

Platoon Grouping Network Offloading Mechanism for VANETs

*Original*

Platoon Grouping Network Offloading Mechanism for VANETs / Kamoi, R. N.; Junior, L. A. P.; Verri, F. A. N.; Marcondes, C. A. C.; GOMES FERREIRA, CARLOS HENRIQUE; Meneguette, R. I.; Cunha, A. M. D.. - In: IEEE ACCESS. - ISSN 2169-3536. - STAMPA. - 9:(2021), pp. 53936-53951. [10.1109/ACCESS.2021.3071085]

*Availability:*

This version is available at: 11583/2932658 since: 2021-10-18T20:10:22Z

*Publisher:*

Institute of Electrical and Electronics Engineers Inc.

*Published*

DOI:10.1109/ACCESS.2021.3071085

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

(Article begins on next page)

Received March 17, 2021, accepted March 27, 2021, date of publication April 5, 2021, date of current version April 13, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3071085

# Platoon Grouping Network Offloading Mechanism for VANETs

ROGER NOBUYUKI KAMOI<sup>1</sup>, LOURENÇO ALVES PEREIRA JÚNIOR<sup>1</sup>,  
FILIPE ALVES NETO VERRI<sup>1</sup>, CESAR AUGUSTO CAVALHEIRO MARCONDES<sup>1</sup>,  
CARLOS HENRIQUE GOMES FERREIRA<sup>2</sup>,  
RODOLFO IPOLITO MENEGUETTE<sup>3</sup>, (Member, IEEE), AND ADILSON MARQUES DA CUNHA<sup>1</sup>

<sup>1</sup>Instituto Tecnológico de Aeronáutica, São José dos Campos 12228-900, Brazil

<sup>2</sup>Department of Computing and Systems, Universidade Federal de Ouro Preto, João Molevade 35931-008, Brazil

<sup>3</sup>Department of Computer Systems, University of São Paulo (USP), São Carlos 13566-590, Brazil

Corresponding author: Lourenço Alves Pereira Júnior (ljr@ita.br)

This work was supported in part by the São Paulo Research Foundation (FAPESP) under Grant #2020/07162-0, and in part by the ITA's Programa de Pós-graduação em Aplicações Operacionais (ITA/PPGAO).

**ABSTRACT** The growth in the number of connected vehicles in ad-hoc networks or VANETs enables the use of increasingly sophisticated services and applications. One emerging application in this scenario is Platooning, a stable stream of connected autonomous vehicles. This application improves safety, vehicle flow on the road, energy consumption, among other variables. However, vehicles using this technology need to share space with vehicles not connected or with other automation levels. Among the bottlenecks in scenarios like this, Platoons formation and splits stand out, as these maneuvers involve costs and risk. Therefore, our contribution is developing a new strategy for Platoons formation in a mixed universe of vehicles to bring some of these desired benefits. The results show that the proposed solution achieved an accuracy of 89% of the entire amount of platoons in the datasets. Our experiments demonstrate cases with collisions, and the proposed solution provides a solution to a more safe operation on roads. Finally, we conclude that our solution provides more determinism in the traffic conditions, as vehicles achieving stationary speeds with low variation.

**INDEX TERMS** Mobile edge computing, vehicular ad-hoc networks, cooperative adaptive cruise control, intelligent transportation systems, smart cities, 5G, beyond-5G.

## I. INTRODUCTION

Vehicular ad-hoc Networks (VANETs) are considered technologies formed by complex systems. Their investigation involves several aspects from protocols, security, and physical layer of communication to final applications of traffic monitoring, services, and leisure [1]. In this way, VANETs allow improvements in safety and traffic management, flow efficiency, vehicle flow, and drivers and passengers' convenience.

VANETs involve communication aspects such as Vehicle-to-Vehicle (V2V) or Vehicle-to-Infrastructure (V2I) [2]. These networks have, as main characteristics: intense production and consumption of local content; anonymity of providers; and collaboration of resources that lead to changes in the computational and network models of VANETs [3]. Besides, the evolution of communication technologies,

mainly involving the new generation of 5G mobile networks, will make these networks more reliable, secure, and ubiquitous, forming Vehicular Cloud Networks [4], [5].

In parallel to VANETs, the evolution of autonomous and semi-autonomous vehicles will make their uses increasingly present and safe [6]. Autonomous vehicles are not necessarily connected, but combining these two technologies enables a series of benefits and applications in urban and road traffic. One of this combination's practical applications is the Platooning method, where vehicles interconnected in a network keep moving in the synchronized queue with constant distances and speeds. Thus, the platoon functionality brings benefits to traffic, increasing safety and improving vehicle flow, avoiding congestion, and saving time and resources.

Platooning represents the coordinated movement of a group of vehicles that follows in the same direction, maintaining the distances between them and the same speeds. This move was originally designed for heavy-duty vehicles and smart highways [7]. However, recent advances in wireless

The associate editor coordinating the review of this manuscript and approving it for publication was Shajulin Benedict<sup>1</sup>.

network communications, automation technologies, sensing devices, and increases in processing capabilities have made it possible to apply Platooning to partially automatic vehicles, such as those equipped with adaptive and cooperative cruise control capabilities, the so-called Cooperative Adaptive Cruise Control (CACC).

The use of the Platooning method and improving the flow of vehicles and their safety can help prevent congestion and provide resource savings, as the smooth and constant movement of vehicles would also reduce fuel consumption and consequent pollution [8]–[10]. In practice, it appears that interconnected vehicles must mutually operate with vehicles that do not yet have these methods and technologies embedded. Therefore, in a universe of vehicles with different technology and automation levels, the application of methods such as Platooning may not yet bring the expected benefits.

Investigations on the Platooning method's application have been carried out under several aspects, such as Market Penetration (MP) when investigating from which percentage of MP Platooning becomes advantageous [11]–[13]. Algorithms or Protocols for the operation and stability of platoons, when investigating the main maneuvers such as junction (merge), lane change and separation (split) between vehicles [7], [14]; or even when investigating the impact of automated vehicles on traffic flow [10], [15]. In more recent work, [16] analyzed the benefits of Platooning at intersections, [17] investigated the impact of the model on junctions on major roads, and [18] and [19] investigated the formation of platoons, analyzing different merge strategies. However, few studies investigate criteria for forming groups of platoons, as proposed in this research.

Dao *et al.* [20] had investigated a strategy for the formation of platoons based on the analysis of the entrances and exits of one-way vehicles, addressing the problem as a linear model, and [21] use the vehicle's speed and position as criteria for selecting platoons treating these characteristics as similarity metrics. In both proposals, the decentralized model for platoons' formation is approached and considered the best. Nevertheless, those models are limited to the range of vehicles and nearby platoons, leaving potential candidates out of the grouping. Moreover, the models were tested without other control models, making them less realistic as automated vehicles' market penetration would be gradual.

Thus, this work aims at developing an architecture to select the best platoon for vehicles entering a road with several lanes. To this end, we use an unsupervised machine learning algorithm (namely, DBSCAN), observing the impact on the flow of vehicles and the road's capacity, where an Edge Computing node designates vehicles to target platoons. We also consider different levels of Market Penetration (MP). Our simulation comprises a map representing a highway with four lanes and the vehicles' flow from real data collected from taxis, the Shenzhen dataset. In this work, we also proposed a new join protocol, which will allow the connection of multiple vehicles on multiple platoons. Therefore the contributions of this work are:

- The analysis of techniques and grouping criteria for the formation of Platoons for their use in machine learning models;
- Improvement of the protocol for the formation of Platoons for join maneuvers (join)
- Development and validation of a real dataset to analyze the performance of the proposed solution

This paper's remainder is structured as follows. We describe the most relevant related solutions in Section II. We present our proposed architecture of platoon in section III. We describe the validation of the propulsion mechanism and its results in Section IV. Finally, we summarize our work and provide direction for future works in Section V.

## II. RELATED WORKS

This section describes the most recent investigations on Platoons regarding their impact on the flow of vehicles and the roads' capacity and their relationship with the Market Penetration (MP) used in this research.

### A. IMPACT OF CACC ON VEHICLE FLOW

The impact of Connected and Automated Vehicles (CAV) on traffic flow are numerous and address this issue in several ways. Rios-Torres and Malikopoulos [22] propose a model and an analytical solution to coordinate the entry of vehicles on the road. They also analyzed the impact of this model on traffic flow [23], where the authors considered it in two scenarios. The first scenario involves 0% MP, that is, only Human-Driven Vehicles (Intelligent Drive Mode - IDM). The second scenario involves 100% MP, all CAV having a parameter of analysis of the flow of vehicles on the roads. In these two studies, vehicle join microscopic traffic, and indicators such as vehicle consumption, travel times, and flow densities are collected, showing that CAVs, especially with a large flow of vehicles, improve traffic flow, mitigating traffic flow congestion. However, the models adopt centralized controls to coordinate vehicle flows and not measured if the current set of vehicles form the best grouping.

Lioris *et al.* [16] analyzed the platoon problem in intersections. The authors adopted approach involves Adaptive Cruise Control (ACC) and Cooperative and Adaptive Cruise Control (CACC), verifying impacts formation Platoons in the crossroads flow. They analyze the queue size and vehicle flow using the ACC and CACC models and conclude that the formation of Platoons can improve vehicles' flow at intersections two to three times. Nevertheless, this work considers a 100% MP for the analysis.

Shladover *et al.* [13] proposed a different approach from these previous works. They consider MP a set of vehicles equipped with CACC, ACC, and manually driven. However, they varied and analyzed them exclusively. We can see in [24]–[27] an effort to collect data from real drivers in vehicles equipped with ACC and CACC technologies. These traces serve as base values for parameterizing models and inputs to microscopic simulation. The results showed that

vehicles equipped with ACC technology did not significantly impact the vehicles' flow. On the other hand, CACC equipped vehicles impacted throughput, increasing from 2,000 to 4,000 vehicles per hour, as the MP ranged from 0% to 100%. Thus, CACC showed as a mechanism to improve traffic quality.

In work of Liu *et al.* [28], addressed a mix of the previous analyzes, based on the parameterizations of [13], which present a more sophisticated and realistic model created to simulate more complex scenarios. As in [23], they add the scenario of an entrance ramp to the model, but the similarity. In [28] also studied the impact of joining an entrance ramp in a scenario with a vehicle chain using CACC technology. They consider different MP values for vehicles that use CACC technology in conjunction with human-driven vehicles, starting to add only more traffic lanes and a lane change algorithm. This study reached a flow rate of around 3,080 vehicles per hour per lane for an MP of 100% of vehicles with CACC. They found that the road's capacity increased quadratically concerning the MP of vehicles with CACC technology. ACC, a chain of vehicles, managed to neutralize waves of movements and stops of vehicles.

Liu *et al.* [29] proposed the Anticipatory Lane Change (ALC) algorithm, improving flexibility and overcoming the shortcomings of the Mandatory Lane Change (MLC) and Discretionary Lane Change (DLC) algorithms. In addition to the ALC maneuver, the authors included the concept of Managed Lane (Managed Lane - ML) as a specific lane for CACC vehicles, which only made sense to be implemented, from a limit of 20% of MP. This article also added a ramp scenario to the model that defines the parameters and the algorithm for carrying out this maneuver. Similar conclusions show an increase of a lower limit of 20% MP with CACC technology. The road's capacity increased in a quadratic way as the MP of the CACC technology also increased. Below this lower limit of 20% MP, the improvement was not significant. Another important conclusion from them was the significant improvement that the ML additions provided in the low MP values. Hence, we consider in our proposal exclusive lanes for CACC as evidence for baseline in road design.

Finally, Kreidieh *et al.* [30] used Reinforcement Learning (Reinforcement Learning - RL) on vehicles using VANETs purely to dissipate the stop waves in a vehicle stream. Based on the concept that even without interference, drivers cannot keep their vehicles' speed and distance from other vehicles constant, creating instabilities in the so-called "walk-and-go wave" in the flow of vehicles, generating congestion, according to [31]. In Kreidieh *et al.* [30] approach, the authors proposed the training of machine learning models with connected autonomous vehicles (CAV) in a simulated environment, assuming a closed network. To this end, the authors used Reinforcement Learning (RL) based on the use of techniques of Markov Decision Problem (MDP) and Transfer Learning (TL). They used the Flow framework [32] as a platform and SUMO for microscopic traffic generators and performed experiments to evaluate RL. By adopting these

techniques, they demonstrated an almost complete dissipation of the "walk-and-go wave" with an MP of only 10%. Moreover, they observed a decrease of those waves' frequency and magnitude after an MP of 2.5%.

## B. FORMATION OF PLATOONS

Studies on the Platoons formation process, that is, on how to direct vehicles to a given Platoon, can be classified according to a centralized [21] or distributed [20], [33], [34]. These studies may have different optimization objectives, such as maximizing the duration or size of the Platoon [20] chain or fuel economy, which is especially important for cargo transport vehicles such as those discussed in [35], which propose an opportunistic pairing of trucks, to optimize consumption.

Thomas and Vidal [34], proposed a optimization of the formation of Platoon. In this approach, they assume that the vehicles that are candidates for the formation of the vehicle chain follow the same route in the same time window. Therefore, with a training proposal ad-hoc, they use a game strategy to define who leads and who follows. In this approach, a kind of all-against-all tournament, based on the scenario of Iterate Prisoner Dilemma [36], forming the Platoon queue. The work's objective is to propose a decentralized alternative so that the process is computationally more efficient than a centralized model for coordinating training.

Another decentralized model that also does not define the optimization goal is presented by Hades and Sommer [33], which proposed a strategy for the formation of Platoons in urban centers. In this approach, the vehicles verify their similarity of routes to form a vehicle chain during stop time at a red beacon. The conclusions are similar to [16], that Platoons at intersections, but focusing on analyzing the flow of vehicles and not on the formation criterion.

In the work of Dao *et al.* [20], the objective is to direct the vehicle to a Platoon. To this end, the vehicle remains in this Platoon until it reaches its point of exit from the road, seeking to optimize Platoons' duration. The idea is to minimize the difference between the Platoon's leader's fate and his followers' future. For this, they use a decentralized approach and adopt, as premises, the use of a Global Positioning System (Global Positioning System - GPS) as a single available sensor and vehicle to vehicle communication (V2V) in a vehicular network ad-hoc networks. Hence, all vehicles have a GPS receiver and an on-board processor to process the Platooning algorithms and exchange information vehicle by vehicle. For inter-vehicular communication, the IEEE 802.11p standard was adopted, allowing a communication radius of around 300 m for a speed of 200 km/h [37].

The Platoon type designation task's formulation can be defined by a Linear Programming problem solved by the [38] simplex method. Considering a road with  $n_e$  entrance ramps and  $n_d$  exit ramps and the route is discretized in segments, each one going from the entrance  $i$  to the next entrance ramp  $i + 1$ . In this case, the optimization problem is solved in real-time. Considering the moment the vehicle is positioned on the entrance ramp, according to the following algorithm:

- When arriving at the road's entry point, the vehicle must communicate with all the Platoons on the road and with all free agent vehicles (vehicles on the entrance ramp that have not yet been assigned to any Platoon) within the  $R$  range. Platoon is considered within scope if its leader is within range;
- The assignment routine is processed and defines which vehicle from the entrance ramp goes to which Platoon, based on the information exchanged between the vehicles of the groups and the free agents; and
- The vehicles then stop being free agents, leave the entrance ramp, and go to their Platoons. Once assigned to a Platoon, the vehicle will not call the assignment routine again.

To guarantee the stability of the Platoons, some policies have been defined: the first policy indicates that the vehicle must remain on the same Platoon throughout its journey on the road; the second policy is related to the maximum size of the Platoon, which must have a maximum of  $\gamma$  vehicles. The third policy concerns Platoon vehicles' destination, which must not exceed a  $r$  limit, which represents the "range of destinations". A fourth policy concerns the ordering of vehicles in the queue Platoon. The vehicles are in the reverse order of exit of the road; that is, the Platoon leader is the last to leave, and the previous vehicle in the queue be the first to abandon Platoon. The fifth policy states that if a vehicle cannot find a Platoon, it will become the new Platoon Leader.

Based on the experiments, varying the number of vehicles and the sizes of the rows, it was possible to verify that the larger Platoons bring more benefit, the primary indicator being the increase in the road's capacity, reaching a capacity of up to 4,880 vehicles/h in these conditions.

In this study, despite not indicating, the author assumes that 100 % of the vehicles are connected, which in practice distances a little from a closer scenario, where connected vehicles will coexist with vehicles that do not yet have this technology. Finally, the author did not define the vehicle control model used, making it difficult to reproduce the work.

In more recent work, [21] aims to compare the two approaches, centralized and decentralized, using the same optimization criterion for both, here called the similarity criterion. In this case, the vehicle's speed and position were used as similarity metrics to avoid the junction of the vehicle with Platoons that were distant.

This study formulated the problem as a linear equation where the cost ( $f_i(x)$ ) of Platoon formation is calculated where a  $i$  vehicle with each  $x$  vehicle in the neighborhood, this cost function being a function of the differences speed ( $d_s$ ) and distance ( $d_p$ ) between vehicles multiplied by a bias. In this case, the vehicles are represented by a set given by:

$$\{id, des, pos\}, \quad (1)$$

where  $id$  represents the vehicle identifier,  $des$  the desired speed and  $pos$  the current position of the vehicle. The objective is to find the best Platoon candidate  $x$  for each vehicle  $i$ , maximizing their similarities. And the function to calculate

the cost of the vehicle  $i$  to join the vehicle or Platoon  $x$  is given by

$$f_i(x) = \alpha \cdot d_s(x, i) + \beta \cdot d_p(x, i), \quad (2)$$

where  $\alpha, \beta \in [0, 1], \alpha + \beta = 1$ .

Once the join maneuver is successfully performed, the  $i$  vehicle will be part of the Platoon with the  $x$  vehicle. Once the Platoon is formed, the vehicles remain in it until they reach their destinations. In this article, the authors assume that all vehicles have the same destination.

In the centralized approach, the optimization problem is realized for all vehicles present in the scene simultaneously. Once all combinations have been calculated, the algorithm selects the best ones based on the slightest deviations of  $f_i(x)$ , eliminating the others. Each  $i$  vehicle performs this analysis with its neighbors in the distributed approach, following a similar algorithm. In this case, vehicles are made aware of their neighborhood by sending beacons. In a neighborhood table, each vehicle stores the set of information required to calculate the function  $f_i(x)$ .

The simulations take place on a highway, throughout 30 km with four lanes and an additional lane for vehicles' entry and exit. All simulations are performed on the PLEXE / VEINS / SUMO platform. In addition to the two scenarios, centralized and decentralized.

When executing the simulations and comparing the approaches and the base scenario, the author concluded that, in the centralized process, there is a lower occurrence of aborted Joins since the centralized version filters out unviable options. However, the distributed version manages to form larger Platoons, leading to a more significant deviation in the speed similarity indicator, which is one of the similarity criteria of the optimization function. With larger groups, more vehicles need to adjust to a single leader. Finally, like the previous one, this study is based on an MP of 100 % and that all vehicles have the same destination.

### C. SUMMARY

Our literature review reveals that the Cooperative Adaptive Cruise Control (CACC) can improve the vehicle's throughput on roads; however, most related works present isolated or too specific results. Differently, we are using real traces to study the influence of market penetration in the Intelligent Transportation System with realistic scenarios. We also see the need for approaching the platoon formation as a network function, allowing us to offload it to the mobile edge computing (MEC) facility. Our system view has an inherent distributed architecture, enabling vehicles to report sensor data to the edge. Thus, the grouping Virtual Network Function (VNF) has access to the road's global state and makes better decisions, both in the accuracy and response-time aspects.

We organized related works in four criteria, as shown in Table 1. It enables us to objectively state the differences in our work when compared to them. Our solution is a hybrid between centralized and distributed. The platoon grouping function is centralized, benefiting from making decisions

**TABLE 1.** Related works summarization for platoons in vehicular networks.

Computing model	Works
Centralized	[21]
Distributed	[20], [33], [34]
Optimization Objective	Works
Platoon Duration	[20]
Fuel consumption	[35]
Computational Efficiency	[34]
Track Capacity	[29]
Bottlenecks of the Roads	Works
Entry and Exit Ramps	[22]
Crossings	[16]
Human Factor with the Waves of walk and stop	[30]–[32], [39]
Action Approach	Works
Merge maneuver	[17]
Lane Change and Managed lane	[28]

from a global state view of the road. On the other hand, it has a distributed architecture, as vehicles send information to the MEC and receive configuration to platoon forming. We focus on the platoon duration, as we considering small vehicles inside Smart Cities. Fuel consumption is a crucial metric for trucks, so we do not consider it in this work. Unlike others, we provide a mix of market penetration but do not analyze the impact of CACC in stop-and-go waves. We use managed lanes and include maneuvers for merging and splitting into/from platoons.

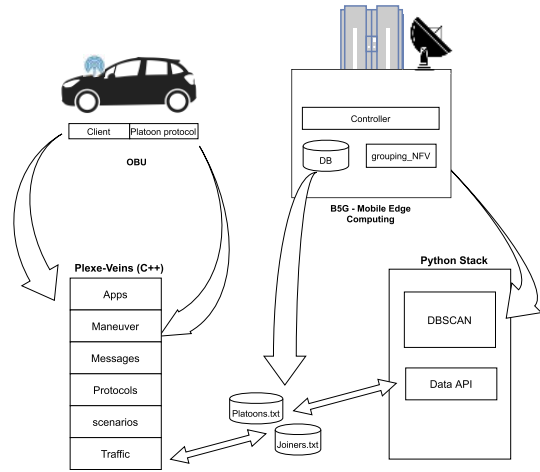
### III. PINION: A PLATOON GROUPING FUNCTION OFFLOADING FOR VANETS

In this section, we describe PINION: a Platoon Grouping Function Offloading for VANETs. PINION resembles a mechanism allowing the cooperation of two gears to transfer power between the systems; in our case, the V2V CACC protocol and the grouping Virtual Network Function (VNF) offloaded to the Beyond-5G Mobile Edge Computing (MEC). Hence, it performs vehicle groupings based on the machine learning algorithms and requests computing in MEC centers.

The main components of the architecture conceived and developed to support this research’s approach are found in Figure 1. Following a centralized approach to Platoon’s formation, the initial module is called Controller and can be executed in a Road Side Unit (RSU), in our case a Beyond-5G Mobile Edge Computing (MEC). We consider Mobile and Multi-access edge computing are equivalent to our purpose. It monitors vehicles and Platoons in its coverage region to direct vehicles with CACC technology to the Platoons present on the road.

The Client module, hosted on vehicles equipped with CACC in its On-Board Unit (OBU), is responsible for identifying the target vehicle and requesting a Platoon so that it can join it. As a basic assumption, all V2I communication with the Controller module is carried out via 5G in the region’s coverage area, and V2V communication will take place via a wireless network interface cards with IEEE 802.11p.

Algorithm 1 describes the Controller module’s function that aims to locate a Platoon for a requesting vehicle. The Controller monitors the vehicles on the road, processes



**FIGURE 1.** PINION architecture.

#### Algorithm 1 Controller - Find Platoon

```

Input Vehicle_Id, Vehicle_Location, Vehicle_Destination
Output Platoon_Id, Status
max_size_platoon ← 10
max_distance_platoon ← 600m
1: FindPlatoon Vehicle_Id; Vehicle_Location,
   Vehicle_Destination {
2: Nearby_Platoons ← FILTER(Vehicle_Location)
3: if Nearby_Platoons is empty then
4:   Platoon_Id ← Vehicle_Id
5:   Return Platoon_Id, “LEADER”
6: else
7:   Ongoing_platoons ← UPDATE(DBSCAN,
   Nearby_Platoons, Vehicle_Destination,
   max_size_platoon)
8:   Platoon_Id ← GET_PlatoonID(Ongoing_platoons,
   Vehicle_Id)
9:   Return Platoon_Id, “JOINER”
10: end if
}

```

the requests received and sends the commands to vehicles and Platoons communicating with it. When a vehicle requests a Platoon, the Controller checks the requesting vehicle’s data and performs an initial filter, selecting only the Platoons within the distance range (line 2). The parameter `max_distance_platoon` specifies the distance criterion for a given platoon to be considered a candidate for the requesting vehicle. By doing so, it is possible to check if there are target platoons. At the very first request, there is no platoon available to join. In such a case, the requesting vehicle will become a leader (lines 3, 4, and 5). Otherwise, the Controller updates the current platoons in progress by maximizing the platoon time considering the candidates’ destination. The cost function is the Platoon lifespan, consisting of a maximization function. Thereby, the task takes nearby vehicles’ destinations (constrained by

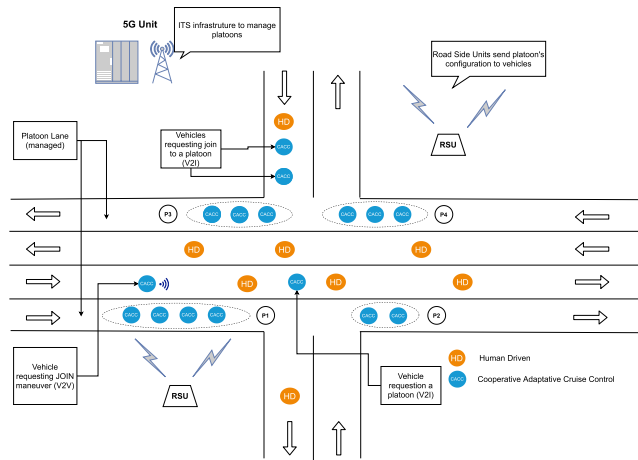


FIGURE 2. Intelligent transportation system scenario: B5G infrastructure, smart-roads, platoons, and vehicles.

the `max_distance_platoon` parameter) and tries to find the better group for that instant (lines 7 and 8). Our approach allows different grouping functions and settings. In this paper, we adopt DBSCAN. Finally, the Controller maps the vehicles' identification to the target platoon as a response to the initial request, and the status is updated (line 9).

The client module, a system hosted on the OBU of the CACC vehicle, is responsible for the platoon forming operation. The platoon protocol is a V2V communication that carries out the necessary Join and Split maneuvers. As the Controller sends the configuration settings via the B5G (beyond-5G) link, the communications happens in V2I.

The platoon has a well-defined life cycle, including its creation and termination, and is persisted locally by the Controller service. As this process is dynamic and real-time, the persistence occurs in a Key-Value database in a Hash Table. Each platoon has a unique identification code, coinciding with the leader code. Thus, the Controller has the lists of all platoons with the vehicles' identification that compose them, respectively. The record comprises a tuple of roles (each vehicle being a leader or follower), platoon size, and location (road and lane).

As shown in Figure 2, we are assuming a scenario where:

- Policies or protocols for the formation of the Platoons in managed roads. For example, the lanes and extension of the road for the Platoons, the maximum number of vehicles per Platoon, and maximum platoon time;
- An Intelligent Transportation System (ITS) - will manage the system, deployed in a B5G vertical service;
- Upon entering the managed lane, the vehicle starts exchanging V2I messages with the lane system;
- Each vehicle with CACC technology, in its V2I communication, will follow the defined protocol and, upon receiving the Platoon, to which it should join, it will start communicating V2V with the other vehicles;
- The join in Platoon will always be at the end of the group and in the inverse sequence of the exit, that is, the last vehicle in the queue will be the first to leave Platoon (Last-In First-Out queuing policy);

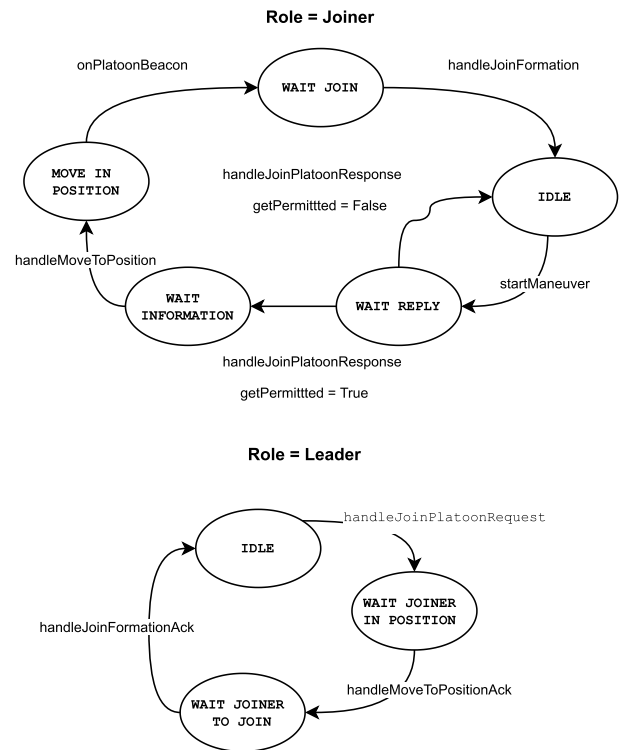


FIGURE 3. Extended version of the PLEXE's simple join protocol, allowing multiples joiners and platoons.

- After designation to one Platoon, the communication becomes V2V. The vehicle will start the join maneuvers, after, the maintenance actions will be carried out by the Platoon ( Split, in the example shown in Figure 2, where the vehicle requests merge on Platoon P1); and
- All vehicles in a Platoon are considered to be equipped with CACC technology and have defined destinations.

### A. MANY TO MANY JOIN PROTOCOL

The join protocol's state machine is based on the existing protocol in PLEXE (Figure 3) that met a simple Join protocol at the back from a vehicle to a Platoon. We extended it by adding new modules and states.

Among the primary adjustments, we highlight the removal of the new Joiners release lock while the leader is in the process of maneuvering with the current Joiner. Given the time required to perform a complete join maneuver, it is not feasible to wait for the entire previous scheme to finish before starting a new one in the context of multiple vehicles are joining to multiple platoons, or simply Multiple Joiners. This process requires a concurrency control among the maneuvers.

Another necessary adjustment was to create a vector of Joiners on the leader's side to send the correct messages and contexts to each Joiner in approaching maneuver. Finally, on the side of the Joiner, it was necessary to adjust the context to know which is the reference vehicle for its approach in Platoon, which will not always be the last in the current Platoon queue. But, eventually, it may be another Joiner that is still in the maneuvering phase.

## B. THE PLATOON GROUPING FUNCTION

Considering that the objective of this research is to optimize the vehicle's permanence time on the Platoon, it is necessary to select the best Platoon for vehicles entering a road with several lanes, aiming, in this way, to obtain improvements in the vehicle flow and the capacity of the streets. Thus, two main characteristics were selected as inputs for the machine learning model, the vehicles' destinations and the time window when these vehicles are present in the simulation.

We modeled this problem as a clustering task in the Machine Learning framework. Specifically, clustering means grouping vehicles with the maximum destination likelihood. Therefore, the unsupervised machine learning model fits that purpose. We use the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) because it can identify clusters and is extensively employed algorithm for this goal.

We structured our model as [40]:

### a: TASKS

Group vehicles according to their destination.

- 1) Grouping Criteria:
  - a) Target range; and
  - b) Possibility in the timeline.
- 2) Restriction:
  - a) Maximum group size = 10;
  - b) The road must support Platoons; and
  - c) CACC vehicles share the same origin, entry-point.

### b: HYPERPARAMETERS

The DBSCAN machine learning model basically uses two hyperparameters  $\epsilon$  and  $MinPts$  [41], where  $\epsilon$  represent the maximum range of the platoon (600 m) and  $MinPts$  the number of vehicles in platoon. In the case,  $MinPts$  was defined with a value equal to two ( $MinPts = 2$ ), defining Platoons with a minimum size equal to two vehicles. The  $\epsilon$  hyperparameter is a user-dependent choice (we evaluate different values to it in Section IV).

## IV. PERFORMANCE EVALUATION AND METHODOLOGY

In this section, we present a case study to prove the efficiency of the proposed approach. We describe our simulation engine, detailing configuration, criteria, levels, parameters, and metrics in the first part. Then, we show the data preparation and the machine learning model's training, specifying the origin of the data, its preparation, and pre-processing. Finally, we test our using the Shenzhen taxi dataset (dataset available at <https://github.com/c2dc/pinion>), real-world collected mobility traffic.

### A. THE IMPLEMENTATION OF PINION

We developed our solution using the PLEXE simulation library. The Client module is implemented in the PLEXE App routines and its protocol in the Maneuver routines. The server is developed partly in PLEXE and partly in Python routines; the former for the Controller deployment platform,

TABLE 2. Factors and levels for experiments.

Factor	Levels
Market Penetration (MP)	25%, 75%
Number of Vehicles	300, 500
DBSCAN range	$\epsilon \in [0.01 - 0.15]$

and the latter for the DBSCAN implementation, scenarios, Traffic modules. Finally, all data exchange between the two stacks is carried out via text files.

### B. SIMULATION ENVIRONMENT

The study of vehicle platooning can involve several complexity levels, and the market penetration is an issue to consider. In a real situation, vehicles with different capabilities levels should co-exist. Regarding communication, vehicles have or have not the functionality to connect to the infrastructure and, in more advanced cases, features to establish inter-vehicle communication. Another influencing factor is the automation levels such as Cruise Control (CC), Adaptive Cruise Control (ACC), and Cooperative and Adaptive Cruise Control (CACC).

For example, [39] demonstrated that with the MP factor of vehicles equipped with CACC below 40%, there is no impact on traffic, and it is not possible to avoid traffic shockwaves in the flow of vehicles. However, in a more recent study, [30] found that with an MP factor of only 10% autonomous vehicles, it is already possible to dissipate these waves. The big difference between these works is that in [39], the execution of lane change maneuvers is taken into account so that vehicles with CACC form the Platoon only then causing the effects on traffic, while in [30] the study is carried out solely with the presence of autonomous vehicles in traffic, maintaining the flow of vehicles.

Our experiment design includes three factors: market penetration (MP), the number of vehicles on the road, and the DBSCAN range ( $\epsilon$ ). Table 2 shows the levels for each factor.

We varied the MP factor level in the simulation with the values of 25% and 75%; that is, considering two scenarios, (i) one with the predominance of vehicles in the Intelligent Driver Model (IDM – human driven vehicles) and (ii) the other with vehicles equipped with CACC. In addition to the Market Penetration factor (MP), the other factor considered was the Number of Vehicles (NoV) in the simulation. The impact of vehicles' volume may vary, depending on the road components (number of lanes, access and exit ramps, the section length). For the clustering algorithm, we changed the range of the cluster ( $\epsilon \in [0.01 - 0.15]$ ), increased by 0.1).

We adopted the PLEXE library for platooning formation, built on top of the VEINS (The Vehicles in Network Simulation) framework. It is a discrete-event simulation engine that uses OMNET++ to implement networking functionalities and SUMO (Simulation of Urban Mobility) to realize the road traffic. The simulation platform's configuration involved integrating these frameworks and their libraries so that the experiments could simulate vehicles' dynamics on maps, responding according to their interactions in V2V or V2I



**TABLE 3. Parameters for the intelligent driven model (IDM) and CACC.**

Attribute	Default value	Range	Description	Models
accel	Vehicle-dependant	$\geq 0$	Vehicle acceleration in $m/s^2$	IDM, CACC
decel	Vehicle-dependant	$\geq 0$	Vehicle deceleration in $m/s^2$	IDM, CACC
emergencyDecel	Vehicle-dependant	$\geq decl$	Maximum deceleration on emergency in $m/s^2$	IDM, CACC
delta	4		Acceleration exponent	IDM
stepping	0.25	$\geq 0$	internal stepping speed to moving forward	IDM
speedControl, GainCACC	-0.4		CACC gain specifying deviance speed (speed control mode)	CACC
gapClosing, ControlGain, Gap	0.005		CACC gain specifying deviance by positioning (proximity control mode)	CACC
gapClosing, ControlGain, GapDot	0.05		CACC gain specifying deviance derivative by position (proximity control mode)	CACC
gapControl, GainGap	0.45		Controller gain specifying deviance rate by positioning (GAP control mode)	CACC
gapControl, GainGapDot	0.0125		Controller gain specifying deviance rate derivative by positioning (GAP control mode)	CACC
collision, Avoidance, GainGap	0.45		Controller gain specifying deviance rate by positioning (collision avoidance mode)	CACC
collision, Avoidance, GainGapDot	0.05		Controller gain specifying deviance rate derivative by positioning (collision avoidance mode)	CACC

**TABLE 4. 802.11p link simulation configuration.**

Parameter	Value
Road length	18 km
Number of lanes	4
Transmission rate	18 Mbts/s
Transmission power	2.2 mW
Communication range	300 m
Updating frequency SUMO	0.1 s
Replication	rep = 100

communication. We represented a road with four lanes and an extension of 18 km, with access and exit routes and replicating the model used by [29]. The SUMO IDM control model parameters were based on the work of [42] and the control model CACC [43]. Table 3 summarizes this set up.

The Market Penetration factor (MP) specifies the number of vehicles with communication modules. We vary it by two levels of percentage: 25% and 75%, representing CACC vehicles' presence. The remaining vehicles implemented the Intelligent Driver Model (IDM), which represents the human-driven vehicle behavior. The total volume of cars in the simulation was adjusted to the levels of 300 and 500 vehicles to verify the flow of vehicles and the road's capacity.

All combinations of the Market Penetration factor (MP) with the Number of Vehicles factor (NoV) represent experiments in the simulations. For each MP versus NoV level, the maximum capacity reached in the road lanes and the vehicle flow's average speed were measured. For the other parameters, we use [44] as a reference; therefore, the bit rate in the MAC layer is 18 Mbit/s, and the transmission power is 2.2 mW, which in the Two-Ray Ground propagation model, provides a Communication Ratio (CR) of 300 meters, as shown in Table 4.

In addition to the communication configuration parameters described above, there are specific parameters for the Platoon settings. The maximum size of Platoon ( $\tau_{max}$ ) used in this research was fixed in 10 vehicles because, as explained in [17], the ideal size should be between 10 and 20 vehicles. Bigger Platoons affect Join's maneuvers, and smaller ones end up canceling their benefits. These simulation parameters are presented in Table 5.

Another parameter used in the simulation was the  $r$  range for selecting the compared Platoons so that the  $v_i$  vehicle could join the Platoon. In this research, a range of  $r = 600$  m was used, taking into account that we use the centralized model to control the Platoons' formation.

**TABLE 5. Platoons parameters.**

Parameter	Notation	Value
Mean vehicle size	$s$	4 m
Distance between platoons	$h$	$\approx 33$ m
Platoons split time	$t_h$	1.2 s
Platoon maximum size	$\tau_{max}$	10 vehicles
Leader maximum speed	$V_{max}$	100 Km/h
Platoon's inclusion range	$r$	600 m

A benefit of the centralized approach to control the Platoons' formation is the possibility of choosing vehicles with similar destinations. Wherefore, for each new incoming vehicle, analyzing the managed route and ones around the reach of  $r$ . However, this alternative is more complicated because, in addition to requiring more analysis attributes, such as the distance between the vehicle (Joiner) and the active Platoons on the road, the Platoon training protocol also becomes more complex.

As the platoon's minimum size is 2 (two), we define the DBSCAN parameter  $MinPts$  to 2 accordingly.

**Indicators** — The following indicators were calculated in the simulations:  $C_{max}$  the lane traffic capacity in vehicles/lane/hour, where the speed  $v$  is in  $km/h$ , the separation time between Platoons  $t_h$  in seconds, and the distance  $h$  between Platoons and the mean vehicle size  $s$  both in meters.

We can define the platoon's influence on the maximum capacity  $C_{max}$  in vehicles/lane/hour of a road according to the maximum size of Platoon  $\tau_{max}$ ; the steady speed of vehicles  $v$  in  $km/h$ ; the separation time between groups  $t_h$  in seconds; the average size of the vehicle  $s$  in meters; the average distance between the groups  $h$ ; and can be given by

$$C_{max} = \frac{3600\tau_{max}v}{3.6[(\tau_{max} - 1)h + s] + t_h v}. \quad (3)$$

With the parameters defined for this research work (Table 5) we have  $C_{max} = 2,991$  vehicles/range/hour. From this reference [20], it was possible to measure whether the proposed approach reached the road's maximum capacity at some point in the simulation.

The size of the Platoon  $\tau_{max}$  allowed us to verify the success rate of the formation of the Platoons because, if a vehicle was unable to join any Platoon, it followed its path to the destination without forming a new one. We assume that all vehicles start the simulation at the road's same entry-point position and the joiners stream arrives after platoons, so not all Platoons reached their potential maximum size.

**TABLE 6.** Pre-processing vehicles characteristics for clustering.

#	Field	Description
1	id	Vehicle identification
2	lat	Destination latitude
3	lon	Destination longitude
4	time	Entry timestamp
5	projection	Distance from entry-point to destination
6	pltId	Platoon identification

Finally, we also measured the vehicles' average speed  $V_m$  to verify the road behavior as the number of vehicles grew. It reveals the impact of the platoons on the vehicle's flow.

### C. DATA PRE-PROCESSING AND EXECUTION OF THE MACHINE LEARNING MODEL

We pre-processed the data to enable its use in the machine learning model before running the simulation. The Python programming language, version 3.8, was used to develop and execute the model. We used Pandas and Numpy libraries for data analysis and pre-processing and ScikitLearn for clustering with DBSCAN.

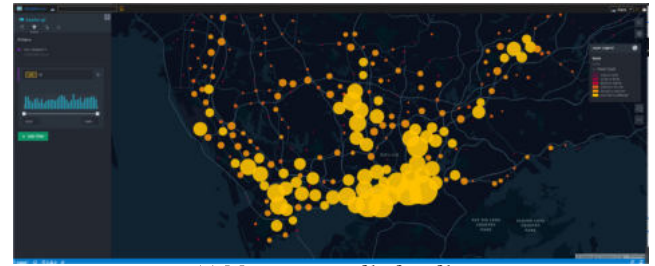
As a heuristic, we adopted the vehicle's longitude destination and the entry-time in the road to defining the best platoon. Thus, at a given simulation time, vehicles more distant tend to form a platoon. We tested our approach against two datasets: a randomly created and Shenzhen's taxi dataset. We used the attributes listed in Table 6. Using a uniform distribution, we generated latitude, longitude, and time fields. And the projection field according to Equation 4 explained below. On the other hand, the actual dataset comprises data collected from taxis in the city of Shenzhen in China, loaded by companies in real-time, periodically measuring taxis' status (GPS and occupation) [45].

Considering the dataset has more than ten million records, we divided it into ten files with one million records. By visual inspection of the concentrations of taxi routes in the city (Figure 4), we could identify the area with a highest vehicle concentration (Figures 4a and 4b). We chose Binhe Blvd as a reference for our simulation because of its several lanes and a great taxi flow concentration (detailed in Figure 5). Figure 5a contains a representation of the traffic concentration, and Figure 5b a highlighted in the boulevard. Finally, in Figure 5c, it is possible to see a satellite view of the region and, in Figure 5d, the road with its lanes, junctions, and access and exit ramps, in one region delimited by the coordinates (22.543145, 114.071804), (22.543145, 114.111372), (22.530162, 114.071804), and (22.530162, 114.111372). We pre-processed Shenzhen dataset as described in Algorithm 2.

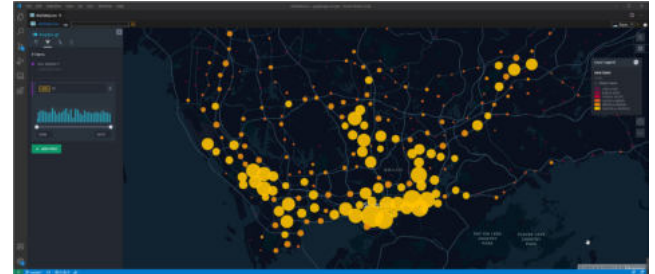
Considering the average latitude for the Binhe Blvd is 22.543145 and the reference longitude is 114.111372, we compute projection by

$$proj_v = (long_{dest} - long_{orig}) \cdot 111, 12 \cdot \cos(lat). \quad (4)$$

All taxis with negative projection were excluded from processing, as they do not pass through the simulation area.



(a) Movement profile first file.



(b) Movement profile second file.

**FIGURE 4.** Taxi movement profile in Nanshan district, Shenzhen, China.

#### Algorithm 2 Shenzhen Dataset Pre-Processing

**Input** CSV-file with Shenzhen dataset.

**Output** CSV-file to DBSCAN.

- 1: **for** taxi input file **do**
- 2:   sort entries ascending by timestamp
- 3:   Origin latitude and longitude  $\leftarrow$  first latitude and longitude
- 4:   Destination latitude and longitude  $\leftarrow$  last latitude and longitude
- 5:   timestamp  $\leftarrow$  most frequent timestamp
- 6:   projection  $\leftarrow$  computeProjection(reference latitude and longitude, destination longitude)
- 7:   Save in output CSV-file
- 8: **end for**

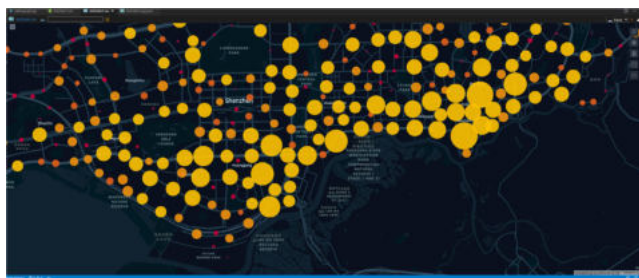
Finally, the reference time used in step 5 was the most frequent timestamp in the dataset.

DBSCAN is an unsupervised clustering algorithm and has two main hyperparameters  $\epsilon$  and  $MinPts$  [41]. In the case of this work,  $MinPts$  was defined with the value 2 (two), meaning the minimum platoon size. However, we varied  $\epsilon$  to find the better configuration.

### D. STUDY CASE

To compare results and obtain improvements in the vehicle flow and the road's capacity, we carried out a case study involving a scenario composed of a four-lane road section with ramps for entry and exit. In this scenario, the insertion of vehicles in the simulations followed a negative exponential distribution, dividing them into three distinct groups:

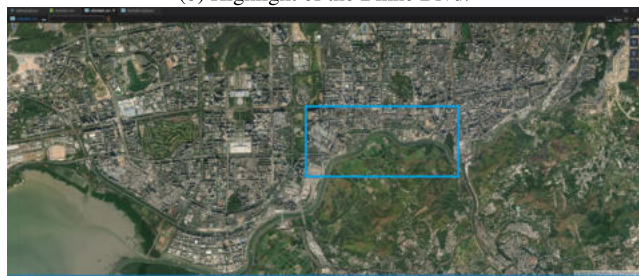
- platoons;
- joiners, composed of vehicles using the Cooperative Automated Cruise Control (CACC) model; and



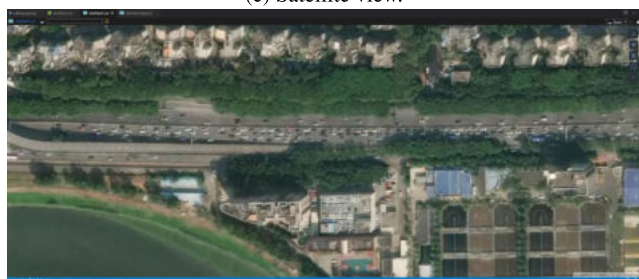
(a) Overall view of the considered region.



(b) Highlight of the Binhe Blvd.



(c) Satellite view.



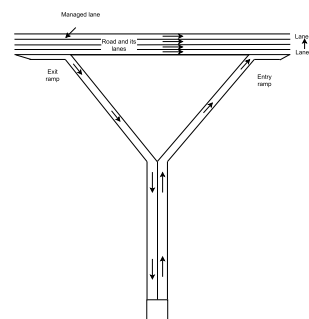
(d) Zoom of the Binhe Blvd satellite view.

**FIGURE 5.** Sequence of the binhe blvd region to represent in our simulation.

- human-driven vehicles, using Intelligent Driver Model (IDM).

According to the Market Penetration levels aforementioned, these two Control Models (i.e., Car-following Models) concurrently present in the simulation.

Similar to [17], we use a Managed Lane—ML with a platoon-exclusive lane. Thus, we assigned vehicles with CACC that lane. In case of saturation, the adjacent lanes receive the surplus contingent as they get overloaded. Using the SUMO framework, we mapped Binhe Blvd and verified our approach using the platoons formation algorithm and the join protocol algorithm. This map was composed of a four-lane traffic lane and a pattern of entry and exit ramps, as shown in Figure 6.



**FIGURE 6.** Exit- and entry-ramp used in the simulation.

**TABLE 7.** Distribution policy for IDM and CACC vehicles.

Lane	IDM	CACC
1	0%	100%
2	0%	100%
3	40%	60%
4	100%	0%

The proportion of IDM and CACC vehicles for each lane follow the policy shown in Table 7. Lanes 1 and 2 are CACC-exclusive, whereas Lane 3 is a mix of IDM and CACC, and Lane 4 is IDM-exclusive. However, in border cases of MP, this policy has been disabled. Thus, for 0%MP, all tracks were made available for vehicles of the IDM type, and, in the case of 100% MP, the first two tracks were made available for platoons, and the CACC vehicles occupied other lanes. In this way, it mitigated the risk of vehicles performing extreme lane-change maneuvers, generating an unnatural disturbance in vehicles' flow at the simulation commencement, which would not occur in a real situation.

The simulation requires two files containing platoons and joiners. They form four sets of vehicles, resulting from the combination of market penetration (with and without CACC) and amount of vehicles (300 and 500). Together they constitute all vehicles in the simulation. The set of vehicles using the CACC model for the platoons file had the following attributes: identifier (*id*), destination,  $lat_n$  and  $long_n$ , and the platoon (*plId*). Before the simulation, we run DBSCAN to assign a vehicle (*id*) to the target platoon (*plId*). Thus, the vector of characteristics of each vehicle in the CACC model presented the format

$$V_v = [id, latd, longd, plId]. \tag{5}$$

On the other hand, the joiners file has two attributes, the target platoon *id* and the vehicle code (*id*).

Two different scenarios were considered in this research and defined based on the *r* range parameter. In the first scenario,  $r = 600$  m was used, equaling twice the V2V transmission radius. In the second scenario,  $r = \infty$ , the entire set of road groups. The difference between these two scenarios was that, in the first, once a platoon was received to join, the candidate vehicle had to send a `join` request to the designated platoon queue. In the second scenario, the join algorithm used an initial approach protocol indicated by the

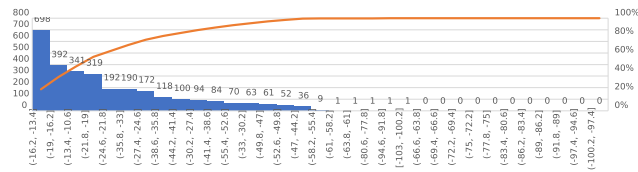


FIGURE 7. The Shenzhen taxi's run approximates to Pareto distribution by Distance.

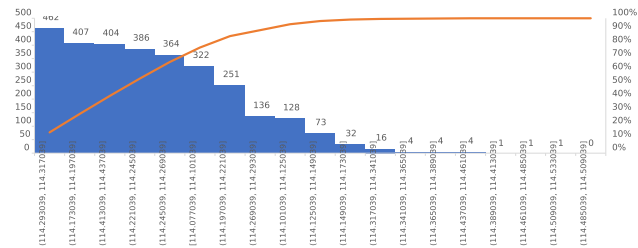


FIGURE 8. Pareto distribution by destination (longitude).

control service so that the vehicle and the Platoon stayed in their communication range to start the V2V protocol then.

To simplify the join and split strategy for vehicles in the target platoons in the simulations:

- 1) The join maneuvers occur at tail only, and the vehicles' queue follows the last-in, first-out (LIFO) policy;
- 2) Vehicles join once to their target platoon, staying in the line until the departure point; and
- 3) New vehicles could join a platoon at any time, as long as they respect the previous rules.

### 1) VEHICLES PROFILING IN DATASET

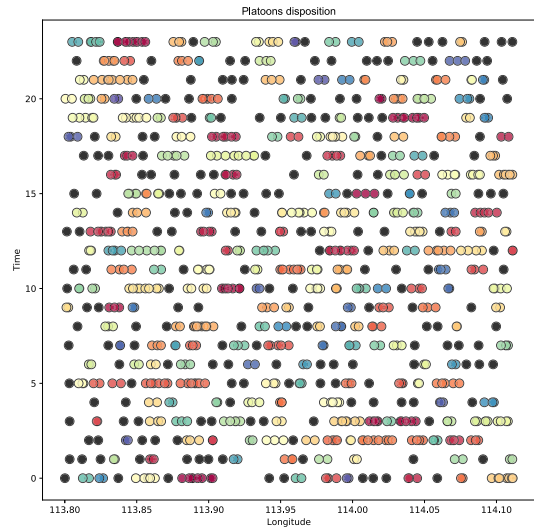
The Shenzhen taxis dataset, made initially available with GPS position information over 24 hours, was processed and transformed into a set of sources and destinations in the format: {id, origin latitude, origin longitude, destination latitude, destination longitude, projection, timestamp}.

In Figure 7, it is possible to see the Pareto distribution of the projections, indicating that most vehicle trips (almost 80%) are on short distances (almost 20% of the routes), between 13 km and 16 km, which indicates the possibility of a constant change of vehicles in the platoons.

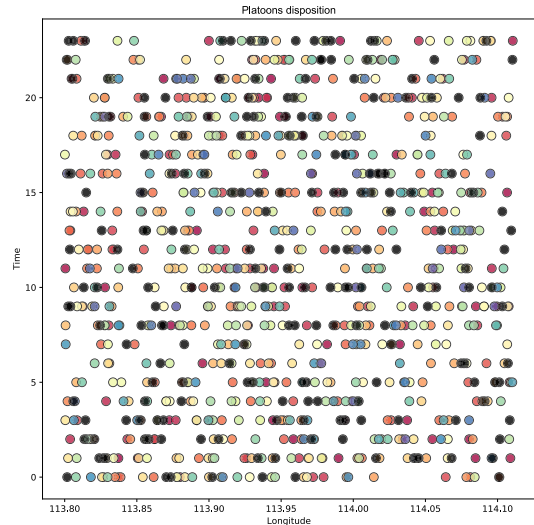
In Figure 8, there is, therefore, an approximate Pareto distribution of vehicles to their destination longitudes. As we can see, it is possible to verify regions with the most significant movement, which helps to infer that some stretches throughout Binhe Blvd will have a greater vehicle flow. Thus, the final vector of the vehicle attributes was in the following format: {id, origin latitude, origin longitude, destination latitude, destination longitude, projection, timestamp, platoon}, and the platoon attribute indicates the target the vehicle must proceed. The output file contains 2,996 records, divided into two datasets, one with 300 vehicles and another with 2000 for input in the DBSCAN. For the simulations, we grouped records by 300 and 500 vehicles.

### 2) DBSCAN BEHAVIOR ANALYSIS

The randomly generated dataset contains platoons with a size of ten or less. The disposition of vehicles to join follows a



(a) Platoons generated by the baseline approach (sort vehicles by longitude and group them by 10 at most).

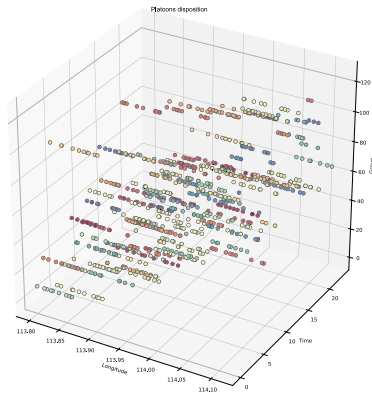


(b) Platoons generated by DBSCAN.

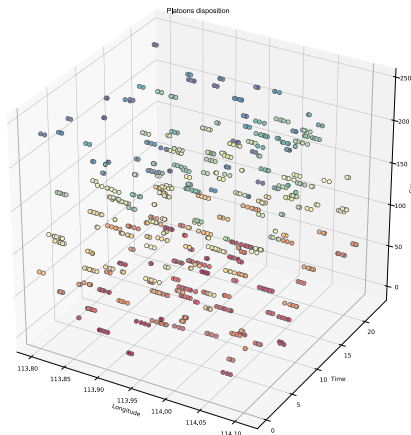
FIGURE 9. Clusters as platoons – generated by DBSCAN.

uniform distribution. Comparing the results of the clusters created by the DBSCAN model's processing with the original Platoons of the data cluster, it is seen in Figure 9 that the models are similar. The graph on the left (Figure 9a) corresponds to data from the uniform distribution, according to attributes described in Table 6. The graph on the right (Figure 9b) to the DBSCAN model with  $\epsilon = 0.07$ , chosen for generating clusters with the maximum of 10 vehicles, the allowed size by platoons.

Fragmentation in clusters yielded by DBSCAN is similar to those randomly generated, and outliers follow the same pattern in both clusters, dispersing about 10% of the number of vehicles. However, we can better visualize the difference between the models a three-dimensional view, from the same graph (Figure 10), including the dimension of the generated clusters, both by traditional programming (Figure 10a from the left) and the one generated via Learning



(a) Platoons generated by the baseline approach (sort vehicles by longitude and group them by 10 at most).



(b) Platoons generated by DBSCAN.

FIGURE 10. 3D-view of the clusters as platoons – generated by DBSCAN.

Machine (Figure 10b on the right). DBSCAN generated sparse Platoons, whereas the traditional one generates platoons more efficiently.

The analysis of the model’s application to the Shenzhen data (Figure 11) indicated a different behavior from the model generated by the uniform distribution. There are more outliers in the Shenzhen dataset than those in uniform distribution. Concerning the platoon formation constrained by the maximum size as the hyperparameter ( $\epsilon$ ) goes to the limit reference, we see a vehicles’ loss in the group of about 50% due to outliers. That was the case for the full dataset (1004 outliers) and the reduced one with 158 outliers.

When comparing the Figures 10b (uniform,  $\epsilon = 0.07$ ) and 11 (Shenzhen  $\epsilon = 0.13$ ), distributions seem different. However, as the former has sparser vehicles, DBSCAN achieve a better performance in clustering them, whereas in the latter there are concentration on some regions which makes harder the platoon forming task. Finally, when analyzing metrics (outliers, Number of clusters, and Maximum size of cluster) and the three clustering variations (datasets: uniform and Shenzhen reduced and complete—346 and 2005 vehicles, respectively), we observed a different behavior insofar as the  $\epsilon$  hyperparameter varies.

We observed two effects in Figure 12 a significant decrease of outliers in the initial stages of variation of the parameter  $\epsilon$

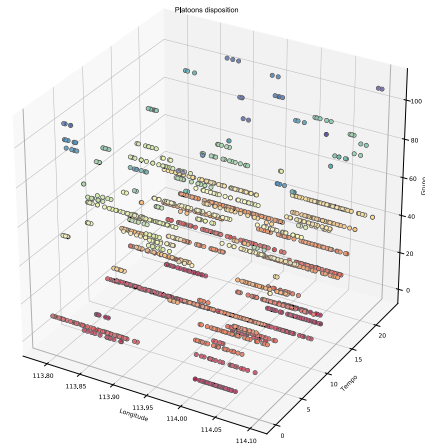


FIGURE 11. 3D-view of platoon formation using DBSCAN with Shenzhen dataset.

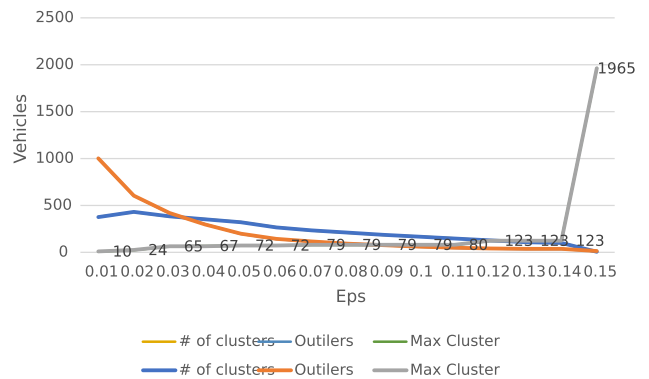


FIGURE 12. Metrics for the Shenzhen full dataset.

and settling after it reaches 0.06 for the complete dataset clustering result. The same characteristic applies to the reduced one (Figure 13). For the uniform dataset (Figure 14), this drop occurs more gradually with a smoother and more linear slope. Regarding the number of clusters, the three have a gradual reduction as the  $\epsilon$  varies, and in the uniform dataset, the number of clusters has tiny variation compared to  $N_{eps}$ . About the cluster’s size, the complete Shenzhen dataset and uniform dataset presented the phenomenon known as percolation, for the same  $\epsilon = 0.14$ . After this limit, there was an almost complete connection in both models, drastically reducing the number of clusters with the exponential increase in the platoon’s size, connecting all vehicles to the same platoon.

### 3) PLATOON’S FORMATION IN THE SIMULATION

We calculated the road capacity metric according to Equation 3 and found 2, 967 vehicles/h/banner is the maximum throughput, considering our simulation parameters (Table 5). The road finds its maximum utilization depending on three conditions when platoon sizes tend to 10 (maximum), vehicle joins as soon as possible its target platoon and leaves as near as possible from its exit ramp. We used DBSCAN to verify the performance on the two datasets: 1,000 and 2,005 vehicles from uniform distribution and Shenzhen. We compare the number of platoons formed

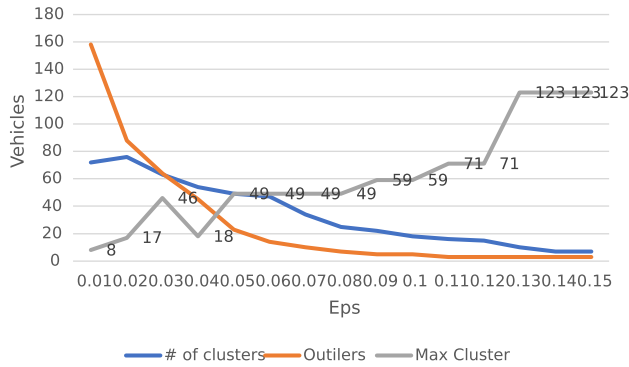


FIGURE 13. Metrics for the Shenzhen reduced dataset.



FIGURE 14. Metrics for the uniform-generated dataset.

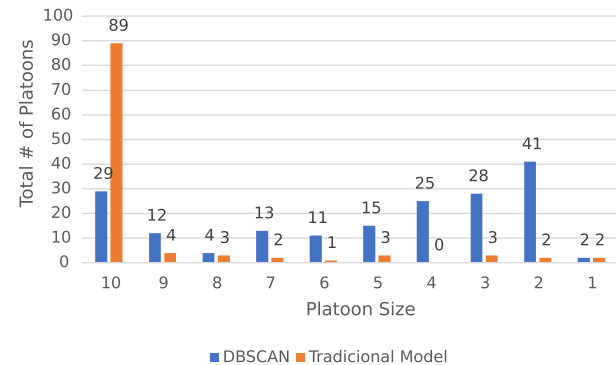


FIGURE 15. Comparison of the Platoons generated by DBSCAN on the grouping of data generated by normal distribution versus the actual grouping of data from Shenzhen.

by PINION against the number of platoons in the uniform dataset, labeled when produced. Results are on Figure 15.

PINION reached 16% of the number of platoons. When using  $\epsilon$  at the percolation limit (meaning all vehicles can form a single platoon), results show PINION distributes vehicles in small platoons. On the other hand, when grouping vehicles by longitude and restricting the platoon to ten participants, it yields 81% of possible platoons in the dataset (uniform-generated). We call this approach baseline.

The PINION model's result using the Shenzhen real data cluster shows a significant improvement, as shown in Figure 16. We see that PINION achieves a concentration of 89%, against a 77% from the baseline; that is, a difference

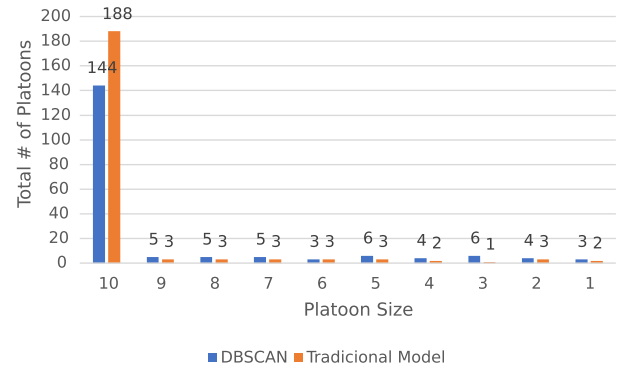


FIGURE 16. Comparison of Platoons generated by DBSCAN x Via programming using the grouping of real data from Shenzhen.

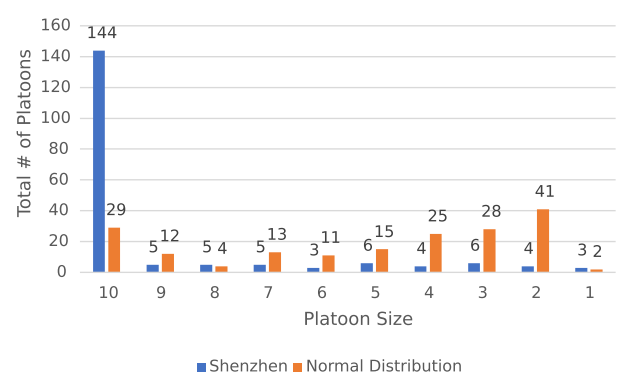


FIGURE 17. Comparison of the Platoons generated by PINION x Via programming using the grouping of Shenzhen Real data.

of 12%. Comparing the results from two data clusters and fixing the PINION algorithm, the difference is evident, as shown in Figure 17.

Formation of the Platoons did not significantly impact the flow of IDM vehicles, that is, for a Market Penetration with a value of 0 (zero). It can be verified numerically in Table 8, with detailed statistical data with means, standard deviations, quartiles, and medians. We see a stationary behavior with low variation, a mean of 27.78 mph, as the median and quartiles are the same. Also, for grouping 300 or 500 vehicles and Market Penetration levels of 25 % and 75%, our experiments yield the same pattern. We conclude that the ITS can incorporate the CACC without compromising operations.

E. JOIN PROTOCOL EVALUATION

To evaluate the Join protocol, in the many-to-many relationship among Joiners and Platoons, we use three performance indicators: Speed, Distance, and Acceleration. To clarify the use these indicators and their relationship with the protocol, Figure 18 shows, as a counterexample, an accident generated by a flawed protocol, where two vehicles (Joiners), using the CACC control model, clash.

The vehicles involved in the accident are represented by nodes one and two, and the zero nodes indicate a vehicle that followed its trajectory without problems. Figure 18a shows the speed indicator where it is already possible to verify that the vehicle (two) has a discrepancy with the other Joiners

TABLE 8. Dispersion of the speed in Binhe Blvd during our simulation scenarios.

	OMP300	OMP500	25MP300	75MP300	25MP500	75MP500
count	127.000000	127.000000	240.000000	382.000000	252.000000	521.000000
mean	28.338179	28.338179	28.129296	28.324124	28.148155	28.263865
std	1.231967	1.231967	0.980720	1.226032	0.935884	1.353442
min	27.777642	27.777642	27.777807	27.777642	27.777642	27.777807
25%	27.780000	27.780000	27.780000	27.780000	27.780000	27.780000
50%	27.780000	27.780000	27.780000	27.780000	27.780000	27.780000
75%	27.780000	27.780000	27.780000	27.780000	27.780000	27.780000
max	32.816957	32.816957	33.454227	33.688365	32.816957	34.987651

{0,25,75}MP: marketing penetration levels; {300,500}: number of vehicles.

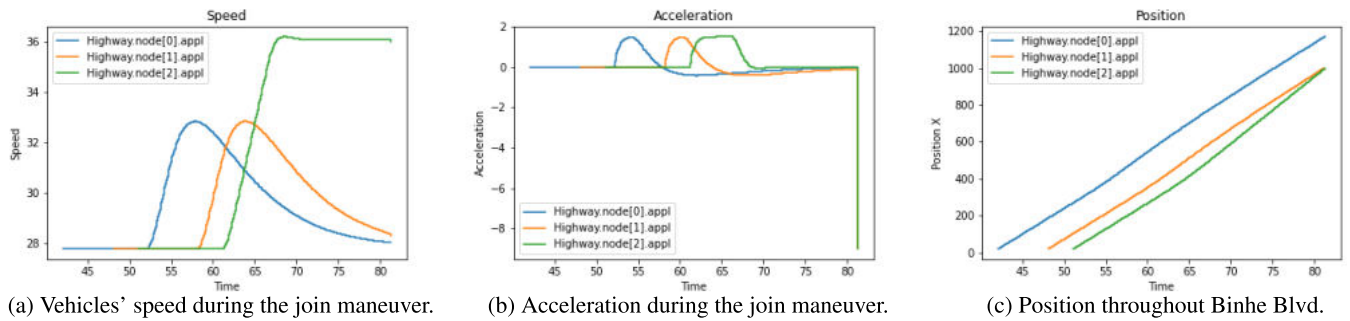


FIGURE 18. Trace of an accident during the join maneuver.

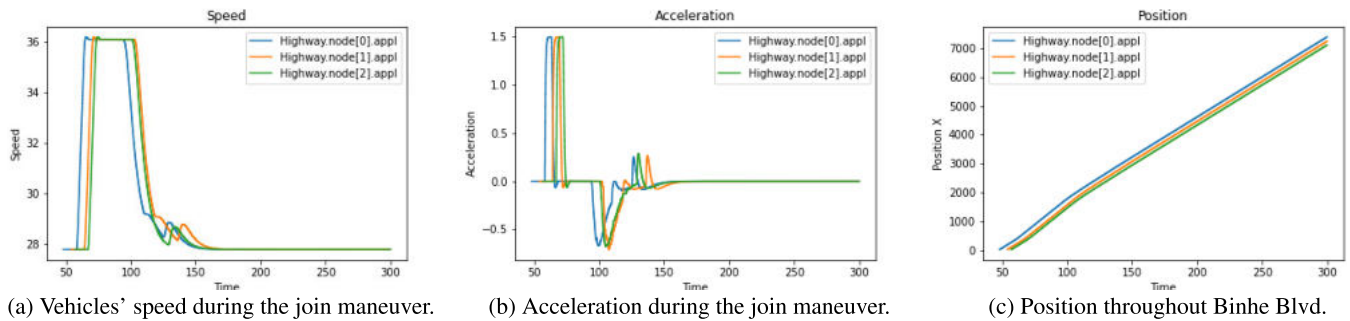


FIGURE 19. Performance indicators for a successful join maneuver.

going much faster. Figure 18b, it is possible to identify the cause of this high speed. Due to the protocol failure, this vehicle had a more significant acceleration than the other simulation’s other Joiners. Finally, in Figure 18c, it is possible to see the consequence of this failure with the collision between vehicles one and two.

Following the same diagram as above, the execution of the protocol in the Shenzhen scenario with the reduced number of vehicles (346) for three randomly selected Joiners is shown in Figure 19. In this Figure, it is possible to verify the same pattern of behavior between the three vehicles in all stages of the maneuver, as much as the Speed shown in Figure 19a, as well as the Acceleration shown in Figure 19b, as well as the Position with different trajectories throughout the simulation shown in Figure 19c. The same occurs with the other simulated scenarios.

V. CONCLUSION

In this paper we present PINION: a Platoon Grouping Function Offloading for VANETs. It resembles a mechanism

allowing the cooperation of two gears to transfer power between the systems; in our case, the V2V CACC protocol and the offloading of the grouping Virtual Network Function (VNF) into the 5G Mobile Edge Computing (MEC). We implemented it as a clustering task in the machine learning framework and demonstrated our solution’s efficiency using realistic traces from Shenzhen taxis’ flow. Moreover, we explore scenarios with different levels of Market Penetration in a road design with managed lane. Such a setup allows us to analyze feasible scenarios to happen in the next ten years.

We demonstrate the performance difference when comparing dense (the Shenzhen dataset) and disperse scenarios (randomly-generated dataset). PINION achieve an accuracy of 89% of the entire amount of platoons in the datasets. Depending on the hyperparameters configuration, the performance can change considerably. Therefore, we see a need to dynamically adapt those values as the road condition changes from one state to another. We also evaluate an extension of the PLEXE join protocol equipping it with multiple joiners vs.

multiple platoons functionality. Our experiments demonstrate cases with collisions, and we provide a solution to a more safe operation on roads. Finally, our solution did not improve the overall road speed; however, the results show a settled speed (stationary and low dispersion), where users can experience a more deterministic traffic condition.

As future work, we plan to implement other functions for the grouping VNF: different clustering algorithms and modeling vehicles using complex networks. In the latter case, implement a graphical neural network to this end. Another point of improvement considers the extensive use of other datasets and verifies their impact on the algorithms, allowing us to analyze how suitable hyperparameters are to dynamic adaptation.

## REFERENCES

- [1] R. I. Meneguette, A. Boukerche, A. H. M. Pimenta, and M. Meneguette, "A resource allocation scheme based on semi-Markov decision process for dynamic vehicular clouds," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.
- [2] R. I. Meneguette, A. Boukerche, G. Maia, A. A. Loureiro, and L. A. Villas, "A self-adaptive data dissemination solution for intelligent transportation systems," in *Proc. 11th ACM Symp. Perform. Eval. Wireless Ad Hoc, Sensor, Ubiquitous Netw.* New York, NY, USA: Association for Computing Machinery, 2014, pp. 69–76, doi: 10.1145/2653481.2653492.
- [3] R. I. Meneguette, A. Boukerche, and A. H. M. Pimenta, "AVARAC: An availability-based resource allocation scheme for vehicular cloud," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3688–3699, Oct. 2019.
- [4] E. Lee, E.-K. Lee, M. Gerla, and S. Oh, "Vehicular cloud networking: Architecture and design principles," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 148–155, Feb. 2014.
- [5] C. Correa, J. Ueyama, R. I. Meneguette, and L. A. Villas, "VANets: An exploratory evaluation in vehicular ad hoc network for urban environment," in *Proc. IEEE 13th Int. Symp. Netw. Comput. Appl.*, Aug. 2014, pp. 45–49.
- [6] R. I. Meneguette, E. R. M. Madeira, and L. F. Bittencourt, "Multi-network packet scheduling based on vehicular ad hoc network applications," in *Proc. 8th Int. Conf. Netw. Service Manage. (CNSM) Workshop Syst. Virtualization Manage. (SVM)*, 2012, pp. 214–218.
- [7] M. Amoozadeh, H. Deng, C.-N. Chuah, H. M. Zhang, and D. Ghosal, "Platoon management with cooperative adaptive cruise control enabled by VANET," *Veh. Commun.*, vol. 2, no. 2, pp. 110–123, Apr. 2015.
- [8] J. I. Ge, S. S. Avedisov, C. R. He, W. B. Qin, M. Sadeghpour, and G. Orosz, "Experimental validation of connected automated vehicle design among human-driven vehicles," *Transp. Res. C, Emerg. Technol.*, vol. 91, pp. 335–352, Jun. 2018.
- [9] V. Milanes, S. E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, and M. Nakamura, "Cooperative adaptive cruise control in real traffic situations," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 296–305, Feb. 2014.
- [10] A. Talebpoor and H. S. Mahmassani, "Influence of connected and autonomous vehicles on traffic flow stability and throughput," *Transp. Res. C, Emerg. Technol.*, vol. 71, pp. 143–163, Oct. 2016.
- [11] A. Ghiasi, O. Hussain, Z. Qian, and X. Li, "A mixed traffic capacity analysis and lane management model for connected automated vehicles: A Markov chain method," *Transp. Res. B, Methodol.*, vol. 106, pp. 266–292, Dec. 2017.
- [12] A. Mostafizi, S. Dong, and H. Wang, "Percolation phenomenon in connected vehicle network through a multi-agent approach: Mobility benefits and market penetration," *Transp. Res. C, Emerg. Technol.*, vol. 85, pp. 312–333, Dec. 2017.
- [13] S. E. Shladover, D. Su, and X.-Y. Lu, "Impacts of cooperative adaptive cruise control on freeway traffic flow," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2324, no. 1, pp. 63–70, Jan. 2012.
- [14] M. di Bernardo, A. Salvi, and S. Santini, "Distributed consensus strategy for platooning of vehicles in the presence of time-varying heterogeneous communication delays," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 102–112, Feb. 2015.
- [15] Y. Zheng, J. Wang, and K. Li, "Smoothing traffic flow via control of autonomous vehicles," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3882–3896, May 2020.
- [16] J. Lioris, R. Pedarsani, F. Y. Tascikaraoglu, and P. Varaiya, "Platoons of connected vehicles can double throughput in urban roads," *Transp. Res. C, Emerg. Technol.*, vol. 77, pp. 292–305, Apr. 2017.
- [17] H. Liu, X. Kan, S. E. Shladover, X.-Y. Lu, and R. E. Ferlis, "Impact of cooperative adaptive cruise control on multilane freeway merge capacity," *J. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 263–275, May 2018.
- [18] S. Maiti, S. Winter, L. Kulik, and S. Sarkar, "The impact of flexible platoon formation operations," *IEEE Trans. Intell. Vehicles*, vol. 5, no. 2, pp. 229–239, Jun. 2020.
- [19] M. Goli and A. Eskandarian, "Merging strategies, trajectory planning and controls for platoon of connected, and autonomous vehicles," *Int. J. Intell. Transp. Syst. Res.*, vol. 18, no. 1, pp. 153–173, Jan. 2020.
- [20] T.-S. Dao, J. P. Huissoon, and C. M. Clark, "A strategy for optimisation of cooperative platoon formation," *Int. J. Vehicle Inf. Commun. Syst.*, vol. 3, no. 1, pp. 28–43, 2013.
- [21] J. Heinovski and F. Dressler, "Platoon formation: Optimized car to platoon assignment strategies and protocols," in *Proc. IEEE Veh. Netw. Conf. (VNC)*, Taipei, Taiwan, Dec. 2018, pp. 1–8.
- [22] J. Rios-Torres and A. A. Malikopoulos, "Automated and cooperative vehicle merging at highway on-ramps," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 4, pp. 780–789, Apr. 2017.
- [23] J. Rios-Torres and A. A. Malikopoulos, "Impact of connected and automated vehicles on traffic flow," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Yokohama, Japan, Oct. 2017, pp. 1–6.
- [24] S. E. Shladover, X.-Y. Lu, D. Cody, C. Nowakowski, Z. Qiu, A. Chow, J. O'Connell, J. Nienhuis, and D. Su, "Development and evaluation of selected mobility applications for VII," Dept. Transp. California, Inst. Transp. Stud., California Path Program, Univ. California, Berkeley, Berkeley, CA, USA, Tech. Rep. UCB-ITS-PRR-2010-25, 2011.
- [25] C. Nowakowski, S. E. Shladover, D. Cody, F. Bu, J. O'Connell, J. Spring, S. Dickey, and D. Nelson, "Cooperative adaptive cruise control: Testing drivers' choices of following distances," Dept. Transp., California, Inst. Transp. Stud., California Path Program, Univ. California, Berkeley, Berkeley, CA, USA, Tech. Rep. UCB-ITS-PRR-2011-01, 2011.
- [26] S. E. Shladover, C. Nowakowski, J. O'Connell, and D. Cody, "Cooperative adaptive cruise control: Driver selection of car-following gaps," in *Proc. 17th ITS World Congress/ITS Japan/ITS America/ERTICO*, Busan, South Korea, 2010, p. 8.
- [27] C. Nowakowski, J. O'Connell, S. E. Shladover, and D. Cody, "Cooperative adaptive cruise control: Driver acceptance of following gap settings less than one second," in *Proc. Hum. Factors Ergonom. Soc. Annu. Meeting*, Sep. 2010, vol. 54, no. 24, pp. 2033–2037.
- [28] H. Liu, X. Kan, S. E. Shladover, X.-Y. Lu, and R. E. Ferlis, "Impact of cooperative adaptive cruise control on multilane freeway merge capacity," *J. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 263–275, May 2018, doi: 10.1080/15472450.2018.1438275.
- [29] H. Liu, X. Kan, S. E. Shladover, X.-Y. Lu, and R. E. Ferlis, "Modeling impacts of cooperative adaptive cruise control on mixed traffic flow in multi-lane freeway facilities," *Transp. Res. C, Emerg. Technol.*, vol. 95, pp. 261–279, Oct. 2018.
- [30] A. R. Kreidieh, C. Wu, and A. M. Bayen, "Dissipating stop-and-go waves in closed and open networks via deep reinforcement learning," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Maui, HI, USA, 2018, pp. 1475–1480.
- [31] G. Orosz and G. Stépán, "Subcritical hopf bifurcations in a car-following model with reaction-time delay," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 462, no. 2073, pp. 2643–2670, Sep. 2006.
- [32] C. Wu, A. Kreidieh, K. Parvate, E. Vinitsky, and A. M. Bayen, "Flow: Architecture and benchmarking for reinforcement learning in traffic control," 2017, *arXiv:1710.05465*, [Online]. Available: <https://arxiv.org/abs/1710.05465>
- [33] T. Hardes and C. Sommer, "Dynamic platoon formation at urban intersections," in *Proc. 44th IEEE Conf. Local Comput. Netw. (LCN)*, Osnabrueck, Germany, Oct. 2019, pp. 101–104.
- [34] R. W. Thomas and J. M. Vidal, "Ad hoc vehicle platoon formation," in *Proc. SoutheastCon*, Huntsville, AL, USA, USA, 2019, pp. 1–5.
- [35] K.-Y. Liang, J. Martensson, and K. H. Johansson, "Heavy-duty vehicle platoon formation for fuel efficiency," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1051–1061, Apr. 2016.
- [36] R. Axelrod and W. D. Hamilton, "The evolution of cooperation," *Science*, vol. 211, no. 4489, pp. 1390–1396, 1981.



[37] *Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems-5 GHz Band Dedicated Short Range Communications (DSRC)*, Standard E2213-03, Medium Access Control and Physical Layer specifications, 2003.

[38] G. B. Dantzig, *Linear Programming and Extensions*. Princeton, NJ, USA: Princeton Univ. Press, 1963.

[39] B. van Arem, C. J. G. van Driel, and R. Visser, "The impact of cooperative adaptive cruise control on traffic-flow characteristics," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 429–436, Dec. 2006.

[40] T. M. Mitchell, *Machine Learning*, 1st ed. New York, NY, USA: McGraw-Hill, 1997.

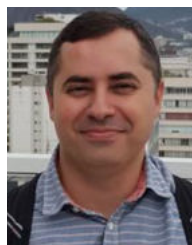
[41] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery Data Mining (KDD)*, Portland, Oregon, 1996, pp. 226–231.

[42] M. Treiber and A. Kesting, "The intelligent driver model with stochasticity-new insights into traffic flow oscillations," *Transp. Res. Procedia*, vol. 23, pp. 174–187, Jan. 2017.

[43] V. Milanés and S. E. Shladover, "Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data," *Transp. Res. C, Emerg. Technol.*, vol. 48, pp. 285–300, Nov. 2014.

[44] J. Costa, D. Rosário, A. M. de Souza, L. A. Villas, and E. Cerqueira, "Protocolo para disseminação de dados em vanets baseado em métricas de redes complexas: Um estudo de caso com sistema de gerenciamento de trânsito," in *Proc. nais do XXXVI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, Porto Alegre, Brasil, 2018. [Online]. Available: <https://sol.sbc.org.br/index.php/sbr/article/view/2478>

[45] D. Zhang, J. Zhao, F. Zhang, and T. He, "UrbanCPS: A cyber-physical system based on multi-source big infrastructure data for heterogeneous model integration," in *Proc. ACM/IEEE 6th Int. Conf. Cyber-Phys. Syst. (ICCPs)*. New York, NY, USA: Association for Computing Machinery, Apr. 2015, pp. 238–247.



**CESAR AUGUSTO CAVALHEIRO MARCONDES** received the Ph.D. degree in computer science from the University of California, Los Angeles (UCLA), in 2008. From 2008 to 2018, he was an Assistant Professor with the Federal University of Sao Carlos (UFSCar), where he was working on future Internet and congestion control. He is currently an Assistant Professor with the Department of Computer Systems, Instituto Tecnológico de Aeronáutica (ITA). He also holds industrial patents from USPTO in congestion control while working at Sun Microsystems. His main research interests include computer networks and security.



**CARLOS HENRIQUE GOMES FERREIRA** received the bachelor's degree in information systems from the Universidade Federal de Viçosa, Brazil, in 2014, and the master's degree in computer science and computational mathematics from the Universidade de São Paulo, Brazil, in 2016. He is currently pursuing the Ph.D. degree with the Universidade Federal de Minas Gerais, Brazil, with emphasis on distributed systems, complex network, and user behavior. In 2019, he became an Assistant Professor with the Universidade Federal de Ouro Preto, Brazil, and a Visiting Scholar with the SmartData Center, Politecnico di Torino.



**ROGER NOBUYUKI KAMOI** received the B.Sc. and M.Sc. degrees in computer technology and software engineering from the Instituto Tecnológico de Aeronáutica (ITA), in 1989 and 2020, respectively. He is currently the Head of IT at Hurmez Supply Chain Solutions. His research interests include machine learning, graph neural networks, platoons, and VANETS.



**LOURENÇO ALVES PEREIRA JÚNIOR** received the B.Sc. degree (Hons.) in computer science from the University of Alfenas (UNIFENAS), in 2006, and the M.Sc. and Ph.D. degrees in computer science and computational mathematics from the University of Sao Paulo (USP), 2010 and 2016, respectively. He is currently an Assistant Professor with the Department of Computer Systems, Brazilian Aeronautics Institute of Technology—ITA. He is also a member of the

Laboratory of Command and Control and Cyber Security. His main research interests include computer networks and cyber security.



**FILIFE ALVES NETO VERRI** received the B.Sc. degree (Hons.) in computer science and the Ph.D. degree (Hons.) in computer science and computational mathematics from the University of São Paulo, in 2014 and 2018, respectively. In 2016, he was a Visiting Scholar with the School of Electrical, Computer and Energy Engineering, Arizona State University, under the supervision of Prof. Ying-Cheng Lai. He is currently an Assistant Professor with the Division of Computer Science,

Department of Computer Methods, Instituto Tecnológico de Aeronáutica (ITA). His research interests include data science, machine learning, complex networks, and complex systems.



**RODOLFO IPOLITO MENEGUETTE** (Member, IEEE) received the bachelor's degree from the Universidade Paulista (UNIP), in 2006, the master's degree from the Federal University of São Carlos (UFSCar), in 2009, and the Ph.D. degree from the Universidade Estadual de Campinas (UNICAMP), in 2013, all in computer science. He was a Postdoctoral Student from the University of Ottawa (UOttawa), in 2017. He is currently a Professor with the Institute of Mathematical and Computer Sciences (ICMC), University of São Paulo (USP). He is also a Leader of the Internet of Things Research Group with a focus on urban computing. His research interests include intelligent transport systems, vehicular networks, clouds, and mobility management.



**ADILSON MARQUES DA CUNHA** received the B.Sc. degree in pilot training from the Brazilian Air Force Academy—AFA, in 1970, the B.Sc. degree in business administration from the Brasilia Unified Learning Center (CEUB), in 1977, the M.Sc. degree in information systems from the United States Air Force Institute of Technology-AFIT, USA, in 1984, and the D.Sc. degree in information systems from George Washington University (GWU), USA, in 1987. He has worked with information systems, software engineering, and artificial intelligence applications. He is currently a Full Professor with the Department of Computer Science, Instituto Tecnológico de Aeronáutica (ITA), Brazilian Aeronautics Institute of Technology.

...