregation and resource management. For the model aggregation, to accelerate the training process, the inner-cluster and inter-cluster training schemes are proposed for vehicles with multi-hop clusters [15]. Asynchronous aggregation methods are developed for vehicles that upload locally trained models at different time slots [16]. For resource management, to handle communication capability variations caused by vehicle mobility, several works focus on optimizing communication, computing resource allocation and privacy budget consumption [12], [17], [18], [19]. The training round duration and the number of local epochs are optimized with the consideration of vehicle mobility [20]. However, these solutions often overlook the inherent heterogeneity of vehicles in terms of varying communication capabilities, underscoring the need for vehicle selection. Furthermore, the computation and communication overhead of AI models in dynamic vehicular networks necessitates techniques for

complexity reduction. Model sparsification, which involves removing redundant weights to reduce model complexity, is a potential technology for significantly cutting down computational burden and communication overhead. As such, on-board model training and transmission can be completed efficiently within the limited communication window mandated by the rapid movement of vehicles in the roadside base station's coverage.

Vehicle mobility poses two fundamental challenges for VFL systems. *Firstly*, dynamic vehicle movement alters vehicle-to-BS distances, causing time-varying communication latency. To mitigate this instability, adaptive vehicle selection can be employed to prioritize participants with more stable connections and longer expected residency times. *Secondly*, vehicles moving across different coverage zones experience intermittent connectivity with the base station and cause inconsistent model updates. Concurrently, to handle these inconsistent updates, adaptive model sparsification can be applied to reduce the communication burden by tailoring model size to each vehicle's channel capacity. However, these two strategies are fundamentally intertwined: vehicles in poor coverage zones often require aggressive sparsification to ensure the successful transmission of models within transient connection windows, but such extreme compression inevitably collectively delays global model synchronization and reduces convergence speed. This critical trade-off necessitates a co-design of vehicle selection and model sparsification to effectively accelerate convergence in dynamic VFL systems.

In this paper, we propose a Mobility-Aware Vehicular Federated Learning (MAVFL) scheme that accelerates model training through dynamic vehicle selection and adaptive sparse training. Firstly, the MAVFL scheme jointly optimizes vehicle selection and assigns a unique sparsification ratio to each vehicle. This tailors the model update size to individual mobility and channel conditions, ensuring timely uploads and maximizing successful participation from vehicles with transient connectivity. Secondly, we provide a rigorous convergence analysis, which theoretically quantifies the joint impact of correlated vehicle participation (due to mobility) and heterogeneous model divergence (due to adaptive sparsification). The proposed convergence analysis establishes a guaranteed performance bound for our dynamic VFL system. Thirdly, we formulate a long-term convergence latency minimization problem and design an efficient online algorithm to solve it. The proposed algorithm uses a Lyapunov framework to decompose the problem into per-round subproblems, which are then efficiently solved using a Generalized Benders Decomposition (GBD) to handle the mixed-integer nature of vehicle selection and resource allocation. Extensive simulation results based on the SUMO simulator and real-world datasets demonstrate 14.8% faster convergence and a 26% reduction in communication overhead compared to conventional VFL approaches. The main contributions of the paper are summarized as follows.

- We propose the MAVFL scheme that integrates model sparsification and mobility-aware vehicle selection to counteract straggler effects and connection instability.
- We establish theoretical convergence bounds characterizing how vehicle mobility and model sparsification collectively influence VFL training performance.
- We formulate a long-term optimization problem with the objective of minimizing training latency.
- We design an efficient algorithm that selects participating vehicles, model sparsification ratio, and bandwidth allocation dynamically in vehicular networks with high mobility.

TABLE I
IMPORTANT SYMBOLS AND NOTATIONS

Notation	Description
Z	The number of zones of covering segment
D_k	The dataset of vehicle k
$x_k^{r,e}$	The location of vehicle k in round r, epoch e
$v_k^{r,e}$	The velocity of vehicle k in round r, epoch e
$\ell(\cdot)$	Local loss function for vehicles
$F(\cdot)$	Global loss function
$\mathcal{S}^r$	The set of selected vehicles in round r
$\mathcal{N}^r$	The set of vehicles successfully uploading model in round r
$\mathbf{w}^r$	The global model in round r
$\mathbf{w}_k^{r,e}$	The local model of vehicle k in round r at epoch e
α_k^r	The sparsity ratio for vehicle k in round r
$\mathbf{m}_k^r$	The mask generated for vehicle k in round r
$L_{k,z}^r$	The distance between vehicle k and BS at round r
Q_k^r	The sparsification error of vehicle k in round r
μ_k^r	The uplink transmission rate of vehicle k in round r
$\mathcal{Z}_k^r$	The set of zones where vehicle z passes through
$T_{k,\mathrm{c}}^r$	The communication time of vehicle k in round r
$T_{k,\mathrm{p}}$	The communication time of vehicle k in round r
b_k^r	The percentage of bandwidth allocated for vehicle k
a_k^r	The indicator of whether the vehicle k is selected in round r

The remainder of this paper is organized as follows. The related works are presented in Section II. In Section III, the MAVFL scheme and convergence analysis are introduced. The delay model and the optimization problem are presented in Section IV. The mobility-aware vehicle selection algorithm is developed in Section V. The simulation results are given in Section VI, and Section VII concludes the paper.

II. Related Work

In the literature, the widespread deployment of on-board sensors for vehicles has been demonstrated to facilitate the implementation of heterogeneous tasks in dynamic vehicular networks, such as target detection [8], [9], beam selection [6], [21], trajectory prediction [4], [22], etc. To improve the performance of sector selection, Salehi et al. [23] proposed a multimodal FL framework that leverages a dataset collected by an autonomous vehicle and a BS, all coordinated by a collaborative server. Salehi et al. [23] proposed a multimodal FL framework with a dataset collected by an autonomous vehicle to improve the performance of sector selection. Xing et al. proposed a joint time-series modeling approach with various driving modes to predict the trajectory of the leading vehicles [22]. Mashhadi et al. [6] proposed a reduced-complexity classifier architecture and a LIDAR preprocessing scheme through FL on their locally available data.

Recent efforts to address mobility challenges in VFL focus on designing vehicle selection and aggregation strategies. The vehicle selection algorithms leverage real-time vehicular metrics, such as energy consumption [24], [25], model similarity [26],

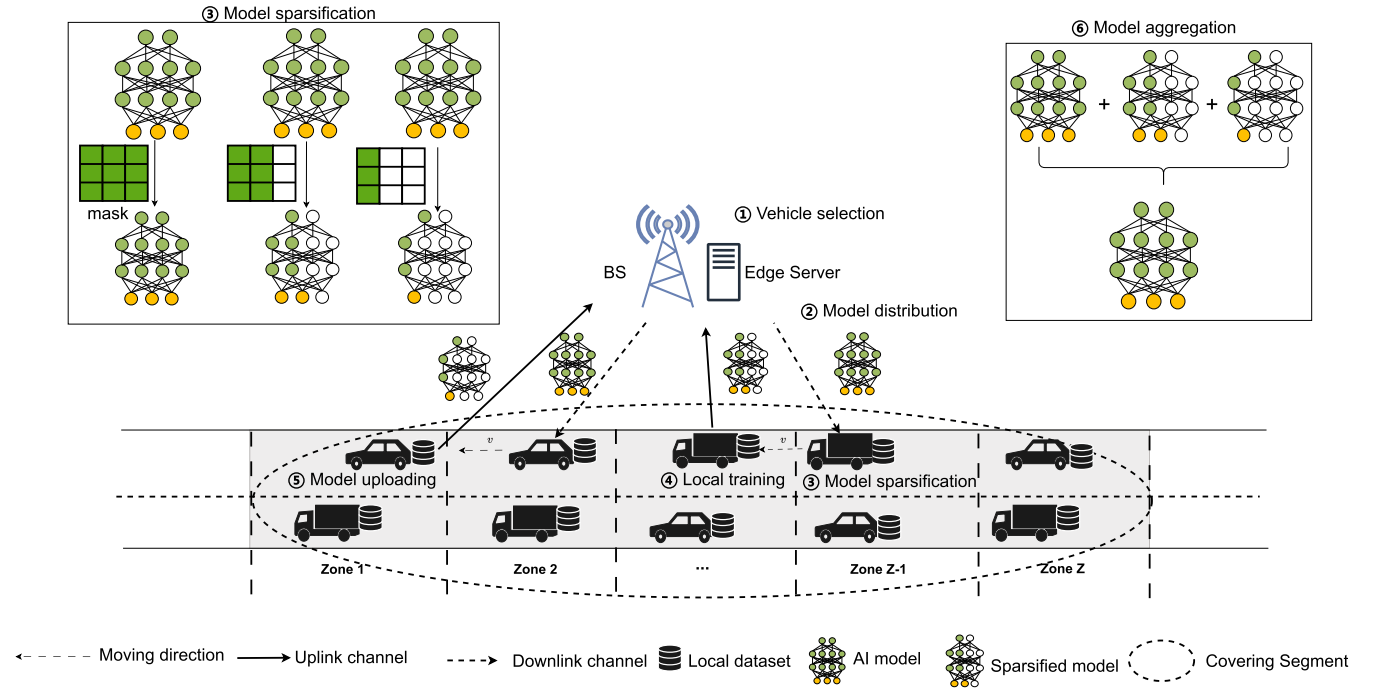


Fig. 1. An illustration of our MAVFL system. The training procedures contain six steps: ① vehicle selection, ② model distribution, ③ model sparsification, ④ local training, ⑤ model uploading, and ⑥ model aggregation.

[27], and training accuracy [28], [29], to dynamically prioritize vehicles. For instance, Xu et al. [25] introduced a long-term optimization framework integrating energy consumption and bandwidth allocation. Fraboni et al. [26] proposed the cluster sampling algorithm based on data distribution and model divergence. Aggregation schemes mitigate mobility impacts by incorporating vehicular participation and dataset scales. For example, the authors in [12] designed weighted aggregation to maximize the successful training participation ratio, and Feng et al. [30] accelerated convergence via hierarchical FL architectures. However, these methods predominantly rely on long-term statistical mobility patterns, without considering short-term movement information, which limits the adaptability to highly dynamic vehicular networks.

Sparsification techniques have gained attention to reduce communication and computation overhead in FL. Jiang et al. [31] introduced a multi-armed bandit-based approach to adaptively select sparsification ratios, balancing training speed and model quality. While quantization methods [32], [33] dynamically adjust compression rates according to network conditions, existing sparsification frameworks [31], [34] focus primarily on static or low-mobility scenarios. In particular, these sparsification techniques do not explicitly model vehicle mobility patterns, such as transient connectivity or spatial distributions, limiting their effectiveness in VFL. This gap underscores the need for mobility-aware sparsification strategies that jointly optimize communication efficiency and model robustness in dynamic vehicular networks.

Different from existing works, we design the vehicle selection algorithm based on real-time vehicle position and training loss values. Considering the straggling effect, we also design sparse training process and a mobility-aware vehicle selection algorithm to accelerate the training process and maintain the training accuracy in VFL.

III. Proposed MAVFL Scheme

A. Considered Scenario

As shown in Fig. 1, we consider a road segment of length L that is covered by the BS with a computing server. During the training process, K_0 vehicles will arrive at this segment of the road in total, and K_r vehicles are available for selection within the BS's covering segment in training round r. The maximum number of vehicles that can be simultaneously present within the BS's covering segment is set as $K = \max_k\{K_r\}$. Consider a deep learning (DL)-based autonomous driving service (e.g., trajectory prediction, image detection) that operates on an edge server and requires continuous model updates relying on collaborative vehicular training. We denote the local dataset as D_k owned by the vehicle $k \in \{1, 2, \ldots, K_0\}$. For simplicity, we divide the road segment into Z fine-grained zones as $\mathcal{Z} = \{1, 2, \ldots, Z\}$. Given that the distance from the base station to any vehicle significantly exceeds the road zone diameter, we assume vehicles within the same zone maintain approximately equal distances to the base station, while vehicles in different zones experience materially different distances.

The server is able to get the location and velocity of vehicle k within the segment during training as x_k^t and v_k^t at $t < T_R$ with the maximum deadline T_R. Vehicle mobility leads to dropouts when vehicles move beyond BS coverage, resulting in incomplete model collection at the server. Then the successful training

ratio p^r is defined by

$$p^r = \frac{|\mathcal{N}^r|}{|\mathcal{S}^r|},$$

where $|\mathcal{N}^r|$ is the number of uploaded models successfully received by the server in round r, and $|\mathcal{S}^r|$ counts downloaded models successfully received by vehicles.

B. MAVFL Procedures

The goal of MAVFL is to minimize the global training loss as

$$F(w) = \sum_{k=1}^{K} \gamma_k f(\mathbf{w}_k, D_k),$$

where $f(\mathbf{w}_k, D_k)$ is the loss function for vehicle k of local model $\mathbf{w}_k$ as $f(\mathbf{w}_k, D_k) = \frac{1}{|\mathcal{D}_k|}\sum_{i\in\mathcal{D}_k} \ell(\mathbf{w}_k; z_k^i)$, $\mathcal{D}_k$ means the sample indices of D_k, γ_k is the aggregation factor for vehicle k, and z_k^i is the data sample representation of $i \in \{1, 2, \ldots, |\mathcal{D}_k|\}$ and ℓ is the local loss function. Then vehicle k trains local model $\mathbf{w}_k$ to obtain the optimal model as $\hat{\mathbf{w}}_k = \arg\min_{\mathbf{w}_k} \frac{1}{|\mathcal{D}_k|}\sum_{i=1}^{|\mathcal{D}_k|} \ell(\mathbf{w}_k; z_k^i)$.

When vehicles move beyond BS coverage during training, their model updates are lost, resulting in a decrease in training accuracy. To solve the problem, we adaptively design the mobility-aware vehicle selection and sparsification ratios distribution algorithm. At the beginning of each training round, the server selects participating vehicles and assigns vehicle-specific sparsification ratios based on their movement patterns and training status. Selected vehicles then sparsify local models and upload compressed parameters via allocated uplink bandwidth. By incorporating vehicle mobility into both selection and sparsification decisions, MAVFL maximizes successful participation per training round. The pseudocode of our MAVFL algorithm is given in Algorithm 1, and the details are described in the following.

1) Vehicle Selection: At the beginning of every training round $r \in \{1, 2, \ldots, R\}$, the server owns the current global model $\mathbf{w}^r$. The server then determines the set of participating vehicles from the connected vehicles to participate in the current round FL training, denoted by $\mathcal{S}^r = \{k, \forall a_k^r = 1\}$. Here, $a_k^r \in \{0, 1\}$ indicates that the vehicle k is selected in the r-th round; otherwise, $a_k^r = 0$.

2) Model Distribution: After vehicle selection, the global model is distributed to vehicles within the set $\mathcal{S}^r$ as

$$\mathbf{w}_k^{r,0} \leftarrow \mathbf{w}^r, \forall k, x_k^{r,0} \in \mathcal{X}_s, k \in \mathcal{S}^r, \tag{1}$$

where $\mathbf{w}_k^{r,0}$ is the local model of vehicle k in round r and epoch 0, $\mathcal{X}_s$ is the coordinate range of covering segment, and $x_k^{r,0}$ is the location of vehicle k at the beginning of round r. Especially when all vehicles are stationary, the ratio p^r is 1; otherwise, when all selected vehicles drive out of the covering segment during local computing, the ratio is 0. When p^r is 0, the server will distribute the global model as $\mathbf{w}^{r+1} = \mathbf{w}^r$.

3) Model Sparsification: The server adaptively determines the sparsification ratio list of selected vehicles at the beginning of round r as

$$\boldsymbol{\alpha}^r = \{\alpha_k^r, \forall k \in \mathcal{S}^r\}.$$

Here, the sparsification ratio $\alpha_k^r \in [\alpha_{\min}, 1]$ means the remaining percentage of weights for vehicle k in round r without sparsification. Then the server sends the list of sparsification ratios with the global model.

When vehicle k receives the global model $\mathbf{w}^r$ and the sparsification ratio α_k^r from the server, it generates a local mask $\mathbf{m}_{k,l}^r \in \{0,1\}^{|\mathbf{w}_l^r|}$ for the weights of each layer l. The elements of the mask $\mathbf{m}_{k,l}^r[i]$ for the i-th weight in layer l are defined by the following piecewise function:

$$\mathbf{m}_{k,l}^r[i] = \begin{cases} 1, & \text{if } 1 \le i \le \lfloor \alpha_k^r |\mathbf{w}_l^r| \rfloor, \\ 0, & \text{otherwise}, \end{cases} \tag{2}$$

where $|\mathbf{w}_l^r|$ denotes the total number of parameters in layer l. This formulation indicates that for every layer, the first $\lfloor \alpha_k^r |\mathbf{w}_l^r| \rfloor$ elements are retained (set to 1), while the remaining elements are pruned (set to 0). After generating the local masks for all layers, the sparse local model $\mathbf{w}_k^{r,0}$ for vehicle k is derived via an element-wise product: $\mathbf{w}_k^{r,0} = \mathbf{w}^r \odot \mathbf{m}_k^r$, where $\mathbf{m}_k^r$ is the concatenation of all layer masks $\mathbf{m}_{k,l}^r \in \{0,1\}^{|\mathbf{w}_l^r|}$.

4) Local Training: After generating the sparsified local model $w_k^{r,0}$, vehicle k performs local stochastic gradient descent (SGD) for E epochs. For local training epoch $e \in \mathcal{E} = \{0, 1, \ldots, E-1\}$, the gradient descent is expressed as $g_k^{r,e} \leftarrow \frac{1}{|\mathcal{D}_k|}\sum_{i=1}^{|\mathcal{D}_k|} \nabla \ell(\mathbf{w}_k^{r,e}; z_k^i)$ and vehicle k performs local SGD as

$$\mathbf{w}_k^{r,e+1} \leftarrow \mathbf{w}_k^{r,e} - \eta g_k^{r,e} \odot \mathbf{m}_k^r, \forall e \in [0, E-1] \tag{3}$$

where η is the learning rate. During local training, the sparsified weights with the element-wise product of the local mask $\mathbf{m}_k^r$ are not updated.

5) Model Uploading: After local training, vehicle k attempts to upload model updates $g_k^r = \sum_{e=0}^{E-1} g_k^{r,e} \odot \mathbf{m}_k^r$ to the server. When vehicles move beyond BS coverage during training, their model updates are lost. We define the indicator $\mathbb{1}_k^r\left(x_k^{r,E}\right) \in \{0, 1\}$ as

$$\mathbb{1}_k^r(x_k^{r,E}) = \begin{cases} 1, & x_k^{r,E} \in \mathcal{X}_s, \\ 0, & x_k^{r,E} \notin \mathcal{X}_s \end{cases}$$

to represent whether vehicle k stays within the covering segment of BS after local training with E epochs in round r.

6) Model Aggregation: The server receives model updates from the selected vehicles within the covering segment of the BS. If the server receives any model updates within $T_{\max}$, it will aggregate these models to get new global model $\mathbf{w}^{r+1}$ as

$$\mathbf{w}^{r+1} = \mathbf{w}^r - \eta \sum_{k\in\mathcal{K}} \mathbb{1}_k^r \left(\frac{g_k^r \odot \mathbf{m}_k^r}{\sum_{k\in\mathcal{K}} \mathbb{1}_k^r} \right), \forall \sum_{k\in\mathcal{K}} \mathbb{1}_k^r \neq 0. \tag{4}$$

C. Convergence Analysis

We analyze MAVFL's convergence under mobility-induced dynamics and sparsification effects, leveraging assumptions and a lemma from [30], [31], [35] for analytical tractability.

Algorithm 1: Mobility-aware vehicular federated learning (MAVFL) scheme.

for *training round* $r = 1, 2, \ldots, R$ **do**
  `// Model Distribution`
  Edge server collects the vehicle information $x_k^{r,0}, v_k^r, h_k, f_k$ and pre-training information $\mathcal{A}^r$;
  Edge server determines the set of vehicle selection and model sparsification ratio as $\mathcal{S}^r$ and $\{\alpha_k^r\}$;
  Edge server distributes the newest global model $\mathbf{w}^r$ and model sparsification ratio $\{\alpha_k^r\}$ to the vehicles in $\mathcal{S}^r$;
  for *Each vehicle* $k \in \mathcal{S}^r$ **do**
    `// Local Training`
    Vehicle k generates the local mask based on (2) to generate the masked model;
    Vehicle k performs local SGD based on (3);
    `// Model Uploading`
    Vehicle k uploads the parameters with sparsification to server in $\mathbf{w}_k^{r,E}$;
  end
  `// Model Aggregation`
  Edge server recovers local models $\mathbf{w}_k^{r,E}, k \in \mathcal{S}^r$ based on sparsification ratio and received parameters;
  Edge server aggregates local models to generate new global model based on (4);
end

Assumption 1: The loss function is L-smooth as $\|\nabla F(\mathbf{w}_1) - \nabla F(\mathbf{w}_2)\| \leq L\|\mathbf{w}_1 - \mathbf{w}_2\|$ for arbitrary given $\mathbf{w}_1$ and $\mathbf{w}_2$.

Assumption 2: The expected squared norm of stochastic gradients for each vehicle k is upper-bounded by $\mathbb{E}\|\nabla f(\mathbf{w}_k^{r,e})\|^2 \leq \tilde{G}^2, \forall k, \forall r, \forall e$.

Assumption 3: The variance of mini-batch gradients is upper-bounded by $\|g(\mathbf{w}) - \nabla f(\mathbf{w})\|^2 \leq \sigma^2$.

Assumption 4: The divergence between local and global loss functions is bounded by $\frac{1}{K}\sum_{k=1}^{K}\|\nabla f(\mathbf{w}) - \nabla F(\mathbf{w})\|^2 \leq \epsilon_g^2, \forall \mathbf{w}$.

Assumption 5: The weights of local and global model is bounded: $\mathbb{E}[\| \mathbf{x} \|^2] \leq \delta^2, \forall \mathbf{x}$.

Lemma 1: The divergence of the global model and sparsified local model is bounded by [31]:

$$\sum_{t=1}^{E}\sum_{k=1}^{K}\mathbb{E}\left[\| \mathbf{w}^r - \mathbf{w}_k^{r,t-1} \|^2\right] \leq \frac{2E^3\gamma^2G^2N}{3} + 2E\delta^2\sum_{k=1}^{K}(1-\alpha_k^r).$$

According to Assumptions 1-5 and Lemma 1, we have the following theorem.

Theorem 1: The convergence rate of the proposed MAVFL scheme satisfies

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\nabla F(\bar{\mathbf{w}}^t)\|^2\right] \leq \frac{1}{T}\sum_{t=1}^{T}\frac{2}{\eta p^t}(\mathbb{E}[F(\bar{\mathbf{w}}^0)] - F_{\text{inf}}) + 2\epsilon_g^2 + \eta(\delta^2+G^2)L + \frac{4E^3\gamma^2G^2}{3} + \frac{4L^2\delta^2}{KT}\sum_{k=1}^{K}\sum_{t=1}^{T}(1-\alpha_k^t). \quad (5)$$

Proof: See Appendix A. □

Remark 1: Theorem 1 establishes that convergence depends critically on the participation ratio p^t (driven by vehicle selection) and the sparsification ratio α_k^t. The convergence upper bound degrades under low p^t and high α_k^t.

Remark 2: The convergence of the proposed scheme is easily observed. When T increases to infinity, the training loss is bounded by $2(\epsilon_g)^2 + \eta(\delta^2+G^2)L + \frac{4E^3\gamma^2G^2}{3}$, which is a constant influenced by only training parameters.

Remark 3: We set ε as the target training loss of the proposed MAVFL scheme. The training loss refers to the total loss value when the training converges within a given round R. Based on the convergence result in Theorem 1, we have the following convergence constraint

$$\sum_{k=1}^{K}\sum_{r=1}^{R}\frac{A}{p(a_k^r)} - \sum_{k=1}^{K}\sum_{r=1}^{R}\frac{P\alpha_k^r}{K} + C \leq \varepsilon, \quad (6)$$

where $A = \frac{2}{\eta}(\mathbb{E}[F(\bar{\mathbf{w}}^0)] - F_{\text{inf}})$, $P = 4L^2\delta^2$ and $C = R[2\epsilon_g^2 + \eta(\delta^2+G^2)L + \frac{4E^3\gamma^2G^2}{3} + 4\tilde{}L^2\delta^2]$. Here, $p(a_k^r)$ represents the successful transmission probability of vehicle k if it is selected to participate in round r. The probability $p(a_k^r)$ is zero if vehicle k is not selected in round r. Considering the relation of vehicle selection and successful training probability, p^r is derived by averaging the individual successful transmission probabilities $p(a_k^r)$ for all selected vehicles in each round as $\sum_k p(a_k^r)/K = p^r$, and the inequality $\frac{1}{a+b} < \frac{1}{2a} + \frac{1}{2b}$ is utilized.

Corollary 1: To evaluate the successful probability p_k^r, we adopt the expectation-maximization method [36] and rewrite the p^r as

$$p_k^r(\theta|\mathcal{A}_k^r) = \frac{\sum_{(i,j)\in\mathcal{A}_k^r}\Pr(j|i,\theta_k)}{\sum_{\theta_k'\in\Theta_k}\sum_{(i,j)\in\mathcal{A}_k^r}\Pr(j|i,\theta_k')}.$$

Here, $\mathcal{A}_k^r$ is the observation of selected vehicle historical successful training states in previous training r rounds, θ_k and Θ_k is the event and set where vehicle k participates training successfully respectively, and $\Pr(j|i,\theta)$ is the probability that vehicle k moves from i to j during training given the successful training information. The successful probability p^r is calculated as a moving average over a historical window of the last r rounds, which prevents it from abrupt mobility patterns. Based on the above formulation, the successful training probability p_k^r is initially derived from a statistical model of historical location transitions and is then dynamically updated using a moving average of recent successful training outcomes.

IV. System Model and Problem Formulation

In this section, we first give the system model of the MAVFL scheme, then we formulate the optimization problem based on the convergence analysis results to minimize the training delay.

A. Training Delay Model

In this subsection, we give the delay analysis of the MAVFL scheme. The distance of vehicle k and BS as L_k^r is related to the movement of vehicles. We define the normalized distance between the vehicle and BS in zone z as $L_{k,z}^r$. The uplink transmission for selected vehicles is considered as an orthogonal frequency-division multiple access (OFDMA), and the total bandwidth is B_0. Then the uplink transmission rate of vehicle k in zone z is

$$\mu_{k,z}^r = b_k^r B_0 \log_2\left(1 + \frac{P h_k (L_{k,z}^r)^{-\beta}}{N_0}\right), \tag{7}$$

where $b_k^r \in [b_{\min}, 1]$ is the percentage of bandwidth allocated for vehicle k, P is the transmission power, h_k is the channel power gain for vehicle k, β is the pass loss exponent and N_0 is the noise power. Then we can get the average uplink transmission rate as

$$\mu_k^r = \frac{\sum_{z \in \mathcal{Z}_k^r} \mu_{k,z}^r}{|\mathcal{Z}_k^r|}, \tag{8}$$

where $\mathcal{Z}_k^r \subseteq \mathcal{Z}_k$ is the set of zones where vehicle k passes through.

Our sparsification method selectively zeros elements in high-dimensional weight matrices. Each vehicle transmits only non-zero weight values and sparse mask indices. The server reconstructs complete local models using this compressed data. While sparse indices incur minimal overhead, we exclude this cost in delay analysis [37]. Then the communication time for vehicle k is expressed as

$$T_{k,\mathrm{c}}^r = \frac{\alpha_k^r M}{\mu_k^r}, \tag{9}$$

where M denotes the transmission size of the original model.

Next, the computation time for vehicle k is given by

$$T_{k,\mathrm{p}} = \frac{E|\mathcal{D}_k| g_k}{f_k}, \tag{10}$$

where g_k is the number of GPU cycles required to train data for one local epoch, E is the local epoch, and f_k is the GPU frequency.

In a synchronous FL framework, the server performs global aggregation only upon receiving all participant models. Thus, MAVFL's convergence time becomes:

$$T_{\mathrm{all}} = \sum_{r \in \mathcal{R}} \max_{k \in \mathcal{K}} \left(a_k^r (T_{k,\mathrm{c}}^r + T_{k,\mathrm{p}})\right). \tag{11}$$

B. Problem Formulation

We formulate a training delay minimization problem for MAVFL, which jointly optimizes the vehicle selection $\{a_k^r\}$, bandwidth allocation $\{b_k^r\}$, and the sparsification ratio $\{\alpha_k^r\}$. The optimization problem is formulated as follows:

$$\mathcal{P}_1: \min_{\{\mathbf{a}^r, \mathbf{b}^r, \{\alpha_k^r\}\}_{r \in \mathcal{R}}} \sum_{r=1}^{R} \max_{k \in \mathcal{K}} \left(a_k^r \left(T_{k,\mathrm{c}}^r + T_{k,\mathrm{p}}\right)\right), \tag{12a}$$

$$\text{s.t.} \quad a_k^r \in \{0,1\}, \forall r \in \mathcal{R}, k \in \mathcal{K}, \tag{12b}$$

$$\sum_{k=1}^{K} a_k^r = K_0, \forall r \in \mathcal{R}, \tag{12c}$$

$$\alpha_k^r \in [\alpha_{\min}, 1], \forall r \in \mathcal{R}, k \in \mathcal{K} \tag{12d}$$

$$B_0 \sum_{k=1}^{K} b_k^r = B, \forall r \in \mathcal{R}, \tag{12e}$$

$$b_k^r \in [b_{\min}, 1], \forall r \in \mathcal{R}, k \in \mathcal{K}^r, \tag{12f}$$

$$(6). \tag{12g}$$

The constraint in (12b) ensures the feasibility condition of the vehicle selection. The constraint in (12c) limits the number of selected vehicles. The constraint (12d) guarantees the feasibility condition on the model sparsification ratio. The constraint (12e) indicates that the sum of bandwidth allocation should be B. The constraint (12f) specifies the feasible condition on the bandwidth allocation. The constraint (12g) ensures that the total training loss of the proposed scheme does not exceed the target upper bound ε.

The above optimization problem is difficult to solve. The first reason is that $\mathcal{P}_1$ is a mixed integer nonlinear programming (MINLP) problem with min-max optimization, which is NP-hard. Consequently, finding its globally optimal solution directly is computationally intractable. Secondly, addressing the above optimization problem $\mathcal{P}_1$ requires offline vehicle information for future R rounds, such as location, communication conditions, etc. Thirdly, constraint (12g) is a long-term constraint requiring future knowledge of the vehicle's training.

V. Proposed Algorithm

In this section, we provide an efficient solution to the aforementioned long-term MINLP delay-efficient problem. The workflow of the proposed algorithm is shown in Fig. 2, which shows the convergence of proposed algorithm. The original problem is transformed into the minimum problem and decomposed into a sequence of sub-problems using the Lyapunov technique. Then, the generated Benders decomposition method is adopted to handle the integer variable and continuous variables respectively: the integer variables (e.g., vehicle selection decisions) are optimized in the master problem through the branch-and-bound method, while the continuous variables (e.g., sparsification ratio and bandwidth allocation) are resolved in the sub-problems via an iteration algorithm. This dual-layer approach iteratively tightens the feasibility and optimality cuts, ensuring convergence to a near-optimal solution with reduced computational complexity compared to monolithic solvers.

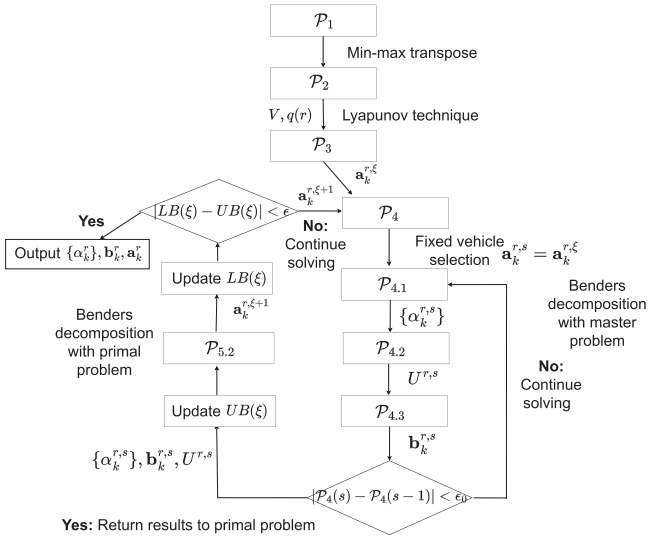


Fig. 2. The logical flow of solving problem $\mathcal{P}_1$.

A. Problem Transformation and Lyapunov-Based Long-Term Optimization

By introducing an auxiliary variable U satisfying $U^r \geq (T_{k,c}^r + T_{k,p}), \forall k \in \mathcal{K}$, the problem $\mathcal{P}_1$ can be equivalently rewritten as

$$\mathcal{P}_2 : \min_{\{\mathbf{a}^r, \mathbf{b}^r, \{\alpha_k^r\}, U^r\}_{r\in\mathcal{R}}} \sum_{r=1}^{R} U^r \tag{13a}$$

$$\text{s.t.} \quad (12b) - (12g) \tag{13b}$$

$$U^r \geq (T_{k,c}^r + T_{k,p}), \forall k \in \mathcal{K}, \forall r \in \mathcal{R}. \tag{13c}$$

Considering the long-term constraint (12g), the vehicle selection decision is coupled with the distribution of sparsification ratio across different training rounds. Assigning a larger sparsification ratio to vehicles reduces the latency, increases the probability of successful transmission of the vehicle and the corresponding training loss, which leads to the reduction of the training loss budget for future rounds. Applying a low sparsification ratio may improve the final training accuracy, but it will increase the training latency. To solve the problem, we apply the Lyapunov technique and establish a virtual convergence deficit queue, which gives guidance on vehicle selection and model sparsification ratio allocation.

For each vehicle k in round r, we propose the virtual training loss queue $q(r)$, which means the gap between the vehicle training loss in round r and the long-term convergence constraint as

$$q(r) = \begin{cases} \max\left\{\sum_{k\in\mathcal{K}} E\left(a_k^{r-1}, \alpha_k^{r-1}\right) - \frac{\varepsilon}{R} + q(r-1), 0\right\}, \\ r > 1; \\ 0, r = 1. \end{cases} \tag{14}$$

Here, we obtain the equation $E(a_k^r, \alpha_k^r) = \frac{A}{p(a_k^r)} - \frac{P\alpha_k^r}{K} + \frac{C}{KR}$ as the training loss for each vehicle k in round r by separating each round respectively in (6). The Lyapunov function can be defined as $L(q(r)) \triangleq \frac{1}{2}q^2(r)$, which means the congestion level in the training loss queue. A small value of $L(q(r))$ means that the queue backlog is insignificant, which implies that the virtual queue is highly stable. During the training, the value of the Lyapunov function should be lower as the stability of the training loss is guaranteed. We then introduce one-shot Lyapunov drift to ensure the stability as

$$\Delta(q(r)) = \mathbb{E}[L(q(r+1)) - L(q(r))|q(r)].$$

The drift is derived as

$$\begin{aligned} \Delta(q(r)) &= \frac{1}{2}\mathbb{E}\left[q^2(r+1) - q^2(r) \mid q(r)\right] \\ &\leq \frac{1}{2}\mathbb{E}\left[\left(\sum_{k\in\mathcal{K}} E(a_k^{r-1}, \alpha_k^{r-1}) - \varepsilon/R \right.\right. \\ &\quad \left.\left. + \; q(r-1)\right)^2 - q^2(r) \mid q(r)\right] \\ &= \frac{1}{2}(\hat{E}^r - Q)^2 + q(r)\mathbb{E}\left[(\hat{E}^r - \varepsilon/R) \mid q(r)\right] \\ &\leq \zeta + q(r)\mathbb{E}\left[(\hat{E}^r - \varepsilon/R) \mid q(r)\right], \end{aligned} \tag{15}$$

where $\hat{E}^r = \sum_{k\in\mathcal{K}} E(a_k^{r-1}, \alpha_k^{r-1})$ and $\zeta = \frac{1}{2}(\hat{E}_{\max}^r - Q)^2$ means the difference between maximum training loss and target loss. Then the drift-plus-cost expression in each training round

Algorithm 2: Adaptive vehicle selection, bandwidth allocation, and sparsification ratio allocation algorithm to solve $\mathcal{P}_2$.

Input: vehicle location $x_k^{r,0}$, velocity v_k^r, communication rate μ_k^r and computation frequency f_k
Output: $\mathbf{a}^r, \mathbf{b}^r, \{\alpha_k^r\}$

```
for r = 1, 2, ..., R do
    if r = zR_t, ∀z ∈ {1, 2, ..., Z − 1} then
        q(r) ← 0;
        V ← V_z;
    end
    Observe vehicle position, velocity, communication, and computation capacity as x_k^{r,0}, v_k^r, h_k, f_k respectively;
    Decompose P_3 into the primal problem with continuous variables and the master problem with integer variables ;
    Solve the primal problem P_4 based on Alg. 3;
    Check the feasibility of P_3 and solve the master problem P_{5.2} based on Alg. 4;
    Update training loss queue via Eq. (14);
end
```

is expressed as

$$\Delta(q(r)) + V \cdot \mathbb{E}\left[U^r \mid q(r)\right] \leq \zeta + q(r)\mathbb{E}\left[(\hat{E}^r - Q) \mid q(r)\right] + V \cdot \mathbb{E}\left[U^r \mid q(t)\right], \quad (16)$$

which means the normalized cost function of the objective function and long-term training loss. Here, V is denoted as the controlling positive parameter to balance the minimization of training delay and training loss. Based on the above derivation, we transform the $\mathcal{P}_2$ into the following per-round problem as

$$\mathcal{P}_3 : \min_{\mathbf{a}^r, \mathbf{b}^r, \{\alpha_k^r\}, U^r} V \cdot U^r + \sum_{k=1}^{K} q(r) E(a_k^r, \alpha_k^r) \quad (17a)$$

$$\text{s.t. (12b)} - \text{(12f)}, \text{(13c)}. \quad (17b)$$

To obtain the value of V, we need to fetch and update it every R_t rounds as $V_0, V_1, \ldots, V_j, \ldots, V_{J-1}, j = \lfloor \frac{r}{R_t} \rfloor, J = \lfloor \frac{R}{R_t} \rfloor$. At the beginning of each time frame j, the value of V is updated as $V = V_j$, and the virtual energy queue $q(r)$ is reset to zero. When the value of $q(r)$ is increasing, the training loss deficit is a dominant factor, and the value of E is reduced through the design of a virtual convergence queue. The algorithm of transforming $\mathcal{P}_2$ into $\mathcal{P}_3$ is described in Algorithm 2. However, the transformed per-round $\mathcal{P}_3$ is still a difficult MINLP problem.

B. Generalized Benders Decomposition

In this subsection, we apply the Generalized Benders Decomposition (GBD) method to solve $\mathcal{P}_3$. The GBD method decomposes the problem into a primal problem with continuous variables and a master problem with integer variables. We solve these two problems iteratively:

- Solving the primal problem provides the lower bound and Lagrange multipliers for constraints like bandwidth allocation and sparsification ratio.
- Solving the master problem yields the upper bound and binary vehicle selection variables used in the primal problem for the next iteration.

The iterative process converges when the upper bound equals the lower bound, which ensures efficient and effective solution of $\mathcal{P}_3$.

1) Solution to Primal Problem: Given the integer variables as the vehicle selection $a^{r,s}$, where s is the current iteration number, we obtain the primal problem $\mathcal{P}_4$ as

$$\mathcal{P}_4 : \min_{\mathbf{b}^r, \{\alpha_k^r\}, U^r} V \cdot U^r + \sum_{k=1}^{K} q^r E(a_k^{r,s}, \alpha_k^r) \quad (18a)$$

$$\text{s.t. (12d)} - \text{(12f)}, \text{and (13c)}. \quad (18b)$$

The problem $\mathcal{P}_4$ has a convex objective function. However, the constraint (13c) is still non-convex, which is difficult to solve efficiently. To address this issue, we use an alternating optimization approach to solve the proposed problem **P4** by fixing the two variables of $\mathbf{b}^r$, $\{\alpha_k^r\}$, or U^r while updating the remaining variable, respectively.

First, we derive the solution of $\{\alpha^r\}$, when $\mathbf{b}^r$ and U^r is fixed, the problem $\mathcal{P}_4$ can be written as

$$\mathcal{P}_{4.1} : \min_{\{\alpha_k^r\}} V \cdot U^r + \sum_{k=1}^{K} q(r) E(a_k^{r,s}, \alpha_k^r) \quad (19a)$$

$$\text{s.t. (12b), (13c)}. \quad (19b)$$

It is obvious to see $\mathcal{P}_{4.1}$ is a convex optimization problem. To solve the $\mathcal{P}_{4.1}$, we apply the Lagrangian function with given $a^{r,s}$ as

$$\begin{aligned} &\mathcal{L}_1(U^r, \lambda, \mathbf{a}^{r,s}, \alpha_1^r, \ldots, \alpha_K^r, \mathbf{b}^r) \\ &= V \cdot U^r + \sum_{k=1}^{K} q(r) E(a_k^{r,s}, \alpha_k^{r,s}) \\ &\quad + \sum_{k=1}^{K} \lambda_{1,k} [T_{k,c}^r + T_{k,p} - U^r], \end{aligned} \quad (20)$$

Here, $\lambda_{1,k} > 0$ is the Lagrangian multiplier vector. Then we apply the Karush-Kuhn-Tucker (KKT) conditions to obtain the following expressions as

$$\begin{aligned} &\frac{\partial \mathcal{L}_1}{\partial \alpha_k^{r,\star}} = -\frac{q(r)P}{KR} + \lambda_{1,k}^{\star} \frac{a_k^{r,s} M |\mathcal{Z}_k^r|}{b_k^r B_0 \sum_{z \in \mathcal{Z}_k^r} \log_2(p_k^z)}, \forall k \in \mathcal{K}, \\ &T_{k,c}^r + T_{k,p} - U^{r,\star} \leq 0, \forall k \in \mathcal{K}, \\ &\lambda_{1,k}^{\star}(T_{k,c}^r + T_{k,p} - U^{r,\star}) = 0, \forall k \in \mathcal{K}, \end{aligned} \quad (21)$$

where $p_k^z = 1 + \frac{P_k h_k (L_{k,z}^r)^{-\beta}}{N_0}$. Considering the conditions in (21), we derive the following results

$$\frac{\partial \mathcal{L}}{\partial \alpha_k^{r,\star}} = \begin{cases} < 0, & \alpha_k^{r,\star} < \alpha_{\min} \\ = 0, & \alpha_{\min} < \alpha_k^{r,\star} \leq 1, \\ > 0, & \alpha_k^{r,\star} > 1, \end{cases} \quad (22)$$

For all the vehicle $k \in \mathcal{K}$ in round r, we consider two subsets of vehicles as $\mathcal{K}_1$ and $\mathcal{K}_2$. The sum of communication and computation time for some vehicle $k_1 \in \mathcal{K}_1$ is less than $U^{r,\star}$, then we can get the below expression $\forall \alpha_k \in [\alpha_{\min}, 1]$ as

$$T_{k,c}^r + T_{k,p} < U^{r,\star}, \forall k \in \mathcal{K}_1. \quad (23)$$

Meanwhile, some vehicle $k_2 \in \mathcal{K}_2$ may not be selected in round r as

$$a_k^{r,s} = 0, \forall k \in \mathcal{K}_2. \quad (24)$$

According to (21), we get the following inequality as

$$\frac{\partial \mathcal{L}_1}{\partial \alpha_k^{r,\star}} < 0, \forall k \in \mathcal{K}_1, \forall k \in \mathcal{K}_2,$$

which means the derivative of $\mathcal{L}_1$ considering sparsification ratio is negative for these vehicles within $\mathcal{K}_1$ or $\mathcal{K}_2$. The results of $\alpha_k^{r,\star}$ can be derived as below.

- *Case I:* The two subsets $\mathcal{K}_1$ and $\mathcal{K}_2$ are both not empty as

$$0 < |\mathcal{K}_1| < K, 0 < |\mathcal{K}_2| < K.$$

Then based on (21), we can get that

$$\alpha_k^{r,\star} = \begin{cases} \alpha_{\min}, & k \notin \mathcal{K}_1, k \notin \mathcal{K}_2, \alpha_k^\star < \alpha_{\min}, \\ \alpha_k^\star, & k \notin \mathcal{K}_1, k \notin \mathcal{K}_2, \alpha_{\min} \le \alpha_k^\star \le 1, \\ 1, & \text{otherwise}, \end{cases} \quad (25)$$

where

$$\alpha_k^\star = \frac{U^{r,\star} c_k f_k b_k^r B_k^{\mathrm{a}} - |D_k| q(r) b_k^r B_k^{\mathrm{a}}}{a_k^r M |\mathcal{Z}_k^r| c_k f_k},$$

and $B_k^{\mathrm{a}} = B_0 \sum_{z \in \mathcal{Z}_k^r} log_2(p_k^z)$.

- *Case II:* The subset $\mathcal{K}_1$ is empty as $\mathcal{K}_1 = \emptyset$ which indicates the sum of computation time and communication time for all vehicle $k \in \mathcal{K}$ is able to equal to U^r as $\alpha_{\min} \le \alpha_k^\star \le 1$, then based on (21), we can get that

$$\alpha_k^{r,\star} = \begin{cases} \alpha_k^\star, & k \notin \mathcal{K}_2, \\ 1, & \text{otherwise}. \end{cases} \quad (26)$$

- *Case III:* The subset $\mathcal{K}_1$ is equal to $\mathcal{K}$ as $\mathcal{K}_1 = \mathcal{K}$ which indicates the sum of computation time and communication time for all vehicle $k \in \mathcal{K}$ is less or larger than U^r as $\alpha_k^\star < \alpha_{\min}$ or $\alpha_k^\star > 1$. Based on (21), we can get that

$$\alpha_k^{r,\star} = \begin{cases} \alpha_{\min}, & k \notin \mathcal{K}_2, \alpha_k^\star < \alpha_{\min}, \\ 1, & \text{otherwise}. \end{cases} \quad (27)$$

Second, we propose the solution of $\mathbf{b}^r$ with fixed $\{\alpha_k^r\}$ and U^r. The problem $\mathcal{P}_4$ can be transformed into

$$\mathcal{P}_{4.2}: \min_{\mathbf{b}^r} V \cdot U^r + \sum_{k=1}^{K} q(r) E(a_k^{r,s}, \alpha_k^r) \quad (28a)$$

$$\text{s.t. (12c), (12d), (13c)} \quad (28b)$$

$\mathcal{P}_{4.2}$ is a convex optimization problem, and we construct the Lagrangian function with given $a_k^{r,s}$ as

$$\begin{aligned} &\mathcal{L}_2(U^r, \lambda, \mathbf{a}^{r,s}, \alpha_1^r, \ldots, \alpha_K^r, \mathbf{b}^r) \\ &= V \cdot U^r + \sum_{k=1}^{K} q(r) E(a_k^{r,s}, \alpha_k^{r,s}) \\ &+ \lambda_2 \left(B_0 \sum_{k=1}^{K} b_k^r - B \right) + \sum_{k=1}^{K} \lambda_{1,k} [T_{k,c}^r + T_{k,p} - U^r], \end{aligned} \quad (29)$$

where $\lambda_2 > 0$ is the Lagrangian multiplier. The KKT conditions are applied to obtain the equations as

$$\begin{aligned} &\frac{\partial \mathcal{L}_2}{\partial b_k^{r,\star}} = \lambda_2 B_0 - \lambda_{1,k}^\star \frac{\alpha_k^r a_k^{r,s} M |\mathcal{Z}_k^r|}{(b_k^r)^2 B_0 \sum_{z \in \mathcal{Z}_k^r} \log_2(p_k^z)} = 0, \forall k \in \mathcal{K}, \\ &B_0 \sum_{k=1}^{K} b_k^r - B = 0, \\ &T_{k,c}^r + T_{k,p} - U^{r,\star} \le 0, \forall k \in \mathcal{K}, \\ &\lambda_{1,k}^\star (T_{k,c}^r + T_{k,p} - U^{r,\star}) = 0, \forall k \in \mathcal{K}. \end{aligned} \quad (30)$$

According to (21), if (23) is satisfied for some $k \in \mathcal{K}_1$, then the corresponding $\lambda_{1,k}^\star = 0$. For vehicle $k \in \mathcal{K}_2$, the value of $\lambda_{1,k}^\star$ can be derived out in (21) as

$$\lambda_{1,k}^\star = \frac{q(r) P b_k^r B_0 \sum_{z \in \mathcal{Z}_k^r} \log_2(p_k^z)}{K R a_k^{r,s} M |\mathcal{Z}_k^r|}. \quad (31)$$

We then put back the result to (30) and the solution of b_k^r is solved as follows:

- *Case I ($0 < |\mathcal{K}_1| < K$):* According to above formulations, we can get

$$b_k^{r,\star} = \begin{cases} b_k^\star, & k \notin \mathcal{K}_1, k \notin \mathcal{K}_2, b_{\min} \le b_k^\star \le 1, \\ 1, & k \notin \mathcal{K}_1, k \notin \mathcal{K}_2, b_k^\star > 1, \\ b_{\min}, & k \notin \mathcal{K}_1, k \notin \mathcal{K}_2, b_k^\star < b_{\min}, \\ 0, & \text{otherwise} \end{cases} \quad (32)$$

where

$$b_k^\star = \frac{q(r) P \alpha_k^r}{K R \sum_{k \in \mathcal{K} \setminus \{\mathcal{K}_1 \cup \mathcal{K}_2\}} \frac{q_k^r P \alpha_k^r}{K R (1 - b^a)}},$$

and $b^a = \sum_{k \in \mathcal{K}_1} b_{\min}$.

- *Case II ($|\mathcal{K}_1| = 0$):* Based on the previous results, we can get the bandwidth for each vehicle should satisfy $b_{\min} \le b_k \le 1$ as follows:

$$b_k^{r,\star} = \begin{cases} b_k^\star, & k \notin \mathcal{K}_2, b_{\min} \le b_k^\star \le 1, \\ 0, & \text{otherwise}. \end{cases} \quad (33)$$

- *Case III ($|\mathcal{K}_1| = K$):* Based on the previous results, we can get the sum of communication and computation time for each vehicle is larger or less than U^r as $b_k^\star > 1$ or $b_k^\star <$

Algorithm 3: Iteration algorithm of generating and solving master problem $\mathcal{P}_{5.2}$

```
Initialize iteration index ξ = 1, the upper bound UB(0) = +∞, the lower bound LB(0) = 0, minimum tolerance τ, and maximum iteration I_max;
while |UB(ξ − 1) − LB(ξ − 1)| > τ and ξ ≤ I_max do
    Solve the primal problem P_4 with the given vehicle selection variable a^r;
    if P_3 is feasible then
        Obtain {α_k^r}, b^r, L^⋆ with Lagrange multiplier λ;
        Update upper bound: UB(ξ) = min{UB(ξ − 1), L^⋆};
    else
        Solve the l_1-minimization problem P_{5.1} to get {ᾱ_k^r}, b̄^r, λ̄;
    end
    Add optimality and feasibility cuts in the master problem;
    Solve the master problem P_{5.2} to update a^r and get LB(ξ) = ω;
    if |UB(ξ) − LB(ξ)| ≤ τ then
        return Optimal results: {α_k^{r,ξ}}, b^{r,ξ}, a^{r,ξ};
    end
    ξ ← ξ + 1;
end
```

b_{min} as follows:

$$b_k^{r,\star} = \begin{cases} 1, & k \notin \mathcal{K}_2, b_k^\star > 1, \\ b_{\min}, & k \notin \mathcal{K}_2, b_k^\star < b_{min}, \\ 0 & \text{otherwise} \end{cases} \tag{34}$$

Thirdly, we derive the solution of U^r with fixed $\{\alpha_k^r\}, \mathbf{b}^r$, then the problem $\mathcal{P}_4$ can be converted as

$$\mathcal{P}_{4.3}: \min_{U^r} V \cdot U^r + \sum_{k=1}^{K} q(r) E(a_k^{r,s}, \alpha_k^r) \tag{35a}$$

$$\text{s.t. (13c).} \tag{35b}$$

It is easy to see that the problem $\mathcal{P}_{4.3}$ is a linear programming problem, and the solution is obtained in the following theorem.

$$U^{r,\star} = \max_{k \in \mathcal{K}} (T_{k,c}^r + T_{k,p}) \tag{36}$$

with given $\{\alpha_k^{r,\star}\}, \mathbf{b}^{r,\star}$. The primal problem $\mathcal{P}_4$ is solved using an iterative alternating optimization approach. Within each iteration, the algorithm sequentially updates three blocks of variables: first, the sparsification ratios $\{\alpha_k^r\}$ are optimized while keeping other variables fixed; second, the bandwidth allocation $\mathbf{b}^r$ is updated; and third, an auxiliary upper bound variable U^r is adjusted. This iterative process continues until the relative improvement in the objective function falls below a predefined tolerance ϵ_0 or a maximum number of iterations is reached.

2) Solution to Master Problem: In this part, we propose an iterative solution for the master problem with integer variables. Firstly, we check the feasibility of the primal problem; if feasible, we update the lower bound with optimality cuts, and if infeasible, we use l_1 minimization to derive feasibility cuts. The upper bound is obtained from the Lagrange function of $\mathcal{P}_3$. Second, we add the optimality and feasibility cuts to the constraints of the master problem. Then, we solve the master problem to minimize the lower bound given the cuts, using the lower bound as the optimization objective. Next, the iteration continues until the upper and lower bounds converge within a specified range. This process ensures efficient convergence while maintaining the feasibility and optimality of the solution.

- *Feasible primal problem:* In this situation, the optimility cut can be derived as

$$\omega \leq \bar{\mathcal{L}}(\mathbf{b}^r, \{\alpha_k^r\}, U^r, \boldsymbol{\lambda}, \mathbf{a}^{r,s}). \tag{37}$$

  Then the above optimility cut is added to the master problem, and the results of the sparsification ratio and bandwidth allocation are given.
- *Infeasible primal problem:* Considering the infeasible primal problem, we formulate the following optimization problem to derive out the feasibility cut as

$$\mathcal{P}_{5.1}: \min_{\bar{\mathbf{b}}^r, \{\bar{\alpha}_k^r\}, \bar{U}^r} \sum_{k \in \mathcal{K}} (c_{1,k} + c_2) \tag{38a}$$

$$\text{s.t. } T_{k,\mathrm{c}}^r + T_{k,\mathrm{p}} - \bar{U}^r \geq c_{1,k}, \forall k \in \mathcal{K}. \tag{38b}$$

$$B_0 \sum_{k=1}^{K} b_k^r - B = c_2, \tag{38c}$$

which means the l_1 minimization problem for all constraints.

Through solving $\mathcal{P}_{5.1}$, the solution of feasible sparsification ratio, bandwidth allocation, and U^r can be derived, then the feasibility cut is formulated as

$$\begin{aligned} 0 &\leq \bar{\mathcal{L}}\left(\bar{\mathbf{b}}^r, \{\bar{\alpha}_k^r\}, \bar{U}^r, \bar{\boldsymbol{\lambda}}, \mathbf{a}^{r,s}\right) \\ &= \sum_{k=1}^{K} \bar{\lambda}_{1,k} [\bar{T}_{k,c}^r + \bar{T}_{k,p} - \bar{U}^r - c_{1,k}] \\ &\quad + \bar{\lambda}_2 \left(B_0 \sum_{k=1}^{K} \bar{b}_k^r - B - c_2 \right), \end{aligned} \tag{39}$$

where $\bar{\mathbf{b}}^r, \{\bar{\alpha}_k^r\}, \bar{U}^r$ are the solutions of l_1 minimization problem in $\mathcal{P}_{5.1}$, $\mathbf{a}^{r,s}$ is the vehicle selection variable at iteration s and $\sum_{k \in \mathcal{K}} \bar{\lambda}_{1,k} + \bar{\lambda}_2 = 1$ are the Lagrange multipliers.

Through the feasible checking of the primal problem, the optimility cut and feasibility cut can be derived, and the original master problem is formulated as

$$\mathcal{P}_{5.2}: \min_{\mathbf{a}^r} \quad \omega \tag{40a}$$

$$\text{s.t.} \quad \omega \leq \mathcal{L}(\mathbf{b}^{r,x_1}, \{\alpha_k^{r,x_1}\}, U^{r,x_1}, \boldsymbol{\lambda}, \mathbf{a}^r), \quad \forall x_1 \in \{1, 2, \ldots, X_1\}, \tag{40b}$$

$$0 \leq \bar{\mathcal{L}}(\mathbf{b}^{r,x_2}, \{\alpha_k^{r,x_2}\}, U^{r,x_2}, \boldsymbol{\lambda}, \mathbf{a}^r), \quad \forall x_2 \in \{1, 2, \ldots, X_2\}, \tag{40c}$$

$$\text{(12b) and (12c),} \tag{40d}$$

where $X_1 + X_2 = \xi$ means the sum of two variables equals the iteration number. The proposed pure integer programming problem $\mathcal{P}_{5.2}$ can be solved by utilizing branch-and-bound, cutting plane, and so on. The details of the GBD-based vehicle selection algorithm are shown in Algorithm 3.

C. Computational Complexity Analysis

In this subsection, we analyze the computational complexity of the proposed optimization algorithm, which mainly consists of the GBD algorithm for solving the primal problem and the master problem. Considering solving the primal problem, we utilize the alternating optimization algorithm. Since each alternating optimization subproblem is convex, we can directly obtain the closed-form solution. The computational complexity of solving the primal problem is $\mathcal{O}(k_0)$, where k_0 is the number of iterations of the algorithm executed to convergence. In addition, the computational complexity of solving the master problem using the branch and bound method is $\mathcal{O}(2^K)$. Note that the value of K is the maximum number of vehicles in the covering segment. Thus, in the worst case, the overall computational complexity of the algorithm can be roughly estimated as $\mathcal{O}(2^K + k_0) = \mathcal{O}(\max\{2^K, k_0\})$.

TABLE II
SIMULATION PARAMETERS

Parameter	Value
Learning rate, η	0.01
Batch size	32
Local epoch, E	5
BS antenna gain,G_v	6 dBi
Pass loss model	128.1+37.6$\log_{10} d$
Bandwidth, B	10 MHz
Noise power, P_n	-114 dBm
Vehicle velocity, v	{60, 80} km/h
GPU frequency, f_k	1.3 GHz
Minimum sparsification ratio	0.1
Minimum bandwidth allocation	0.05
The controlling parameter, V	4
Length of covering segment	1,000 m
The height of BS	25 m
The number of zones, Z	20
Virtual queue reset rounds, R_t	10
Max alternating update iterations, S_{max}	200
The convergence loss, ϵ	210

VI. SIMULATION RESULTS

A. Simulation Setup

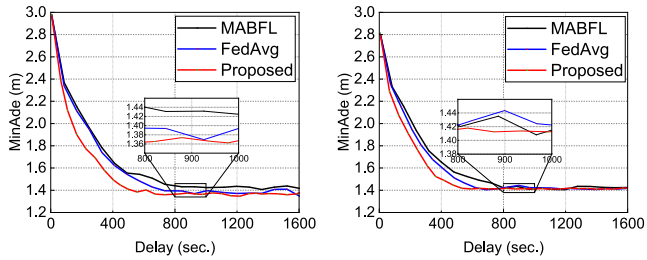
In the simulation, a flow of vehicles is driving through a straight road with a length of 1,000 m in a single, unidirectional lane, which can be easily extended to multiple lanes. The road is divided into 20 zones, and different zones are considered to have varying distances between vehicles and the server. Vehicle trajectories follow the SUMO mobility simulator using the Intelligent Driver Model at 60 km/h or 80 km/h. The learning rate is 0.01, and the batch size is 32. Key parameters are summarized in Table II.

The simulations are conducted with two FL tasks:

- CIFAR-10@ResNet-18 for image recognition: CIFAR-10 is an image dataset that includes color images of 10 classes with 50,000 training and 10,000 test samples. In our simulation, we assign 600 dataset samples to each vehicle. The training model for CIFAR-10 is ResNet-18.
- Argoverse@LaneGCN for trajectory prediction [38]: The Argoverse motion forecasting dataset contains 205,942 training samples and 39,472 validation samples. The LaneGCN model is used for trajectory prediction, and performance is evaluated using the minimum Average Displacement Error (minADE), defined as the average ℓ_2 distance between predicted and ground truth locations.

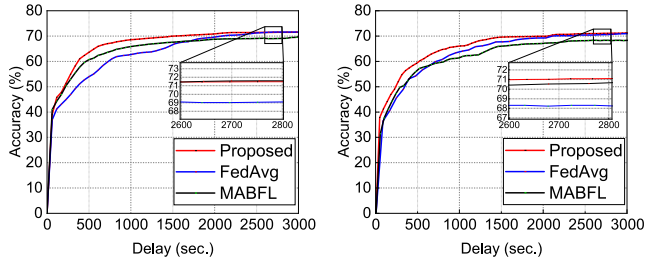
The proposed scheme is compared with the following baselines:

- *Multi-Armed Bandit-based FL (MABFL) [31]:* In this scheme, the sparsification ratio of each participant is chosen from the given sequence of finite partitions of sparsification ratio through MAB.
- *FedAvg:* The models are trained and uploaded without local sparsification, and vehicles are selected with the proposed vehicle selection scheme.
- *Heuristic:* In this scheme, the vehicles are selected within better communication conditions with sparse training.
- *Random:* The vehicles are selected randomly with sparse training in this scheme.



(a) Argoverse with 60 km/h (b) Argoverse with 80 km/h

Fig. 3. Training performance of different training algorithms on Argoverse@LaneGCN.



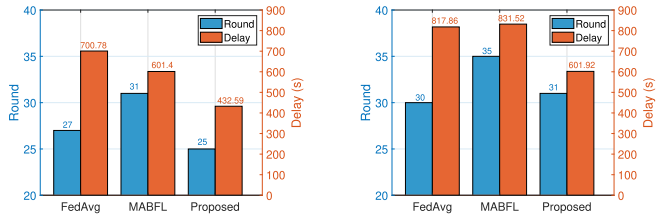
(a) CIFAR-10 with 60 km/h (b) CIFAR-10 with 80 km/h

Fig. 4. Training performance of different training algorithms on CIFAR-10@ResNet-18.

B. Performance of Proposed MAVFL Scheme

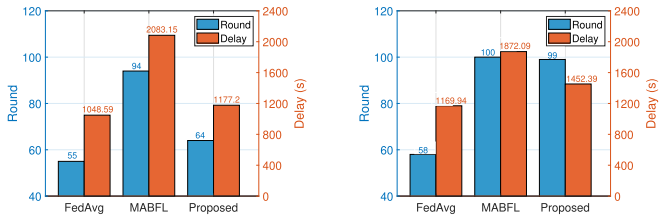
1) Accuracy Performance: Figs. 3 and 4 illustrate the accuracy and minADE versus time curve of our scheme for Argoverse and CIFAR-10 datasets compared to the benchmark schemes. From this figure, it is observed that the MABFL and the proposed scheme with sparsification converge faster than FedAvg. The reason is that the sparsification of the model effectively reduces the communication time. Meanwhile, the proposed scheme achieves nearly the same accuracy as FedAvg, while MABFL achieves lower accuracy than FedAvg. For example, considering the CIFAR-10 dataset, the final accuracy of the proposed scheme is larger than FedAvg by 0.5%, and larger than MABFL by nearly 3 percent. The reason is that our scheme incorporates mechanisms to adaptively adjust the sparsification ratio and communication strategies based on instant vehicle and network conditions, such as vehicle location, velocity, and channel quality.

2) Delay Performance: The performance of training delay and round achieving target metric is further evaluated, including given accuracy and minADE. Figs. 5 and 6 illustrate the number of rounds and training latency of different schemes when reaching the target metric. For the CIFAR-10 dataset, the target accuracy is set as 69%, and the target minADE is 1.44 m for the Argoverse dataset. The training loss is derived from the loss curve upon achieving the specified metrics. First, the results demonstrate that the number of rounds and latency required by our proposed scheme are the minimum in both datasets. Especially for the Argoverse dataset, our proposed scheme reduces up to 27.9% delay and up to 6 rounds compared to MABFL and reduces 38.2% delay and two rounds compared



(a) Argoverse with 60 km/h (b) Argoverse with 80 km/h

Fig. 5. Convergence time and training rounds on Argoverse@LaneGCN.



(a) CIFAR-10 with 60 km/h (b) CIFAR-10 with 80 km/h

Fig. 6. Convergence time and training rounds on CIFAR-10@ResNet-18.

to FedAvg. For the CIFAR-10 dataset, our proposed scheme reduces up to 43.5% delay and up to 30 rounds compared to MABFL and reduces up to 9 rounds compared to FedAvg. The delay of our proposed scheme is larger than FedAvg for the target accuracy of CIFAR-10. The reason is the loss of accuracy due to sparse training. The performance results show the robustness of our proposed scheme in vehicular networks.

3) Impact of Vehicle Velocity: The impact of vehicle velocity on the performance of the proposed schemes is further investigated. As illustrated in Figs. 3 and 4, the simulation results demonstrate the effects of different velocities, specifically 60 km/h and 80 km/h, on system behavior. Firstly, it is evident that as the velocity increases, the accuracy improvement for each scheme diminishes substantially. This trend indicates that higher vehicle speeds introduce additional challenges to maintaining consistent performance. Secondly, for vehicles traveling at 80 km/h, both the number of rounds required to achieve target accuracy and the associated delay are more pronounced compared to those at 60 km/h. This phenomenon can be attributed to the fact that at higher velocities, fewer vehicles remain within the covering segment for an extended period. Consequently, the amount of data available for simultaneous training decreases, leading to reduced efficiency in accuracy performance.

4) Impact of the Bandwidth: The influence of bandwidth is further evaluated. Fig. 8(a) shows the latency of two tasks during convergence. First, it can be observed from the graph that the latency for different tasks decreases as bandwidth increases. This behavior is expected because a higher bandwidth allows for faster data transmission between vehicles and the central server, reducing the time required for model updates and synchronization. Additionally, it can be seen that within the same task, as the velocity of the vehicle increases, the latency also increases due to a decrease in the number of vehicles on the same road segment. This occurs because higher velocities lead to shorter dwell times for vehicles within a communication range, resulting in fewer opportunities for collaborative training and increased reliance on individual contributions, which are generally less efficient. Furthermore, the reduction in the number of vehicles participating at higher speeds exacerbates the issue of limited data availability. With fewer vehicles contributing to the training process, the system must wait longer to aggregate sufficient data for meaningful updates, thereby increasing overall latency.

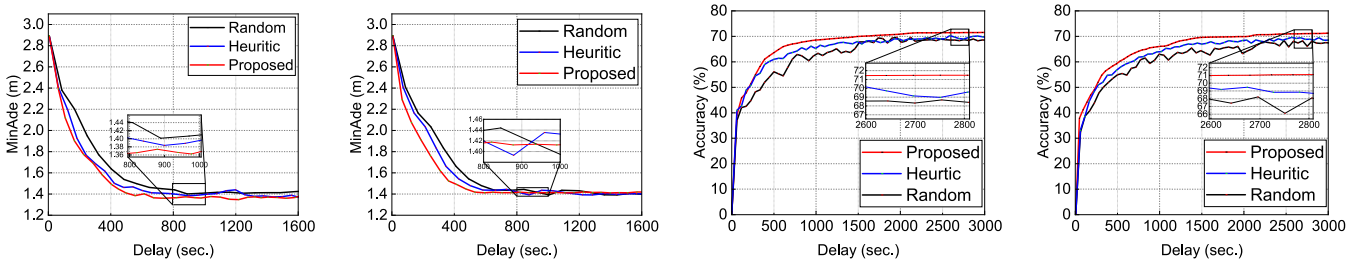
C. Performance of Proposed Optimization Algorithm

In Fig. 7, the training metrics for different vehicle selection algorithms are compared with varying velocities, such as 60 km/h and 80 km/h, with Argoverse and CIFAR-10 datasets. From Fig. 7(a) and (b), we first observe that the convergence speed of the proposed vehicle selection algorithm is faster, and the overall time of the proposed scheme can be substantially reduced compared to heuristic and random algorithms. This is because the proposed optimization algorithm considers the influence of bandwidth allocation and sparsification ratio. Secondly, the proposed scheme loses an average of 0.03 m in minADE on the first dataset compared to the baselines for Argoverse, while the final accuracy improves by 2 to 5 percentage points for CIFAR-10. This scenario is because we consider the impact of long-term optimization aims of training loss on vehicle selection.

In Fig. 8(b), the convergence performance of the GBD algorithm is demonstrated through the evolution of the lower bound and upper bound. Following the algorithmic framework, the upper bound is initialized to positive infinity and the lower bound to zero in the first iteration. These initial values reflect the absence of any constraints or information in the first iteration. Subsequent iterations begin with the initialization of integer variables as vehicle selection choices, triggering progressive updates to both bounds through the solution of the primal and master problems. This dual process ensures that the bounds are tightened iteratively, bringing them closer together. The convergence is achieved within 18 iterations, where the upper bound and lower bound effectively coincide, signifying that the algorithm has reached an optimal or near-optimal solution of vehicular selection choice.

We compare the solution of our proposed algorithm with the exhaustive search one, which is obtained through continuous sub-problems with the fixed vehicle selection variable and exhaustively traversing all possible integer combinations of selection. As shown in Fig. 8(c), our algorithm converges very quickly, and the solution is very close to the exhaustive search one. Our algorithm converges rapidly within 5 iterations to a stable value. Quantitatively, the final converged value of our proposed algorithm is approximately 31.0, while the optimal value found via exhaustive search is approximately 29.2, indicating an optimality gap of about 6.1%. This result demonstrates that the performance of the proposed algorithm is comparable to the optimality.

Table III shows the results of convergence performance considering the delay, convergence metrics, and communication overhead for all the baselines with two datasets. First, our proposed scheme achieves higher convergence accuracy than MABFL, heuristic, and random algorithms for the CIFAR-10 dataset. The accuracy of the proposed scheme is slightly lower

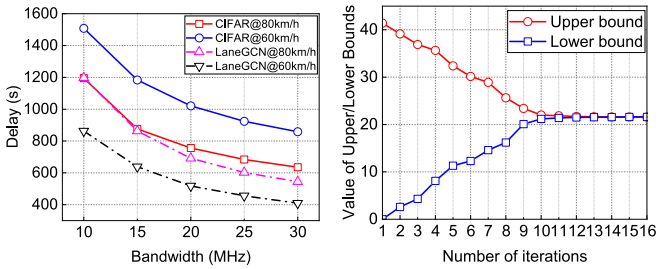


(a) Argoverse with 60 km/h (b) Argoverse with 80 km/h (c) CIFAR-10 with 60 km/h (d) CIFAR-10 with 80 km/h

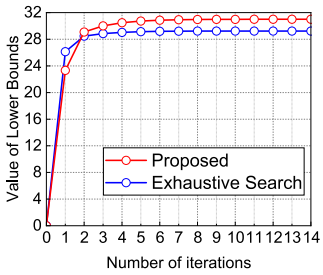
Fig. 7. Training performance of different vehicle selection algorithms for Argoverse and CIFAR-10.

TABLE III
CONVERGENCE PERFORMANCE COMPARISON ON CIFAR-10 AND ARGOVERSE DATASETS

Method	CIFAR-10@ResNet-18				Argoverse@LaneGCN			
	Training Delay (sec.) 60 km/h	Training Delay (sec.) 80 km/h	Convergence Accuracy (%)	Communication Overhead (MB)	Training Delay (sec.) 60 km/h	Training Delay (sec.) 80 km/h	Convergence minADE (m)	Communication Overhead (MB)
Proposed	**2092.2**	**2285.16**	**71.4**	**4445.84**	**674.11**	**721.32**	**1.43**	**367.57**
FedAvg	2372.87 (1.13×)	2770.68 (1.21×)	71.9	6377.80	733.65 (1.08×)	899.54 (1.24×)	1.41	645.30
MABFL	1660.82 (0.79×)	2573.71 (1.13×)	68.1	3880.20	844.85 (1.25×)	805.04 (1.12×)	1.43	361.37
Heuristic	2421.6 (1.16×)	2620.91 (1.15×)	69.4	4763.40	780.5 (1.16×)	959.40 (1.33×)	1.42	408.41
Random	2202.3 (1.05×)	2805.90 (1.23×)	68.2	5049.21	970.73 (1.44×)	1087.81 (1.51×)	1.42	490.10



(a) Impact of bandwidth allocation (b) Convergence of the GBD algorithm



(c) The comparison of the proposed and exhaustive lower bound

Fig. 8. Training performance of the proposed scheme.

than that of FedAvg, but it achieves a 28.8% reduction in communication overhead. Second, the convergence time of the MABFL scheme is faster, but the convergence accuracy is lower than that of the proposed scheme. This is because the model is sparsified with higher rates in MABFL, which cannot meet long-term loss constraints. Third, in both datasets, our scheme shows lower communication overhead with an average of 26% than FedAvg, heuristic, and random algorithms.

VII. CONCLUSION

In this paper, we have proposed a mobility-aware VFL scheme that accelerates training through adaptive vehicle selection and sparse training. By adapting vehicle-side local training duration within its network connection window, the MAVFL mitigates the straggler effect and ensures efficient FL in dynamic vehicular networks. A Lyapunov-driven and GBD-based optimization framework is presented to solve the training delay minimization problem. The proposed optimization method can adapt to vehicular networks with high mobility, while reducing communication overhead and preserving training accuracy. For future work, we will investigate a multimodal VFL scheme that effectively utilizes multimodal sensing data to enhance training accuracy in vehicular networks.

REFERENCES

[1] G. Karagiannis et al., "Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions," *IEEE Commun. Surveys Tuts.*, vol. 13, no. 4, pp. 584–616, 4th Quart. 2011.
[2] X. Shen, J. Gao, W. Wu, M. Li, C. Zhou, and W. Zhuang, "Holistic network virtualization and pervasive network intelligence for 6G," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 1, pp. 1–30, firstquarter 2022.
[3] K. Qu, W. Zhuang, Q. Ye, W. Wu, and X. Shen, "Model-Assisted learning for adaptive cooperative perception of connected autonomous vehicles," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 8820–8835, Aug. 2024.
[4] L. Sequeira and T. Mahmoodi, "The role of machine learning for trajectory prediction in cooperative driving," in *Proc. ACM MobiHoc*, Oct. 2020, pp. 368–373.
[5] R. Zhu et al., "Boosting collaborative vehicular perception on the edge with vehicle-to-vehicle communication," in *Proc. ACM SenSys*, 2024, pp. 141–154.
[6] M. B. Mashhadi, M. Jankowski, T.-Y. Tung, S. Kobus, and D. Gündüz, "Federated mmWave beam selection utilizing LiDAR data," *IEEE Wireless Commun. Lett.*, vol. 10, no. 10, pp. 2269–2273, Oct. 2021.
[7] X. Ye, K. Qu, W. Zhuang, and X. Shen, "Accuracy-Aware cooperative sensing and computing for connected autonomous vehicles," *IEEE Trans. Mobile Comput.*, vol. 23, no. 8, pp. 8193–8207, Aug. 2024.

[8] D. Dooley, B. McGinley, C. Hughes, L. Kilmartin, E. Jones, and M. Glavin, "A blind-zone detection method using a rear-mounted fisheye camera with combination of vehicle detection methods," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 264–278, Jan. 2016.
[9] S. Wang et al., "End-to-end target liveness detection via mmwave radar and vision fusion for autonomous vehicles," *ACM Trans. Sen. Netw.*, vol. 20, no. 4, pp. 1–26, May 2024.
[10] W. Wu et al., "Dynamic RAN slicing for service-oriented vehicular networks via constrained learning," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 7, pp. 2076–2089, Jul. 2021.
[11] J. Jiang, K. Han, Y. Du, G. Zhu, Z. Wang, and S. Cui, "Optimized power control for over-the-air federated averaging with data privacy guarantee," *IEEE Trans. Veh. Technol.*, vol. 72, no. 2, pp. 2728–2733, Feb. 2023.
[12] M. F. Pervej, R. Jin, and H. Dai, "Resource constrained vehicular edge federated learning with highly mobile connected vehicles," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 6, pp. 1825–1844, Jun. 2023.
[13] W. Wu et al., "Split learning over wireless networks: Parallel design and resource management," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 4, pp. 1051–1066, Apr. 2023.
[14] Q. Kong et al., "Privacy-Preserving aggregation for federated learning-based navigation in vehicular fog," *IEEE Trans. Ind. Inf.*, vol. 17, no. 12, pp. 8453–8463, Dec. 2021.
[15] B. Li, Y. Jiang, Q. Pei, T. Li, L. Liu, and R. Lu, "FEEL: Federated end-to-end learning with non-IID data for vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 16728–16740, Sep. 2022.
[16] F. Liang, Q. Yang, R. Liu, J. Wang, K. Sato, and J. Guo, "Semi-Synchronous federated learning protocol with dynamic aggregation in Internet of Vehicles," *IEEE Trans. Veh. Technol.*, vol. 71, no. 5, pp. 4677–4691, May 2022.
[17] T. Zeng, O. Semiari, M. Chen, W. Saad, and M. Bennis, "Federated learning on the road autonomous controller design for connected and autonomous vehicles," *IEEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 10407–10423, Dec. 2022.
[18] X. Chen, G. Xu, X. Xu, H. Jiang, Z. Tian, and T. Ma, "Multicenter hierarchical federated learning with fault-tolerance mechanisms for resilient edge computing networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 1, pp. 47–61, Jan. 2025.
[19] W. Zhu et al., "Randomized DP-DFl: Towards differentially private decentralized federated learning via randomized model interaction," *IEEE Trans. Mobile Comput.*, vol. 24, no. 12, pp. 12802–12817, Dec. 2025.
[20] B. Xie et al., "MOB-FL: Mobility-aware federated learning for intelligent connected vehicles," in *Proc. IEEE Int. Conf. Commun.*, 2023, pp. 3951–3957.
[21] L. Li, Y. Li, and R. Hou, "A novel mobile edge computing-based architecture for future cellular vehicular networks," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2017, pp. 1–6.
[22] Y. Xing, C. Lv, and D. Cao, "Personalized vehicle trajectory prediction based on joint time-series modeling for connected vehicles," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 1341–1352, Feb. 2020.
[23] B. Salehi, J. Gu, D. Roy, and K. Chowdhury, "FLASH: Federated learning for automated selection of high-band mmWave sectors," in *Proc. IEEE Conf. Comput. Commun.*, 2022, pp. 1719–1728.
[24] Y. Li, S. He, Y. Li, Y. Shi, and Z. Zeng, "Federated multiagent deep reinforcement learning approach via physics-informed reward for multimicrogrid energy management," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 5, pp. 5902–5914, May 2024.
[25] J. Xu and H. Wang, "Client selection and bandwidth allocation in wireless federated learning networks: A long-term perspective," *IEEE Trans. Wireless Commun.*, vol. 20, pp. 1188–1200, Feb. 2021.
[26] Y. Fraboni, R. Vidal, L. Kameni, and M. Lorenzi, "Clustered sampling: Low-variance and improved representativity for clients selection in federated learning," 2021, *arXiv:2105.05883*.
[27] C. Zhang, Y. Xie, T. Chen, W. Mao, and B. Yu, "Prototype similarity distillation for communication-efficient federated unsupervised representation learning," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 11, pp. 6865–6876, Nov. 2024.
[28] H. Xiao, J. Zhao, Q. Pei, J. Feng, L. Liu, and W. Shi, "Vehicle selection and resource optimization for federated learning in vehicular edge computing," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 11073–11087, Aug. 2022.
[29] Z. Zhai, Q. Wu, S. Yu, R. Li, F. Zhang, and X. Chen, "FedLEO: An offloading-assisted decentralized federated learning framework for low earth orbit satellite networks," *IEEE Trans. Mobile Comput.*, vol. 23, no. 5, pp. 5260–5279, May 2024.
[30] C. Feng, H. H. Yang, D. Hu, Z. Zhao, T. Q. S. Quek, and G. Min, "Mobility-aware cluster federated learning in hierarchical wireless networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 10, pp. 8441–8458, Oct. 2022.
[31] Z. Jiang et al., "Computation and communication efficient federated learning with adaptive model pruning," *IEEE Trans. Mobile Comput.*, vol. 23, no. 3, pp. 2003–2021, Mar. 2024.
[32] L. Cui, X. Su, Y. Zhou, and J. Liu, "Optimal rate adaption in federated learning with compressed communications," in *Proc. IEEE Conf. Comput. Commun.*, 2022, pp. 1459–1468.
[33] R. Chen, L. Li, K. Xue, C. Zhang, M. Pan, and Y. Fang, "Energy efficient federated learning over heterogeneous mobile devices via joint design of weight quantization and wireless transmission," *IEEE Trans. Mobile Comput.*, vol. 22, no. 12, pp. 7451–7465, Dec. 2023.
[34] Y. Jiang et al., "Model pruning enables efficient federated learning on edge devices," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 12, pp. 10374–10386, Dec. 2023.
[35] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAvg on non-IID data," 2020, *arXiv:1907.02189*.
[36] K. Liu, C. Liu, G. Yan, V. C. S. Lee, and J. Cao, "Accelerating DNN inference with reliability guarantee in vehicular edge computing," *IEEE/ACM Trans. Netw.*, vol. 31, no. 6, pp. 3238–3253, Dec. 2023.
[37] A. Li et al., "LotteryFL: Empower edge intelligence with personalized and communication-efficient federated learning," in *Proc. EEE/ACM Symp. Edge Comput.*, 2021, pp. 68–79.
[38] M. Liang et al., "Learning lane graph representations for motion forecasting," in *Proc. 16th Eur. Conf. Comput. Vis.*, 2020, pp. 541–556.

Haoyu Tu (Graduate Student Member, IEEE) received the BS degree in information engineering from Xidian University, Xi'an, China, in 2019, and the MS degree in communication and information engineering from the Shanghai Institute of Microsystem and Information Technology, Shanghaitech University, Shanghai, China, in 2022. He is currently a joint Ph.D. student in information and communication engineering with the School of Computer Science, Sun Yat-sen University, Guangzhou, China, and Pengcheng Laboratory, Shenzhen, China. His research interests include federated learning and vehicular networks.

Wen Wu (Senior Member, IEEE) received the PhD degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2019. He is currently an associate researcher with Frontier Research Center, Pengcheng Laboratory, Shenzhen, China. He has authored or coauthored more than 130 refereed IEEE journal and conference papers, and six books/book chapters, filed ten+ PCT patents, and issued two U.S. patents. His research interests include 6G networks, edge LLM, split learning, network intelligence, and network virtualization. He was the recipient of the OJVT Best Paper Award from the IEEE Vehicular Technology Society in 2025, and Award for Excellence (Early Career Researcher) from the IEEE Hyper-Intelligence Technical Committee in 2022. He also was the recipient of the World Top 2% Scientist Award in 2023-2025, USENIX Security Distinguished Paper Award in 2024, ComSoc TEA Excellence Camp Runner-Up Award in 2023, IEEE ICCT Best Student Paper Award in 2025, and IEEE CIC/ICCC Best Paper Award in 2022. He serves as the track co-chair of IEEE VTC 2022-2025, and workshop co-chair of IEEE INFOCOM 2022-2025, IEEE Globecom 2024, IEEE MASS 2025, and IEEE ICCC 2023-2025. He serves as the editor or guest editor of *IEEE Transactions on Mobile Computing*, *IEEE Networking Letters*, Hindawi WCMC, *China Communications*, *Electronics*, and Springer PPNA.

Lin Chen (Member, IEEE) received the BSc degree in radio engineering from Southeast University, Nanjing, China, in 2002, the MSc degree in networking from the University of Paris 6, in 2005, the Engineer and Ph.D. degrees in computer science and networking from Telecom ParisTech, in 2005 and 2008, respectively. From 2009 to 2019, he was an associate professor with the University of Paris-Sud. From 2019 to 2025, he was a professor with Sun Yat-sen University. Since 2025, he has been a professor with Macao Polytechnic University. His research interests include centered around algorithm design and analysis in networked systems.

Liang Li (Member, IEEE) received the PhD degree from the School of Telecommunications Engineering, Xidian University, China, in 2021. From 2018 to 2020, she was a visiting PhD student with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX, USA. From 2021 to 2023, she was a postdoctoral faculty member with the School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing, China. She is currently an assistant researcher with the Department of Strategic and Advanced Interdisciplinary Research, Pengcheng Laboratory, Shenzhen, China. Her research interests include edge intelligence, distributed learning, data-driven robust optimization, and differential privacy.

Xu Chen (Senior Member, IEEE) received the PhD degree in information engineering from the Chinese University of Hong Kong in 2012. From 2012 to 2014, he was a postdoctoral research associate with Arizona State University, Tempe, AZ, USA, and Humboldt scholar fellow with the Institute of Computer Science, University of Goettingen, Germany, from 2014 to 2016. He is currently a full professor with Sun Yat-sen University, Guangzhou, China, director of Institute of Advanced Networking and Computing Systems, and vice director of the National Engineering Research Laboratory of Digital Homes. He was the recipient of the prestigious Humboldt research fellowship awarded by Alexander von Humboldt Foundation, 2014 Hong Kong Young Scientist Runner-up Award, 2022 IEEE Communications Society Asia Pacific Outstanding Paper Award, 2022 IEEE Internet of Things Journal Best Paper Runner-Up Award, 2022 IEEE Computer Society Best Paper Awards Runner-up, 2017 IEEE Communication Society Asia-Pacific Outstanding Young Researcher Award, 2017 IEEE ComSoc Young Professional Best Paper Award, Honorable Mention Award of 2010 IEEE international conference on Intelligence and Security Informatics (ISI), Best Paper Runner-up Award of 2014 IEEE International Conference on Computer Communications (INFOCOM), and Best Paper Award of 2017 IEEE International Conference on Communications (ICC). He is the area editor of *IEEE Open Journal of the Communications Society*, associate editor for *IEEE Transactions on Mobile Computing*, *IEEE Transactions on Wireless Communications*, *IEEE Transactions on Vehicular Technology*, and *IEEE Journal on Selected Areas in Communications* Series on Network Softwarization and Enablers.

Xuemin (Sherman) Shen (Fellow, IEEE) received the PhD degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is currently a University professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research interests include network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular networks. He is a registered professional engineer of Ontario, Canada, Engineering Institute of Canada fellow, Canadian Academy of Engineering fellow, Royal Society of Canada fellow, Chinese Academy of Engineering foreign member, and Engineering Academy of Japan International fellow. He was the recipient of the "West Lake Friendship Award" from Zhejiang Province in 2023, Canadian Award for Telecommunications Research from the Canadian Society of Information Theory (CSIT) in 2021, R.A. Fessenden Award in 2019 from IEEE, Canada, Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society (ComSoc), and Technical Recognition Award from Wireless Communications Technical Committee (2019) and AHSN Technical Committee (2013). He was the recipient of the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada. He serves/served as the general chair of 6G Global Conference'23, and ACM Mobihoc'15, Technical Program Committee chair/co-chair of IEEE Globecom'24, '16 and '07, IEEE Infocom'14, and IEEE VTC'10 Fall. He is the past president of IEEE ComSoc, vice president of Technical & Educational Activities, vice president of Publications, Member-at-Large on the Board of Governors, chair of the Distinguished Lecturer Selection Committee, and member of IEEE Fellow Selection Committee of the ComSoc. He served as the editor-in-chief of *IEEE IoT Journal*, *IEEE Network*, and *IET Communications*.