DNN-Controlled Multi-Technology Platooning

Christian Quadri[1], Manuel Dileo[1], Vincenzo Mancuso[2,3], Marco Ajmone Marsan[2]
[1]*Computer Science Department*, *Università degli Studi di Milano*, Milan, Italy
[2]*IMDEA Networks Institute*, Madrid, Spain, [3]*Università degli Studi di Palermo, Palermo, Italy*

***Abstract*—We look at platooning applications that leverage two wireless technologies for the coordination of vehicle movements. The first technology is DSRC (dedicated short-range communications), based on IEEE 802.11p, which allows a distributed implementation of the platoon coordination. The second technology is 5G (and beyond). In the latter case, the platoon control algorithm is centralized in an edge computing facility close to the platoon (or in the cloud, if latency allows). Trying to maximize platoon performance and overcome the unpredictability of the radio channel, both radio access technologies are simultaneously active, and a deep neural network (DNN) is used to decide which of the two should be relied on for platoon control, at each point in time. The proposed platooning architecture is compared against previously proposed alternatives, investigating performance with detailed simulation tools. Results show significant advantages in terms of accuracy and safety in inter-vehicle distance for all vehicles within the platoon.**

***Index Terms*—Platooning; DSRC: 5G; DNN; Simulation.**

I. Introduction

Driving together at tight and controlled inter-vehicle distance through platooning is realized by relying on connected autonomous vehicles (CAVs) that use wireless communications to exchange sensor data. Traditionally, platooning systems rely on dedicated short-range communications (DSRC) for managing the platoon in a distributed fashion [1]. In particular, the ETSI ITS-G5 standard, based on 802.11p, allows vehicles to exchange cooperative awareness messages (CAMs) carrying their motion information (e.g., speed, acceleration, position, and maneuver intentions) to other platoon members. Upon receiving a CAM, each vehicle evaluates a control law that provides the desired longitudinal acceleration to maintain platoon stability.

With the emergence of 5G radio access networks (RANs), centralized approaches based on cellular communications and multi-access edge computing (MEC) become possible [2]. Some recent works [3], [4] have shown the feasibility and the benefits of edge-assisted platooning, overcoming limitations of DSRC such as limited radio coverage, medium access contention, shadowing, and multi-hop forwarding for managing long platoons. In spite of the advantages of centralized platoon management, the MEC-based approach is completely dependent on the mobile RAN, and even short periods without radio coverage can lead to severe consequences. In addition, communications through the RAN and backhaul network may suffer unexpected long delays due to temporary traffic peaks.

As a result, designing a platooning system that relies on a single radio access technology (RAT) is likely not to meet the reliability and safety requirements that must be guaranteed for the entire duration of the platoon journey, regardless of the level of quality of service (QoS) offered by the specific RAT. This is especially true in the vehicular context, where network conditions are continuously changing and there can be situations in which some RATs cannot provide a suitable level of QoS. Hence, vehicles need a set of instruments to measure how much they can *"trust"* a platoon control system (PCS) based on a specific RAT in the short-term future, to be able to select the most suitable platoon control, or even disable the platooning driving mode when safety is at risk.

In this work, we discuss the effectiveness of an onboard system that exploits a deep neural network (DNN) for the prediction of the short-term reliability of the PCSs available to the vehicle. In the inference, we consider both the kinematic and communication contexts of the vehicle. In particular, we use measurements directly collected on the vehicle, e.g., speed, acceleration, 5G radio channel quality, and DSRC channel contention time, as well as the age of information (AoI) of data coming from other vehicles/edge servers. The goal is to provide the vehicle with the ability to choose which PCS to use in the next few actions.

As metrics for the estimation of the effectiveness of the operation of the PCS, we consider the divergence of the platoon control instruction with respect to the ideal instruction that would be computed with a perfect zero-latency and lossless RAT. However, it must be noted that a coupling exists between kinematic and communication contexts. The same level of network QoS can have very different impacts on the platoon behavior, depending on its kinematics context. For example, strong acceleration/deceleration phases are very sensitive to network delays, while cruising periods at quasi-constant speed can tolerate higher latencies.

The contributions of this work are the following.

1) We define a partly distributed multi-RAT PCS monitoring and decision system (PMDS), which operates under the control of a DNN;
2) We show that a proactive approach can be very effective in choosing the control actions to be implemented in the short-term future, thanks to DNN predictions;
3) We study the proposed platooning approach with very detailed simulations based on tools that have gained credibility in the vehicular domain;

This work was supported by project *FORESEEN* (Grant: P2022WYAEW, CUP: G53D23006720001) under MUR NRRP (Call PRIN 2022 PNRR) and by project *SERICS* (PE00000014, CUP: G43C22002580001) under the MUR NRRP, funded by the European Union NextGenerationEU.

This paper has been also funded by project PID2022-140560OB-I00 (DRONAC) funded by MICIU/AEI /10.13039/501100011033 and ERDF, EU.

4) We compare our proposal against existing alternatives, and we show that it provides better performance in terms of inter-vehicle distance discrepancies with respect to those that would be obtained with perfect information.

II. Related Work

The combination of RATs (from WiFi to cellular to LiFi) to improve performance has already been widely investigated in vehicular networking and was demonstrated to provide significant gains in terms of data rate, robustness to interference, and link reliability [5], [6], [7].

Sepulcre *et al.* [8] proposed a distributed and decentralized algorithm that allows each vehicle to autonomously and dynamically select the most adequate communication technology. The proposed algorithm maximizes the network capacity while satisfying the application requirements. The communication technology options are limited to V2V paradigms, considering different portions of radio spectrum (DSRC, WiFi, TV White Spaces). Jacob *et al.* [9] investigated a combination of IEEE 802.11p and LTE-V2V (Mode 4) to improve the reliability of cooperative automated driving applications. To increase the packet reception probability, multiple copies of the same packet are sent over different RATs. Segata *et al.* [10] presented *SafeSwitch*, an onboard system specifically designed for platooning, which determines the reliability of multi-RAT by constantly monitoring the packet delivery ratio (PDR) of three distributed communication technologies, IEEE 802.11p, VLC, and LTE C-V2X (Mode 3). Moreover, SafeSwitch suspends cooperative driving if no RATs provide a suitable level of PDR. Yacheur *et al.* [11] proposed a decentralized RAT selection strategy that uses Deep Reinforcement Learning aiming to limit resource consumption and channel load, offering reliable and high throughput communication. Differently from our work, the solution is not tailored to platooning and considers V2V approaches only, ITS-G5 and LTE-V2X (Mode 3). Recently, [12] proposed a dual-link-enabled vehicular network, to support inter-vehicular communication exploiting both DSRC and LTE (Uu). The proposed approach optimizes the traffic steering between the two technologies based on the network load in both DSRC and LTE cellular networks.

Different from previous works, we combine DSRC and 5G communication technologies to design an onboard PMDS that monitors and selects actions computed by multiple available PCSs, tailored to platoon longitudinal control. The reliability of the two PCSs based on either DSRC or 5G-Edge is measured w.r.t. their direct impact on platoon performance, rather than communication QoS parameters. Finally, by predicting the reliability of platooning systems, our solution aims to prevent the actuation of incorrect instructions.

III. System model

We consider a platoon of vehicles in which each member is equipped with two radio interfaces: DSRC (802.11p) and cellular (5G), as depicted in Fig. 1. We assume that the platoon was formed before the time window we consider and that no vehicle joins or leaves the platoon during the observation period. Each vehicle relies on two parallel and independent

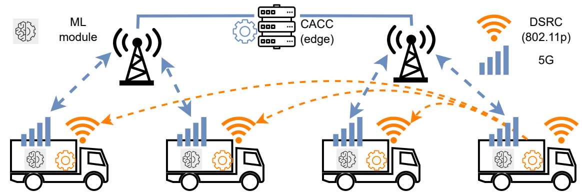

Fig. 1: ML assisted Multi-RAT platoon.

PCSs. One PCS is the traditional distributed platooning control, in which each vehicle computes a control law to adjust its acceleration, using data read from its onboard sensors and those received from the other vehicles through V2V communication. The other PCS delegates the computation of the control law to an edge controller that receives sensors' data via the mobile radio network and sends back acceleration instructions to vehicles, e.g., as described in [3]. Hence, we describe such PCS as based on '5G-Edge' (the controller could also be in the cloud, if the latency between controller and vehicles is within few tens of milliseconds [3]). Both PCSs use the Cooperative Adaptive Cruise Control (CACC) [13] control law with the same parameters. The control law adopts a constant space policy and the leader–predecessor–follower topology, i.e., it computes the desired acceleration for each vehicle using the data of the vehicle itself, the one provided by the preceding and leader vehicles, with the target of maintaining a fixed distance between each pair of adjacent vehicles.

In addition to PCSs, each vehicle is also equipped with a machine learning (ML) module for the evaluation of the reliability of PCSs. Each vehicle performs the evaluation locally, using only pieces of information that are available onboard, without cooperating with other platoon members. Based on the evaluation performed by the ML module, a vehicle can decide whether or not to actuate the instructions computed by either PCS, with the objective of preventing unstable and dangerous situations. In particularly critical contexts, when no PCS is believed to provide suitable reliability, a vehicle can temporarily adopt a non-cooperative control law, e.g., Adaptive Cruise Control (ACC). We remark that the decision of which PCS to use or to temporarily suspend the cooperative control is taken individually and independently by each platoon member. This design choice guarantees a secure execution environment, enforcing data protection and privacy. Moreover, in this work, we assume that the local decision is not broadcast to other platoon members.

IV. Reliability of Platooning Instructions

Executing platooning instructions blindly can have serious consequences on the safety of vehicles and passengers. It is therefore critical that vehicles can trust the results of the adopted control law. To that purpose, it is important to render vehicles capable of evaluating the quality and correctness of received data and instructions, which depends on network issues and delays, as we discuss below.

A. Impact of Network Performance

Under ideal communications conditions, the CACC control law guarantees two properties. The first is *individual vehicle* *stability*, according to which, vehicle spacing errors converge to zero if the platoon leader speed is constant. The second is *string stability*, according to which, spacing errors do not amplify as they propagate towards the tail of the platoon [13].

Kinematic properties of vehicles are a source of uncertainty, but existing control laws are flexible enough to support a wide variety of vehicles and engines, under ideal conditions or in the presence of limited noise [13]. However, under non-ideal communications conditions in which link quality can degrade significantly, at least some of the vehicles might receive data that leads to inaccuracy in platoon control. In the case of DSRC, vehicles may miss timely updates of the input of the control law, and compute inaccurate control decisions. In the case of 5G-Edge, vehicles may fail to receive platoon control commands or receive them when it is too late. In these contexts, it is natural to resort to the evaluation of performance metrics that can quantify the quality of the received data for each of the available communication technologies. A possible performance indicator is the packet delivery ratio (PDR), i.e., the fraction of packets that are not lost due to communication errors. Another metric is the Age of Information (AoI), that represents the freshness of data, which in our case are used for platoon control [14]. High values of AoI lead to the use of outdated information, which might jeopardize performance and safety. In this paper we consider both PDR and AoI.

The link quality is affected by various factors strictly related to the radio technology used by the platooning system. For example, a DSRC-based PCS is affected by limited communication range, co-channel interference, uncoordinated channel access, and shadowing effects [15], [16]. Instead, the PCS relying on 5G-Edge suffers from unstable channel conditions due to vehicle mobility and poor radio coverage areas. Besides, both technologies suffer when approaching congestion.

B. Instruction reliability measure

To provide the vehicle with a PCS reliability measure, we consider as reference the ideal case of instantaneous lossless data transfer. With DSRC, this means that the control law is computed onboard, using data instantaneously read from the vehicles' sensors. With 5G, the ideal case implies that data are transferred from vehicles to the edge computing facility with no delay and that the platoon control instructions are computed at the edge in zero time, and reach vehicles with no delay. This represents the best possible operating conditions for the platoon control law, in which all the AoI values are zero and the actuation lag is the only delay component in the system, which is already taken into account in the design of the control law [13]. The control law operating under these conditions always provides the best instruction, a^*, guaranteeing both individual vehicle and string stability properties.

Unfortunately, this is not what happens in practice. Every PCS relying on network communications introduces delays and possibly also losses, hence leading to platoon control instructions that differ from the ideal case. Intuitively, the larger the difference between the instruction provided by the platoon system and the ideal one, the less reliable the PCS.

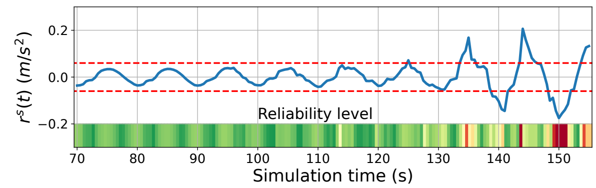

Fig. 2: Platoon system reliability example.

Formally, we define the level of reliability of PCS s (either based on DSRC or 5G-Edge) at time t as follows:

$$r^s(t) = a^s(t) - a^*(t), \tag{1}$$

where $a^s(t)$ is the instruction provided by PCS s at time t and $a^*(t)$ is the ideal instruction at time t. In Fig. 2, we show an example of the evolution of the reliability value over time for the 6th follower in a platoon of 8 vehicles, with a PCS using DSRC in which the leader is traveling with a sinusoidal speed pattern. The red dashed lines represent empirical reliability thresholds ($\pm 0.06 m/s^2$, in figure). In this example, the PCS provides reliable instructions until 130 s. From that time on, the level of reliability is not sufficient, exceeding the threshold multiple times, meaning that the communication system fails to provide a suitable level of QoS to manage the platoon.

Unlike previous works, in which system reliability is based on the evaluation of network parameters that indirectly degrade the platoon performance, e.g., PDR [10], we measure the divergence of the provided instruction w.r.t. the ideal one, that directly affects the platoon performance. Unfortunately, we cannot directly obtain the value of $r^s(t)$, because it is impossible to compute the ideal instruction $a^*(t)$ on board at time t. For this reason, we need to solve a regression task to infer reliability, based on information available on board. Moreover, we aim to predict the reliability of the system in the short-term future, to prevent the vehicle from applying unreliable instructions. Formally, the prediction task to evaluate the reliability of PCS s performed by a vehicle at time t is defined as follows:

$$\langle r^s(t), \ldots, r^s(t+\tau) \rangle = f_{\boldsymbol{\theta}^s}(\langle \mathbf{x}^s(t-\sigma), \ldots, \mathbf{x}^s(t) \rangle) \tag{2}$$

where $\mathbf{x}^s(t)$ represents the vector of the features related to system s observed at time t, σ represents the duration of the window of past observations and τ stands for the span of the prediction window. Finally, $f_{\boldsymbol{\theta}^s}(\cdot)$ represents the predictor function given model parameters $\boldsymbol{\theta}^s$.

V. PCS Monitoring and Decision System (PMDS)

In this section, we describe the onboard PMDS responsible for evaluating the reliability of the available PCSs, thus providing the vehicle with reliable instructions in the near future. Our approach continuously predicts the reliability level of each PCS using a neural network architecture, based on a set of features, organized in time series, extracted from onboard sensors, and checking the status of the radio channel. Given the differences between communication technologies, each vehicle has its onboard component for monitoring and predicting the reliability of the two PCSs. At training time,

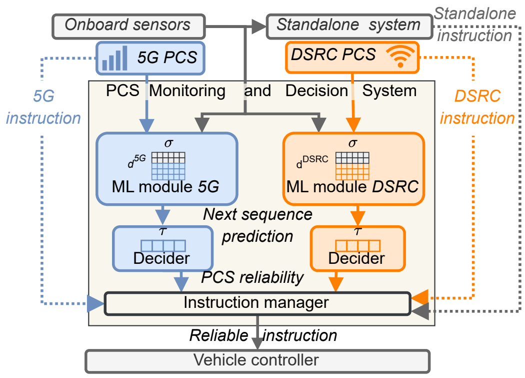

Fig. 3: ML-based onboard PMDS.

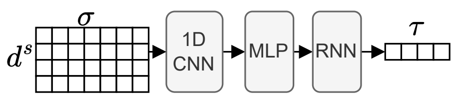

Fig. 4: Neural network architecture.

two different neural network models are used, one for 5G and one for DSRC, relying on both common and technology-specific feature sets. At inference time, each vehicle predicts the reliability level of the two PCSs, based on its own view of the current status of the system. Fig. 3 shows the schema of the proposed onboard PMDS. Starting from data derived from onboard sensors and system status, which, for each PCS s, can be represented as a matrix of d^s features along a time window of span σ, a neural network predicts the next sequence of reliability levels, whose span is equal to τ. Subsequently, a heuristic algorithm decides the next reliability level of the two PCSs, based on the prediction made by the neural network. Finally, the reliability of the PCSs is given to an instruction manager, which is responsible for selecting which PCS to use, i.e., CACC with 5G-Edge, or CACC with DSRC, or neither of those, switching to ACC (we refer to this case as *standalone*).

A. Input features and preprocessing

The feature set used for training the two models includes common kinematic information and system-specific features.

1) Common kinematics features: vehicle acceleration, distance from the preceding vehicle, and relative position of the vehicle within the platoon.

2) 5G-Edge features: channel quality indicator (CQI) in uplink and downlink direction, round-trip time (RTT), and AoI values accounting for the freshness of the status information of the leader, the preceding car and the vehicle itself, as well as the AoI of the platoon controller instruction.[1]

3) DSRC features: MAC layer queueing time for sending the packets, packet drop events, received signal power of the leader and preceding vehicle messages, AoI values of status information of the leader, and of the preceding vehicle.

Due to the features' different and irregular sampling rates, we align them by performing a resampling with a fixed period of 250 ms. The chosen resampling period is longer than the typical PCS control loop, around 100 ms, avoiding missing values under reliable system conditions. However, prolonged instability of the platoon may cause missing values for some features, e.g., a lack of mobile network coverage leads to no data about RTT and AoI values. To handle these cases, we estimate the missing data with linear interpolation and we add extra binary features to record the missing value event.

[1] In the 5G-Edge PCS, AoI values are included in the instruction message.

B. Neural network architecture

Fig. 4 shows the architecture of the neural network. The input consists of matrices of size $d^s \times \sigma$, where d^s represents the number of features for PCS s, and σ is the time window span, i.e., each feature has its time series of σ elements. The architecture is composed of three main components:

- **1D Convolutional Neural Network (1D CNN)**: The first stage of the network employs a 1D CNN for identifying short-term dependencies and feature interactions within the fixed time window σ.
- **Multilayer Perceptron (MLP)**: The intermediate layers of the network consist of an MLP that serves to model higher-level interactions among the input features and project the features in a latent space that represents the current reliability status of the system.
- **Recurrent Neural Network (RNN)**: The final stage leverages a Gated Recurrent Unit (GRU) to output a predicted sequence of size τ, representing the system's reliability levels over the next τ time steps, starting from the current reliability status of the system.

C. Loss function

Given the PCS s, its corresponding ML module can be optimized using gradient descent, minimizing the loss function:

$$L(f(\mathbf{x}), f_{\boldsymbol{\theta}^s}(\mathbf{x})) = \frac{1}{n}\sum_{i=1}^{n}\|f(\mathbf{x}_i) - f_{\boldsymbol{\theta}^s}(\mathbf{x}_i)\|_1 + \alpha\frac{1}{n}\sum_{i=1}^{n}\|f'(\mathbf{x}_i) - f'_{\boldsymbol{\theta}^s}(\mathbf{x}_i)\|_2, \quad (3)$$

where $\mathbf{x}_i$ represents the vector of the features related to the system observed in a time window i, $f_{\boldsymbol{\theta}^s}(\mathbf{x}_i)$ is the next sequence predicted by the model for time window i, and $f(\mathbf{x}_i)$ is the true next sequence after i. The first term represents the average mean absolute error (MAE) between the true and predicted next sequence, which is a standard loss function term for multi-regression tasks. The second term acts as a regularization term on the shape of the predicted sequence, where α controls the strength of the regularization. It is the average mean squared error (MSE) between the values of the derivatives of the next sequence and the learned function. This regularization term prevents the predicted sequence from drastically mismatching the shape of the true reliability function. This mitigates the risk of obtaining a reliability predictor that is close, on average, to the true value but disagrees with the reliability level along the time window.

D. Reliability decision

Based on the prediction of reliability levels provided by the neural network, the *decider* module makes the final decision using a threshold-based heuristic. We compute the absolute value of the predicted reliability level and consider a PCS s reliable if the following conditions are met: (i) the mean of absolute values of reliability levels r^s is below a threshold δ;

TABLE I: Instruction manager system selection.

	5G-Edge	DSRC	System used
Reliability	yes	yes	5G-Edge
	yes	no	5G-Edge
	no	yes	DSRC
	no	no	Standalone

(ii) the average of the subset of values of r^s that exceed δ is lower than a second threshold $\Delta > \delta$. Otherwise, the PCS is considered unreliable. The first condition evaluates overall predicted values, while the second focuses on peak values.

E. Instruction manager

The last step is to provide the vehicle controller with reliable instructions based on the deciders' evaluation. The *instruction manager* is the component of the PMDS that collects the instructions and the reliability evaluation of the PCSs and selects the most appropriate option. Table I shows the system selection based on the evaluation of the reliability of the PCSs. We prioritize the *5G-Edge* PCS over *DSRC* and we select the *Standalone* system only when both PCSs are unreliable. The motivation for giving priority to 5G derives from the fact that, in larger platoons, distances between vehicles may render DSRC links unstable. Lastly, the instruction manager implements a hysteresis mechanism to prevent continuous switching among PCSs.

The PMDS operates at fixed time intervals (0.5 s) and the evaluation is performed using only the input time windows, disregarding the reliability evaluation of the previous step. Considering the hysteresis mechanism and the PMDS operating period, the handover between control systems takes at most 1.5 seconds[2]. Lastly, the fallback to the standalone system is possible at any moment, with no latency, as the instruction is always computed and sent to the instruction manager.

VI. Simulation Setup

We consider a highway scenario with a fleet of 8 light-duty commercial vehicles. We develop a full-fledged simulator framework using OMNeT++ on top of the SUMO simulator [17]. In particular, we rely on Veins [18] and Simu5G [19] to model vehicles and 5G-Edge network, respectively. The ML module is implemented using PyTorch and is integrated with OMNeT++, allowing the simulation of onboard inference. Table II reports the main simulation parameters.

A. Training settings

The training phase is performed independently for the 5G-Edge and DSRC PCSs. To collect meaningful training datasets specific to each PCS, we run simulation campaigns by constructing network scenarios having different levels of criticalities in terms of network coverage, background traffic, and radio interference, using combinations of the parameters reported in Table II. The resulting datasets provide a wide variety of reliability levels for both PCSs.

The hyperparameters of the model and training settings are reported in Table III. In particular, the ML module observes a time window of 5 seconds and predicts the reliability level for the next 5 seconds. The resulting model contains around 77k parameters, requiring less than 1 MB of memory for running both ML models, and is suitable for onboard deployment.

[2]In this work, we perform switches among systems if the reliability conditions hold for at least 3 consecutive evaluation steps.

TABLE II: Simulation parameters

General parameters	
Simulated road	Straight 3-lane highway
Simulation time (repetitions)	300 s (60 s of warm-up time) (10 repeats)
Platoon parameters	
Number of platoon members	8
Leader speed pattern	Sinusoidal 90 km/h ($\pm$ 5 km/h), 0.1 Hz
CACC spacing policy	Constant space (15 m)
ACC spacing policy	Constant ahead time (0.7 s)
Decision system parameters	
Input time window size (σ)	5 s (20 time steps)
Prediction time steps (τ)	5 s (20 time steps)
Decider thresholds (δ, Δ)	$\delta = 0.06 m/s^2$, $\Delta = 0.07 m/s^2$
DSRC configuration	
TX power, Radio sensitivity	20 dBm, -95 dBm
Pathloss model	Rician (k = 8 dB)
Obstacle loss	Model from [16]
Channel band (bandwidth)	5.9 GHz (10 MHz)
Number of RSUs (area size)	0, 10, 20, 30 (250 m x 40 m)
RSUs traffic	3kB, exponential(20 ms)
5G network configuration	
Base station physical resource	3 RBs per TTI (1 ms)
UE Tx power (gain)	26 dBm (+0dBi)
Base station Tx power (gain)	46 dBm (+18dBi)
Carrier frequency	800 MHz, 2100 MHz
Base station model	*ITU-Urban* & *ITU-Rural* macrocell
Pathloss model	Rural: Free Space $\alpha = 2.5$ Urban: Free Space $\alpha = 3.5$
Base station scheduler	Max Channel Indicator
Number of background devices	0, 40 UEs
Packet size (UL/DL)	10, 500 byte
Packet frequency (UL/DL)	20 pkt/s (UPD Constant Bit Rate)
Generation starting/ending time	U(120 s, 150 s) / U(220 s, 250 s)
Congestion-free 5G-Edge RTT	20 $\pm$ 5 ms

TABLE III: Training settings and hyperparameters

1D-CNN	Kernel size: 3, Max-polling size: 3
MLP	Layers: [128, 128, 64], dropout: 0.1, Relu
GRU hidden size	1
Training/Test splitting	80/20, 5-folds Cross-val.
Optimizer (learning rate, batch)	Adam (0.001, 128)

B. Evaluation settings

To evaluate the performance of the proposed ML approach, we design a simulation scenario that incorporates challenging situations for both communication systems. The goal is to define an evaluation scenario in which the vehicles are expected to use all operating modes, including standalone, across the simulation. Fig. 5 shows the schema of the evaluation scenario. We deploy 4 base stations (BSs) in two groups, creating a mobile network coverage hole between the second and the third BS. Moreover, we generate background traffic at the BS level which causes temporary saturation of the physical resources of BSs. The background traffic generation starts approximately when the platoon vehicles are under the coverage of *BS 2* and ends when they are served by *BS 3*. To challenge also the DSRC PCS, we deploy a variable number of roadside units (RSUs) close to the highway that interferes with the platoon communications. While the number of RSUs that we use in experiments may be larger than in most operating conditions, we must consider the possibility of interference from other platoons, or users located in proximity of a urban highway segment.

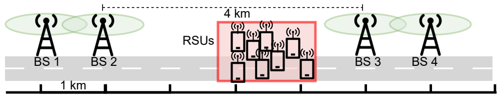

Fig. 5: Evaluation scenario.

We expect that the platoon vehicles operate CACC using the 5G-Edge PCS in the first part and progressively switch to the DSRC PCS due to the combined effect of the background traffic and the weakening of the cellular coverage. Eventually, vehicles should temporarily suspend platooning, adopting the standalone system when traversing the RSUs' area. Later, vehicles are expected to rejoin cooperative driving using DSRC or 5G-Edge depending on the channel conditions. Finally, the 5G-Edge PCS should be adopted until the end of the simulation.

VII. Results

We use simulation experiments to evaluate PCS kinematics performance, PMDS accuracy, and safety.

A. Baseline decision systems

We compare our *ML-based* PMDS approach with two baseline PMDS alternatives, each focusing on just one feature (either PDR or AoI). Both baselines determine the reliability of the PCSs using a simple threshold-based algorithm:

1) PDR-based PMDS (*PDR* for short). This baseline stems from [10]. It monitors the arrival ratio of platoon messages (5G-Edge instructions and DSRC status messages) and considers a PCS reliable if the PDR is above a threshold (we use 0.85, which we observed to provide the best overall results).

2) AoI-based PMDR (*AoI* for short). This approach computes the mean AoI of vehicle status messages and platoon control instructions over a time window. Then it takes the highest mean AoI across the evaluated features and considers a PCS reliable if the result is below 100 ms (as in standards).

Both baselines adopt the same instruction manager policy (see Section V-E) to select the acceleration instruction sent to the vehicle controller.

B. Distance error and platoon stability

One of the main performance indicators of a platoon control system is its ability to guarantee low inter-vehicle spacing error and preserve string stability. In Fig. 6 we report the box plot of the distribution of the distance error w.r.t. the front vehicle, for each follower vehicle, in different mobile network scenarios. We recall that CACC adopts a constant spacing policy, i.e., a fixed inter-vehicle distance, regardless of the traveling speed. In this analysis, we exclude the data points corresponding to periods when vehicles disconnect the platoon and use the *standalone* system because the ACC control law employs a constant ahead-time spacing policy and the platoon coordination is temporarily suspended.

The results show that the *ML-based* PMDS is able to maintain the vehicle spacing error within $\pm$ 1.5 m in all scenarios. Moreover, it preserves the string stability property, with the only minor exception of the *Urban 2100 MHz* environment, in which we can observe a slight increase in the spacing error of the last follower. On the contrary, the *PDR* and *AoI* baselines

TABLE IV: Decision system accuracy (classes percentage)

Rural	800 MHz			2100 MHz		
	Correct	*Missed*	*Hazard*	*Correct*	*Missed*	*Hazard*
ML based	97.4	0.5	2.1	95.1	1.1	3.8
PDR (0.85)	91.0	2.1	6.9	87.2	4.5	8.3
AoI (100ms)	95.0	3.7	1.3	93.2	5.5	1.3
Urban	800 MHz			2100 MHz		
	Correct	*Missed*	*Hazard*	*Correct*	*Missed*	*Hazard*
ML based	94.4	1.4	4.2	92.7	2.2	5.1
PDR (0.85)	88.6	6.3	5.1	82.0	9.2	8.8
AoI (100ms)	91.7	6.7	1.6	90.1	8.2	1.7

fail to maintain a bounded spacing error and to preserve string stability. In particular, the inter-vehicle distance error grows significantly for the last two vehicles. In addition, while AoI exhibits more consistent performance across the mobile network scenarios, the PDR approach worsens its performance under more challenging 5G channel conditions, as we observe in scenarios with a carrier frequency of 2100 MHz.

In Fig. 7, we report the evolution of the distance error in meters versus time in seconds, focusing on the 6th follower. At the bottom of each figure, we also report the PCS used by the vehicle, where the color intensity reflects the agreement across the simulation runs. As we can observe, the overall behavior is what we expect, as described in Section VI-B. We see a first transition from 5G-Edge to DSRC due to the progressive lack of mobile network coverage. This happens around 130 or 140 s in the 800 MHz case, and around 110-120 s in the 2100 MHz case. Then the platooning is temporarily suspended (around 180 s) for a short interval, causing an increase in inter-vehicle distance due to the control law change. Finally, the vehicle rejoins platooning using DSRC or 5G-Edge PCS, according to channel conditions. From the figure, we can observe that *PDR* and *AoI* are more conservative, using the standalone mode for a longer time, causing a later realignment to the target distance. Moreover, the *AoI* baseline switches more often to the DSRC PCS, as it is more sensitive to 5G delay variations.

C. Decision system accuracy

Results presented in the previous section highlight that the three decision systems exhibit distinct and sometimes contrasting operational patterns, leading to different platoon performances. To investigate the accuracy of the decision systems, we classify the decision based on the instantaneous reliability level of the selected technology. More specifically, the technology s is reliable at time t if $|r^s(t)| < 0.06$ m/s^2, (i.e., below the threshold δ used in the PMDS to select the PCS) otherwise it is labeled as unreliable. By combining the reliability of the two radio technologies, we divide the system decision into three classes: *correct*, *missed*, and *hazard*. *Correct* means that the selected technology is reliable or standalone mode is used when no radio technology provides a suitable reliability level. *Missed* refers to missed platooning opportunities, i.e., the standalone mode is used when at least one radio technology is reliable. Finally, *Hazard* refers to decisions that mistakenly select unreliable radio technologies.

In Table IV, we report the percentage of decisions in the three classes in all network scenarios. Our *ML-based* PMDS is the one producing most correct decisions, outperforming base-

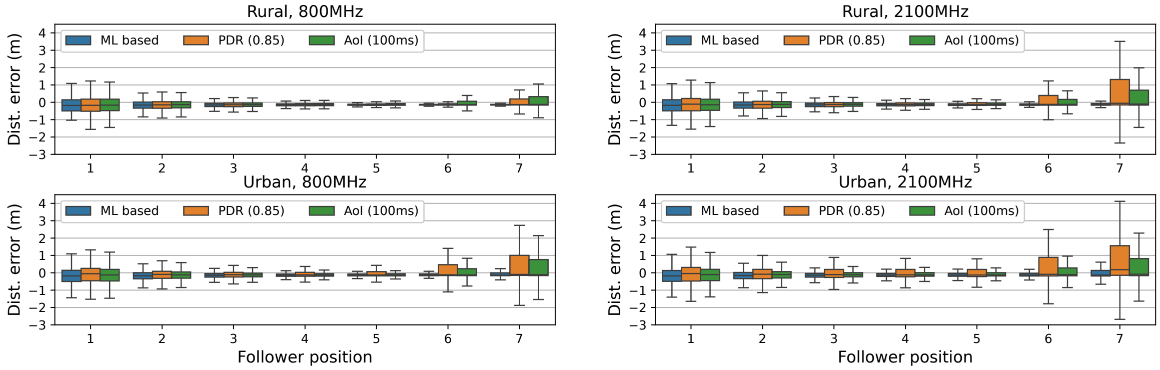

Fig. 6: Box plot of distance error of the whole platoon, only when a platoon system is used (No *Standalone* system).

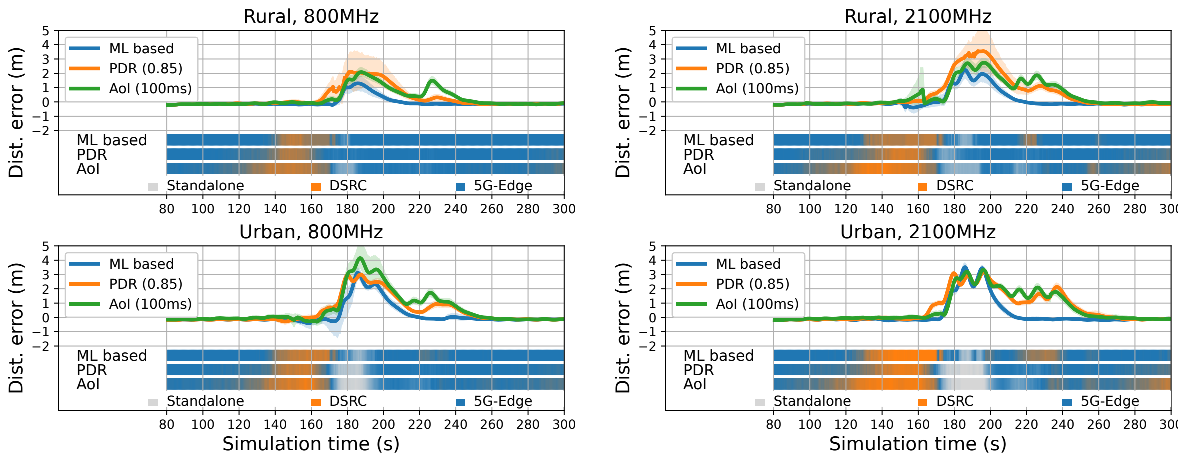

Fig. 7: Temporal evolution of distance error of 6th follower. Areas represent 95% confidence interval.

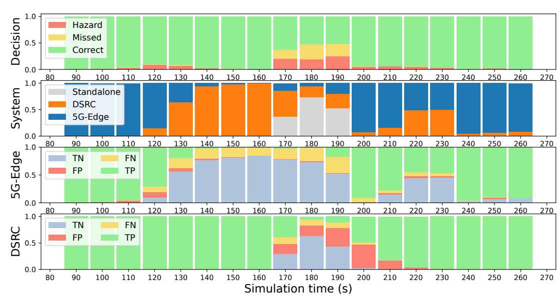

Fig. 8: PMDS outputs of 6th follower (*Urban-2100 MHz*).

lines in all scenarios, with a correct decision rate ranging from 92.7% to 97.4%. It also limits missed platooning opportunities, thus improving platoon efficiency. However, we can observe a non-negligible percentage of hazard decisions, up to 5.1% in the most challenging scenario. *AoI* is the most inefficient approach, as it misses a considerable amount of platooning opportunities, from 3.7% up to 8.2%. *PDR* shows the poorest performance, with the highest rate of hazard choices.

To better understand the hazard decision of the *ML-based* decision system, in Fig. 8 we report the evolution of decision classes over time for the 6th follower in the urban-2100 MHz scenario, along with the technology used and the confusion matrix of the reliability evaluation of PCSs using either 5G-Edge or DSRC, separately. Each bar aggregates 10 seconds of simulated time. What emerges from the figure is that the concentration of hazard decisions between 170-190 s is mainly due to the misclassification of the reliability of the PCS based on DSRC technology, combined with a too-conservative evaluation of the PCS based on 5G-Edge. Differently, the effect of the large number of false positive outcomes of the DSRC-based PCS around 200 s is mitigated by the prioritization of the PCS relying on 5G-Edge rather than DSRC.

Overall, the *ML-based* approach provides the highest rate of correct decisions and better platooning efficiency, to the detriment of hazard decisions in the most challenging conditions, caused by inaccurate evaluation of the reliability of the DSRC PCS, which requires further development and refinements.

D. Platoon safety

The last aspect we analyze is the level of safety that the onboard PMDS can guarantee. For each simulation run, we measure the evolution of the minimum inter-vehicle distance over time across the whole platoon. In Fig. 9, we report the distance error w.r.t. the target distance (15 m) in the urban-2100 MHz scenario. Colored areas represent the range between the 5th and 95th percentiles across all simulation runs. Lines report averages. Results show that the *ML-based* approach offers the highest level of safety with a limited displacement of 3 meters below the target distance at around 170-180 s. On the contrary, *PDR* and *AoI* fail to provide the same level of safety, in particular at the end of the simulation, with *PDR* distance errors falling dangerously below -6 meters. The reason of the poor performance of the baselines can be ascribed to the lack

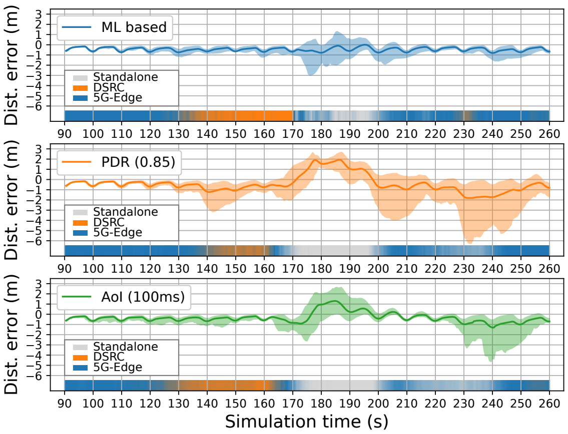

Fig. 9: Safety evaluation (*All followers, urban-2100 MHz*). Areas represent the 5th to 95th percentile interval.

of agreement among the followers about the platoon system to use. We recall that each vehicle makes independent decisions and does not share them with other followers. By focusing on the platoon system used over time, which is indicated with colors at the bottom of each graph, we can observe that vehicles equipped with *PDR* and *AoI* operate using a mix of 5G-Edge and standalone systems (deduced from the low color intensity). The different spacing policies used by CACC and ACC (see Table II) cause perturbations in the platoon, leading to instability and unsafe displacement. Conversely, our *ML-based* PMDS switches between 5G-Edge and DSRC that use the same control law, thus limiting perturbations.

VIII. Conclusions

In this paper, we presented an onboard PCS monitoring ad decision system based on ML, meant to predict the short-term level of reliability of the available PCSs. We proposed a metric for measuring the reliability of a platoon system, which accounts for the divergence of the acceleration instructions provided by the platoon system and ideal ones. The simulation results have shown the effectiveness of the proposed approach, outperforming *PDR* and *AoI* based baselines. In particular, our *ML-based* approach guarantees the highest level of decision accuracy and platooning efficiency. Lastly, the proposed approach exhibits a more consistent behavior among the followers, avoiding extra perturbation along the platoon.

Simulation results are promising, although they highlight minor misclassification errors under challenging radio channel conditions, especially for the reliability of the DSRC-based PCS. However, thanks to the availability of two PCSs in our proposal, infrequent inaccurate predictions do not lead to unsafe situations. Still, we remark that models can be further improved through an additional fine-tuning process which was out of the scope of our work in this paper. Moreover, we foresee that adopting explainable AI techniques could provide a meaningful overview of which features are the most significant for a better design of an ML-based PCS monitoring and decision algorithm, which we leave for future work.

Finally, another key aspect of the monitoring and decision system is the reliability threshold value δ. Tuning or even dynamically adapting its value is not straightforward, and we leave it for future work as well; indeed, it directly affects the platoon performance and even small variations could lead to significantly different platoon behavior. In particular, the selected values should guarantee suitable performance in a wide variety of speed profiles. In this context, the detailed simulation of the vehicle kinematics plays a central role in evaluating the tolerable degree of instruction divergence.

Extensions of this work shall consider the effect of interference in the cellular system, the impact of larger platoon setups and of platoon maneuvers, as well as varied environments, so as to obtain more general conclusions.

References

[1] S. Santini, A. Salvi, A. S. Valente, A. Pescapé, M. Segata, and R. Lo Cigno, "A consensus-based approach for platooning with intervehicular communications and its validation in realistic scenarios," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 3, pp. 1985–1999, 2017.

[2] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1657–1681, 2017.

[3] C. Quadri, V. Mancuso, M. Ajmone Marsan, and G. P. Rossi, "Edge-based platoon control," *Elsevier Com. Com.*, vol. 181, pp. 17–31, 2022.

[4] A. Virdis, G. Nardini, and G. Stea, "A framework for MEC-enabled platooning," in *IEEE WCNCW Workshop*, pp. 1–6, IEEE, 2019.

[5] E. Ndashimye, S. K. Ray, N. I. Sarkar, and J. A. Gutiérrez, "Vehicle-to-infrastructure communication over multi-tier heterogeneous networks: A survey," *Computer Networks*, vol. 112, pp. 144–166, 2017.

[6] A. Zekri and W. Jia, "Heterogeneous vehicular communications: A comprehensive study," *Ad Hoc Networks*, vol. 75-76, pp. 52–79, 2018.

[7] M. A. Altahrawi, N. F. Abdullah, R. Nordin, and M. Ismail, "Multi-radio access software-defined vehicular network," *IEEE Trans. on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 10030–10048, 2022.

[8] M. Sepulcre and J. Gozalvez, "Heterogeneous V2V Communications in Multi-Link and Multi-RAT Vehicular Networks," *IEEE Transactions on Mobile Computing*, vol. 20, no. 1, pp. 162–173, 2021.

[9] R. Jacob, W. Anwar, G. Fettweis, and J. Pohlmann, "Exploiting Multi-RAT Diversity in Vehicular Ad-Hoc Networks to Improve Reliability of Cooperative Automated Driving Applications," in *IEEE VTC2019-Fall*, pp. 1–7, 2019.

[10] M. Segata, R. L. Cigno, T. Hardes, J. Heinovski, M. Schettler, B. Bloessl, C. Sommer, and F. Dressler, "Multi-Technology Cooperative Driving: An Analysis Based on PLEXE," *IEEE Transactions on Mobile Computing*, vol. 22, no. 8, pp. 4792–4806, 2023.

[11] B. Y. Yacheur, T. Ahmed, and M. Mosbah, "Efficient DRL-Based Selection Strategy in Hybrid Vehicular Networks," *IEEE Trans. on Network and Service Management*, vol. 20, no. 3, pp. 2400–2411, 2023.

[12] Z. H. Mir, N. Dreyer, T. Kürner, and F. Filali, "Investigation on cellular lte C-V2X network serving vehicular data traffic in realistic urban scenarios," *Future Generation Comp. Sys.*, vol. 161, pp. 66–80, 2024.

[13] R. Rajamani, *Vehicle dynamics and control*, ch. 7. Springer, 2012.

[14] S. Zhou, S. Li, and G. Tan, "Age of Information in V2V-Enabled Platooning Systems," *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 4015–4028, 2024.

[15] R. Jacob, W. Anwar, N. Schwarzenberg, N. Franchi, and G. Fettweis, "System-level Performance Comparison of IEEE 802.11p and 802.11bd Draft in Highway Scenarios," in *ICT*, pp. 1–6, 2020.

[16] M. Boban, T. T. V. Vinhoza, M. Ferreira, J. Barros, and O. K. Tonguz, "Impact of vehicles as obstacles in vehicular ad hoc networks," *IEEE JSAC*, vol. 29, no. 1, pp. 15–28, 2011.

[17] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using SUMO," in *IEEE ITSC*, 2018.

[18] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10, pp. 3–15, Jan 2011.

[19] G. Nardini, D. Sabella, G. Stea, P. Thakkar, and A. Virdis, "Simu5G–An OMNeT++ Library for End-to-End Performance Evaluation of 5G Networks," *IEEE Access*, vol. 8, pp. 181176–181191, 2020.