Federated Learning over MU-MIMO Vehicular Networks

Raftopoulou, Maria; da Silva, José Mairton B.; Litjens, Remco; Poor, H. Vincent; Van Mieghem, Piet

DOI
10.3390/e27090941

Publication date
2025

Document Version
Final published version

Published in
Entropy

entropy

MDPI

Article

Federated Learning over MU-MIMO Vehicular Networks

Maria Raftopoulou [1,2], José Mairton B. da Silva, Jr. [3], Remco Litjens [1,2,*], H. Vincent Poor [4] and Piet Van Mieghem [1]

[1] Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, 2628 CD Delft, The Netherlands; maria.raftopoulou@tno.nl (M.R.); p.f.a.vanmieghem@tudelft.nl (P.V.M.)
[2] Department of Networks, Netherlands Organisation for Applied Scientific Research (TNO), 2595 DA The Hague, The Netherlands
[3] Department of Information Technology, Uppsala University, 751 05 Uppsala, Sweden; mairton.barros@it.uu.se
[4] Department of Electrical and Computer Engineering, Princeton University, Princeton, NJ 08544, USA; poor@princeton.edu
* Correspondence: r.litjens@tudelft.nl

Abstract

Many algorithms related to vehicular applications, such as enhanced perception of the environment, benefit from frequent updates and the use of data from multiple vehicles. Federated learning is a promising method to improve the accuracy of algorithms in the context of vehicular networks. However, limited communication bandwidth, varying wireless channel quality, and potential latency requirements may impact the number of vehicles selected for training per communication round and their assigned radio resources. In this work, we characterize the vehicles participating in federated learning based on their importance to the learning process and their use of wireless resources. We then address the joint vehicle selection and resource allocation problem, considering multi-cell networks with multi-user multiple-input multiple-output (MU-MIMO)-capable base stations and vehicles. We propose a "vehicle-beam-iterative" algorithm to approximate the solution to the resulting optimization problem. We then evaluate its performance through extensive simulations, using realistic road and mobility models, for the task of object classification of European traffic signs. Our results indicate that MU-MIMO improves the convergence time of the global model. Moreover, the application-specific accuracy targets are reached faster in scenarios where the vehicles have the same training data set sizes than in scenarios where the data set sizes differ.

Keywords: vehicle selection; resource allocation; MU-MIMO; federated learning; wireless networks; vehicular networks

Academic Editors: Shenghao Yang and Jun Chen

Received: 17 July 2025
Revised: 24 August 2025
Accepted: 28 August 2025
Published: 9 September 2025

Citation: Raftopoulou, M.; da Silva, J.M.B., Jr.; Litjens, R.; Poor, H.V.; Van Mieghem, P. Federated Learning over MU-MIMO Vehicular Networks. *Entropy* **2025**, *27*, 941. https://doi.org/10.3390/e27090941

1. Introduction

In recent years, many advances have been made in the field of autonomous vehicles. Autonomous vehicles rely on the information they receive from sensors, as well as from other vehicles and the network, through vehicle-to-vehicle and vehicle-to-infrastructure wireless links, respectively, to make real-time decisions, such as route planning, speed adjustment, and collision avoidance [1]. To ensure driving safety, driving decisions should be accurate, and communication via vehicle-to-vehicle and vehicle-to-infrastructure links should be fast and reliable. To address these challenges, machine learning (ML) algorithms are widely applied. Examples of such ML applications include the optimal assignment of radio resources and optimal handover control [1].

When driving, the environment is very dynamic and can change drastically over time. In addition, driving must be adjusted according to the location/area and the enforced driving rules. Hence, the applied ML algorithms should be constantly updated based on new sensor data and can greatly benefit from using data from other vehicles. For example, vehicles can share their camera data to enhance environmental perception, thus allowing vehicles to observe obstacles or dangerous situations that are out of the reach of their own cameras but still in close proximity [1].

Due to the distributed nature of the data, federated learning (FL) is a promising method for collaborative learning in vehicular networks [2]. Specifically, FL allows the training of a centralized global model using decentralized data samples from distinct vehicles without the need to upload the individual samples to a centralized server. A range of applications has been discussed in the literature that can benefit from FL in vehicular networks, with one example being co-operative environmental perception [3,4]. In particular, the FL global model can provide higher accuracy in detecting and localizing obstacles or hazardous situations compared to a local model trained on a single vehicle. This is because knowledge from multiple vehicles driving in the same area contributes to the global model and hence provides information that other vehicles do not have. Another example is the improvement of navigation systems, because vehicles from different areas, which experience different driving conditions, contribute to the global model [3]. More examples include traffic prediction for resource management [3] and steering wheel angle precision [5].

Some major challenges with FL in wireless networks are the high communication cost for exchanging the model parameters between the FL server and the vehicles, the wireless channel quality variations, and the limited transmission resources. To overcome these challenges, a subset of vehicles is selected to take part in each training round of the learning process. In [6], we addressed the agent (and in this context, vehicle) selection problem for FL in resource-constrained wireless networks by providing an agent selection framework based on distinct agent characteristics while also considering an application-specific latency budget. In this work, we extend the previously proposed framework to address the challenge of agent selection and resource management in the context of vehicular wireless networks. Given this context, we consider road and mobility models and a multi-cell network with multi-user multiple input multiple output (MU-MIMO)-capable base stations.

The integrated agent selection and resource management problem for wireless networks has been addressed in the literature, but mainly for stationary agents and in contexts other than vehicular networks [7]. For example, Chen et al. [8] address the minimization of training loss while considering parameters related to the wireless channels, whereas Zeng et al. [9] focus on minimizing energy consumption. Shi et al. [10] focus on latency-constrained networks and propose a policy to optimize the global model accuracy under a given latency constraint. Fan et al. [11] claim to have published the first work that addresses the minimization of the time duration of each communication round while also considering a practical mobility model. However, their evaluation does not consider a learning task related to vehicular networks or MU-MIMO-capable base stations. Moreover, the mobility model used is simpler and less realistic than the 3GPP-based model used in this study. One of the few works that addresses FL in vehicular networks is by Deveaux et al. [12]. Specifically, they highlight the need for algorithms addressing unevenly distributed data and propose a high-level protocol that allows the network to retrieve information about the type of data within each vehicle. The prior art on applying FL for beam assignment in MU-MIMO systems is very limited. A work addressing MU-MIMO systems is by Guan et al. [13], who propose an access scheduling algorithm. Unlike this work, they consider full-duplex transmissions and focus on Internet of Things applications.

The main contributions of this study are as follows:

- We evaluate the performance of FL over vehicular scenarios, which are realistically modeled using the road and mobility models from 3GPP [14]. These models are more complex and realistic than mobility models typically used in the literature. Additionally, we consider MU-MIMO-capable base stations, which are not frequently considered in FL-related studies. Moreover, we consider the learning task of object classification on the European traffic sign data set, which is a relevant data set for vehicular applications. This data set is statistically and geographically more diverse, and therefore more challenging to train on, than commonly used data sets such as MNIST and CIFAR-10.
- Based on the defined MU-MIMO vehicular scenario, we investigate the challenge of vehicle selection and resource management by characterizing vehicles based on their importance in the learning process and their wireless channel quality. We then propose the "vehicle-beam-iterative" (VBI) algorithm to approximate the solution of the defined optimization problem. The evaluation of the VBI algorithm provides insights into the novel and realistic scenario under investigation.
- We show that MU-MIMO-capable base stations improve the convergence time of the global model by enabling the selection of multiple vehicles on the same time–frequency resources and improving the achievable vehicle data rates.
- We show that the local loss is an effective vehicle selection metric for scenarios with non-independent and identically distributed (IID) data, assuming that all vehicles have the same training times. When vehicles have different training times, e.g., due to different data set sizes and/or processing capabilities, the loss-based policies do not provide substantial gains.
- We demonstrate, through realistic numerical evaluations, that convergence time in scenarios where vehicles have different data set sizes is longer than in scenarios where vehicles have the same data set sizes.

The outline of the paper is as follows. Section 2 describes the network, learning, and communication models. Section 3 then derives the joint vehicle selection and resource allocation optimization problem, and Section 4 describes the VBI algorithm. Section 5 presents the configuration of the considered evaluation scenarios, and Section 6 provides the numerical evaluation of the VBI algorithm. Finally, conclusions and proposals for future works are given in Section 7.

2. System Model

This section describes the considered network and learning models, as well as the communication model between vehicles and base stations. For clarity, a short description of the most commonly used symbols in this paper is given in Table 1.

Table 1. List of most commonly used symbols.

Symbol	Description
Γ_{vb}	Estimated uplink SNR at vehicle v from beam b in [dB]
ρ	Constant tuning the relative significance of the learning importance and the resource consumption
$\tau_{\mathrm{APP,MAX}}$	Application-specific latency budget in [s]
τ_{DL}	Broadcast time of the global model in [s]
$\tau_{\mathrm{T+UL}}$	Time for all selected vehicles to train and upload their local models in [s]
$\tau_{\mathrm{L},b}$	Start time of uplink transmission to beam b in [s]
$\tau_{\mathrm{T},v}$	Training time of vehicle v in [s]

Table 1. *Cont.*

Symbol	Description
$\tau_{\mathrm{UL},vb}$	Upload time of vehicle v on beam b in [s]
$\boldsymbol{\tau_B} \in \mathbb{R}^{B_{\mathrm{TOT}}}$	Upload latency budget at each beam in [s]
$\boldsymbol{\tau_L} \in \mathbb{R}^{B_{\mathrm{TOT}}}$	Start time of uplink transmissions on each beam in [s]
$\boldsymbol{\tau_T} \in \mathbb{R}^{V}$	Training times of vehicles in [s]
$\mathbf{A} \in \mathbb{R}^{V \times B_{\mathrm{TOT}}}$	Optimization matrix with beam associations between the vehicles and the base station beams
$\mathbf{Q} \in \mathbb{R}^{V \times B_{\mathrm{TOT}}}$	Importance of vehicles at each beam
$\mathbf{T_{UL}} \in \mathbb{R}^{V \times B_{\mathrm{TOT}}}$	Upload times of vehicles at each beam in [s]
$\mathbf{W}_G$	The weights of the global model
$\mathbf{W}_v$	The weights of the local model at vehicle v
$\mathbf{s} \in \{0,1\}^V$	Optimization vector for vehicle selection
$B_M = \lvert\mathcal{B}_m\rvert$	Number of beams at each base station m
$B_{\mathrm{TOT}} = \lvert\mathcal{B}_{\mathrm{TOT}}\rvert$	Total number of base station beams in the network
$C_{\mathrm{R,MAX}}$	Available transmission resources
$C_{R,vb}$	Consumption of transmission resources of vehicle v on beam b
$F(\cdot)$	The loss function of the model
$K = \lvert\mathcal{K}\rvert$	The total number of samples
$K_v = \lvert\mathcal{K}_v\rvert$	Number of training samples at vehicle v
$K_{T,v} = \lvert\mathcal{K}_{T,v}\rvert$	Number of testing samples at vehicle v
$M = \lvert\mathcal{M}\rvert$	Number of base stations in the network
$P_{\mathrm{NF},M}$	Noise figure at the base stations in [dB]
$P_{\mathrm{V,MAX}}$	Maximum transmit power of vehicles in [dBm]
Q_{TOT}	Total vehicle importance
R_{vb}	Bit rate at vehicle v on beam b in [Mbps]
$V = \lvert\mathcal{V}\rvert$	Number of vehicles in the network
$V_G = \lvert\mathcal{V}_G\rvert$	Number of selected vehicles for training
Z	Size of the FL model in [Mbits]
f_{BW}	System bandwidth in [MHz]
q_{vb}	Importance of vehicle v on beam b

2.1. Network Model

Consider a cellular network with one FL server, a set $\mathcal{M}$ of base stations and a set $\mathcal{V}$ of vehicles, where $M = |\mathcal{M}|$ and $V = |\mathcal{V}|$ are the number of base stations and vehicles in the network, respectively. The vehicles and the FL server collaboratively train a global model, without requiring the transmission of the data sets gathered by the vehicles, while the base stations facilitate communication between the vehicles and the FL server. For this, we assume that the FL server is connected to all base stations with fiber, hence their backhaul communication latency is negligibly small and that communication between the FL server and the base stations is synchronized.

Figure 1 shows a schematic overview of a communication round i, assuming a simple network with $V = 2$ vehicles and $M = 1$ base station. First, the FL server selects and notifies, via the base station, the vehicles that will participate in the learning. The vehicle selection and notification, potentially via broadcast transmission, are performed within a time interval of duration τ_{SCH}. Each selected vehicle $v \in \mathcal{V}_G[i]$ then trains its local model, where $\mathcal{V}_G[i]$ is the set of vehicles selected in communication round i. A vehicle $v \in \mathcal{V}_G[i]$ has a training time $\tau_{\mathrm{T},v}$, which can be different from other vehicles, as shown in Figure 1, due to different data set sizes and/or processing capabilities.

Once each selected vehicle $v \in \mathcal{V}_G[i]$ finishes its local training, it transmits its local model using the assigned uplink transmission resources to its serving base station, which then forwards the model to the FL server. The uplink transmission time $\tau_{\mathrm{UL},v}$ for vehicle v depends on its wireless channel quality, discussed in Section 2.3. In Figure 1, the transmissions of the two vehicles are assumed to be time-multiplexed, hence vehicle 2 does not

initiate its uplink transmission until the completion of the uplink transmission of vehicle 1. Once all local models are aggregated at the FL server, the global model is updated for the next communication round $i+1$. The time duration of this process is τ_{AGG}. Finally, the FL server broadcast in the downlink, with duration τ_{DL}, the new global model to each vehicle $v \in \mathcal{V}$. The process repeats until sufficient accuracy is achieved for the global model, verified using an FL server-specific testing data set.

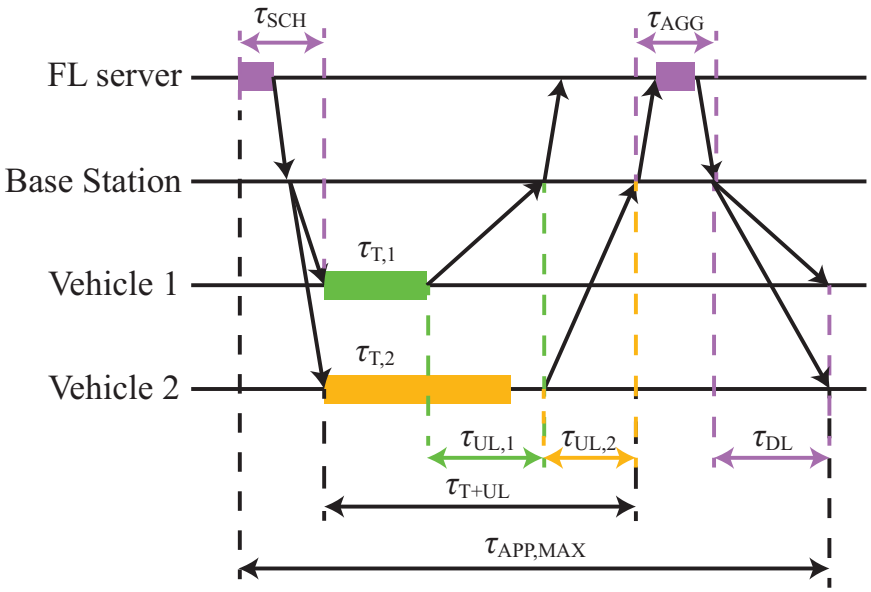

Figure 1. Schematic overview of the different steps involved in a single communication round and the corresponding time intervals.

An application-specific deadline budget $\tau_{\mathrm{APP,MAX}}$ can be set on the time duration of each communication round i to prevent the selection of vehicles with limited processing power and/or poor wireless channel quality, thus

$$\tau_{\mathrm{SCH}} + \tau_{\mathrm{T+UL}} + \tau_{\mathrm{AGG}} + \tau_{\mathrm{DL}} \leq \tau_{\mathrm{APP,MAX}}, \tag{1}$$

where $\tau_{\mathrm{T+UL}}$ is the time needed for all selected vehicles to perform local training and upload their local models to the FL server, as shown in Figure 1. We assume that the processing times at the FL server are negligible because it is likely to have significantly more powerful hardware compared to the vehicles. Moreover, we assume that the time duration of the broadcast to notify the selected vehicles for training is negligible because the control data transmitted is very small in size. Therefore, $\tau_{\mathrm{SCH}} \approx \tau_{\mathrm{AGG}} \approx 0$. Additionally, we assume that the network is configured to ensure a minimum bit rate at the cell edge and thus, the broadcast time τ_{DL} is fixed for every communication round i. Therefore, using (1), we perform vehicle selection and resource allocation over a time period fulfilling the inequality:

$$\tau_{\mathrm{T+UL}} \leq \tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}. \tag{2}$$

Moreover, the FL process can be bound to the available transmission resources $C_{\mathrm{R,MAX}}$ allocated to the FL task, e.g., in a slice in 5G networks, which can restrict the number of selected agents per communication round.

In practical wireless communication scenarios, multiple scheduling time slots exist within the time interval $\tau_{\mathrm{T+UL}}$, occurring on a millisecond scale. These scheduling decisions depend on the experienced signal-to-noise ratio (SNR) of the vehicles, which varies within milliseconds. In this work, we focus on the resource allocation problem from a higher time-scale perspective. We perform periodic resource allocation over a period $\tau_{\mathrm{T+UL}}$ and

we assume that the effects occurring on the millisecond scale, e.g., multipath fading, can be averaged.

2.2. Learning Model

In the considered learning model, vehicle $v \in \mathcal{V}$ has training data set $\mathcal{K}_v$ and testing data set $\mathcal{K}_{T,v}$, with $K_v = |\mathcal{K}_v|$ and $K_{T,v} = |\mathcal{K}_{T,v}|$ denoting the respective number of samples in each data set. Vehicle v's input training data samples are given by $\mathbf{X}_v = [\mathbf{x}_{v1}, \cdots, \mathbf{x}_{vK_v}]$, with $\mathbf{x}_{vk} \in \mathbb{R}^{n_X}$ as the k^{th} input vector to its model and n_X as the length of the input vector. Additionally, the output data samples are given by $\mathbf{Y}_v = [\mathbf{y}_{v1}, \cdots, \mathbf{y}_{vK_v}]$, with $\mathbf{y}_{vk} \in \{0,1\}^{n_C}$ as the real output vector corresponding to the k^{th} input vector $\mathbf{x}_{vk}$, and n_C as the number of model outputs.

In local training conducted by vehicle v, predictions (model output) $\hat{\mathbf{Y}}_v = [\hat{\mathbf{y}}_{v1}, \cdots, \hat{\mathbf{y}}_{vK_v}]$ are generated using the vehicle's available training data samples in $\mathcal{K}_v$, with $\hat{\mathbf{y}}_{vk} \in \mathbb{R}^{n_C}$ denoting the predicted output vector associated with input vector $\mathbf{x}_{vk}$. The obtained local model is characterized by the derived parameter weights $\mathbf{W}_v$, which are used such that, given the input data $\mathbf{X}_v$, the predictions $\hat{\mathbf{Y}}_v$ represent the real output $\mathbf{Y}_v$. The closeness between the predictions $\hat{\mathbf{Y}}_v$ and the real output $\mathbf{Y}_v$ is generally expressed by the loss function $F(\mathbf{W}_v; \mathbf{X}_v, \mathbf{Y}_v)$, which depends on the input $\mathbf{X}_v$ and the real output $\mathbf{Y}_v$. For notational simplicity, we will omit this dependency in the remainder of the text and simply denote the loss function by $F(\mathbf{W}_v)$.

The objective of training the local model at vehicle v is

$$\min_{\mathbf{W}_v} F(\mathbf{W}_v) = \frac{1}{K_v} \sum_{k \in \mathcal{K}_v} f_k(\mathbf{W}_v), \tag{3}$$

where $f_k(\mathbf{W}_v)$ denotes the loss function of sample k, which for image classification problems is commonly defined as the cross-entropy loss [15]. To obtain the weights $\mathbf{W}_v$ that minimize the loss function $F(\mathbf{W}_v)$, a number of iterations (local epochs) n_{LE} are performed. Assuming the stochastic gradient descent (SGD) optimizer [16], the weights $\mathbf{W}_v$ are adapted at every local epoch based on the applied learning rate η.

In FL, the training data set $\mathcal{K} = \cup_{v \in \mathcal{V}} \mathcal{K}_v$ (with $K = |\mathcal{K}|$) is the union of the vehicle-specific training data sets and the global objective function $F(\mathbf{W}_G[i])$ at communication round i is approximated by the weighted average of losses for the vehicle-specific local models:

$$F(\mathbf{W}_G[i]) \approx \sum_{v \in \mathcal{V}_G[i]} \frac{K_v}{K} F(\mathbf{W}_v[i]). \tag{4}$$

Given the SGD optimizer, the `FedAvg` method [17] determines the global model weights $\mathbf{W}_G[i]$ at the end of communication round i as the weighted average of the local model weights:

$$\mathbf{W}_G[i] \leftarrow \sum_{v \in \mathcal{V}_G[i]} \frac{K_v}{K} \mathbf{W}_v[i], \tag{5}$$

The updated global weights are then broadcast to all the vehicles for the next communication round.

2.3. Communication Model

We assume that both base stations and vehicles are equipped with beamforming antenna arrays to form narrow and strong beams. Specifically, beam pairs are formed between the base stations and vehicles, and the same beam pair is used for both uplink and downlink transmissions [18]. Moreover, we assume that the base station and vehicle beams are directly pointing at each other and interference between different transmissions

is neglected. For the base station antenna array, we assume use of a grid-of-beams mode, i.e., a base station $m \in \mathcal{M}$ can form a pre-defined set of beams $\mathcal{B}_m$ in the three-dimensional space. We further assume that all base stations have the same set of beams and thus each base station m has $B_M = |\mathcal{B}_m|$ beams. Regarding the antenna array of the vehicles, we assume a single beam that can be steered in any direction. A detailed description of the antenna array models is provided in Appendix A.1.

We assume an OFDMA-based (orthogonal frequency division multiple access) access technology and MU-MIMO-capable base stations, thus, spatial-multiplexing, i.e., multiple vehicles can transmit at a serving cell on the same time-frequency resources. Moreover, we consider time-multiplexing, thus, beams from different vehicles can be paired to the same base station beam. Additionally, wideband transmissions are assumed. Therefore, during communication round i, a vehicle $v \in \mathcal{V}_G[i]$ is assigned to beam $b \in \mathcal{B}_{\text{TOT}}$ (from a base station $m \in \mathcal{M}$) for a fraction of the time period $\tau_{\text{T+UL}}$, where $\mathcal{B}_{\text{TOT}} = \cup_{m\in\mathcal{M}}\mathcal{B}_m$ is the set with all base station beams and $B_{\text{TOT}} = |\mathcal{B}_{\text{TOT}}| = B_M M$ is the total number of beams in the network. Finally, we assume that during the period $\tau_{\text{T+UL}}$, vehicles stay connected to the same beam. This assumption is further discussed in Section 5.4.

For the uplink transmission of the local model, and as an input to the periodic resource assignment, we estimate the bit rate R_{vb} of vehicle v from beam $b \in \mathcal{B}_{\text{TOT}}$ as

$$R_{vb} = f_{\text{BW}} \min\left(\log_2\left(1 + 10^{\Gamma_{vb}/10}\right), 15\right), \tag{6}$$

where 15 bits/Hz/s is the target peak spectral efficiency in the uplink channel in 5G [19], f_{BW} denotes the system bandwidth in MHz, and Γ_{vb} is the estimated uplink SNR at vehicle v from beam b given, in dB, by

$$\Gamma_{vb} = P_{\text{V,MAX}} + G_{T,vb} + G_{V,vb} + G_{M,vb} - P_{\text{NOISE}} - P_{\text{NF},M}, \tag{7}$$

where $P_{\text{V,MAX}}$ is the maximum transmit power of the vehicle in dBm, $G_{T,vb}$ is the transmission gain between vehicle v and the base station that beam b belongs to in dB, $G_{V,vb}$ and $G_{M,vb}$ are the vehicle and base station antenna gains in dBi, respectively, P_{NOISE} is the thermal noise power in dBm, and $P_{\text{NF},M}$ is the noise figure in dB at each base station. The channel gain $G_{T,vb}$ is modeled by [20]

$$G_{T,vb} = 20\log\left(\frac{c}{4\pi f_C}\right) - 10\gamma\log(d_{vb}) + \psi, \tag{8}$$

(in dB) with c the speed of light (in m/s), f_C the carrier frequency (in Hz), d_{vb} the 3D distance between vehicle v and its serving base station (in m), γ the path loss exponent, and ψ as a zero-mean Gaussian random variable with standard deviation σ, included to model shadow fading. Due to the periodic nature of the resource assignment approach and the assumption that vehicles stay connected to the same beam during the period $\tau_{\text{T+UL}}$, the SNR Γ_{vb} and bit rate R_{vb} are assumed to be constant during the period $\tau_{\text{T+UL}}$.

Finally, for the broadcast transmission, we assume that all base stations use all their beams because there are many vehicles in the network, spread in different directions. We derive the broadcast bit rate based on the network layout and the antenna array configuration, in Section 5.4.

3. Problem Formulation

In real-world applications, vehicles are diverse in terms of their training data, processing capabilities to train their local model, and wireless channel quality. In this section, we define the so-called *vehicle importance* metric to characterize the vehicles. We present the latency considerations, which depend on the processing capabilities and the wireless

channel quality of the vehicles. Furthermore, we combine the vehicle importance with the latency considerations to formulate the joint vehicle selection and resource allocation optimization problem.

For the joint vehicle selection and resource allocation problem, we consider two optimization parameters; one for the vehicle selection and one for the resource allocation. Specifically, we define the optimization vector $\mathbf{s}[i] \in \{0,1\}^V$, during communication round i, in which $s_v[i] = 1$ when vehicle v is selected for training, and $s_v[i] = 0$ otherwise. Additionally, we define the optimization matrix $\mathbf{A}[i] \in \{0,1\}^{V \times B_{\text{TOT}}}$, in which $A_{vb}[i] \in \{0,1\}$, holds the beam associations between the selected vehicles and the base station beams. Because all selected vehicles remain connected to a single beam during the time interval $\tau_{\text{T+UL}}$, the following equality must hold

$$\mathbf{A}[i] \cdot \mathbb{1}_{B_{\text{TOT}} \times 1} = \mathbf{s}[i], \tag{9}$$

where $\mathbb{1}_{B_{\text{TOT}} \times 1}$ denotes the all ones $B_{\text{TOT}} \times 1$ vector.

3.1. Vehicle Importance

To capture the diversity of vehicles, we introduce the *vehicle importance* $q_{vb}[i] \in \mathbb{R}$, which is the metric governing the vehicle selection and resource allocation process at communication round i. Specifically, the vehicle importance $q_{vb}[i]$ captures the trade-off between the importance of vehicle v in the learning process against its consumed transmission resources on the, potentially assigned, beam b. The definition of this metric is essential to the vehicle selection and resource allocation problem, as both learning and wireless aspects play a significant role in the accuracy and convergence time of the global model. For example, if only learning aspects are considered, vehicles with poor channels may be selected for training, which will lead to long upload times and eventually a long global model convergence time. However, if a vehicle with a poor channel holds a data set belonging to a class with few samples across the system, i.e., a non-IID data scenario with varying sample counts across agents, it is important to include that vehicle in the FL training despite its wireless conditions. The vehicle importance $q_{vb}[i]$ metric also allows for configuring the relative significance of the learning and wireless aspects, as shown below.

We express the importance of vehicle v in the learning process [6] by the locally computed *loss* function $F(\mathbf{W}_{G,v}[i])$, determined based on the testing data set $\mathcal{K}_{T,v}$ and global weights $\mathbf{W}_G[i]$, as generated and broadcast at the end of communication round i. The computed loss $F(\mathbf{W}_{G,v}[i])$ is conveyed to the central FL server and used in the vehicle selection and resource allocation process for upcoming communication round $i+1$. We disregard the corresponding transmission time, considering that the loss is just a scalar value.

The time-frequency resources $C_{R,vb}[i]$ consumed by vehicle v for the upload of the locally derived model weights $\mathbf{W}_v[i]$ on candidate beam b in communication round i. We then consider $C_{R,vb}[i]$ as:

$$C_{R,vb}[i] = \tau_{\text{UL},vb}[i] f_{\text{BW}} = \frac{Z}{R_{vb}[i]} f_{\text{BW}}, \tag{10}$$

where $\tau_{\text{UL},vb}[i]$ is the transmission time for vehicle v on beam b in seconds, $R_{vb}[i]$ is the bit rate given by (6), and Z is the model size in Mbits. Calculation of the resource consumption $C_{R,bv}[i]$ does not require additional vehicle-to-base station communication since the bit rates can be estimated based on the periodic channel quality indicator (CQI) feedback that all vehicles report to their serving base station.

We define the importance $q_{vb}[i]$ of vehicle v on beam b at communication round i as

$$q_{vb}[i] = \frac{F(\mathbf{W}_{G,v}[i])^{\rho}}{C_{R,vb}^{1-\rho}[i]}, \tag{11}$$

where $\rho \in [0,1]$ is a constant that can be configured to set the relative significance of the learning importance and the resource consumption. We further define the matrix $\mathbf{Q}[i] = [\mathbf{q}_1[i], \cdots, \mathbf{q}_{B_{\text{TOT}}}[i]] \in \mathbb{R}^{V \times B_{\text{TOT}}}$, where $\mathbf{q}_b[i]$ is a column vector holding the importance $q_{vb}[i]$ of each vehicle v on beam $b \in \mathcal{B}_{\text{TOT}}$.

3.2. Latency Considerations

We assume that a fixed amount of uplink transmission resources $C_{\text{R,MAX}}$ are allocated to the FL task, which are expressed as a product of the bandwidth and the maximum allowed aggregate upload time. Therefore, the selected vehicles should perform their uplink transmission within the available transmission resources $C_{\text{R,MAX}}$, where the uplink transmission resources $C_{R,vb}[i]$ of vehicle v on beam b at communication round i are given by (10). Because we assume wideband transmissions, for simplicity, we will express from here onward the transmission resources $C_{\text{R,MAX}}$ and $C_{R,vb}[i]$ only in terms of time. Additionally, the selected vehicles should train and transmit their local models within the latency budget $\tau_{\text{APP,MAX}} - \tau_{\text{DL}}$, as previously captured in (2). In this section, we first derive the training time $\tau_{\text{T},v}$ of vehicle v, and then elaborate on the two latency constraints.

The training time $\tau_{\text{T},v}$ of vehicle v depends on the vehicle's processing capability g_v, as well as on its data set size K_v, and other training-related parameters, e.g., number of local epochs n_{LE}. The processing capability g_v of vehicle v is measured in floating point operations (FLOPs) per second as [21]

$$g_v = n_{\text{CORES},v}\, \nu_v\, \omega_v, \tag{12}$$

where $n_{\text{CORES},v}$ is the number of central processing unit (CPU) cores at vehicle v, ν_v is the CPU clock frequency at vehicle v in cycles per second and ω_v is the number of FLOPs per cycle at vehicle v. Then, the training time $\tau_{\text{T},v}$ of vehicle v is

$$\tau_{\text{T},v} = \left\lceil \frac{K_v}{s_B} \right\rceil \frac{n_{\text{FLOP},G}\; n_{\text{LE}}}{g_v}, \tag{13}$$

where $n_{\text{FLOP},G}$ denotes the number of FLOPs to train the model for a batch of size s_B and $\lceil \cdot \rceil$ represents the ceiling operation.

The first latency constraint relates to the available transmission resources $C_{\text{R,MAX}} = \tau_{\text{T+UL}} = \tau_{\text{APP,MAX}} - \tau_{\text{DL}}$ that are available per communication round i. Considering spatial- and time-multiplexing, multiple vehicles can be scheduled on a given base station beam and share the frequency resources over time. To perform this co-scheduling of vehicles on the same beam, the training time $\tau_{\text{T},v}$, as defined in (13), of each vehicle should be taken into consideration. That is because each base station beam becomes active, i.e., receives uplink data, for the first time when the vehicle with the shortest training time that is assigned to that beam finishes its local training. We define the vector $\boldsymbol{\tau}_{\mathbf{L}}[i] = [\tau_{\text{L},1[i]}, \cdots, \tau_{\text{L},B_{\text{TOT}}}[i]]^{\intercal} \in \mathbb{R}^{B_{\text{TOT}}}$ to indicate the start time of the uplink transmissions per beam at communication round i, where $^{\intercal}$ denotes the transpose operation, and

$$\tau_{\text{L},b}[i] = \min_{1 \le vs. \le V} \hat{\boldsymbol{\tau}}_{\mathbf{L},b}[i], \tag{14}$$

where $\hat{\boldsymbol{\tau}}_{\mathbf{L},b}[i] \in \mathbb{R}^V$ indicates the training times of the vehicles assigned to beam b. The vector $\hat{\boldsymbol{\tau}}_{\mathbf{L},b}[i]$ is defined from the auxiliary matrix $\hat{\mathbf{T}}_{\mathbf{L}} = [\hat{\boldsymbol{\tau}}_{\mathbf{L},1}, \cdots, \hat{\boldsymbol{\tau}}_{\mathbf{L},B_{\mathrm{TOT}}}] \in \mathbb{R}^{V \times B_{\mathrm{TOT}}}$ which associates the training time $\tau_{\mathrm{T},v}$ of each vehicle v to its assigned beam as follows:

$$\hat{\mathbf{T}}_{\mathbf{L}}[i] = (\boldsymbol{\tau}_{\boldsymbol{T}} \cdot \mathbb{1}_{1 \times B_{\mathrm{TOT}}}) \circ \mathbf{A}[i], \tag{15}$$

where $\boldsymbol{\tau}_{\boldsymbol{T}} = [\tau_{\mathrm{T},1}, \cdots, \tau_{\mathrm{T},V}]^{\intercal} \in \mathbb{R}^V$ and $\circ$ is the Hadamard product. Given that uplink transmissions are starting at a different time per beam, vehicles can be co-scheduled on the same beam if they can jointly finish their uplink transmissions within the remaining transmission resources. Hence, the condition for co-scheduling is

$$\mathbb{1}_{1 \times V} \cdot (\mathbf{T}_{\mathbf{UL}}[i] \circ \mathbf{A}[i]) \preccurlyeq (\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}})\, \mathbb{1}_{1 \times B_{\mathrm{TOT}}} - \boldsymbol{\tau}_{\boldsymbol{L}}{}^{\intercal}(\mathbf{A}[i]), \tag{16}$$

where $\mathbf{T}_{\mathbf{UL}}[i] = [\boldsymbol{\tau}_{\mathbf{UL},1}[i], \cdots, \boldsymbol{\tau}_{\mathbf{UL},B_{\mathrm{TOT}}}[i]] \in \mathbb{R}^{V \times B_{\mathrm{TOT}}}$ with $\boldsymbol{\tau}_{\mathbf{UL},b}[i] \in \mathbb{R}^V$ holding the upload time duration $\tau_{\mathrm{UL},vb}$ of each vehicle v on beam b, $\preccurlyeq$ denotes the element-wise inequality and $\boldsymbol{\tau}_{\boldsymbol{L}}(\mathbf{A}[i])$ denotes the dependence on $\mathbf{A}[i]$.

Figure 2a,b show two examples of assigning two vehicles to the same beam. In both examples, the start time of the uplink transmissions is equal to $\tau_{\mathrm{L},b} = \min\{\tau_{\mathrm{T},1}, \tau_{\mathrm{T},2}\} = \tau_{\mathrm{T},1}$. Thus, the joint upload time interval is $\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}} - \tau_{\mathrm{T},1}$. Figure 2a illustrates that the two vehicles cannot be co-scheduled on the same beam because $\tau_{\mathrm{UL},1} + \tau_{\mathrm{UL},2} > \tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}} - \tau_{\mathrm{T},1}$, which violates the constraint in (16). Conversely, Figure 2b shows that co-scheduling is possible and that the constraint in (16) is fulfilled.

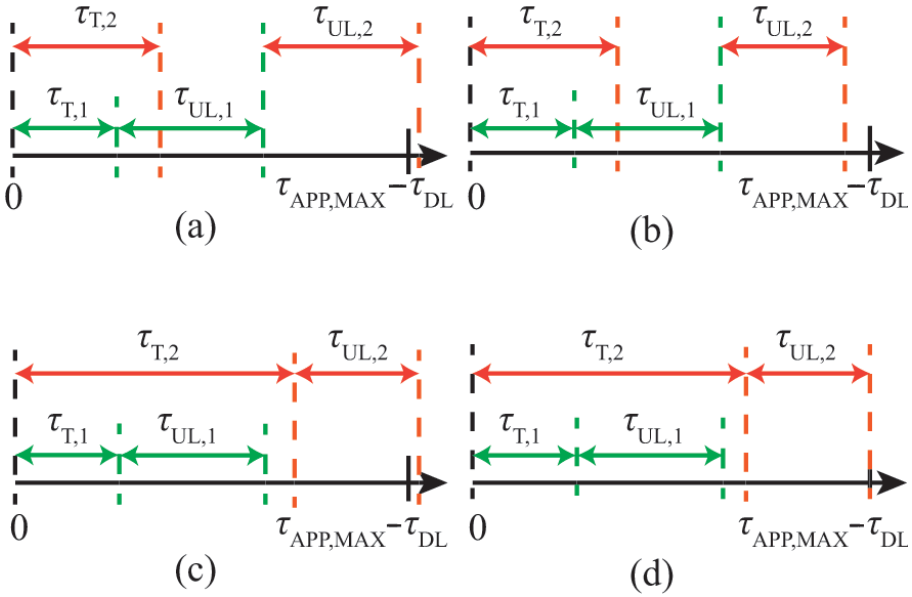

Figure 2. (**a**) The two vehicles cannot be co-scheduled on the same beam because the constraint in (16) is violated; (**b**) The two vehicles can be co-scheduled on the same beam because the constraint in (16) is fulfilled; (**c**) The two vehicles cannot be co-scheduled on the same beam because the constraint in (17) is violated; (**d**) The two vehicles can be co-scheduled on the same beam because the constraint in (17) is fulfilled. In all four sub-figures, the color refers to a different vehicles.

The fulfillment of (16) alone does not guarantee that vehicles can train and transmit within the time interval $\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$, which was captured in (2). Therefore, a second latency constraint is needed, as follows:

$$\boldsymbol{\tau}_{\boldsymbol{T}} + (\mathbf{T}_{\mathbf{UL}}[i] \circ \mathbf{A}[i]) \cdot \mathbb{1}_{B_{\mathrm{TOT}} \times 1} \preccurlyeq (\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}) \mathbb{1}_{B_{\mathrm{TOT}} \times 1}. \tag{17}$$

Consider again the example of assigning two vehicles to one beam and that $\tau_{\mathrm{L},b} = \min\{\tau_{\mathrm{T},1}, \tau_{\mathrm{T},2}\} = \tau_{\mathrm{T},1}$, as shown in Figure 2c,d. Even though (16) is fulfilled, the example in Figure 2c illustrates that vehicle 2 should not be scheduled on that particular beam, because its training and upload time is $\tau_{\mathrm{T},2} + \tau_{\mathrm{UL},2}$, which exceeds the time interval $\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$ and violates the constraint in (17). On the contrary, the constraint in (17) is fulfilled in the example in Figure 2d.

3.3. Problem Formulation

For a given communication round i, we formulate the following joint vehicle selection and resource allocation optimization problem to maximize the total vehicle importance:

$$\max_{\mathbf{s}[i],\mathbf{A}[i]} \quad \mathrm{Tr}(\mathbf{Q}[i] \cdot \mathbf{A}^{\mathsf{T}}[i]) \tag{18}$$

$$\text{subject to} \quad \mathbf{A}[i] \cdot \mathbb{1}_{B_{\mathrm{TOT}} \times 1} = \mathbf{s}[i], \tag{19}$$

$$\mathbb{1}_{1 \times V} \cdot (\mathbf{T_{UL}}[i] \circ \mathbf{A}[i]) \preccurlyeq (\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}) \mathbb{1}_{1 \times B_{\mathrm{TOT}}} - \boldsymbol{\tau_L}^{\mathsf{T}}(\mathbf{A}[i]), \tag{20}$$

$$\boldsymbol{\tau_T} + (\mathbf{T_{UL}}[i] \circ \mathbf{A}[i]) \cdot \mathbb{1}_{B \times 1} \preccurlyeq (\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}) \mathbb{1}_{B_{\mathrm{TOT}} \times 1}, \tag{21}$$

$$\mathbf{s}[i] \in \{0,1\}^{V \times 1}, \tag{22}$$

$$\mathbf{A}[i] \in \{0,1\}^{V \times B_{\mathrm{TOT}}}, \tag{23}$$

where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix, i.e., the sum of the elements on the diagonal. The binary optimization variable $\mathbf{s}[i]$ indicates whether a vehicle is selected and the binary optimization matrix $\mathbf{A}[i]$ indicates the beam on which the selected vehicles are assigned to. Therefore, the total vehicle importance is defined as the summation of distinct vehicle importance values from a set of vehicles, that would potentially jointly be selected for training. Thus, the optimization problem aims to select the set of vehicles that provides the maximum total vehicle importance over other vehicle sets, subject to constraints (19)–(23). With objective function (18), the given optimization problem takes both learning and wireless aspects into account, and its solution directly affects the accuracy and convergence time of the global model. Constraint (19) indicates that vehicles participating in the learning process must be associated with at most one beam in one cell. Constraint (20) shows that vehicles can be assigned to, and time-multiplexed on, the same beam only if both can finish their uplink transmissions within the related time interval. Finally, constraint (21) indicates that all selected vehicles need to train and transmit their local models within the time interval $\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$.

In the optimization problem in (18)–(23), the selection of vehicles $\mathbf{s}[i]$ is defined based on the beam associations $\mathbf{A}[i]$ using (19). Therefore, we can reduce the optimization parameters by combining (19) and (22), leading to:

$$\max_{\mathbf{A}[i]} \quad \mathrm{Tr}(\mathbf{Q}[i] \cdot \mathbf{A}^{\mathsf{T}}[i]) \tag{24}$$

$$\text{subject to} \quad \mathbf{A}[i] \cdot \mathbb{1}_{B_{\mathrm{TOT}} \times 1} \preccurlyeq \mathbb{1}_{V \times 1}, \tag{25}$$

$$\mathbb{1}_{1 \times V} \cdot (\mathbf{T_{UL}}[i] \circ \mathbf{A}[i]) \preccurlyeq (\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}) \mathbb{1}_{1 \times B_{\mathrm{TOT}}} - \boldsymbol{\tau_L}^{\mathsf{T}}(\mathbf{A}[i]), \tag{26}$$

$$\boldsymbol{\tau_T} + (\mathbf{T_{UL}}[i] \circ \mathbf{A}[i]) \cdot \mathbb{1}_{B_{\mathrm{TOT}} \times 1} \preccurlyeq (\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}) \mathbb{1}_{V \times 1}, \tag{27}$$

$$\mathbf{A}[i] \in \{0,1\}^{V \times B_{\mathrm{TOT}}}. \tag{28}$$

The optimization problem in (24)–(28) is non-linear because the vector $\boldsymbol{\tau_L}^{\mathsf{T}}(\mathbf{A}[i])$ requires the evaluation of a $\min(\cdot)$ function, as shown in (14). Typically, non-linear integer optimization problems are more difficult to solve than linear integer problems, even if the solution space is relatively small. That is because they are often non-convex and reaching a global optimum cannot be guaranteed. To eliminate the nonlinearity, we assume the same

training time τ_{T} for all vehicles and define a relaxed version of the optimization problem in (24)–(28). Such a relaxation is achieved by assuming that all vehicles have the same data set size K_v and processing capabilities g_v. Additionally, constraint (27) is not needed in the relaxed problem because it is implied by constraint (26). The relaxed optimization problem is then given by

$$\max_{\mathbf{A}[i]} \quad \mathrm{Tr}(\mathbf{Q}[i] \cdot \mathbf{A}^{\mathsf{T}}[i]) \tag{29}$$

$$\text{subject to} \quad \mathbf{A}[i] \cdot \mathbb{1}_{B_{\mathrm{TOT}} \times 1} \preccurlyeq \mathbb{1}_{V \times 1}, \tag{30}$$

$$\mathbb{1}_{1 \times V} \cdot (\mathbf{T}_{\mathbf{UL}}[i] \circ \mathbf{A}[i]) \preccurlyeq (\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}} - \tau_L)\mathbb{1}_{1 \times B_{\mathrm{TOT}}}, \tag{31}$$

$$\mathbf{A}[i] \in \{0,1\}^{V \times B_{\mathrm{TOT}}}. \tag{32}$$

For problems with small numbers of variables and constraints, the solution to the relaxed optimization problem in (29)–(32) is derived using integer linear programming solvers. In this work, we consider the COIN-OR branch and cut (CBC) solver [22]. However, once the number of variables and constraints increases, e.g., in scenarios where a large number of vehicles are considered for the learning, the CBC solver either requires long runtimes or fails to converge to the optimal solution. In Section 4, we propose a heuristic algorithm, namely the VBI algorithm, to approximate the solution of both optimization problems (24)–(28) and (29)–(32). The advantage of a heuristic algorithm is that it provides a near-optimal solution, even in scenarios where the solver cannot converge to the optimal solution. Additionally, the algorithm provides its solution in a shorter time than the solver, especially in scenarios with large number of variables and constraints.

4. Proposed Solution

In this section, we propose the VBI algorithm as an approximated solution to the optimization problems (24)–(28) and (29)–(32). First, we provide a description of the algorithm and then, we address the impact of the vehicle importance q_{vb} in (11) on the behavior of the VBI algorithm.

4.1. Algorithmic Description

The VBI algorithm indicates which vehicles are selected for training at each communication round and assigns the selected vehicles to an appropriate base station beam. Hence, the VBI algorithm forms vehicle–beam pairs with the goal of maximizing the total vehicle importance. The VBI algorithm is described in Algorithm 1 and its explanation is as follows.

First, in lines 1–4, the initialization is performed. In line 1, the vehicle-beam matrix $\mathbf{A}$ is initialized by setting $A_{vb} = 0$ to all vehicle-beam pairs that do not satisfy constraint (27), i.e., $\tau_{\mathrm{T},v} + \tau_{\mathrm{UL},vb} \leq \tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$. The remaining entries of matrix $\mathbf{A}$ are temporarily set to the "-1" value and they will be later updated by the algorithm. In line 2, a vector $\boldsymbol{\tau_B}$, holding the upload latency budget for every beam is defined and it is initialized at $\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}}$. Finally, lines 3 and 4, initialize to zero the vector $\boldsymbol{\tau_L}$ holding the start time of the uplink transmissions and the total vehicle importance Q_{TOT}, respectively.

Algorithm 1 Vehicle-Beam-Iterative (VBI) Algorithm

Input: Training time $\boldsymbol{\tau_T}$ of vehicles, upload time $\mathbf{T_{UL}}$ of vehicles per beam, importance $\mathbf{Q}$ of vehicles per beam and time period $\tau_{\text{APP,MAX}} - \tau_{\text{DL}}$
Output: Vehicle selection and beam allocation $\mathbf{A}$ and total vehicle importance Q_{TOT}

1: Set $A_{vb} = 0$ if $\tau_{\text{T},v} + \tau_{\text{UL},vb} > \tau_{\text{APP,MAX}} - \tau_{\text{DL}}$, else $A_{vb} = -1$, for each vehicle $v \in \mathcal{V}$ and beam $b \in \mathcal{B}_{\text{TOT}}$
2: Set $\tau_{\text{B},b} = \tau_{\text{APP,MAX}} - \tau_{\text{DL}}$, for each beam $b \in \mathcal{B}_{\text{TOT}}$
3: Set $\tau_{\text{L},b} = 0$, for each beam $b \in \mathcal{B}_{\text{TOT}}$
4: Set $Q_{\text{TOT}} = 0$
5: **while** matrix $\mathbf{A}$ contains an entry with value "-1" **do**
6: **for** every vehicle $v \in \mathcal{V}$ not yet scheduled **do**
7: Obtain $b^* = \text{argmax}_{b \in \mathcal{B}_{\text{TOT}}}(q_{vb})$, given that $A_{vb} = -1$
8: **end for**
9: **for** every beam $b \in \mathcal{B}_{\text{TOT}}$ **do**
10: Set scheduled = `False`
11: Obtain $v^* = \text{argmax}_{v \in \mathcal{V}_b}(q_{vb})$, where $\mathcal{V}_b$ holds the vehicles selecting beam b as their b^*
12: **if** v^* is the first vehicle scheduled on beam b, i.e., $\tau_{\text{L},b} = 0$ **then**
13: Set training time $\tau_{\text{L},b} = \tau_{\text{T},v^*}$
14: Set scheduled = `True`
15: **else if** $\tau_{\text{UL},v^*b} \leq \tau_{\text{B},b} - \min(\tau_{\text{L},b}, \tau_{\text{T},v^*})$ **then**
16: Set training time $\tau_{\text{L},b} = \min(\tau_{\text{L},b}, \tau_{\text{T},v^*})$
17: Set scheduled = `True`
18: **end if**
19: **if** scheduled = `True` **then**
20: Set $A_{v^*b} = 1$
21: Set $A_{v^*\hat{b}} = 0$ for all other beams $\hat{b} \in \mathcal{B}_{\text{TOT}} \setminus b$
22: Update $\tau_{\text{B},b} -= \tau_{\text{UL},v^*b}$
23: Update total importance $Q_{\text{TOT}} += q_{v^*b}$
24: **end if**
25: **end for**
26: **for** every vehicle $v \in \mathcal{V}$ not yet scheduled **do**
27: **for** every beam $b \in \mathcal{B}_{\text{TOT}}$ **do**
28: **if** $\tau_{\text{UL},vb} > \tau_{\text{B},b} - \tau_{\text{L},b}$ **then**
29: $A_{vb} = 0$
30: **end if**
31: **end for**
32: **end for**
33: **end while**
34: **return** $\mathbf{A}$, Q_{TOT}

After the initialization step, from line 5 onwards the algorithm repeats continuously until all entries of the matrix $\mathbf{A}$ are set to 0 or 1. The algorithm assigns at most one vehicle per beam per iteration and hence at most V iterations are performed. Each iteration consists of the following five steps:

- **STEP 1 (lines 6–8)**: For each vehicle $v \in \mathcal{V}$ that has not already been selected for training, the beam b^* that maximizes the importance q_{vb} of the vehicle v is obtained, assuming that $A_{vb} = -1$.
- **STEP 2 (lines 9–11)**: From line 9 onwards, the algorithm iterates over all beams to define per beam $b \in \mathcal{B}_{\text{TOT}}$ whether or not a vehicle will be assigned to it and which vehicle that will be. Depending on whether or not a vehicle is assigned to the beam, different steps are followed later on. Therefore, in line 10 a decision variable is initialized to `False`. Then, in line 11, based on the derived potential vehicle-beam pairs from step 1 (line 7), the vehicle v^* that has the highest importance q_{vb} on each beam b is selected.

- **STEP 3 (lines 12–18)**: In this step a decision is taken on whether or not the selected vehicle v^* can be scheduled on beam b. In line 12, it is checked whether or not vehicle v^* is the first vehicle to be scheduled on beam b. If it is the first one, line 13 sets the training time $\tau_{\mathrm{L},b}$ at beam b equal to the training time τ_{T,v^*} of vehicle v^* and line 14 sets the decision variable to `True`. If vehicle v^* is not the first vehicle to be scheduled on beam b, line 15 evaluates according to constraint (26) if vehicle v^* can be co-scheduled with the other vehicle(s) already scheduled on beam b. If vehicle v^* can be co-scheduled, in line 16, the training time $\tau_{\mathrm{L},b}$ at beam b is set to the minimum time between the training time $\tau_{\mathrm{L},b}$ set in a previous iteration, when scheduling a different vehicle, and the training time τ_{T,v^*} of the newly scheduled vehicle v^*. Line 17 sets the decision variable to `True`.
- **STEP 4 (lines 19–25)**: If vehicle v^* is scheduled on beam b, i.e., the decision variable is `True`, in lines 20 and 21, the VBI algorithm updates accordingly the entries of matrix $\mathbf{A}$ involving vehicle v^* to ensure that it is assigned only to beam b. Next, line 22 reduces the total available uploading latency budget $\tau_{\mathrm{B},b}$ accordingly and line 23 increases the total vehicle importance Q_{TOT} with the importance q_{v^*b} of the newly scheduled vehicle. In case that vehicle v^* is not scheduled, i.e., the decision variable is `False`, no action is taken and the vehicle can be re-considered for scheduling in a later iteration.
- **STEP 5 (lines 26–32)**: After iterating over all beams, and before starting a new iteration as a result of line 5, an update step takes place. Specifically, lines 26–32 discard vehicle–beam pairs that cannot fulfill constraint (27) due to the newly scheduled vehicles in the given algorithm iteration.

Finally, all iterations are completed in line 33 and then line 34 returns matrix $\mathbf{A}$ and the total vehicle importance Q_{TOT}.

The complexity of the algorithm is split into two parts. The first part relates to the initialization steps, which have a complexity of $\mathcal{O}(VB_{\mathrm{TOT}})$ due to the calculations in line 1. The second part relates to STEPS 1–5, which also have a complexity of $\mathcal{O}(VB_{\mathrm{TOT}})$ for a single beam iteration. As mentioned above, the algorithm performs at most V iterations, and hence the complexity is $\mathcal{O}(V^2B_{\mathrm{TOT}})$.

4.2. Algorithm Behavior

Because the VBI algorithm depends on vehicle importance q_{vb} in (11), the value of the tuning parameter ρ influences its behavior. Specifically, when $\rho = 0$, the importance q_{vb} depends only on the resource consumption $C_{R,vb}$, which essentially depends on the bit rate R_{vb} and varies per beam. To maximize the total vehicle importance Q_{TOT}, the algorithm selects vehicles with the strongest wireless channels and assigns them to the beams that provide the highest bit rate R_{vb}. Hence, the VBI algorithm maximizes the number of selected vehicles. We will refer to the solution of the VBI algorithm with $\rho = 0$ as `VBI-rate`.

On the other hand, when $\rho = 1$, the vehicle importance q_{vb} depends only on the training loss $F(\mathbf{W}_{G,v})$, which is independent of the beam. In this case, the VBI algorithm prioritizes vehicles with high training loss $F(\mathbf{W}_{G,v})$, thus meaning that they may be assigned on a sub-optimal beam in terms of their resource consumption $C_{R,vb}$. Nevertheless, latency constraints are still considered, and the shorter the latency budget $\tau_{\mathrm{APP,MAX}}$, the more likely a vehicle will be assigned to the beam with the lowest resource consumption $C_{R,vb}$ (and highest bit rate R_{vb}). If a vehicle is assigned to a sub-optimal beam in terms of the resource consumption $C_{R,vb}$, more resources will be consumed, thus limiting the total number of selected vehicles. We will refer to the solution of the VBI algorithm with $\rho = 1$ as `VBI-loss`.

Notably, the VBI algorithm is general, as it allows redefining vehicle importance q_{vb} in terms of either learning metrics (numerator) or wireless resource metrics (denomina-

tor) in Equation (11). In addition, it is flexible, since different values of $\rho \in [0,1]$ yield variants of VBI that emphasize learning performance or wireless resource consumption to different degrees.

5. Scenario Configuration

This section presents the considered scenarios to evaluate the performance of the VBI algorithm. First, we present the learning task and then introduce the configuration of four learning scenarios. Next, we provide the baseline algorithms, which will be compared against the VBI algorithm and finally, we present the wireless scenario.

5.1. Learning Task

The learning task considered in this paper is the object classification of traffic signs. This learning task is a relevant FL application in vehicular networks because different countries use different traffic signs. Therefore, an algorithm that very accurately detects the meaning of a traffic sign in one country may not be able to detect the traffic sign that has the same meaning in a different country, or a traffic sign that does not exist in its origin country. With FL, a global model can be trained based on knowledge from both countries, which will allow for all vehicles to accurately detect traffic signs from both countries. For this work, the European traffic sign data set is used, comprising 164 classes of traffic signs originating from six distinct European countries [23]. Considering that some of the classes contain very few training samples, we select the $n_C = 10$ classes with the highest number of available samples for our study.

The object classification task utilizes a convolutional neural network (CNN) architecture similar to that used in research by Serna and Yuichek [23] and Chiamkurthy [24], which are in turn both inspired by the Visual Geometry Group (VGG) architecture [25]. Figure 3 depicts the assumed CNN architecture, which applies the rectified linear unit (ReLU) function, batch normalization, max pooling and dropout regularization. The final layer is activated by a softmax with 10 outputs indicating the per-class likelihoods. In total, the CNN comprises 3349418 trainable parameters. Assuming a 32-bit precision per trainable parameter, this translates to a model size of $Z \approx 107$ Mbits.

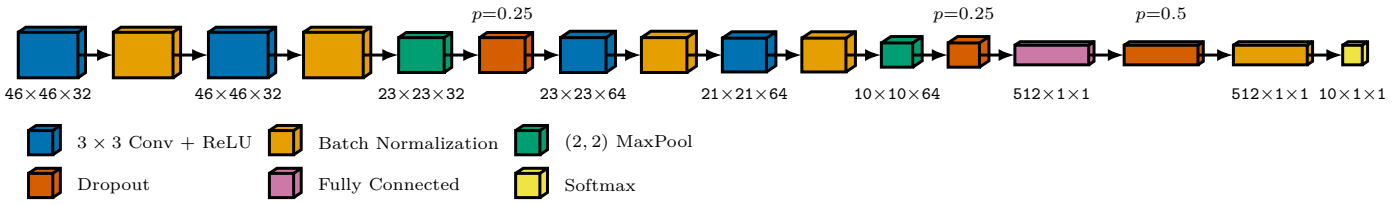

Figure 3. The CNN architecture assumed to perform the object classification task.

5.2. Learning Scenarios

In our analysis, we consider scenarios with $V = 50$ vehicles and both IID and non-IID data. Moreover, we address both the scenarios where vehicles have the same and different training data set sizes, which are described by the problems in (29)–(32) and (24)–(28), respectively. In total, we consider four learning scenarios, whose configurations are summarized in Table 2. Apart from the scenario with same data set size and IID data, in the other three scenarios, the number of training samples are unevenly split over the number of assigned classes. For the scenario with the same data set size and non-IID data, this split is performed such that on average all classes are equally represented in the training data set $\mathcal{K}$. For scenarios with different data set sizes, the number of samples per assigned class per vehicle is drawn from a Poisson distribution with a rate of 15 and 75, for IID data and non-IID data, respectively, as also shown in Table 2.

Table 2. Configuration of the four learning scenarios.

Parameter	Same Data Set Size		Different Data Set Sizes	
	IID	**Non-IID**	**IID**	**Non-IID**
Number of classes per vehicle	10	2	10	2
Training samples K_v per vehicle	150	150	150 (average)	150 (average)
Training samples per class per vehicle	15	75 (average)	15 (average)	75 (average)
Testing samples $K_{T,v}$ per vehicle	50	50	50 (average)	50 (average)
Testing samples per class at FL server	100	100	100	100

For calculation of the loss $F(\mathbf{W}_{G,v}[i])$ of vehicle v at communication round i, the categorical cross-entropy loss function is applied on the testing data set $\mathcal{K}_{T,v}$, which is unique for every vehicle and three times smaller than the training data set $\mathcal{K}_v$. Moreover, the split of the testing data set among the vehicles and the classes is similar to the split of the training data set. Finally, the accuracy of the global model in all four scenarios is measured at the FL server based on an FL server specific testing data set, which consists of 100 samples per class. Note that the accuracy of the global model is defined as the proportion of correctly categorized testing samples divided by the total number of classification instances, where for each classification instance a single testing sample from the FL server is used.

For the training, the vehicles use the SGD optimizer with learning rate $\eta = 0.05$, batch size $s_B = 64$ and with each vehicle performing $n_{\text{LE}} = 2$ local epochs. The number of FLOPs required from the vehicles to train the CNN for a batch size $s_B = 64$ is measured by the Keras library, in Python version 3.6.8, which is $n_{\text{FLOP},G} = 6.55$ GFLOPs. Regarding the hardware of the vehicles, we consider the processing capabilities $g_v = 64$ GFLOPs per second. Therefore, the training time $\tau_{\text{T},v}$ of vehicle v, as given by (13), depends on the number of training samples K_v at vehicle v, which depends on the learning scenario.

5.3. Baseline Algorithms

To evaluate the performance of the VBI algorithm, we compare it with two baseline algorithms, viz. the `max-loss-rate` and the `random-rate` algorithms. The `max-loss-rate` algorithm aims to maximize the sum of the losses over all selected agents based on a rate-based beam assignment. Thus, it treats the loss and rate as fixed metrics for vehicle importance. First, it sorts the vehicles in descending order based on their loss $F(\mathbf{W}_{G,v})$ and selects as many vehicles as possible until the constraints in (24)–(28) are violated. Each selected vehicle v is assigned to the beam b^* that it experiences the lowest resource consumption C_{R,vb^*}. If a vehicle v cannot be assigned to its best beam b^*, the vehicle v is not selected for training. The `random-rate` algorithm is implemented similarly, but the algorithm iterates over the vehicle list in a random order.

One widely used agent selection algorithm from the literature, is the `FedCS` algorithm, as introduced by Nishio and Yonetani [26], which is based on a greedy method to maximize the number of selected agents. When extending this algorithm to our considered scenario, i.e., vehicle selection and beam allocation, the vehicles need to be assigned to the beam that provides the highest bit rate. Therefore, the `FedCS` algorithm is almost identical to our proposed `VBI-rate` algorithm (VBI algorithm when $\rho = 0$). This shows the adaptability of the VBI algorithm in different scenarios via the appropriate configuration of the tuning parameter ρ.

5.4. Wireless Scenario

For the wireless communication scenario, we consider an urban macro environment at $f_C = 3.5$ GHz and a bandwidth of $f_{\text{BW}} = 50$ MHz [19,27]. For the wireless propagation in (8), we assume a path loss exponent $\gamma = 3.7$ and shadowing with $\sigma = 8$ dB, which are

typical values for outdoor dense urban environments [20]. The considered area is covered by $M = 7$ three-sectorized base stations, placed on a hexagonal grid with an inter-site distance of 500 m at a height of 25 m [19]. Each sector is equipped with a 4×4 uniform planar rectangular array (UPRA) configured in a grid of beams comprising $B_M = 12$ beams.

In urban macro deployments, a high number of vehicles is expected to drive around an urban grid that consists of three blocks, each measuring 433 m by 250 m, making up a total area of 433 m by 750 m [19]. Each street around the block has a total of four lanes and there are two lanes per driving direction. The lane width is 3.5 m [19]. Moreover, the vehicles are driving at a speed of 60 km per hour and their antennas are placed at a height of 1.6 m [27]. At the intersections, the vehicles have a probability of 0.5 to keep driving straight ahead, 0.25 to go left and 0.25 to go right [27]. Finally, each vehicle is equipped with a 2×2 UPRA, which can steer the beam to the direction of the beam formed at the serving sector.

Based on the considered UPRA model, as defined in Appendix A.1, an analysis is conducted in Appendices A.2 and A.3 to define the beam directions in the grid of beams at each base station. The derived beam directions are the same at all sectors and they are the following: $-45°, -15°, 15°$ and $45°$ in the azimuth plane and $17°, 47°$ and $77°$ in the elevation plane. A coverage analysis revealed that all roads are covered by the beams pointing at the cell edge. Thus, we only consider the cell edge beams for our evaluation. The remaining UPRA parameters, and in particular the maximum transmit power, noise figure and antenna gain, are derived in Appendix A.3 and summarized in Table 3. These parameters are presented for both the base stations and the vehicles.

Table 3. Parameters of the UPRA at the base stations and the vehicles.

Base Station		Vehicle	
Parameter	**Value**	**Parameter**	**Value**
$P_{\mathrm{M,MAX}}$	49 dBm	$P_{\mathrm{V,MAX}}$	23 dBm
$P_{\mathrm{NF},M}$	5 dB	$P_{\mathrm{NF},V}$	9 dB
$G_{\mathrm{M,MAX}}$	12 dBi	$G_{\mathrm{V,MAX}}$	6 dBi

Recall from Section 2 that during a given communication round, the vehicles stay connected to one and the same beam. In Appendix A.4, we approximate the time that a vehicle stays connected to a single beam. From the analysis, it is estimated that vehicles stay connected to a cell edge beam between 10.4 s and 17.0 s. This time interval upper bounds the latency budget $\tau_{\mathrm{APP,MAX}}$ to ensure that the assumption of staying connected to one beam is not violated.

In Appendix A.5, we calculate the downlink bit rate at the cell edge at 105 Mbps, which can also serve as a broadcast bit rate. Considering that the FL model size is $Z \approx 107$ Mbits the broadcast time duration is $\tau_{\mathrm{DL}} \approx 1.02$ s. Finally, we set the latency budget $\tau_{\mathrm{APP,MAX}} = 2.5$ s, which is long enough to allow vehicles to train and upload their local models.

6. Results and Discussion

This section evaluates the performance of the VBI algorithm. First, Section 6.1 examines its relative performance compared to the optimal solution of the problem in (29)–(32). Then, we assess the accuracy of the global model when solving the problems in (24)–(28) and (29)–(32). Recall that the accuracy of the global model is defined as the proportion of correctly categorized testing samples divided by the total number of classification instances. For this evaluation, we consider the four learning scenarios described in Section 5.2 and the baseline algorithms presented in Section 5.3. Sections 6.2 and 6.3 show results

for scenarios where vehicles have the same and different data set sizes, respectively. Finally, Section 6.4 compares the four learning scenarios in terms of the time required to reach a certain accuracy level. We present the results of the four learning scenarios as an average of 15 independent simulations. The source code used to generate these results is available in [28].

6.1. VBI Algorithm Relative Performance

We evaluate the VBI algorithm with respect to the problem in (29)–(32), i.e., when vehicles have the same training data set size. This comparison demonstrates that the VBI algorithm can provide an accurate approximation to the optimal solution of the problem in (29)–(32). For evaluation, we compare the performance of the VBI algorithm, in terms of the total vehicle importance Q_{TOT}, against the optimal solution given by the CBC solver. Note that the problem in (29)–(32) aims to maximize the total vehicle importance Q_{TOT} and thus both the VBI algorithm and the CBC solver return this value as part of their solution. Therefore, the ratio of the total vehicle importance Q_{TOT} as returned by the VBI algorithm divided by the total vehicle importance returned by the CBC solver indicates how similar the two values are. In the case where the ratio equals to one, the two values are equal and the VBI algorithm returns the same value as the CBC solver. We define this ratio as the 'relative performance' metric.

For the comparison, three values of the tuning parameter ρ are considered: the two extreme cases of $\rho = 0$ (i.e., `VBI-rate`) and $\rho = 1$ (i.e., `VBI-loss`) and the case of $\rho = 0.8$, which leads to a similar value range for the local loss $F(\mathbf{W}_{G,v})$ and the resource consumption $C_{R,vb}$. Moreover, the number of vehicles V in the network is also varied. Finally, the obtained results are averaged over 1000 independent simulations.

Figure 4 shows that for the extreme case of $\rho = 0$ (i.e., `VBI-rate`), the VBI algorithm provides a near-optimal solution, regardless of the number of vehicles in the network. This is because the `VBI-rate` algorithm selects vehicles with the strongest wireless channels and assigns them to beams with the lowest resource consumption $C_{R,vb}$. Consequently, the number of scheduled vehicles is maximized, leading to a near-optimal total vehicle importance Q_{TOT}.

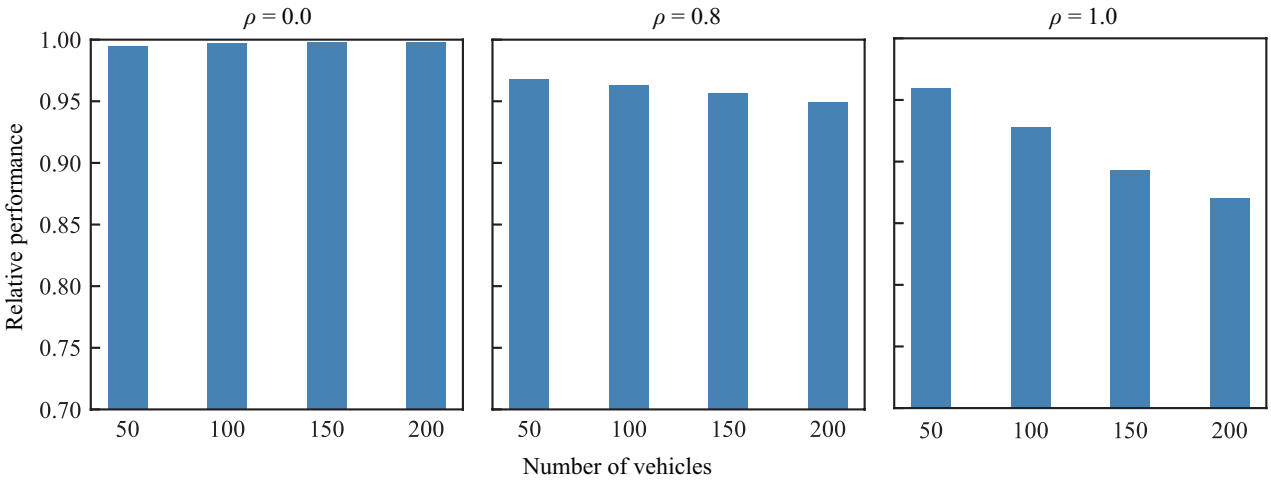

Figure 4. Relative performance of the VBI algorithm to the optimal solution, for the problem where vehicles have the same training data set size.

At the other extreme case of $\rho = 1$ (i.e., `VBI-loss`), the VBI algorithm prioritizes the selection of vehicles with high loss $F(\mathbf{W}_{G,v})$, which might be assigned to a sub-optimal beam, as it was explained in Section 4. Figure 4 shows that the relative performance of the VBI algorithm when $\rho = 1$ is lower than when $\rho = 0$, which is a result of the sub-optimal

beam assignment. Specifically, the sub-optimal beam assignment leads to a higher resource consumption which in turn limits the total number of vehicles that can be scheduled and consequently the total vehicle importance Q_{TOT}. Figure 4 also shows that the performance further decreases with the number of vehicles in the network, because there is a higher probability that a vehicle will be assigned to a sub-optimal beam.

When $\rho \in (0, 1)$, and specifically $\rho = 0.8$ in this comparison, Figure 4 shows that the relative performance of the VBI algorithm is lower than with $\rho = 0$ but higher than with $\rho = 1$. This is because the tuning parameter ρ configures the vehicle importance q_{vb} to account almost equally for both the resource consumption $C_{R,vb}$ and the loss $F(\mathbf{W}_{G,v})$. Therefore, during beam assignment, there is some distinction among the beams to determine the best serving beam in terms of the resource consumption $C_{R,vb}$. However, this distinction is not as prominent as when $\rho = 0$. Hence, the larger the ρ, the less distinction there is among the beams, which consequently leads to a sub-optimal beam assignment. Simulations showed that when $\rho = (0, 1)$, the accuracy of the global model falls between the accuracies obtained at $\rho = 0$ and $\rho = 1$. Therefore, in the remainder of this evaluation, we only show the performance of the VBI algorithm when $\rho = 0$ and $\rho = 1$, i.e., the `VBI-rate` and `VBI-loss` algorithms, respectively. Therefore, the key takeaway from these evaluations is summarized below:

Result 1. *The VBI algorithm provides a near-optimal solution when $\rho \to 0$ (i.e., `VBI-rate`), regardless of the number of vehicles, because it leverages the distinction of beams in terms of resource consumption $C_{R,vb}$. The performance of the VBI algorithm drifts from the optimal solution when increasing the parameter ρ, and in combination with the number of vehicles. Overall, the VBI algorithm offers a good approximation of the optimal solution, and its results will be used in the further evaluations.*

However, the relative performance of the VBI algorithm is not directly related to the accuracy of the global model. Hence, in the following sections, we evaluate the VBI algorithm with respect to global model accuracy.

6.2. Same Data Set Size

We evaluate the performance of the VBI algorithm in terms of the accuracy of the global model for the problem in (29)–(32), where the vehicles have the same training data set size. For this purpose, we compare the `VBI-rate`, `VBI-loss`, `max-loss-rate` and `random-rate` algorithms under scenarios with IID and non-IID data.

6.2.1. IID Data

Figure 5 shows accuracy over time and illustrates that all four algorithms achieve similar performance. The comparable results of the `VBI-loss` and `max-loss-rate` algorithms are expected, as both select vehicles with the highest loss $F(\mathbf{W}_{G,v})$. Additionally, both algorithms select approximately the same number of vehicles per communication round. This implies that the two algorithms are almost identical and that the `VBI-loss` algorithm mostly assigns, to the selected vehicles, the beam that leads to the lowest resource consumption $C_{R,vb}$. This efficiency arises from the short latency budget $\tau_{\text{APP,MAX}} = 2.5$ s, which enforces that only vehicles with favorable wireless channels can participate in the learning process. Due to vehicle mobility, channel quality varies over time, and hence the channel quality limitation due to the latency budget $\tau_{\text{APP,MAX}}$ applies to a different set of vehicles per communication round. Therefore, all four algorithms achieve resource-efficient beam assignments, and all vehicles have fair chances, over time, to be selected.

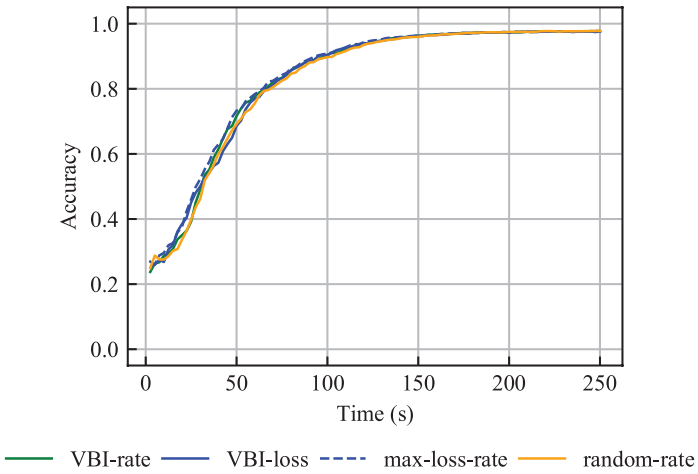

Figure 5. Accuracy over time for the problem with the same training data set sizes and IID data, averaged over 15 independent simulations.

Moreover, Figure 5 shows that although the `VBI-rate` and `random-max` algorithms do not take the loss $F(\mathbf{W}_{G,v})$ into account, they perform similarly to the loss-aware `VBI-loss` and `max-loss-rate` algorithms. This is because all vehicles have samples from every class, thus making the choice of vehicles less crucial for the learning process. Therefore, the main result herein is the following:

Result 2. *When vehicles have the same data set size and IID data are considered, the choice of vehicles is not crucial, provided that resource-efficient beam assignment is performed.*

6.2.2. Non-IID Data

When non-IID data are considered, vehicles contain samples from only two classes; therefore, the selection of vehicles in a given communication round becomes more crucial than in scenarios with IID data. Figure 6 illustrates that, under non-IID data, the loss-aware `VBI-loss` and `max-loss-rate` outperform the loss-unaware `VBI-rate` and `random-rate` algorithms. This is because the former algorithms take both learning and channel aspects into account. The learning aspect ensures that vehicles carrying samples that contribute more to the learning process are selected more frequently, whereas the channel aspect ensures resource-efficient beam assignment. Recall that the `VBI-loss` algorithm implicitly takes the channel quality into account via the latency budget $\tau_{\mathrm{APP,MAX}}$. Additionally, Figure 6 shows that the `VBI-loss` and `max-loss-rate` algorithms behave almost identically, for the same reason as in the IID data scenario.

Moreover, Figure 6 illustrates that although the four algorithms behave differently, they eventually converge to the same accuracy level. Specifically, an accuracy of 96% is reached within 250 s. Convergence to the same accuracy level occurs because many vehicles are selected per communication round, resulting in sufficient training across all distributed samples. The use of MU-MIMO plays a key role in this outcome by providing two main benefits. First, enhanced throughput allows the FL server to select vehicles that are located at the cell edge. It can be qualitatively argued that these vehicles would not have been selected in a single-antenna system, thus reducing the overall vehicles participation. Second, MU-MIMO enables multiple vehicles per base station to be selected in the same training round, with each vehicle transmitting its model to the base station via a different beam. These benefits allow training on a larger number of samples per communication round, and on a wider sample set. This eventually improves the convergence time of the global model.

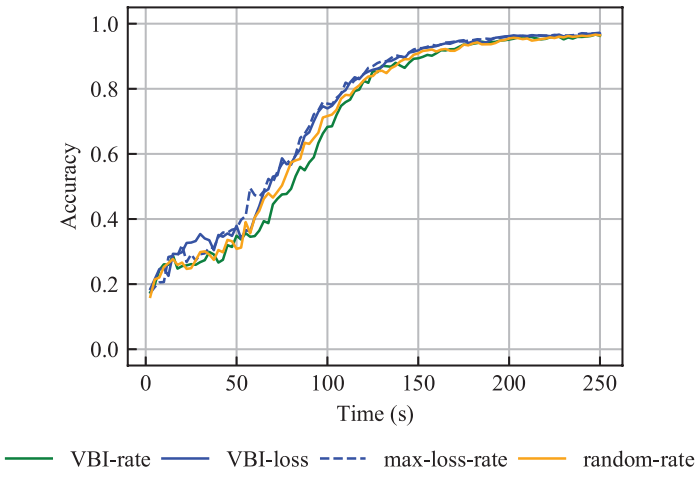

Figure 6. Accuracy over time for the problem with the same training data set sizes and with non-IID data, averaged over 15 independent simulations.

Therefore, the key takeaway results are:

Result 3. *When vehicles have the same data set size and non-IID data are considered, loss-aware algorithms provide higher accuracy during the initial learning phase, assuming that resource-efficient beam assignment is performed.*

Result 4. *MU-MIMO-capable base stations improve the convergence time of the global model, as they allow training on a larger number of samples at each communication round, and on a wider sample set. This results from the enhanced quality of the wireless channels, selection of agents at the cell edge, and exploitation of the spatial separation of the vehicles.*

6.3. Different Data Set Sizes

We now evaluate the performance of the VBI algorithm in terms of global model accuracy for the problem in (24)–(28), where vehicles contain different training data set sizes. Thus, the vehicles have different training times $\tau_{\mathrm{T},v}$. Although the VBI algorithm does not explicitly select vehicles based on their training time $\tau_{\mathrm{T},v}$, vehicles with shorter training times $\tau_{\mathrm{T},v}$ have a higher selection probability. This is due to constraint (27), enforcing that the selected vehicles need to train and upload their local model within the latency budget $\tau_{\mathrm{APP,MAX}} - \tau_{\mathrm{DL}} \approx 1.5$ s. Therefore, vehicles with short training times $\tau_{\mathrm{T},v}$ can be selected even if their channel quality is not very good. On the other hand, vehicles with long training times $\tau_{\mathrm{T},v}$ can only be selected when they have a very good channel quality.

6.3.1. IID Data

Figure 7 shows the accuracy over time for the four considered algorithms and demonstrates that the `VBI-rate`, `VBI-loss` and `max-loss-rate` algorithms perform similarly, while the `random-rate` algorithm underperforms. The similarity in performance between the `VBI-loss` and `max-loss-rate` algorithms is explained by the applied short latency budget $\tau_{\mathrm{APP,MAX}}$, as noted earlier in Section 6.2.1. Moreover, the design of the two loss-based algorithms and the `VBI-rate` algorithm allows them to more frequently select vehicles with longer training times $\tau_{\mathrm{T},v}$ compared to the `random-rate` algorithm. Thus, the `random-rate` algorithm trains more frequently on a specific set of samples, resulting in a slower convergence compared to the other three algorithms.

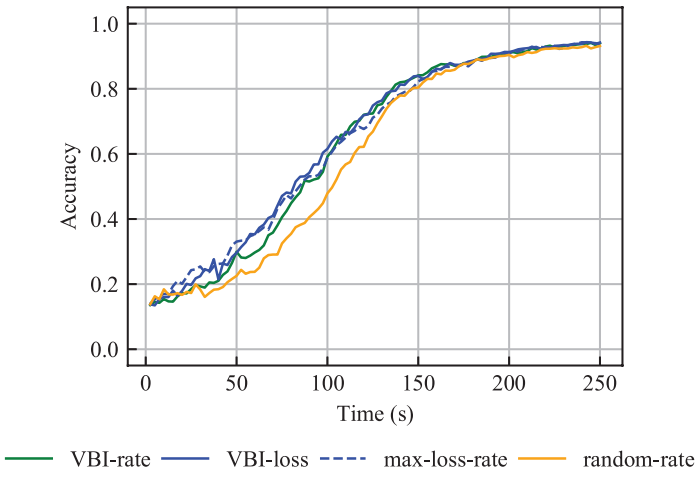

Figure 7. Accuracy over time for the problem with different training data set sizes and with IID data, averaged over 15 independent simulations.

Specifically, the `VBI-rate` algorithm prioritizes vehicles with good channel quality. Therefore, when vehicles with long training times $\tau_{\mathrm{T},v}$ experience good channels, they are likely to be selected. Additionally, the loss-based algorithms prioritize vehicles with higher loss $F(\mathbf{W}_{G,v})$. Consequently, vehicles with a high loss $F(\mathbf{W}_{G,v})$ are selected for training once their channel quality allows for it. Overall, Figure 7 shows that the three algorithms perform similarly. We conclude that the vehicle selection in scenarios with IID data is not crucial, as long as all vehicles contribute to the learning process, which is consistent with Result 2.

Moreover, vehicles with shorter training times $\tau_{\mathrm{T},v}$ have fewer training samples K_v. Thus, their contribution to the global model is not very significant. Considering that vehicles with short training times $\tau_{\mathrm{T},v}$ are often selected, the global model does not change significantly per communication round. Therefore, the convergence time is longer compared to the scenario where all vehicles have the same training times $\tau_{\mathrm{T},v}$. The main message in this scenario is:

Result 5. *The good performance of the `VBI-rate`, `VBI-loss` and `max-loss-rate` algorithms, in scenarios where vehicles have different data set sizes and IID data are considered, is attributed to their ability of frequently selecting vehicles with high training times and thus, allow all vehicles to participate in the learning process.*

6.3.2. Non-IID Data

Figure 8 shows the accuracy over time for the scenario with non-IID data. Here, the `random-rate` algorithm again converges slower than the other three algorithms. Similar to the IID case, in Section 6.3.1, the performance difference of the `random-rate` algorithm compared to the other three algorithms is attributed to not selecting as frequently vehicles with high training time $\tau_{\mathrm{T},v}$. However, this performance difference is more modest in scenarios with non-IID data compared to scenarios with IID data. This occurs because most vehicles are important for the learning process, as their local losses are computed over two classes using a global model averaged across multiple classes. As a result, most vehicles have comparable learning importance, giving the `random-rate` algorithm a higher likelihood of selecting important ones.

As noted in Result 3, with non-IID data, loss-based `VBI-loss` and `max-loss-rate` algorithms provide advantages over `VBI-rate` and `random-rate` algorithms. However, when vehicles have different data set sizes, the loss-based algorithms do not provide significant performance gains. The fact that some vehicles have small testing data set size

$K_{T,v}$, implies that those vehicles calculate their loss $F(\mathbf{W}_{G,v})$ inaccurately. Combined with the non-IID data, where only two classes are represented per vehicle, the loss values in this scenario are both high and highly variable. These two factors indicate that most vehicles are important to participate in the FL training process. As a result, the loss-based algorithms tend to select vehicles more evenly, which reduces the performance gap between them and the `VBI-rate` and `random-rate` algorithms, as shown in Figure 8.

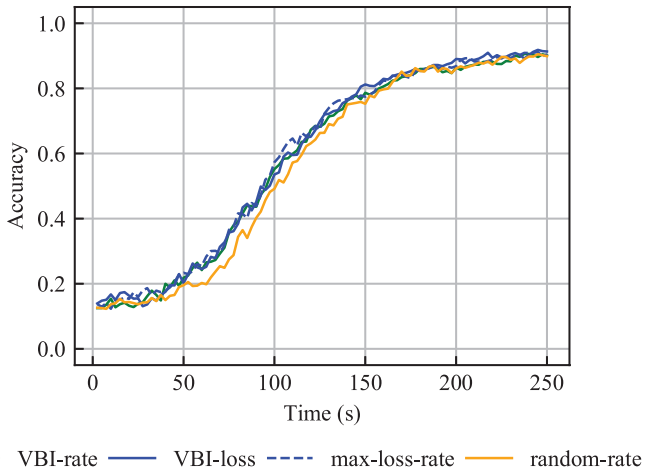

Figure 8. Accuracy over time for the problem with different training data set sizes and with non-IID data, averaged over 15 independent simulations.

Nevertheless, both loss-based algorithms and `VBI-rate` outperform `random-rate`, demonstrating that learning importance (loss metric) and wireless importance (rate as wireless resource consumption metric) are equally relevant for FL training on non-IID data with varying training times. Therefore, we conclude that the loss $F(\mathbf{W}_{G,v})$ and the rate are similarly important metrics for identifying vehicle importance in the learning process when some vehicles have small testing data set sizes $K_{T,v}$ under non-IID conditions.

To further distinguish among vehicles, one option is to consider their testing dataset sizes $K_{T,v}$. However, this may not yield substantial gains, as it would implicitly prioritize vehicles with long training times $\tau_{T,v}$. Because the uploading time is limited, fewer vehicles would be scheduled. This reduction in the number of scheduled vehicles hinders the learning process and does not allow for faster learning. Thus, we highlight that the loss metric remains a strong indicator of a vehicle's importance in the learning process, as it selects vehicles with lower training times while achieving the highest accuracy.

The key take away message is:

Result 6. *When vehicles have different data set sizes and non-IID data are considered, the loss $F(\mathbf{W}_{G,v})$ remains an effective metric for the learning process, yielding the highest accuracy. However, the loss metric is equally important as the rate metric.*

6.4. Comparison of Learning Scenarios

Some applications require training the global model until reaching a specific accuracy target. Therefore, we compare the four algorithms in terms of how much time is needed to reach the 85% and 90% accuracy levels. To average out simulation noise, we consider that the accuracy level is reached if the average accuracy is above the accuracy target for 30 s. Table 4 shows the time in seconds to reach each accuracy level, where a hyphen indicates that the accuracy level could not be reached within the simulated 250 s and have maintained that level for at least 30 s. Specifically, for the scenario addressing problem (24)–(28), where vehicles have different data set sizes and non-IID data, the `VBI-loss` and `max-loss-rate` algorithms reach the 90% accuracy target after about 245 s, whereas the `VBI-rate` and

`random-rate` algorithms have not reached the target yet. However, it can be seen from Figure 8 that all four algorithms reach approximately the same accuracy after 250 s.

Table 4. Time, in seconds, needed to reach the 85% and 90% accuracy levels for every algorithm in each learning scenario, where the shortest time per level and scenario is marked with bold.

Algorithm	Same Data Set Size and IID Data		Same Data Set Size and Non-IID Data		Different Data Set Sizes and IID Data		Different Data Set Sizes and Non-IID Data	
	85%	90%	85%	90%	85%	90%	85%	90%
`VBI-rate`	95	115	145	170	**170**	208	198	-
`VBI-loss`	95	112	**137**	157	172	**206**	**190**	**243**
`max-loss-rate`	**92**	**110**	**137**	**155**	175	208	193	248
`random-rate`	97	117	145	162	180	212	195	-

Moreover, we conclude from Table 4 that the `VBI-loss` and `max-loss-rate` algorithms consistently reach the 90% accuracy target and do so quicker than the other two algorithms. However, the differences between the algorithms are not significant after the initial learning phase, regardless of the learning scenario. This is attributed to the use of MU-MIMO, which improves the quality of the wireless channels and allows the selection of many vehicles per communication round, as also highlighted in Result 4.

Additionally, Table 4 shows that it takes longer to reach the accuracy targets when vehicles have different data set sizes compared to when they have the same data set size. As mentioned in Section 6.3.1, in scenarios with different data set sizes, vehicles with shorter training times are more often selected for training, which then requires more communication rounds to reach a certain accuracy target. Therefore, the comparison among scenarios has the following important take aways:

Result 7. *The two loss-based algorithms, i.e., `VBI-loss` and `max-loss-rate`, are stable in terms of reaching the* 90% *accuracy target more quickly than the `VBI-rate` and `random-rate` algorithms across all learning scenarios. This is particularly evident with varying data set sizes and non-IID data, where loss-based algorithms show performance similar to `VBI-rate` but are more stable.*

Result 8. *In scenarios where vehicles have the same data set size, hence the same training times, the accuracy target is reached faster than when vehicles have different data set sizes, hence different training times.*

7. Conclusions

This work investigated the joint vehicle selection and resource allocation problem for FL in vehicular networks with MU-MIMO-capable base stations. Specifically, we described the related optimization problem under two scenarios: when vehicles have the same data set sizes and when they have different data set sizes. To approximate the solution of these optimization problems, we proposed the VBI algorithm. By conducting extensive simulations, we evaluated the VBI algorithm in various learning scenarios and highlighted the key results 1–8. Overall, these results 1–8 showed that loss-based algorithms consistently achieve high accuracy more quickly than the other algorithms, although their performance gains are limited in scenarios where vehicles have different data set sizes and non-IID data. Moreover, we showed that MU-MIMO improves the convergence time of the global model, as highlighted in Result 4.

We further showed that local loss alone does not adequately characterize importance in the learning process under non-IID data and different data set sizes. For future work, it will be of interest to investigate additional learning importance metrics, or combinations

of learning and wireless metrics, that may provide further improvements in challenging scenarios with variable data set sizes and non-IID data. In addition, it is worth further investigating which vehicular applications can benefit from federated learning and deriving application-specific latency budgets. Finally, future work should also consider scenarios with energy implications, e.g., optimizing the energy consumption of the network.

Author Contributions: Conceptualization, J.M.B.d.S.J., H.V.P. and P.V.M.; methodology, M.R., J.M.B.d.S.J. and R.L.; software, M.R. and J.M.B.d.S.J.; validation, M.R.; formal analysis, M.R., J.M.B.d.S.J. and P.V.M.; investigation, M.R., J.M.B.d.S.J., R.L. and H.V.P.; resources, J.M.B.d.S.J. and H.V.P.; data curation, M.R.; writing—original draft preparation, M.R.; writing—review and editing, J.M.B.d.S.J., R.L., H.V.P. and P.V.M.; visualization, M.R.; supervision, R.L., H.V.P. and P.V.M.; project administration, M.R. and P.V.M.; funding acquisition, J.M.B.d.S.J., H.V.P. and P.V.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research is part of NExTWORKx, a collaboration between TU Delft and KPN on future communication networks. José Mairton B. da Silva Jr. was jointly supported by the European Union's Horizon Europe research and innovation program under the Marie Skłodowska-Curie project FLASH, with grant agreement No 101067652; the Ericsson Research Foundation, and the Hans Werthén Foundation. H. Vincent Poor is supported by the U.S National Science Foundation under Grants CNS-2128448 and ECCS-2335876. Piet Van Mieghem is supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant Agreement 101019718). The APC was funded by the Delft University of Technology (TU Delft).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data can be re-produced by the published code in [28].

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

3GPP	Third-Generation Partnership Project
CBC	COIN-OR Branch and Cut
CNN	Convolutional Neural Network
CPU	Central Processing Unit
FL	Federated Learning
FLOP	Floating Point OPeration
GoB	Grid of Beams
IID	Independent and Identically Distributed
ML	Machine Learning
MU-MIMO	Multi-User Multiple Input Multiple Output
OFDMA	Orthogonal Frequency Division Multiple Access
ReLU	Rectified Linear Unit
SGD	Stochastic Gradient Descent
SNR	Signal-to-Noise Ratio
UPRA	Uniform Planar Rectangular Array
VBI	Vehicle Beam Iterative
VGG	Visual Geometry Group

Appendix A

Appendix A.1. Antenna Model

In this work, we assume for both the base stations and the vehicles, uniform planar rectangular array (UPRAs) with a total of $N_{E,M}$ and $N_{E,V}$ antenna elements, for the base

station antenna arrays and the vehicle antenna arrays, respectively. The antenna elements are positioned at $d_E = 0.5\lambda$ spacing in both the horizontal and vertical plane, where λ is the wavelength. We assume that each antenna element has an omnidirectional radiation pattern with a gain of $G_{AE} = 0$ dBi per antenna element, and the number of antenna elements in the horizontal and vertical plane are equal.

The antenna array at each base station is configured in the grid of beams (GoB) mode. We approximate the beams formed by the vehicles and all beams in the GoB are assumed to have the same half-power beamwidth. Moreover, all beams in the GoB, regardless of their direction, are approximated to have one continuous uniform side lobe. Additionally, given the symmetry of the antenna arrays, the half-power beamwidth $\Delta\varphi_k$ and the first-null beamwidth $\varphi_{0,k}$ are assumed to be equal in both the azimuth and elevation planes and they are defined as [29,30]

$$\Delta\varphi_k \approx 2\left[\frac{\pi}{2} - \cos^{-1}\left(\frac{1.391\lambda}{\pi d_E\sqrt{N_{E,k}}}\right)\right] \approx \sqrt{\frac{3}{N_{E,k}}}, \tag{A1}$$

$$\varphi_{0,k} = 2\left[\frac{\pi}{2} - \cos^{-1}\left(\frac{\lambda}{d_E\sqrt{N_{E,k}}}\right)\right]. \tag{A2}$$

The number of beams in the GoB is given by $B_M = B_{A,M}B_{E,M}$, where $B_{A,M}$ and $B_{E,M}$ denote the number of beams in the azimuth and elevation planes, respectively, and the number of beams in the GoB is derived from the first-null beamwidth $\varphi_{0,M}$ and the targeted angular range in the azimuth and elevation planes. Specifically, setting the boresight direction of each beam at the null of its adjacent beam, we obtain an angular resolution of $\phi_{B,M} = \frac{\varphi_{0,M}}{2}$, for both the azimuth and elevation planes. Thus, the number of beams in a given plane, i.e., $B_{A,M}$ and $B_{E,M}$, is given by dividing the angular range of the antenna (e.g., for three-sectorized antennas, the angular range in the azimuth plane is 120°) by the angular resolution $\phi_{B,M}$.

The effective beam gains $G_{A,vb}$ and $G_{E,vb}$ experienced by vehicle v from beam b in the azimuth and elevation planes, respectively, are given by [31]

$$\begin{aligned} G_{A,vb} &= -\min\left\{12\left(\frac{\varphi_{vb}}{\Delta\varphi_M}\right)^2, \mathrm{FBR}_M\right\} + G_{M,\mathrm{MAX}}, \\ G_{E,vb} &= \max\left\{-12\left(\frac{\vartheta_{vb}}{\Delta\varphi_M}\right)^2, \mathrm{SLL}_M\right\}, \end{aligned} \tag{A3}$$

where φ_{vb} is the angle off the boresight direction of beam b, ϑ_{vb} is the negative elevation angle relative to the direction of beam b, FBR_M is the front back ratio in dB, $G_{M,\mathrm{MAX}}$ is the maximum gain in dBi and SLL_M is the side lobe level in dB, relative to the maximum gain of the main lobe. The maximum beam gain $G_{M,\mathrm{MAX}}$ is given, in dBi, by

$$G_{M,\mathrm{MAX}} = 10\log(N_{E,M}) + G_E. \tag{A4}$$

The side lobe gain $G_{S,M}$ of each beam for the UPRA is given, in dBi, by [29]

$$G_{S,M} = 10\log\left(\frac{\sqrt{N_{E,M}} - \frac{\sqrt{3}}{2\pi}N_{E,M}\sin\left(\frac{\sqrt{3}}{2\sqrt{N_{E,M}}}\right)}{\sqrt{N_{E,M}} - \frac{\sqrt{3}}{2\pi}\sin\left(\frac{\sqrt{3}}{2\sqrt{N_{E,M}}}\right)}\right). \tag{A5}$$

Therefore, the side lobe level is given, in dBi, by

$$\mathrm{SLL}_M = G_{M,\mathrm{MAX}} - G_{S,M}. \tag{A6}$$

We further assume [29] that $\mathrm{SLL}_M = -\mathrm{FBR}_M$. The total beamforming gain at vehicle v from beam b is then given, in dB, by

$$G_{M,vb} = G_{A,vb} + G_{E,vb}. \tag{A7}$$

For the transmission beams formed at the vehicles, we assume that they can be steered to the direction of the serving base station beam b. Therefore, the vehicle antenna gain $G_{V,vb}$ is equal to the maximum vehicle beam gain $G_{V,\mathrm{MAX}}$, given in dBi by

$$G_{V,vb} = G_{V,\mathrm{MAX}} = 10\log(N_{E,V}) + G_E. \tag{A8}$$

Appendix A.2. Cell Edge Beam Downtilt

This section derives the required beam downtilt to ensure coverage at the cell edge. Based on the angle of the beam on the azimuth and elevation planes, a certain area on the ground can be served. A simplified model to calculate the area on the ground which is served by a beam is to assume a trapezoid ground area, whose size depends on the half-power beamwidth $\Delta\varphi_M$ and the downwards elevation angle θ of the beam [32]. Figure A1 illustrates on the left hand side the azimuth projection of the trapezoid ground area, traced with a blue color. The right-hand side of Figure A1 shows with a solid blue line the direction θ of a given beam and the distances d_1 and d_2 defining the size of the trapezoid. Specifically, distance d_1 denotes the distance from the base station until the small base of the trapezoid, and distance d_2 denotes the distance from the base station until the big base of the trapezoid. Both the distances d_1 and d_2 are a function of the beam direction θ, which is constrained as $\frac{\Delta\varphi_M}{2} \leq \theta \leq 90^\circ - \frac{\Delta\varphi_M}{2}$. Additionally, the angles θ_{d_1} and θ_{d_2}, which define the distances d_1 and d_2, are equal to $\theta_{d_1} = \theta + \frac{\Delta\varphi_M}{2}$ and $\theta_{d_2} = \theta - \frac{\Delta\varphi_M}{2}$. Then, the distances d_1 and d_2, for the range $\frac{\Delta\varphi_M}{2} \leq \theta \leq 90^\circ - \frac{\Delta\varphi_M}{2}$, are given in meters by

$$d_1 = \frac{h_M}{\tan\left(\theta + \frac{\Delta\varphi_M}{2}\right)}, \tag{A9}$$

$$d_2 = \frac{h_M}{\tan\left(\theta - \frac{\Delta\varphi_M}{2}\right)}. \tag{A10}$$

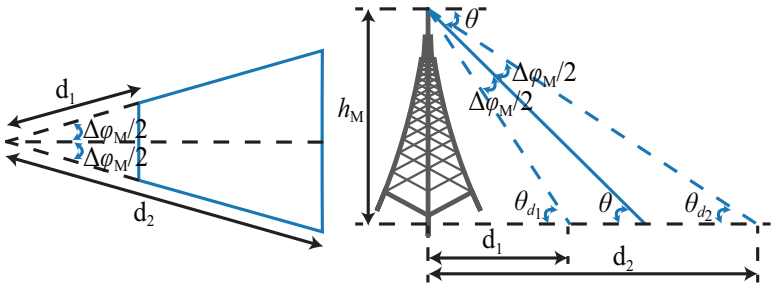

Figure A1. The trapezoid ground area served by a beam with a downwards elevation angle θ and with half-power beamwidth $\Delta\varphi_M$, is given by the distances d_1 and d_2.

The cell edge is served by the beam with the smallest downtilt in the elevation plane. Using the method of the trapezoid ground area, we approximate the downtilt, denoted as θ_{CE}, of the beam serving the cell edge. Using a hexagonal layout, the distance d_c from the base station until the cell edge is given by

$$d_c = \frac{2}{3}\mathrm{ISD}, \tag{A11}$$

where ISD is the inter-site distance. Setting $d_2 = d_c$ in (A10), the downtilt θ_{CE} is given by

$$\theta_{\text{CE}} = \arctan\left(\frac{h_M}{d_c}\right) + \frac{\Delta\varphi_M}{2}. \tag{A12}$$

Appendix A.3. Antenna Array Configuration

We consider 4×4 UPRAs for the base stations and thus $N_{E,M} = 16$. Then, the half-power beamwidth $\Delta\varphi_M \approx 25^\circ$ and the first-null beamwidth $\varphi_{0,M} = 60^\circ$, based on (A1) and (A2), respectively. Therefore, the angular resolution $\phi_{B,M} = \varphi_{0,M}/2 = 30^\circ$ in both the azimuth and elevation planes.

Considering three-sectorized cells, the antenna arrays need to cover a range of 120° in the azimuth plane. Consequently, there are $B_{A,M} = 120^\circ/30^\circ = 4$ beams in the azimuth plane. On the left hand side of Figure A2, the four beams covering the azimuth plane are shown, pointing at angles -45°, -15°, 15° and 45°.

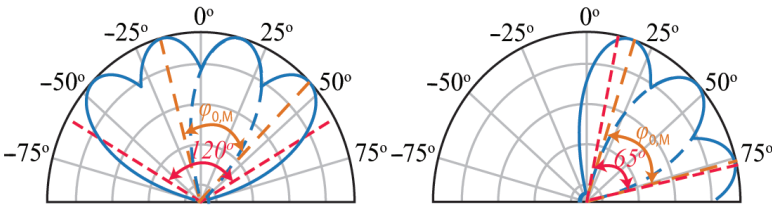

Figure A2. Direction of beams in the GoB for an 4×4 UPRA, in the (**left**) azimuth and (**right**) elevation planes.

In the elevation plane, the antenna array needs to cover the vehicles on the ground and thus cover a range of θ less than 90°. Because each base station needs to cover a specific area on the ground, the beams need to point close to the cell edge to ensure that they do not introduce significant interference to the adjacent cells. Based on an ISD = 500 m and antenna height $h_M = 25$ m, the angle to the cell edge is equal to 4.3°. Therefore, the beams in the elevation plane need to cover a range of 85.7°. Consequently, there are $B_{E,M} = \lceil 85.7^\circ/30^\circ \rceil = 3$ beams in the coverage range. The beam direction for the cell edge is given by (A12) and thus, the cell edge beam is at $\theta_{\text{CE}} \approx 17^\circ$. Based on the angular resolution $\phi_{B,M} = 30^\circ$, the other two beam directions are 47° and 77°. The right hand side of Figure A2 illustrates the three beams and their direction.

Based on the derived beam directions, the UPRA at the base stations can simultaneously form and transmit $B_M = B_{A,M}B_{E,M} = 12$ beams. Furthermore, using (A4), the maximum beam gain $G_{M,\text{MAX}} = 12$ dBi and using (A5) and (A6), we calculate $\text{FBR}_M = \text{SLL}_M = 19.1$ dB.

Regarding the vehicles, they are equipped with a 2×2 UPRA and thus $N_{E,V} = 4$. Thus, the half-power beamwidth $\Delta\varphi_V \approx 50^\circ$, as given by (A1). Finally, using (A8), the maximum beam gain $G_{V,\text{MAX}} = 6$ dBi.

Appendix A.4. Beam Connection Time

A vehicle stays connected to the same beam when moving in the area covered by the given beam. The area served by a beam can be calculated using (A9) and (A10) when the half-power beamwidth $\Delta\varphi_M$ is substituted by the angular resolution $\phi_{B,M}$. Figure A3 shows the ground coverage area of each of the three elevation beams calculated in Appendix A.3 for a given azimuth direction. Each beam coverage area is a trapezoid and it is described by the distances d_1 and d_2, as defined in Appendix A.2. Hence, the two bases and the height of the trapezoid can be calculated. Table A1 shows the length of the distances d_1 and d_2 as

well as the length of the long base and the height of the trapezoid, considering the angular resolution $\phi_{B,M} = 30°$ (The distance d_2 for the beam pointing to the cell edge was set equal to the cell edge distance d_c, as given in (A11).).

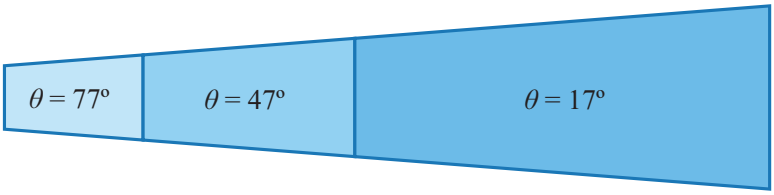

Figure A3. Ground coverage area for every beam in the elevation plane (not in scale), at a given azimuth direction.

Table A1. Geometry description of the trapezoid coverage areas and beam connection time derivation, for the given beam directions.

Beam Direction θ	Distance d_1 [m]	Distance d_2 [m]	Long Base [m]	Height [m]	Maximum Connection Time [s]
16.8°	40.3	333.3	172.5	283.0	10.4–17.0
46.8°	13.4	40.3	20.9	26.0	1.3–1.6
76.8°	0.0	13.4	6.9	12.9	0.4–0.8

Table A1 also shows a rough approximation of up to how much time a vehicle will stay connected to the same beam, which depends on the elevation angle of the beam. The maximum connection time interval is calculated as the length of the long base and height of the trapezoid divided by the speed of the car, which is assumed to be 60 km/h.

Appendix A.5. Broadcast Bit Rate

The broadcast bit rate is chosen such that the associated SNR requirement can be met. In this work, we associate the broadcast bit rate to the cell edge bit rate and hence to the cell edge SNR. We define the cell edge SNR as

$$\Gamma_{\mathrm{CE}} = P_B + G_T + G_{V,\mathrm{MAX}} + G_M - P_{\mathrm{NOISE}} - P_{\mathrm{NF},V}, \tag{A13}$$

where P_B denotes the broadcast beam transmit power and $P_{\mathrm{NF},V} = 9$ dB is the noise figure at the vehicles [27]. Assuming that all four beams at each cell are used for the broadcast and that the maximum transmit power $P_{M,\mathrm{MAX}} = 49$ dBm is equally shared among the four beams, we calculate the power $P_B = 43$ dBm. The distance to the cell edge is $d_c = 333$ m and thus the path gain $G_T = -137.8$ dB. Moreover, from Appendix A.3, the maximum beam gain $G_{V,\mathrm{MAX}} = 6$ dBi. For the transmit antenna gain G_M, we assume the worst case scenario in which the angle off the boresight direction of the beam is equal to $\varphi = \frac{\Delta\varphi_M}{2}$ and $\vartheta = \frac{\varphi_{0,M}}{2}$ in the elevation and azimuth planes, respectively. Using (A7), the transmit antenna gain $G_M = 6$ dB. The noise power is given by

$$P_{\mathrm{NOISE}} = -174 + 10\log_{10}(B), \tag{A14}$$

and for a bandwidth of $f_{\mathrm{BW}} = 50$ MHz, the noise power $P_{\mathrm{NOISE}} = -97$ dBm. Then, using (A13), the cell edge SNR $\Gamma_{\mathrm{CE}} = 5.2$ dB. Therefore, the broadcast bit rate is calculated using (6) as 105 Mbps.

References

1. Balkus, S.V.; Wang, H.; Cornet, B.D.; Mahabal, C.; Ngo, H.; Fang, H. A survey of collaborative machine learning using 5G vehicular communications. *IEEE Commun. Surv. Tutor.* **2022**, *24*, 1280–1303. [CrossRef]
2. Zhang, H.; Bosch, J.; Olsson, H.H. End-to-end federated learning for autonomous driving vehicles. In Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, 18–22 July 2021. [CrossRef]
3. Du, Z.; Wu, C.; Yoshinaga, T.; Yau, K.L.A.; Ji, Y.; Li, J. Federated learning for vehicular internet of things: Recent advances and open numbers. *IEEE Open J. Comput. Soc.* **2020**, *1*, 45–61. [CrossRef] [PubMed]
4. Lim, W.Y.B.; Luong, N.C.; Hoang, D.T.; Jiao, Y.; Liang, Y.C.; Yang, Q.; Niyato, D.; Miao, C. Federated learning in mobile edge networks: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2031–2063. [CrossRef]
5. Zhang, H.; Bosch, J.; Olsson, H. Real-time End-to-End Federated Learning: An Automotive Case Study. In Proceedings of the 2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC), Madrid, Spain, 12–16 July 2021. [CrossRef]
6. Raftopoulou, M.; da Silva, J.M.B., Jr.; Litjens, R.; Poor, H.V.; Van Mieghem, P. Agent selection framework for federated learning in resource-constrained wireless networks. *IEEE Trans. Mach. Learn. Commun. Netw.* **2024**, *2*, 1265–1282. [CrossRef]
7. Hellström, H.; da Silva, J.M.B., Jr.; Amiri, M.M.; Chen, M.; Fodor, V.; Poor, H.V.; Fischione, C. Wireless for Machine Learning: A Survey. *Found. Trends Signal Process.* **2022**, *15*, 290–399. [CrossRef]
8. Chen, M.; Yang, Z.; Saad, W.; Yin, C.; Poor, H.V.; Cui, S. A joint learning and communications framework for federated learning over wireless networks. *IEEE Trans. Wireless Commun.* **2021**, *20*, 269–283. [CrossRef]
9. Zeng, Q.; Du, Y.; Huang, K.; Leung, K.K. Energy-Efficient Radio Resource Allocation for Federated Edge Learning. In Proceedings of the 2020 IEEE International Conference on Communications Workshops (ICC Workshops), Dublin, Ireland, 7–11 June 2020. [CrossRef]
10. Shi, W.; Zhou, S.; Niu, Z.; Jiang, M.; Geng, L. Joint Device Scheduling and Resource Allocation for Latency Constrained Wireless Federated Learning. *IEEE Trans. Wireless Commun.* **2021**, *20*, 453–467. [CrossRef]
11. Fan, K.; Chen, W.; Li, J.; Deng, X.; Han, X.; Ding, M. Mobility-Aware Joint User Scheduling and Resource Allocation for Low Latency Federated Learning. In Proceedings of the 2023 IEEE/CIC International Conference on Communications in China (ICCC), Dalian, China, 10–12 August 2023. [CrossRef]
12. Deveaux, D.; Higuchi, T.; Uçar, S.; Wang, C.H.; Härri, J.; Altintas, O. On the orchestration of federated learning through vehicular knowledge networking. In Proceedings of the 2020 IEEE Vehicular Networking Conference (VNC), New York, NY, USA, 16–18 December 2020. [CrossRef]
13. Guan, Z.; Wang, Z.; Cai, Y.; Wang, X. Deep reinforcement learning based efficient access scheduling algorithm with an adaptive number of devices for federated learning IoT systems. *Internet Things* **2023**, *24*, 100980. [CrossRef]
14. 3GPP. About 3GPP. 2024. Available online: https://www.3gpp.org/about-us (accessed on 1 April 2025).
15. Janocha, K.; Czarnecki, W.M. On Loss Functions for Deep Neural Networks in Classification. *arXiv* **2017**, arXiv:1702.05659. [CrossRef]
16. Murphy, K.P. *Probabilistic Machine Learning: An introduction*; MIT Press: Cambridge, MA, USA, 2022.
17. McMahan, H.B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017; pp. 1273–1282.
18. Dahlman, E.; Parkvall, S.; Sköld, J. *5G NR: The Next Generation Wireless Access Technology*, 1st ed.; Elsevier Academic Press: Cambridge, MA, USA, 2018.
19. 3GPP. TR 38.913; 5G; Study on Scenarios and Requirements for Next Generation Access Technologies; 3GPP Technical Report; v.17; 2022. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2996 (accessed on 1 April 2025).
20. Goldsmith, A. *Wireless Communications*; Cambridge University Press: Cambridge, UK, 2005.
21. Zeng, Q.; Du, Y.; Huang, K.; Leung, K.K. Energy-efficient resource management for federated edge learning with CPU-GPU heterogeneous computing. *IEEE Trans. Wireless Commun.* **2021**, *20*, 7947–7962. [CrossRef]
22. Forrest, J.; Ralphs, T.; Santos, H.G.; Vigerske, S.; Forrest, J.; Hafer, L.; Kristjansson, B.; jpfasano; EdwinStraver; Lubin, M.; et al. coin-or/Cbc. 2023. Available online: https://zenodo.org/records/7843975 (accessed on 1 April 2025). [CrossRef]
23. Serna, C.G.; Ruichek, Y. Classification of traffic signs: The European dataset. *IEEE Access* **2018**, *6*, 78136–78148. [CrossRef]
24. Chilamkurthy, S. Keras Tutorial-Traffic Sign Recognition. 2017. Available online: https://chsasank.com/keras-tutorial.html (accessed on 1 April 2025).
25. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556. [CrossRef]
26. Nishio, T.; Yonetani, R. Client selection for federated learning with heterogeneous resources in mobile edge. In Proceedings of the 2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019. [CrossRef]

27. 3GPP. TR 37.885; Study on Evaluation Methodology for New Vehicle-to-Everything (V2X) Use Cases for LTE and NR; 3GPP Technical Report; v.15.3.0; 2019. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3209 (accessed on 1 April 2025).
28. Raftopoulou, M.; da Silva, J.M.B., Jr. FLoverWireless: System-Level Simulator for FL over Wireless Networks. 2024. Available online: https://zenodo.org/records/12506109 (accessed on 1 April 2025). [CrossRef]
29. Venugopal, K.; Valenti, M.C.; Heath, R.W. Device-to-device millimeter wave communications: Interference, coverage, rate and finite topologies. *IEEE Trans. Wireless Commun.* **2016**, *15*, 6175–6188. [CrossRef]
30. Balanis, C.A. *Antenna Theory: Analysis and Design*, 4th ed.; John Wiley and Sons Inc.: Hoboken, NJ, USA, 2016.
31. Gunnarsson, F.; Johansson, M.N.; Furuskar, A.; Lundevall, M.; Simonsson, A.; Tidestav, C.; Blomgren, M. Downtilted base station antennas—A simulation model proposal and impact on HSPA and LTE performance. In Proceedings of the 2008 IEEE 68th Vehicular Technology Conference, Calgary, AB, Canada, 21–24 September 2008. [CrossRef]
32. Lin, B.; Wang, W.; Guo, J.; Fei, Z. Outage performance for UAV communications under imperfect beam alignment: A stochastic geometry approach. In Proceedings of the 2021 IEEE 21st International Conference on Communication Technology (ICCT), Tianjin, China, 13–16 October 2021. [CrossRef]