REPORT ON T3.4 ON DISTRIBUTED COOPERATION TECHNIQUES FOR THE USE CASES

Report type	Deliverable D3.4
Report name	Report on T3.4 on distributed cooperation techniques for the Use Cases
Dissemination level	PU
Report status:	Draft
Version number:	1.0
Date of preparation:	2018.03.30

REVISION CHART AND HISTORY LOG

Version	Date	Reason
0.1	2017/09/12	Initial table of contents
0.2	2018/02/15	Contributions from DTU, IMPARA and ISEP
0.3	2018/03/16	First draft
0.4	2018/03/21	Review among WP3 partners
0.5	2018/03/26	Final Draft
0.6	2018/03/29	Final review among the partners
1.0	2018/03/30	Final release

CONTRIBUTORS

ISEP (Leader)	Instituto Superior de Engenharia do Porto, Portugal
DTU	Technical University of Denmark
CNR	National Research Council of Italy
IMPARA	Impara S.r.l.

TABLE OF CONTENTS

1 Introduction

The problems of traffic congestion, pollution, and people safety are becoming more and more important, since the transportation infrastructure remains almost the same, but the number of cars are increasing. A common phenomenon of congested roads is the formation of traffic shock waves, i.e., high-density waves traveling backwards with respect to the cruising direction. Traffic shock waves cause sudden emergency braking that can lead to chain collisions, when the vehicle doesn't follow a safety distance with its preceding vehicle. This is clearly a safety risk.

Platooning can be a solution to this problem. A platoon includes a leading vehicle which is operated by a professional driver, and one or more other vehicles, which are (semi-) autonomously driven, following the leader in close proximity Moreover, several studies have shown considerable reductions in fuel consumption for platooning scenarios. In [1], Bonnet and Fritz could show a 21% fuel reduction for trailing trucks travelling at 80 km/h with an inter-vehicle gap of 10 m. Even the lead truck showed a fuel reduction of 6%. With heavy vehicles accounting for 5% of global carbon emissions, there is a clear environmental and economic incentive for transport industries to implement such type of solutions.

To implement a platooning application, we need technologies to be able to follow the preceding vehicle in the platoon while keeping a safe and small inter-vehicle gap. The platooning benefits is proportional to the distance between platoon members. The smaller the distance, more vehicles we can have on the road (better traffic efficiency), and more aerodynamic drag is reduced (better fuel efficiency). However, reducing the distance clearly causes safety issues. Therefore, a tradeoff is required between inter-vehicle distances and safety to provide fuel efficiency, while preserving safety [2]. Sensor-based cruise control (CC) systems are widely deployed worldwide right now as driver assistance systems. CC helps the driver to maintain a predefined speed to reduce the driver workload in free flowing traffic.

Adaptive cruise control (ACC) is the next generation of CC, which maintains the cruising speed set by the driver and automatically keeps a safety distance from the preceding vehicle [3]. ACC is also deployed and available right now on the market. ACC systems uses radars, which are mounted on the front of the vehicle, to measure the distance from the preceding vehicle. These systems maintain the time headway (a distance in time - the spacing is the time headway times the cruising speed: the higher the speed, the larger the spacing.), which is comparable to the one of human drivers (roughly between 1s to 2s). ACC systems cannot avoid the effect of shockwaves, because of significant propagation delay of radar measurements along the string of vehicles [4]. Smaller inter-vehicle gaps can be realized by using vehicle-to-vehicle (V2V) communication.

To improve the control system reaction time and safely reduce the distance between the vehicles, each vehicle must be aware of the position, status and intention of its surrounding vehicles through message broadcasting. The combination of ACC and wireless V2V communication is referred to as Cooperative Adaptive Cruise Control (CACC). ACC and CACC have automatic longitudinal control only, assume that the driver controls the vehicle using the steering wheel. CACC can be further enhanced to be used for platooning when automated lateral control is also provided. Moreover, further intelligence is required, e.g. protocols for joining/leaving the platoon or assisting other vehicles during on-ramp highway merging.

Even putting aside safety and fuel efficiency benefits, platooning has more benefits for fleet owners and vehicle manufacturers. A report from Scania (2014) on the Swedish truck market shows that the total costs of trucking for a fleet owner consists of fuel, driver, vehicles, tires, administration and repair & maintenance. A third of a fleet owner's costs corresponds to fuel consumption and that is the reason why fuel efficiency is always mentioned together with safety as two important benefits for platooning application. Driver wages also represent a third of the total costs. Platooning can also cause reductions in driver tiredness and increase the output of transportation system, since a driverless truck can drive nearly 24 hours per day, whereas drivers are restricted by law from driving more than 11 hours per day without taking an 8-hour break. Hence, the case for automated driving is strong and increasingly gaining momentum.

In this report, we mainly focus on this particular case of cooperative systems which in SafeCOP is

instantiated mainly in Use Case 3 and 6. These are the focus of the contributions presented in this report.

The report begins with an overview of the state-of-the-art regarding wireless cooperation mechanisms in mobile robots and vehicle platooning, which is fundamental to contextualize the proposals that follow in Section 3. Section 3 goes on presenting the technical contributions of cooperative systems, followed by a brief overview of these and conclusion in Section 4.

2 Overview of the State of the Art on distributed cooperation mechanisms

Wireless Co-CPS, have been receiving significant attention due to its promise of enabling new cooperative applications such as vehicle platooning, and other Intelligent Transportation Systems (ITS) paradigms. However, the idea of having machines cooperate to perform a task has been explored significantly in the robotics community since several years ago. As an introduction to the topic, we begin the section with an overview of the state-of-art on the domain of mobile robots and then advance into the ITS topics.

2.1 Mobile Robots

Towards cooperative cyber-physical systems (CPS), the need to apply robust control in conjunction with wireless communications for efficient mobility of robot teams or group of robots was highlighted a few years ago [5]. It is required a system architecture that provides end-to-end connectivity for autonomous teams of robots as they pursue operator-assigned cooperative tasks. Such an architecture is composed of a cyber component that determines the configuration of the wireless network and a physical component that handles mobility. The challenge of using wireless communication to support cooperative mobile robots is related to the performance of point-to-point wireless link, where the medium is fluctuating from time to time. Thus, the authors adopt a stochastic model for supported rates and develop optimal robust solutions to the wireless routing problem. The problem of cooperative mobile robots can demonstrate many recent studies to address the communication impairments such as delay.

In [6], the authors propose a synchronous formation control approach using distributed controller architecture with sampled-data and communication delays. A sufficient condition has been presented to guarantee exponential convergence of both position errors and synchronization errors. Based on the proposed sufficient condition, given control parameters, an upper bound of the sampling period and the coupling delay that guarantees mobile robots to converge to the desired formations can easily be obtained. In [7], the optimal control framework for leader–follower formation control is provided in the presence of obstacles, where in [8], the same model is assumed by the same authors with additional a limited wireless communication that is utilized to construct the leader's dynamics at the follower by exchanging the leader's orientation, control input, and velocity vectors. To solve the obstacle avoidance problem, the separation and bearing of the follower are modified using a potential function that ensures the new desired position is ahead of the robot in contrast to previous works.

In [9], the authors presented a simulator for rapid prototyping of networked mobile robots and studying their performance. Specifically, the formation control from a consensus problem point of view under a wide range of network conditions and multiple experiments can be studied with this platform. Fixed or time-varying delays, packet dropout rates, the topology of the network, just to mention a few, are accessible parameters to the user. They tested in different use cases developing particular experiments showing the robustness of its own functionality.

Further to the direction of simulating networked mobile robots, in [9], the authors address a new type of application scenario for CPS, which is related to disaster relief situations. To this end, they assume a type of cellular network with control stations, where multiple robots belong to, i.e. they are operating under its own coverage area. Their main goal is to build a network that can provide a robust coverage area by optimizing the quality of service between the control stations. The architecture could vary according to particular use cases. In their works, they assume rescue staff that are in a distance from control stations and non-flow robots, which play the role of alarm robots. The first stage of the problem formulation is to merging robots in distant groups, where the second stage is to optimizing a communication flow. In the first stage, robots in distant groups are merged into one group that covers a desirable area. Since connectivity alone cannot guarantee a high communication quality, in the second stage, the flow between any two stations is further optimized in terms of expected number of transmissions per successfully delivered packet. In both stages, a distributed collision-free controller is proposed to regulate the interactive force among robots. The stability issues of WRNs, where the

proposed controller together with a class of interaction models based on an acute angle test is implemented for robots, are analyzed under both fixed and switching topology. Simulation results show that the expected number of transmissions per successfully delivered packet of the flow is decreased significantly while the team remains to cover the desired area. In addition, the robustness of WRNs is demonstrated by the network reconfiguration capability after one robot fails.

Finally, in [11], authors investigate nodes that are swarm robots with communications and sensing capabilities. Each robot in the swarm may operate in a distributed and decentralized manner to achieve some goal. This paper presents a novel approach to dynamically adapting control parameters to achieve mesh configuration stability. The presented approach to robot interaction is based on spring force laws (attraction and repulsion laws) to create near-optimal mesh like configurations. The simulation results have demonstrated that adaptive control parameters are an effective approach for transitioning a mesh from a quasi-stable to a stable state while significantly reducing the convergence time. For instance, 120 robots in a multiple-obstacle environment reached convergence more than twice as fast as with static control parameters.

Another interesting topic in the area of cooperative mobile robots is related to the former GPS bases solutions, where a GPS-based localization system provides the global position of a mobile robot or object in outdoor environment. However, the GPS-based system has an inherent disadvantage because the GPS signal cannot be available in indoor scenarios. Therefore, a wireless sensor network (WSN) has emerged as one of promising localization technologies for mobile robot applications.

To this end, a localization technique of the mobile robot using a H∞ (check [45] for an overview of H∞) filter that removes the manual effort to tune the noise covariance parameters is proposed in [12]. To avoid GPS, they also assume each robot carries a camera and two markers [13]. The robots move within the field of views (FOVs) of stationary robots. The stationary robots observe the moving robots and record the positions of markers of moving robots. Based on the trajectories of markers, i.e., spatiotemporal features, all the robots are localized using multi-view geometry. Localization requires recovering relative positions, i.e., translation and orientation. In [14], a localization strategy for a group of mobile robots via multiple intra-communications and a formation coordination concept. As aforementioned, the main goal pursued in the formation coordination is to coordinate mobile agents to form a specific formation via local information exchanges without any global reference signals. It is revealed that the localization problem is closely associated with the formation problem of multi-agent coordination systems.

2.2 Vehicular Platooning (ITS)

An Automated Highway System (AHS) is an emergent research topic as it provides attractive services for connected vehicles aiming at creating a new and promising transportation system [24]. Such systems are expected to significantly optimize the road capacity and safety.

The vehicle platoon concept appears as an alternative to the conventional transportation systems. It is composed of a set of vehicles which are automatically controlled and driving at a constant distance by following a leading vehicle. Vehicles are driven closely to each other, which reduces the aerodynamic drag and hence the fuel consumption. Safety critical systems require significant advances in many areas including the architecture. Some related works propose various architectures to improve the safety of vehicular systems. A safety architecture based on hybridization was carried out in the Karyon project [21]. The architecture is divided into two parts. The first one is composed of predictable components for safety properties and it is validated during the design stage. The second part is concerned with components that may be affected by run-time uncertainties. The Karyon safety architecture also contains a safety kernel that defines different levels of service (LoS). At run-time, an LoS that guarantees the performance of the intended functionality in a safe way despite faults and uncertainties is selected.

The system inputs provided to the control system may be collected locally (depending on the vehicle itself perception) or globally (depending on a global reference). Choosing one of the aforementioned control types is still opened for investigation due to many reasons. The global control generates a good trajectory matching but requires reliable V2V (Vehicle To Vehicle) communication [20] while local approaches may only depend on low cost sensors but impact the quality of trajectory matching [34].

That is to say, maneuvers such as joining and splitting require a well-managed coordination between vehicles to be safely fulfilled. A V2V communication network is then indispensable. In order to enhance the reliability of global-based platoon models, considerable effort s have been devoted in the last decades toward improving the V2V communication network as reported in [54]. By contrast, a local approach only relies on the local perception of the vehicle (i.e., always decentralized). Generally, vehicles are equipped with sensors which produce the necessary measurements.

Due to the development of global approaches and the V2V communication improvement, the proposition of platoon models relying on local policies has decreased except for a few studies such as the one in [47] where a possible communication degradation is envisaged and thus on-board sensors are used to estimate the preceding vehicle acceleration.

An inter-vehicle wireless communication in addition to on-board sensors were implemented to attenuate the effect of disturbances along the vehicle string. This approach has proven its efficiency as well in the experimentation conducted in [36].

In the literature, several research works propose different solutions to the control of a platoon. Since the control model strongly affects the string stability (i.e., the error propagation through the platoon), the authors in [52] provide a sliding-mode controller to design the adaptive cruise control from the perspective of the practical string stability. The work in [52] study as well the negative effect of time delay and lag on the string stability. In [53], the sliding mode control is also proposed to control the longitudinal distance while lateral control is ensured with a PID controller. The work reported in [45], the design of the platoon control is based on the robust H_∞ control theory and the vehicle's speed is chosen depending on the road inclinations to decrease the number of unnecessary accelerations and brakes.

Adopting the global approach requires a reliable V2V communication network. In [29], the authors optimize the communication channel scheduling when accessing the communication medium. In [39], a different communication topology is used. Since the distance between the leader and the last vehicle in the platoon may affect the communication quality, a new ring flow graph is proposed to reduce the communication distances.

The previous mentioned works are mainly interested in improving the communication reliability and preventing failure cases. Proposing an alternative behavior when faults occur is not well investigated in the literature. Many other factors may cause the platoon performance degradation other than the communication deterioration such as sensors and actuators malfunction. According to Aradi et al. [19], we categorize failures into two classes: (1) hard faults concerning the vehicle components such as sensors, communication nodes, or any other devices and (2) soft faults mainly concerning weather conditions. In [44] , the authors distinguish a laser range finder failure and propose a recovery approach by reconstructing the sensor-based tracking control, meanwhile only vehicles velocity are involved.

In order to easily handle the vehicle coordination in the platoon, a multi-agent system is proposed as a model to manage platoon tasks in [26] but no data communication is established between the agents. Fernandes and Nunes consider as well the car platoon as a collaborative multi-agent system in [30] where new algorithms to maintain the inter-vehicle constant spacing are proposed, as well as the joining maneuver algorithm. Some works such as Kamali et al. [35] and Colin et al. [25] use formal verification to ensure that the autonomous behavior never violates the safety requirements.

3 Technical Proposals

3.1 Dual mode vehicular platooning control model

Technology Description Table – TDT19
Title: Dual mode vehicular platooning control model
Property: Safety
Type: Control Model
Description: A vehicular platoon control model supporting normal and degraded operational modes
Provider: ISEP
Provided as SafeCOP Technology Brick: YES (TB019)
Readiness: Complete
Integration Status: In progress in UC3
Additional Details: [4]

3.1.1 Motivation

The motivation of this work is to investigate and evaluate the safety of a vehicular platoon under normal and degraded operational modes, which is a crucial problem, as any unexpected error during the platoon operation, whether it is in control and join/leave operation would lead to vehicle collisions.

Our approach defines the concept of adaptive platoon, which uses an auto-reconfiguration based on the operational state of the platoon whether being in the normal or degraded mode. We use Vehicle-to-Vehicle (V2V) communication in addition to perception sensors for increasing reliability of the model. Systems reconfigurations shows its effectiveness in various research areas such as [15]-[18].

We address the problem of re-configurable platoon safety from two perspectives: (1) using a formal model checking approach of vehicular management operation, namely, join/leave operation, (2) using a simulation model for evaluating the safety of a reconfigurable PID controller.

In order to solve disturbance problems, we propose a reconfigurable platoon model. We adopt a switching approach between two platoon modes to handle the communication quality degradation. Our adopted PID controller parameters change according to the activated mode. The platoon runs under the normal mode when no disturbances occur. On the other hand, a degraded mode is activated when a communication problem is detected. We assume that the maximum number of vehicles in a platoon is ten vehicles and we consider the platoon model as a multi-agent system composed of two types of agents: a leader agent and a follower agent. In fact, the platoon system comprehends the characteristics of a multi-agent system including the collective decision making: the decision component consists in selecting a set of actions to be executed by agents in order to collectively progress towards the achievement of an objective, while maintaining a global organization.

Multi-agent systems are characterized by self-organization, robustness, adaptability, simplicity, and redundancy of the agents. It has been shown that this approach is efficient for tackling complex problems such as life-systems simulation [24]. Finally, each platoon model proposition should be evaluated in terms of performance and safety. The contributions hereby presented is three-fold:

1. First, we specify a platoon model and define the main software module to ensure a safe platoon system between the followed and the following vehicles. The novelty of the platoon model is that it incorporates a failure management module that deals with different types of failures, including local and global ones. The platoon model also considers two operational modes, namely the normal mode and the degraded mode, in any case of failure.
2. Second, we formally verify the safety properties in normal and degraded modes for vehicles joining and leaving the platoon. This is achieved through the analysis of the platoon management model that follows the Model Checking approach. We validate and evaluate the performance of this model using UPPAAL model checker software.
3. Third, we investigate the safety of platoon control using a simulation model that we developed

with the Webots simulator[1]. This environment provides reliable vehicle model in terms of dynamics and communication disturbances. We simulate a platoon with 10 vehicles using a PID controller that we design for the platoon control. The simulation addresses the safety of the platoon operation in both normal and degraded modes and provides a statistical analysis on the platoon behavior. This is reported under Section 3.2.

3.1.2 Common platoon model

We developed two different platoon models for both formal verification and simulation. The first model includes additional features such as joining and splitting maneuvers. Thus, we start by clarifying the basic platoon model. This section reviews its main characteristics, the multi-agent system and the failure management.

3.1.2.1 Platoon characteristics

The platoon model in this paper supports global and local decentralized approaches for different situations. In a decentralized approach, the control is distributed among vehicles and is not located in one of them. The considered platoon, denoted by P, is composed of n identical autonomous vehicles V $i\ with\ i\ \in\ [1..n\,]$ and one different vehicle V_0 the global leader (GL): it is driven manually by a professional driver and it is responsible for defining the platoon properties such as the inter-distance, the speed and the followed trajectory. Each vehicle is defined by its position in a global referential $(Xi\,, Yi)$ as shown in Figure 1.

A vehicle V_i is considered as a follower of the vehicle V_{i-1} and a local leader (LL) of the vehicle V_{i+1}. The decentralized approach may increase the dependency in the platoon since each vehicle makes its own decisions and transfers data to the next car that may be affected by the decision error of preceding local leaders unless a powerful control model is set up.

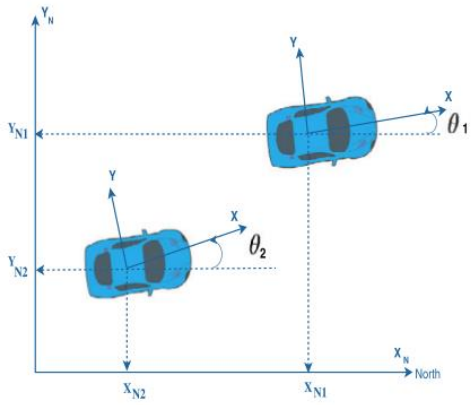

Figure 1 - Follower and leader vehicles.

[1] Commercial Mobile Robot Simulation Software: http://www.cyberbotics.com

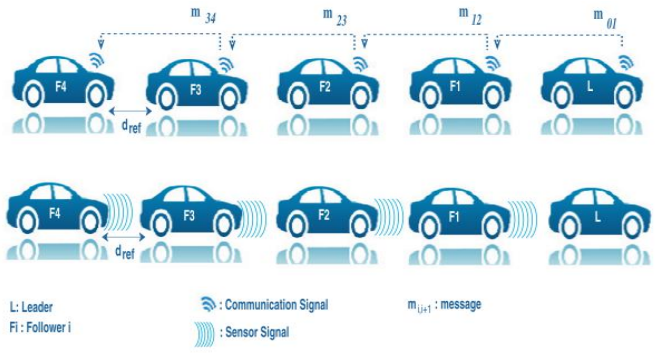

Figure 2- Platoon model.

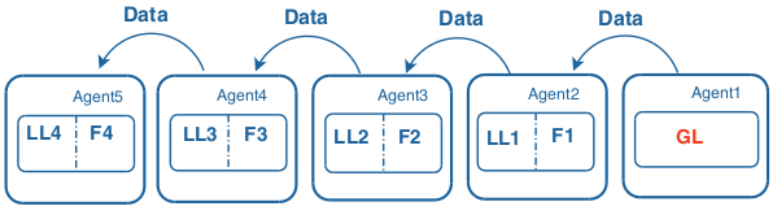

Figure 3 - Platoon multi-agent system

In the flip side, it reduces the crucial role of the global leader. In the global mode, each two consecutive vehicles exchange messages through the V2V communication as shown in Fig. 2. Let us suppose a data message $m_{i,i+1}$ sent by a local leader V_i to its follower. Each $m_{i,i+1}$ contains the following elements: the local leader orientation θ_i^G and the local leader GPS coordinates (Xi, Yi). Once the vehicle V_{i+1} receives $m_{i,i+1}$, it performs the control process to accomplish the tracking goal. The follower should gather the following information: the vehicle orientation (its direction to the north) θ_{i+1}^G, the inter vehicle distance between two vehicles D_{i+1}^G and $m_{i,i+1}$ data. Since the communication network itself is sufficient to ensure a good tracking quality, we limit the use of sensors to the degraded mode. This is a personal choice to simplify the computation of the output data in the normal mode. A vehicle direction θ_i^G in the global mode is given by

$$\theta_i^G = atan\left(\frac{X_{Ni}}{Y_{Ni}}\right)$$

where X_{NI} and Y_{NI} are the vehicle position obtained by the compass using the vehicle GPS coordinates. The inter vehicle distance in the global mode D_i^G is computed using the GPS values of both the follower and leader (transferred by V2V communication) i.e.,

$$D_i^G = \sqrt[2]{(X_i - X_{i-1})^2 + (Y_i - Y_{i-1})^2}$$

In the local mode, the communication network is no more exploited and $m_{i,i+1}$ transferring is interrupted. In Fig. 2 , the second platoon is performing in the local mode. Each vehicle V i is equipped with sensors denoted by S_i to detect: (i) the local leader orientation and (ii) the inter vehicle distance between two vehicles. The switching between local and global modes is clarified in the failure management section.

The proposed multi-agent system is composed of two types of agents: a leader and a follower or followers as shown in Fig. 3. The agent architecture is inspired from a previous research [37]. It is based on three levels:

Communication Unit: On this level, communicated messages are managed depending on the source, destination and request types (e.g., leading, splitting, joining and following). Data are then (i) transferred to the management unit, (ii) sent to other agents, or (iii) ignored if the source is unknown. On the basic platoon model, we suppose that the follower agents identify only the preceding and the following vehicles.

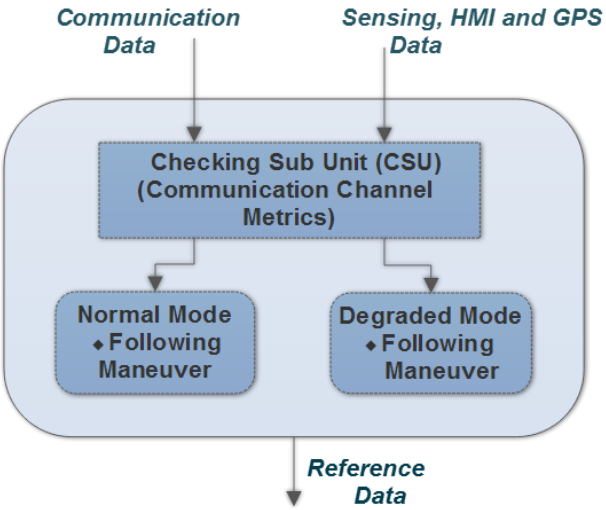

Figure 4 - Follower management unit.

Management Unit: This unit is the decision entity; it is responsible for the agent goal achievement. It receives the information from the communication unit, sensors, HMI (Human Machine Interface) and GPS. This unit is composed of two hierarchical sub-units: The first has a main goal of the decision-making concerning the mode to activate and the second is running the adequate maneuver algorithm depending on notification messages. When receiving a message from the previous vehicle, the Checking Sub Unit (CSU) decides which mode to activate depending on a model described later in this paper. The following process uses the PID controller to compute the reference data which are the vehicle acceleration and its orientation θ. The unit output is basically the reference data to apply to actuators on the purpose of driving the vehicle to the desired position.

Actuators Control Unit: This is the low level control. This unit receives reference data and converts them to applicable variables to steering, throttle and braking actuators.

3.1.2.2 Failure management

Several factors may influence the normal functioning of the platoon system which causes the degradation of the control performance. We categorize these failures into (i) local: affect the functioning of one agent or (ii) global: affect the functioning of the entire platoon. This failure organization offers more flexibility to the platoon. For instance, if a vehicle receives inaccurate data, only one agent switches to the degraded mode (DM) without deteriorating the whole platoon performance.

We suppose that the normal mode is actually a global mode (GM) characterized by a good communication quality. On the other hand, the DM is a result of the communication degradation. Thus, it represents the local mode (LM).

Since platoon systems are proposed to be functional under any condition, our control model as well as the computational architecture has to handle these failures without deteriorating the system performance. Therefore, we enable the agent to change its behavior depending on the selected mode and consequently obtain differently the reference data to keep the platoon on a safe and reliable level. We carry out that switching ability by implementing a sub-unit in the management level responsible for the detection of

failures or anomalies in the car system. Once the degradation condition is detected, the DM is activated.

In order to maintain the good performance of the platoon in GM, we implement (m, k) model in the checking sub unit (CSU) to detect the communication deterioration as illustrated in Fig. 4. We adopt this model since it responds to our model specifications: the CSU requires a predefined number m of lost packets among each k successive packets. We suppose that V_i is a local leader and V_{i+1} is the follower. When using the (m, k) model, we suppose that (i) m is the number of consecutive packets sent by V_i and received by V_{i+1} and (ii) k is a predefined number of successive sent packets. The mode choice for V_{i+1} is given by

$$CSU_{out\,put} = \begin{cases} DM & \text{if } (m-k,k)_{i,i+1} \\ NM & \text{if } (m,k)_{i,i+1} \end{cases}$$

3.1.3 Platoon management modeling

In this section, we detail the platoon model that we implement for the formal verification. We begin by presenting the added features compared with the model presented previously. Next, we investigate the joining and splitting maneuvers. Finally, we expose the formal modeling of the platoon system and present the formal verification results.

3.1.3.1 Platoon model

Joining and splitting require an important amount of notifications and messages among vehicles. Thus, the communicated network and the leader role are developed.

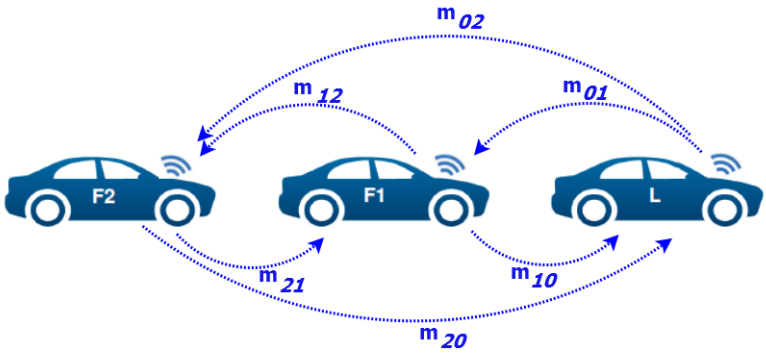

Figure 5 - Interaction between vehicles in the first platoon model.

Table 1 - Leader table.

Characteristics	V_1	V_2	V_3
Destination	{11,0.4,16}	{17,0.4,21}	{100,0.4,50}
$Follower_{id}$	2	3	
$Local\_Leader_{id}$	0	1	2
Mode	Normal	Degraded	Normal

Communication network: On the basic platoon model, each vehicle is only able to receive data from its local leader and send in its turn to its follower. In this model, each vehicle sends and receives messages data from the GL, the preceding and the follower, as shown in Fig. 5. Most of these messages are for the notification purpose when a vehicle is splitting or joining the platoon.

Leader role: The leader has in possession the entire platoon element states in a table in order to manage the vehicular system. Table 2 represents the Leader table of a platoon. It owns each vehicle details: its destination, its follower identity, its local leader identity if the car is not at the platoon front and the mode: normal/degraded. In this example, the platoon comprises four vehicles where the second follower performs a degraded mode. The destination is expressed as GPS coordinates.

3.1.3.2 Joining and splitting maneuvers

The Global Leader (GL) is responsible for managing the joining/splitting request. The GL manages three types of lists:

(i) *Plt _List*: the GL enqueues this list when a joining success notification is received. It contains the identity number i of each vehicle composing the platoon. In the other side, this list is dequeued by GL when it receives a splitting success notification; (ii) *Join_List*: contains identity numbers of vehicles that send joining requests; (iii) *Split_List*: contains identity numbers of vehicles that send splitting requests.

3.2.1. Joining maneuver

We consider a platoon with k vehicles and a vehicle V_{k+1} is not in the platoon. The GL accepts a joining request from V_{k+1} if the platoon fulfills these conditions: the global leader is not treating any request, the V_{k+1} destination $\in$ GL trajectory, both *Join_List* and *Split_List* are empty and the number of platooning vehicles performing under the degraded mode respects the (m, k) firm model. We consider k ($k \geq 0$) as the total number of vehicles and m the number of vehicles performing under the normal mode.

3.2.2. Splitting maneuver

A splitting request is triggered by the GL that checks constantly follower positions and compares them with their destinations.

We suppose that V_s is a platoon vehicle that reaches its position. GL sends a split request to V_s before it reaches its destination by taking into account the splitting process delay. A split request is always prioritized to other requests unless the *split_List* already contains split requests.

A vehicle has the ability to split from the platoon regardless its position (at the rear or in the middle). When splitting from the back, the vehicle decreases the speed and changes its direction. Once out of the platoon, it notifies the leader of the split success. When splitting from the bottom, the vehicle does not change its direction until informing its local leader of the new follower identity and informing its follower as well of the new local leader identity.

3.1.4 Formal modeling

The multi-agent system is useful when using the formal verification approach. In fact, it facilitates the vehicles handling and modeling as well as managing interactions between the different platoon elements. The implemented model is based on the multi-agent architecture. According to the platoon model, the follower is composed of two units and two sub units.

Therefore, a single follower agent is represented by four automata as illustrated in Fig. 6: the follower communication unit automaton, the follower checking unit automaton, the follower management unit automaton and the follower actuator unit automaton.

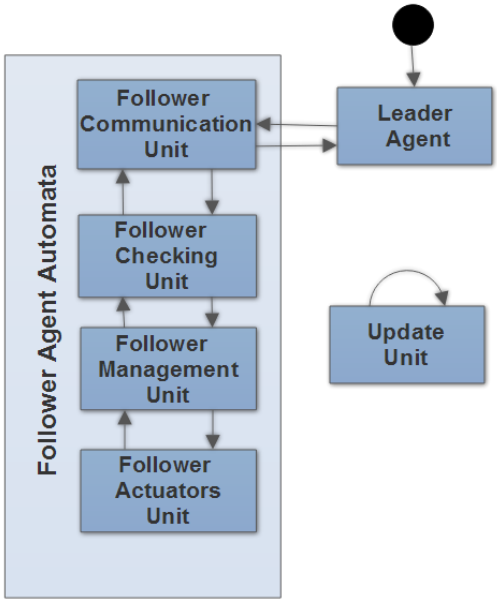

Figure 6 - Interaction model automaton.

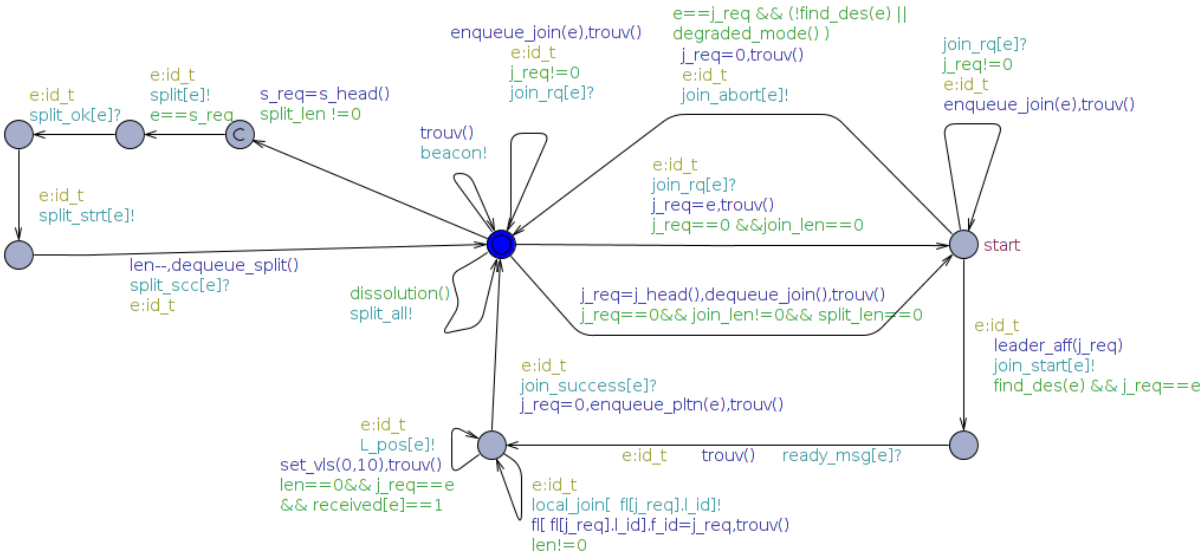

Figure 7 - Leader automaton.

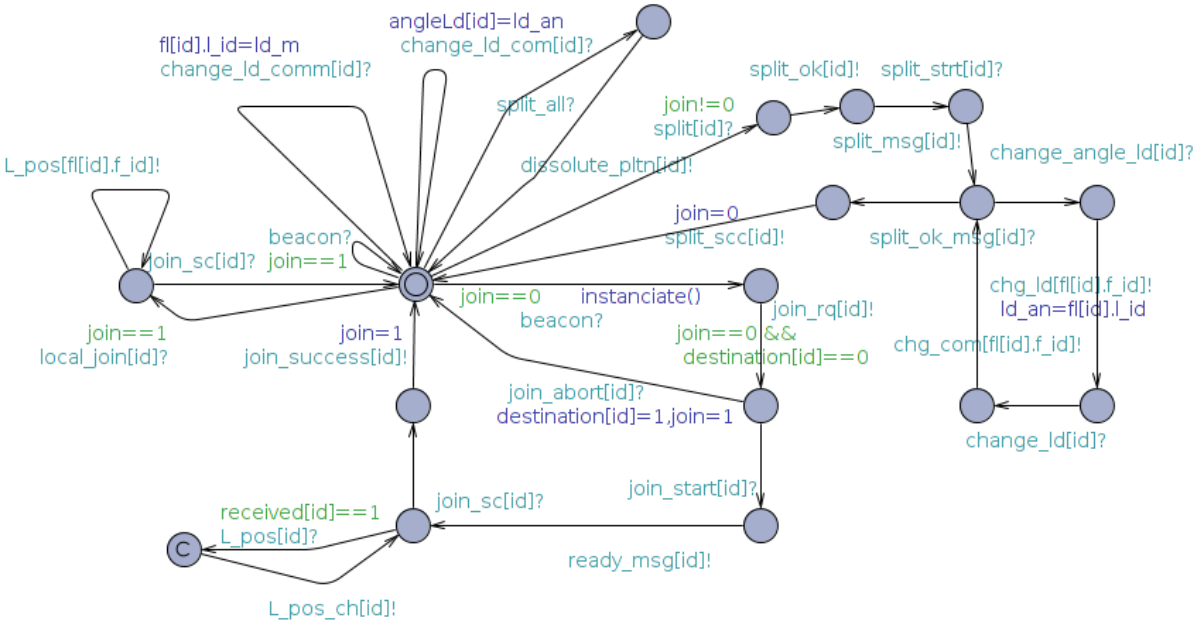

Figure 8 - Follower communication unit automaton.

On the other hand, the leader is represented by one automaton only since it is manually driven. One last update automaton is added to update distances and position values. The leader automaton, illustrated in Fig. 7, interacts with the follower communication unit automaton as visualized in Fig. 8. The follower communication unit sends to and receives messages from the follower checker unit automaton as shown in Fig. 9, which in its turn sends either *normal_mode* or *degraded_mode* signal to the management unit. The management unit, as shown in Fig. 10, consists of three main branches: the joining, the following and the splitting. The reference data computed by the management unit is transferred to the actuator unit, depicted in Fig. 11. Finally, Fig. 12 shows the update unit mostly responsible for computing the different inter-distances and updating the leader and followers positions using respectively the *computeL()* and *compute()* functions.

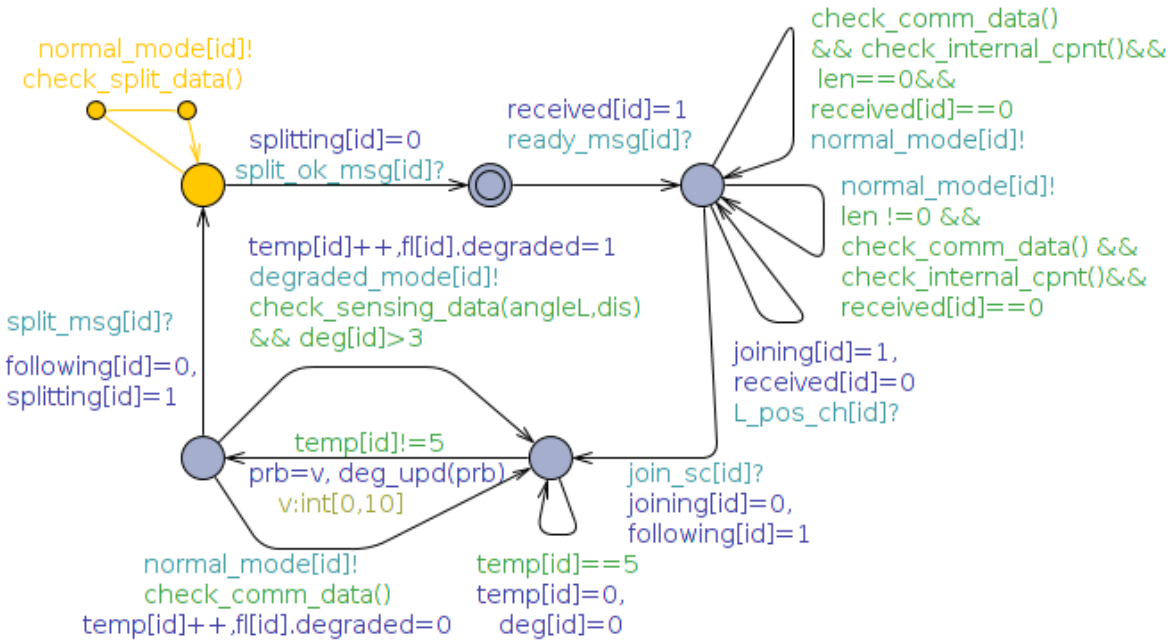

Figure 9 - Follower checker unit automaton.

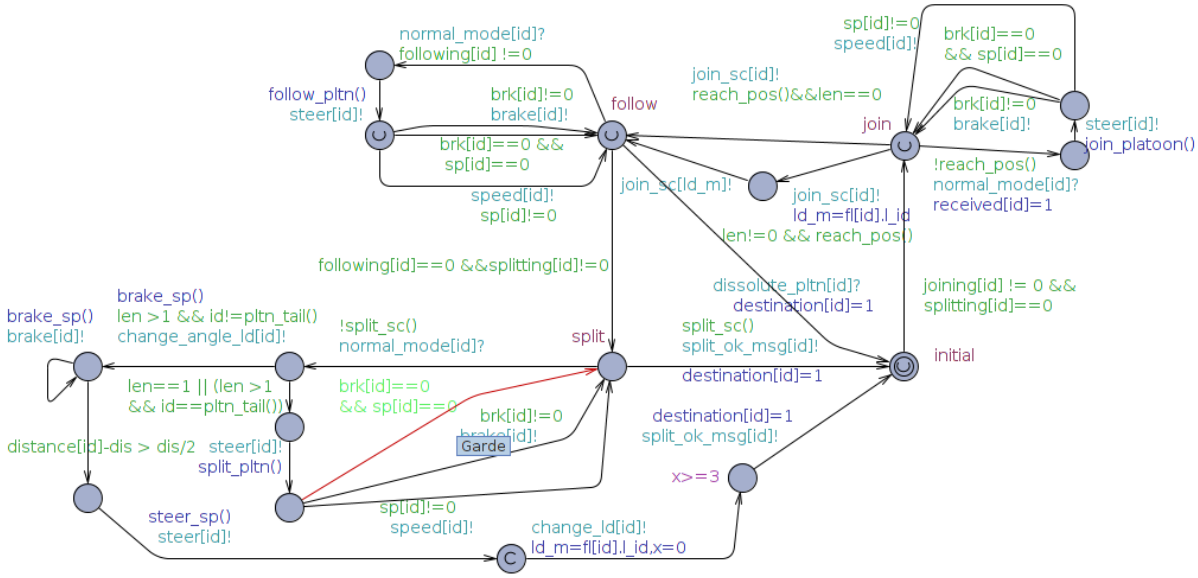

Figure 10 - Follower management unit automaton.

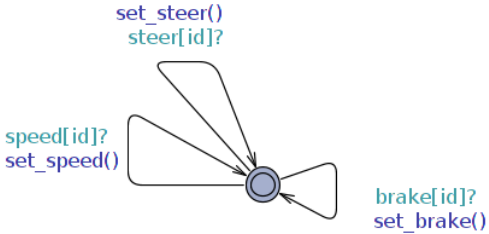

Figure 11 - Actuators automaton.

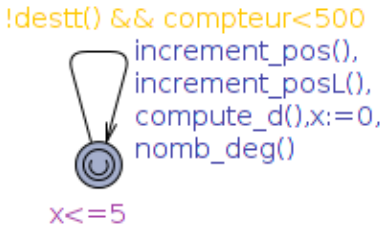

Figure 12 - Update automaton.

3.1.5 Performance evaluation

In the first part of this section, we evaluate the variation of the inter-distance for both modes and for different maneuvers using the UPPAAL software. We implement a four-vehicle platoon (i.e., one leader and three followers). The leader is supposed to be driven at a constant speed equal to 90 km/h. At the start of the simulation, the vehicles are initiated with different GPS coordinates and a redefined inter-distance. In Fig. 13, the three following vehicles are running under a normal mode. The inter-distance of the different vehicles remains generally stable as desired. It oscillates around 10 m value since, as we defined in the following algorithm that *dref* ± 2 m is acceptable to avoid the excessive speed variation. These results prove the efficiency of the implemented PID controller as well as the good reception of data through the agent architecture.

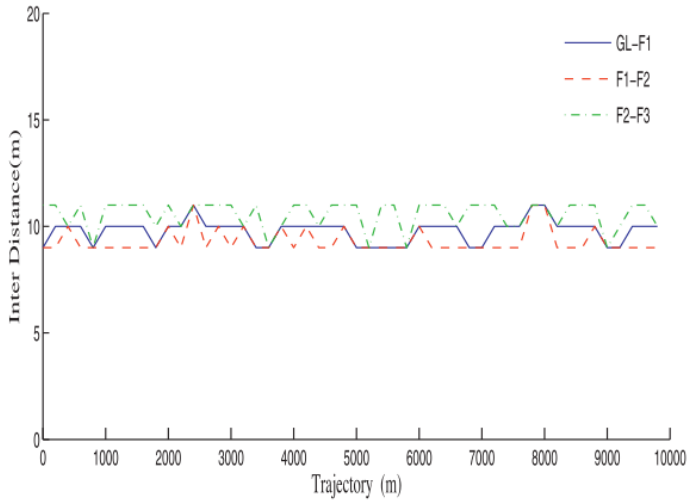

Figure 13 - Inter-distances in the normal mode.

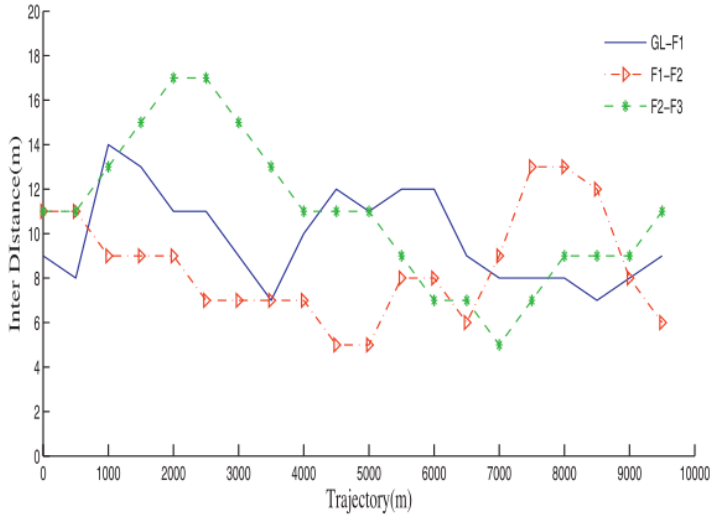

Figure 14- Inter-distances in the degraded mode.

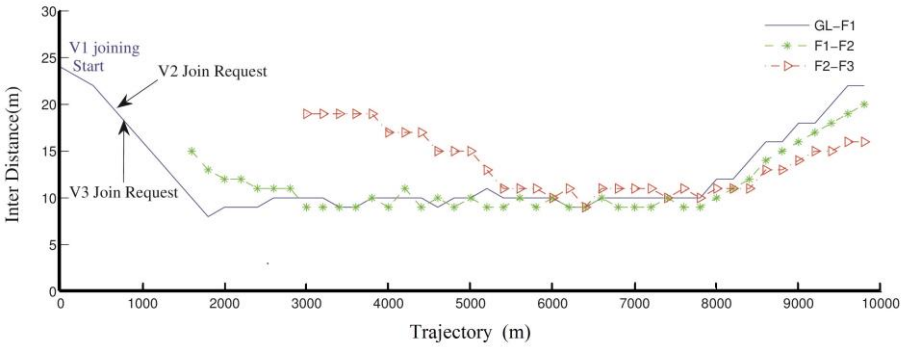

Figure 15 - Inter-distances within splitting and joining maneuvers.

Fig. 14 shows inter-distances while the three following vehicles are performing in the degraded mode which is considered as the worst case. Since data are collected with a sensor in the LM, we add random errors to implement the inaccuracy of sensors. For each iteration, we compute the real inter-vehicle distance and add a random number included in [-2,2]. The longitudinal control is clearly less efficient than that for the global control. However, the safety property during the following maneuver is ensured.

In Fig. 15, V_1 is the first vehicle joining the platoon. Both V_2 and V_3 send joining requests while the leader is already treating the V_1 request. Once the latter reaches the reference distance (10 m), the leader examines the joining list requests with respect to the first in, first out (FIFO) priority. For the splitting process, V_1 is the first to leave. Since V_1 is the local leader of V_2, the distance between V_2 and the leader increases when V_1 is performing the splitting. The same thing occurs with V_3 when V_2 splits from the platoon. As we can see, inter-distances never fall below the reference distance for any type of maneuver.

We show now the formal verification results made with the model checking with a five-vehicle platoon. The model properties are the same previously used. The first property is the safety expressed by the non-collision (i.e., distance between vehicles is above the safety distance). The query is expressed as follows:

$$A[\,](\,forall\,(\,i:\,int\,[1,4])\,distance\,[\,i\,]\,>\,5$$

The term A[ ] means that the property is invariantly true. Distance [i] is a list containing the inter-distances between vehicles. The second property proves the ability of switching the mode when the communication is degraded. The query is expressed as follows:

$$A[\,](\,f\,orall\,(\,i:\,int[1\,,4]\,)\,deg[\,i\,]\,>\,3\,imply\,(\,fl[\,i\,]\,.degraded\,==\,1\,))$$

$Deg\,[i]$ is a counter of degraded quality messages. The property proves that whenever the communication is no more reliable, the vehicle mode is switched to be degraded. The third property describes the dissolution ability of the platoon when the number of DM vehicles exceeds that predefined by the (m, k) firm model. The query is expressed as follows:

$$A[\,](\,nb\,\_\,deg\,>\,0\,imply\,len\,==\,0\,)$$

Len is the length of the platoon. The platoon length is null when more than three vehicles are performing in a degraded mode.

We increase now the number of vehicles to ten in order to assess the impact of the platoon length on its performance. We start by checking the inter-distance values with the following query:

$$A\,[\,](\,forall\,(\,i:int[1\,,9]\,)\,(\,foll\_management\,(i)\,.follow\,\&\&\,fl\,[i]\,.degraded\,==\,0\,)\,imply\,distance\,[i]\,<\,15\,)$$

This query is not satisfied, which means that the inter-distance of each vehicle during the following maneuver (i.e., $follmanagement\,(i).follow$) and performing in the normal mode (i.e., $[i].degraded\,==\,0\,)$ may exceed 14 m. This result proves that the maintenance of the predefined

inter-distance is effected by the number of vehicles. However the platoon safety is insured. Finally, we test the impact of a full-brake on the safety criterion within the ten-vehicle platoon. The property to verify is given by:

$A\,[\,]\,(\,forall\,(\,i:\,int[1,9]\,)\,(\,foll\_management(i)\,.\,follow\;\&\&\;brake\;==\;1)\;imply\;distance\,[i]\;>\;2\,)$ where brake is a global variable that is automatically set to 1 if the counter reaches 400. In this case, the leader speed is decreased to 20 km/h and then to 0 km/h in order to simulate the full-brake. During the full-brake, the inter-distance does not fall behind 2 m. In the next section, we evaluate the impact of the platoon length in a more realistic approach using the Webots software.

3.1.6 Platoon control modeling

We are interested now in the control quality and its effect in the following maneuver. We study as well the impact of the communication disturbance on the following performance. We start by detailing the PID controller.

Once the management unit decides which mode to adopt, the tracking algorithm is performed using the PID controller. Each vehicle is in charge of determining its own reference data (direction and acceleration) using a control model based on the collected information from V_{i-1} ($i \in [1..n]$). In the platoon system, the tracking depends on two control types: longitudinal control and lateral control.

3.1.6.1 Longitudinal Control

It concerns the braking and throttle actions. To ensure the safety of the platoon, the vertical distance between two successive vehicles should not be less than the safety distance fixed by the global leader. It is represented by *dref* in Fig. 2.

The inter-distance is bounded for both modes (G: Global, L: local) by *dref*. In the global control, we use the leader coordinate values to compute the inter-vehicle distance since it provides us with a more accurate PID output. The inter-vehicle distance computed with vehicles coordinates is more precise than the distances measured with a sensor due to the noise that may be added. Nevertheless, when communication network data are not more fully reliable, we use sensor outputs in a local control and change the gain values (used in the PID formula) for a better control result. The controller output is the needed acceleration that leads to the safety distance between vehicles. The PID controller equation applied at time t for vehicle V_i is presented next for both normal and degraded modes, i.e.,

$$u_i(t) = K_P^{G,L} * e_i(t)^{G,L} + K_I^{G,L} * \int e_i(t)^{G,L} dt + K_D^{G,L} * \frac{de_i(t)^{G,L}}{dt}$$

where $u_i(t)$ is the acceleration value of vehicle V_i , d_{ref} is the set point which is the reference distance between two vehicles (limit distance), $K_P^{G,L}, K_I^{G,L}, K_D^{G,L}$ denote respectively the Proportional, Integrator and Derivative gain constants in both modes, $e_i\ (t)^{G,L}$ is the error value at time t and is the difference between the measured inter-vehicle distance and reference distance, and D_i is the process variable which is the current distance between V_i and V_{i-1}.

3.1.6.2 Lateral Control

Lateral Control is related to the steering action. Each vehicle has to apply the same deviation angle as its local leader when it reaches the same position as the preceding vehicle. For both local and global control, we have the same lateral PID since the deviation of the leader is measured in both cases. The vehicle V_i changes its direction as follows:

$$\forall\, i > 1 \begin{cases} For\ t = t_0,\ \theta_{i-1}^{G,L}(t) = \theta_c \\ For\ t = t_0 + \delta t,\ \theta_i^{G,L}(t) = PID_{Steer}(\theta_c) \end{cases}$$

where δ t is the needed time to travel the inter vehicle distance between V_i and V_{i-1}, and PID Steer is the control function defined later. For this control, the set point is the leader angle at t − δ t and the error is then the difference between the current follower angle and the set point. The PID Steer function is given by:

$$\theta_{PID} = K_{pa} * e_i(t) + K_{ia} * \int e_i(t)dt + K_{da} * \frac{de_i(t)}{dt}$$

where, $e_i\ (t)$ is the error value of vehicle V_i, θ_L is the set point which is the direction of vehicle V_{i-1} at $t - \delta t$, θ_F is the process variable which is the vehicle V_i direction at t, and K_{pa}, K_{ia}, K_{da} denote respectively the Proportional, Integral and Derivative gain constants.

3.2 Platoon control modeling in Webots

Technology Description Table – TDT20
Title: Vehicle Platooning Simulator in Webots
Property: Safety
Type: Tools
Description: A vehicular platoon simulation.
Provider: ISEP
Provided as SafeCOP Technology Brick: YES (TB020)
Readiness: Complete
Integration Status: Being used to test control models in UC3
Additional Details: [4]

3.2.1 Webots simulation model

We simulate the proposed PID controller and multi-agent architecture of the platoon presented in Section 3.1 using the Webots simulator, which provides a realistic simulation framework for mobile robots. We use a 3D model of a BMWX5 car for all platoon's vehicles.

The vehicle is modeled with realistic physics properties. Our platoon model is composed of ten vehicles (i.e., a leader and nine followers) with the same physics properties. The experimentation track has a square shape to correctly evaluate the lateral control. A vehicle is equipped with an emitter, a receiver, a sickLms laser rangefinder, a distance sensor, a GPS and a compass. Bellow, we present a screeshot of the simulator with a platooning doing a slalom course.

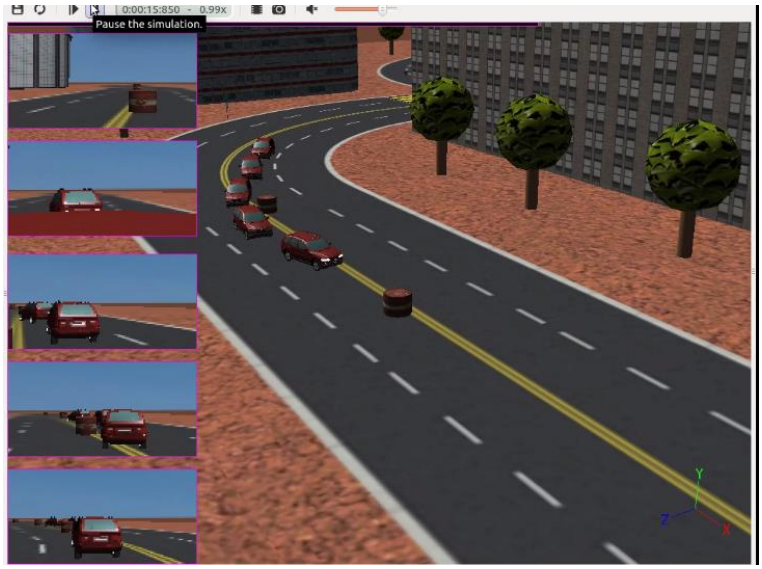

Figure 16 - Platooning simulation doing a slalom course

We assume that all vehicles in the platoon are identical and the nine followers use the same controller. The safety distance is predefined to 7 m and the average speed of the vehicle is up to 50 km/h. In order to demonstrate the advantage of the proposed platoon model, we have performed simulations for both normal and degraded modes.

Normal mode:

For the next scenarios, the communication quality is stable and no disturbances occur. The longitudinal PID controller and the lateral PID controller gains are respectively defined as follows: $K_P = 2.0; K_I = 0.005; K_D = 2.0; K_{pa} = 2.0; K_{ia} = 0.005; K_{da} = 1.0$

Fig. 16 describes vehicles' trajectories when the platoon drives around the track. The leader and the first four follower's trajectories are almost overlapped. On the other hand, the next four followers score mismatched trajectories compared with the leader mainly when they reach sharp turns. The last follower

trajectory is distinctly different. Moreover, its trajectory is not complete because of the large track deviation. We can also notice that the discrepancy between trajectories is very small between two following vehicles and their trajectories are very similar to the leading one. This is due to the accurate information that are received from the preceding vehicle and also the well tuning of the PID controller.

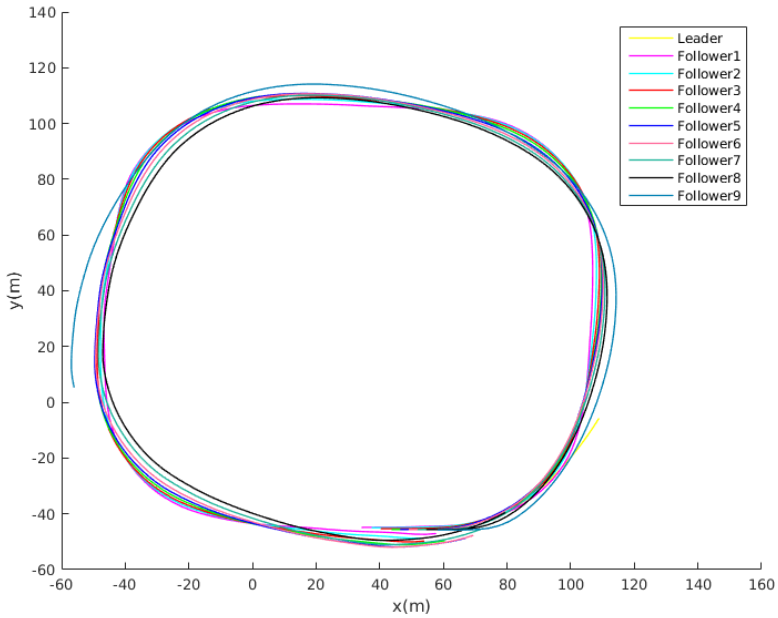

Figure 17- Trajectory matching of different vehicles.

Fig. 17 describes the different lateral error of platooning vehicles. In fact, this figure reveals a very important characteristic of our platoon. We notice that different error plots have the same shape but with different values: it shows the light propagation of error through the platoon that we cannot see in Fig. 16. This error growing forces us to limit the platoon length for the safety condition. We have taken into consideration the mechanical constraints for the controller and engine by putting some constrains on the maximum value of the steering radius.

Fig. 18 describes different inter-distances between vehicles. Values vary between 6 m and 8 m except for the last inter-distance value at t = 110 s that grows quickly to reach 12 m. This is due to deviation of the last follower.

Fig. 19 describes inter-distances when a full brake occurs. In fact, the number of vehicles does not affect the longitudinal control: vehicles are able to keep a safe distance even in a full brake. We notice that the distances are still in the average of [6, 10] m for all the vehicles and the lines end at different time. In fact, as a result of the predecessor-following communication model, vehicles receive the full-brake alert at different moments. Therefore, the period needed to completely stop varies from one car to another.

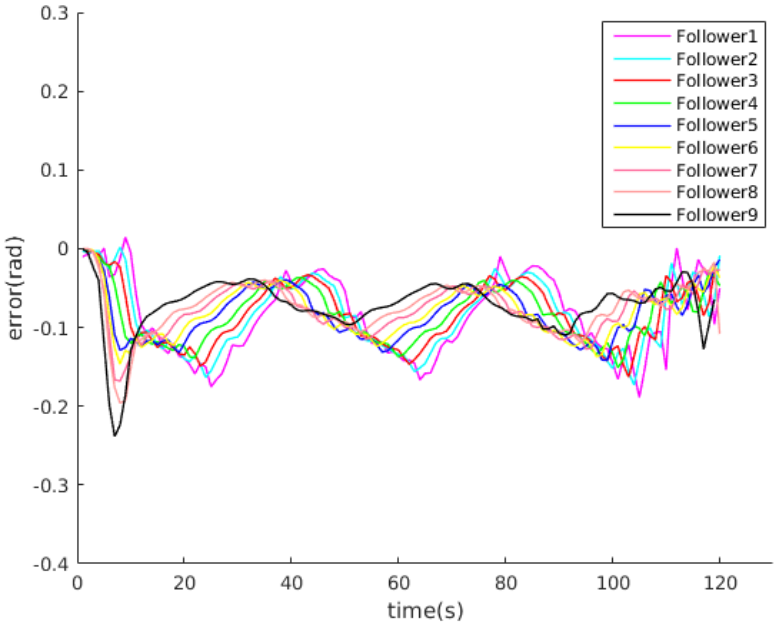

Figure 18 - Lateral errors of different vehicles.

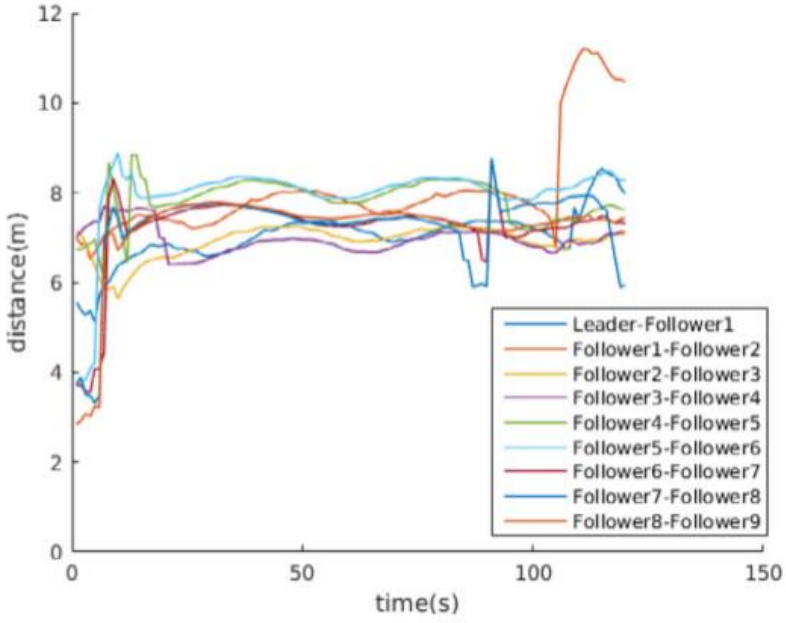

Figure 19- Inter-distances.

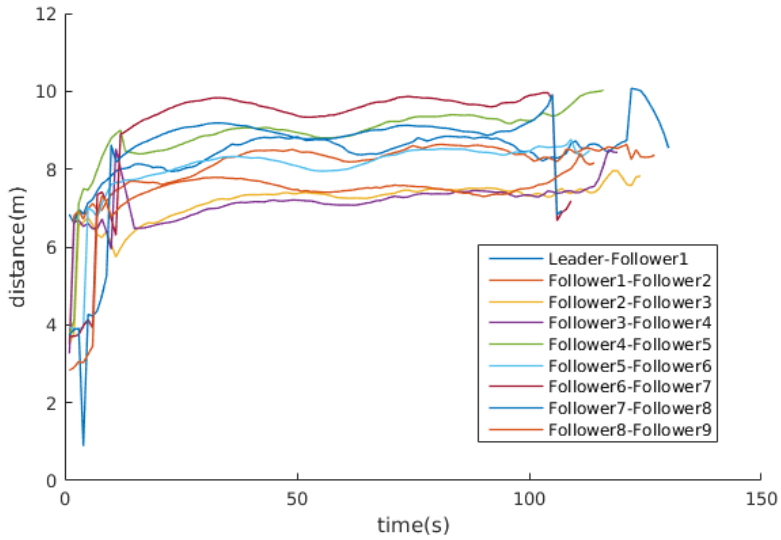

Figure 20 - Inter-distances during a full-brake.

From these results, we conclude that the number of vehicles has no strong impact on the longitudinal control quality except that values may be slightly larger than the safety distance. On the other hand, the tracking quality is much more effected due to the error propagation. Relating to our platoon, four followers are capable of following the leader vehicle with high performance. A platoon length of eight followers is acceptable but with a less efficient lateral control. Finally, adding a tenth follower is not permitted since the trajectory mismatching is large and the safety criterion may be neglected.

The degradation of tracking quality for the last follower may lead us to question the quality of the PID controller and the adequacy of the gains. The tuning of PID gains is performed using the trial and error method. In fact, we performed scenarios for different gains, and the stated one offer the best tracking quality. When comparing two by two the trajectories of vehicles, we notice that they almost overlap which proves the effectiveness of the PID control. On the other hand, the followers are not following the reference path of the GL; in fact, the first follower is the only one tracking the GL path.

The rest of vehicles are following their predecessor path (i.e., local leader) which explains the degradation of the tracking quality compared with the GL path. This basically stems from the following predecessor communication topology although this approach has the advantage of decreasing the number of communication messages.

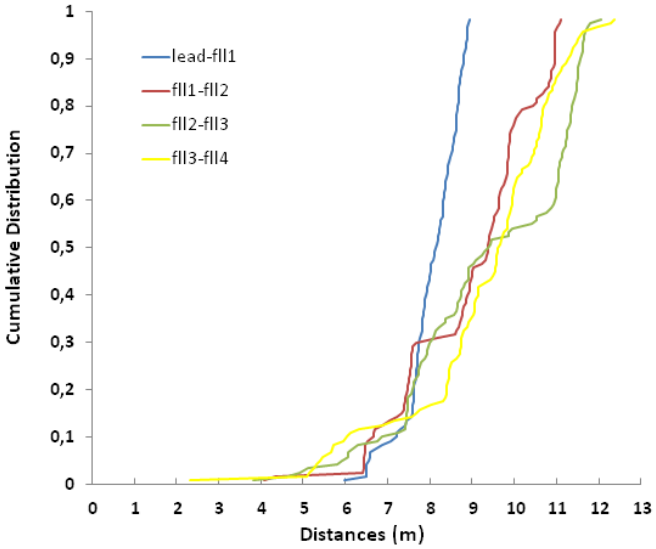

Figure 21 - Cumulative distribution probability of distances in degraded mode.

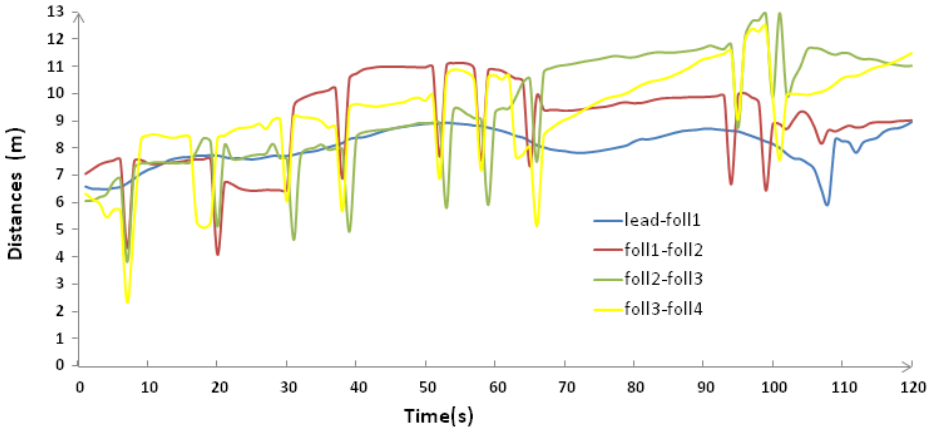

Figure 22 - Inter-distances in degraded mode. (For interpretation of the references to color in the text, the reader is referred to the web version of this article.)

Degraded mode:

In order to simulate the communication degradation, we apply the Bernoulli distribution with a probability of packet loss equal to 20%, implying that only 80% of the packets are correctly exchanged between two vehicles. We set then a random variable depicting the packet loss probability. We emphasize that we study the worst case that all the vehicles are switching the degraded mode at the same time. The longitudinal PID controller and the lateral PID controller gains are respectively defined as follows:

$$K_P = 0.7; K_I = 0.005; K_D = 1.0;$$

$$K_{pa} = 0.8; K_{ia} = 0.008; K_{da} = 0.0.$$

For the next scenarios, we limit the number of vehicles to five. We observe that a platoon with a larger number of vehicles simultaneously performing in a degraded mode is no more predictable. We start by studying the inter-distances variation in Fig. 20 that describes the cumulative distribution function of distances. This figure gives an indication on the inter-distances boundaries. Note that 20% of distances are estimated to be around 7.5 m. Most importantly, values do not fall below 2 m and exceed 13 m. It is important to notice that these values are the same in the normal mode. Fig. 21 exposes more clearly the distances variation from one follower to another. We notice disturbances for different slots of time, e.g.,

at t = 8 s, 20 s and 30 s. These disturbances are due to the mode switch. While activating the degraded mode, the vehicle does not apply any vehicle control which explains the rapid decrease of distance. However, these fluctuations have no strong impact on the platoon safety. In Fig. 22 , the trajectories of the vehicles are almost overlapped, except for the deviation on the second turn. It is probably caused by a switch to the degraded mode when turning. On the basis of this result, we conclude that our platoon model safely supports five vehicles that perform simultaneously in the degraded mode.

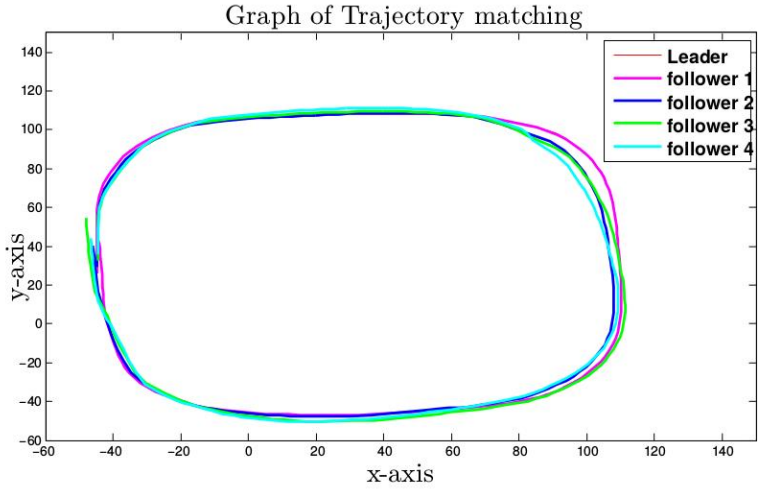

Figure 23 - Trajectory matching of different vehicles in degraded mode.

The switching between modes causes an instantaneous perturbation in the platoon. However, the platoon rapidly recovers itself and ensures the platoon safety.

String stability:

String stability is a significant property of a platoon. It refers to the capability of vehicles in attenuating the error propagation. In general, string stability is defined with regard to spacing errors, i.e., the spacing errors between vehicles are not amplified when propagated throughout the platoon. Denote by e_i, e_{i+1} the spacing error between two adjacent vehicles.

Then, string stability implies that

$$\left\| \frac{E_{i+1}(s)}{E_i(s)} \right\|_\infty \leq 1$$

where $E(s)$ is the Laplace transform of the spacing error.

We assess the string stability in our platoon for both modes by analyzing the inter-distances plots. In Fig. 19, we notice the expansion of distances for different vehicles. However, the distance expansion does not decrease from one vehicle to its follower. In other words, the error is not expending through the platoon. We notice the same behavior for the degraded mode in Fig. 21 where the green line (follower2-follower3) is beyond the yellow line in some parts of the plot. The control model has a great impact on the string stability as well as other factors. For our case, each vehicle is measuring the input data regardless the situation of other vehicles in the platoon which guaranties the string stability. From the figures of several simulations, the convergence rate of spacing errors is fast under the designed PID controller.

Using a dual mode controller can substantially reduce the unpredictability of spacing, which is mainly caused by the changing velocity of the vehicles in the platoon, and the spacing error is not amplified along the platoon, which can guarantee the stability of the platoon system.

3.2.2 Discussion

It is known that safety is one of the most inevitable concerns in the development of more autonomous road vehicles. Due to their complex behaviors and hybrid nature, the implementation of a safe platoon in different situations requires further testing and validation. In this work, a new methodology that has the potential to certify the safety of platoon systems has been proposed by proceeding various steps.

This methodology: starts from the formal verification which applies an exhaustive verification of the most primary safety requirements in the platoon. Then, a reconfigurable controller is modeled, implemented and simulated on Webots. Therefore, the safety is approved by the experimental results.

In the literature, it is known that local approaches suffer from the anticipation error problem. In fact, the majority of the available studies focus on platoons performing in a straight line (i.e., no lateral control) and asses the string stability. Only a few deal with both lateral and longitudinal control.

In [28] , the authors implemented a three-vehicle platoon using a local control approach. The simulation results exposed a lateral error that reaches 0.4 rad. On the other hand, by our control model, vehicle perception capabilities in the DM which is a sensor-based solution has a good quality in terms of inter-vehicle distance error and of lateral errors, i.e., the inter-vehicle distances do not fall below 2 m for both modes and the lateral error obtained did not exceed 0.2 rad.

Consequently, we have proved that the PID controller is not merely convenient for lateral control like the work in [53] where they choose the sliding mode control rather than a PID controller for the longitudinal control and the number of vehicles was limited to four.

The advantage of this methodology over other approaches is that during the GM, LM and in different scenarios, the safety of the platoon is well-studied and guaranteed thanks to the proposed methodology of verification where safety properties are addressed formally and experimentally. In [40]a bond graph model-based reconfiguration is proposed where the model can switch to the DM when a fault occurs. However, only one scenario is studied when a fault (i.e., a fault is identified based on the residuals of the wheels) occurs. Another recent work in [27] uses a reset control technique to adapt the inter-distance based on a simple bicycle model. Regarding the inter-distances, the performance of the platoon was better than the one with a linear control. However, more experiments are required to ensure the safety of the platoon. For example, the lateral control is not considered.

In [47], a new approach is presented to predict acceleration values in case of the communication loss. However, through all the simulation results, the plots are drawn only for 20 s which is considered as a small period to assess the quality of control. To the best of our knowledge, there is no study in the automated highway system field that analyzes the relation between the platoon length and the tracking quality. Moreover, the simulation models are generally based on different simulators that require the implementation of the car physical model such as in [41].

The proposed approach to implement a cooperative and safe platoon is expected to shape a methodology and furthermore open a new line in this direction by properly modeling and analyzing the safety properties of a reconfigurable controller. This can be done by following the proposed methodology that formally verifies safety properties, and then implements the model and evaluates various scenarios. Thus, one can be confident that the platoon has the potential to be safe especially when abrupt threats appear.

3.2.3 Conclusion

We presented an approach that deals with the linear platoon safety. The approach is based on a reconfigurable architecture enabling vehicles to withstand the communication failure. We propose an efficient PID controller for both longitudinal and lateral control models to ensure a safe tracking and following maneuver. We also validate joining/splitting maneuvers using the formal verification approach. Safety properties and the dissolution feature are verified as well using the model checking.

Using the Webots software, we prove the impact of the platoon length on the quality of the tracking and define the limited number of vehicles in the platoon. Our longitudinal controller proves to be efficient to support a great number of followers. However, the lateral error propagation through the platoon causes a trajectory mismatching between vehicles. The Webots software provides better evaluation of

both longitudinal and lateral controller compared with the Uppaal software. This is due to the vehicle dynamics abstraction when using the formal verification.

In a future work, we intend to grasp the communication quality and classify different types of the communication degradation to increase the platoon robustness with a large number of vehicles. We will be focusing then on the V2V communication layer and develop a suitable algorithm for better management of the delay and also the possible disturbances. In addition, we will study the platoon deadlock control and scheduling problem from a logical viewpoint by using Petri nets [22],[23], [42].

3.3 Cooperative resource allocation and scheduling for 5G eV2X Services

Technology Description Table – TDT21
Title: Cooperative resource allocation and scheduling for 5G eV2X Services
Property: Timeliness
Type: Protocol
Description: Cooperative resource allocation and scheduling procedure for supporting V2X communications.
Provider: DTU
Provided as SafeCOP Technology Brick: NO
Readiness: N/A
Integration Status: N/A
Additional Details: N/A

The requirements to enhance 3GPP support for 5G eV2X service are described in [55], where design requirements for 25 different 5G V2X use cases are presented. Such extensive requirements depend on different features of the system design. A short list of the most important 5G eV2X use cases as specified in 3GPP Rel.15 is given below:

- eV2X support for vehicle platooning: information exchange such as join and leave, announcement warning etc.
- eV2X support for remote driving: remote driving is quite different from autonomous driving where a vehicle is controlled remotely.
- Automated cooperative driving for short distance grouping: automated cooperative driving is considered a combination of vehicle platooning with high demanding communication among the vehicles.
- Collective perception of the environment: vehicles can exchange real time information, collected by vehicle sensors, among each other.
- Cooperative collision avoidance: vehicles should be able to evaluate the probability of an accident by coordinating manoeuvres using cooperative aware messages and data from sensors.

Table 2 summarizes the different use cases with the corresponding key performance indicators (KPIs) such as message payload size, payload reliability and latency. 5G use cases should be enabled with technologies, which can guarantee the requirements under an "out of 5G coverage" application scenario.

Table 2 - Payload message size, reliability and latency 5G eV2X requirements

5G Use Case	Size (Bytes)	Reliability (%)	Latency(ms)
Vehicle Platooning	300-400	90	25
Remote driving	300-400	99.99	5
Aut. Coop. Driving	1200	99.99	10
Coll. Perc. Of Envir.	1600	99	100
Coop. Collis. Avoid.	2000	99.99	10

3.3.1 Cooperative resource allocation and scheduling

In the standardized SPS resource allocation approach transmission collisions among the different vUEs may occur in the reselection window region found in the SPS procedure (Figure 24). To this end, the management of concurrent reselection avoidance is required. Our proposal consists of transmitting the counter values in each packet transmission so that the vUEs are aware of future concurrent reselection.

The vUEs will trigger counter reselection, if any of the received counters in the last RRI coincide with their own current counter. In this way, the system performs resource reselection close in time leading to a more free-of-collision reselection window.

We present below the counter learning and reselection (CLR) mechanism. The counters considered for counter reselection (set A, hereafter) consists of the counters lower than the current counter. The interest in choosing only between the lower counters is based in not delaying the resource reselection triggering. In this way, if the transmission of the UE involved in the counter reselection is colliding with another transmission, we do not prolong the collision time. Then, considering the received counter values during the last RRI as $C_{RX} = [c_0, c_1, .., c_{N-1}]$, where is the i-th received counter value and N is the number of received counters, the non-available counters (set B, hereafter) consists of C_RX \belongs (C_RX-1), where $C_{RX} - 1 = [c_0 - 1, c_1 - 1, .., c_{N-1} - 1]$. $C_{RX} - 1$ are also considered in set B because, in case one of them is chosen, it will coincide with some surrounding UE counter when this surrounding UE has gone through another RRI, causing the same problematic situation that we are trying to avoid, which is that two close UEs have the same counter value simultaneously. Finally, when performing counter reselection, the UEs will randomly choose, with equal probability, one of the counters in the set Z=A\B.

Another difference comparing to the standardized SPS approach is based on the counter value that is chosen when the current counter reaches 0 value. In the standardized approach, a counter range is specified so that the UEs randomly choose, with equal probability, one of the counters in the counter range defined as [C1,C2]. In our proposal, we fix the chosen counter to 63, i.e. C1=C2=63. This is because in case that the counter values among close UEs are already separated by the counter reselection approach they will probably continue to be separated in case they choose a fixed counter when the current counter expires, but they could 'collide' again if they choose a new counter randomly between a specific range. Finally, the parameter p_{RK} is removed from the system (or set to 0) since it no longer improves the performance of the algorithm. The operational procedure of the overall proposal running in each UE is given in Fig.6 below, where C represents the current counter value of the UE and C_{RX} represents the received counters from the surrounding UEs during the last RRI period.

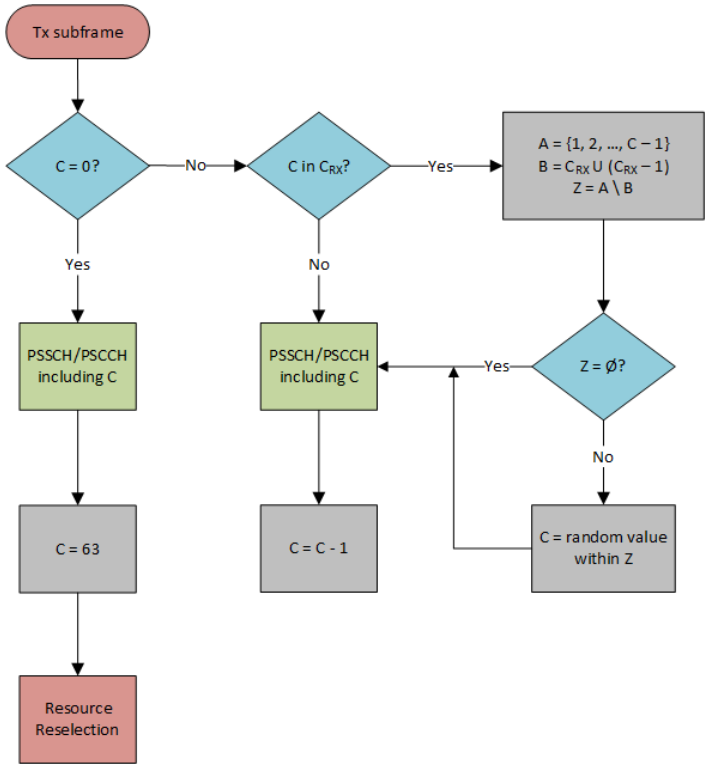

Figure 24 - CLR scheduling mechanism for eV2X services.

Figure 25 depicts results for the proposed approach using the parameter values from Table 2 (apart from C1 and C2) compared to the obtained results for the simulated standardized approach (in this case, using p_{RK} =0.4), We also assume ideal channel conditions, i.e. SNR values with no errors.

We can see that the proposed approach clearly achieves lower collision probability (0.34 % after 1000 secs in the simulated scenario) than the standardized approach (1.43 %) in the dynamic scenario. Considering a static scenario, i.e., without UEs leaving and joining, we can see that the difference is even higher between both approaches. In fact, we can observe that in the static scenario the average collision probability is decreasing throughout all the simulation in our proposal. This is because except for some collisions at the beginning of the simulation (caused by a random initialization of the UEs resources in the simulator) there are no more collisions afterwards.

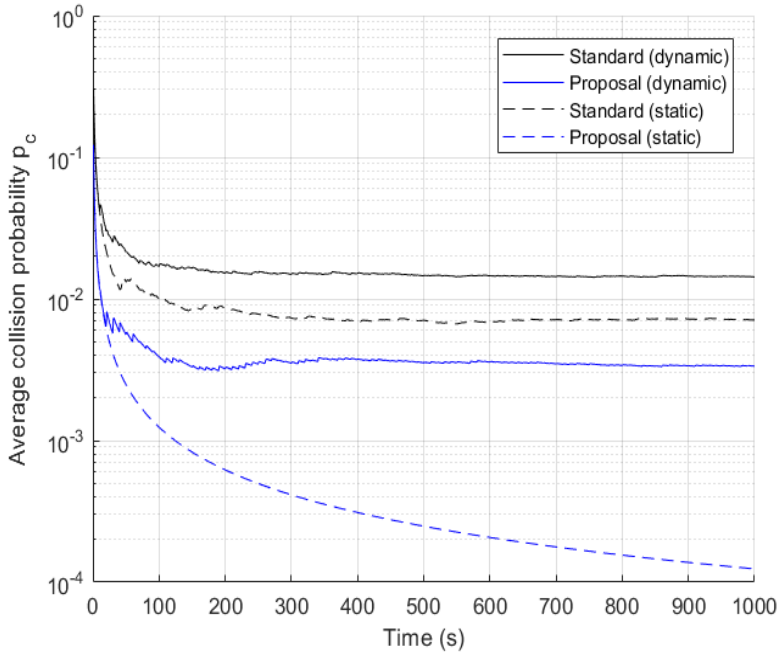

Figure 25- Average collision probability of standardized SPS and proposed scheduling for dynamic and static applications scenario

We assume now two application scenarios (ASs), which can accommodate some of the features listed in the 5G use cases.

In the first application scenario (AS1), we consider a use case similar to "vehicle platooning" where some vUEs periodically transmit packets with a payload of 3240 bits (about 400 bytes) while fulfilling a 25 ms latency requirement. In order to guarantee this latency requirement, we choose RRI=20ms and T2=20ms. QPSK is employed for AS1 due to the better BLER performance compared to 16QAM. We simulate two grid configurations with two different N_{subch} values (5 and 2) to test the impact of this parameter into the system, even though $N_{subch} = 2$ is not a standardized value [56]. Using $N_{subch} = 5$ (AS1a), the packet occupies 3 subchannels ($L_{subch} = 3$), which limits the number of possible allocated UEs to ten in case they perform packet retransmission. This is because the UEs transmit two packets every 20 ms and only one packet per subframe can be transmitted without collision.

Five UEs are assumed for AS1a since, as seen in Figure **26**, the number of UEs must be lower than the limit. In case of $N_{subch} = 2$ (AS1b), the packet occupies 1 subchannel $L_{subch} = 1$). This enables twice as many UEs to be allocated compared to the previous case for the fact that two packets can be transmitted per subframe instead of one.

In this case, 15 UEs are assumed. Comparing AS1a and AS1b shows that by using $N_{subch} = 2$ the collision probability is clearly lower than by using $N_{subch} = 5$, even though the number of UEs is three times higher. This demonstrates the importance of the grid configuration as well as that $N_{subch} = 2$ can

be a useful value for this configuration even if not standardized.

Regarding AS1b and AS2, it is shown that the less-stringent latency requirement does not make the collision probability to be lower for AS2. In fact, the results show that collision probability for AS2 is higher than for AS1b. This is because each packet in AS2 occupies all the subchannels in the grid, whereas in AS1b only half of them. More generally, it can also be seen that the collision probability is higher when using retransmission (Rt = On); however, it is below 1 % in in the long run for all simulated scenarios, which guarantees a reliability of 99 % in good channel conditions as shown in D3.2 about the BLER results achieved using SL D2D modes 3 and 4.

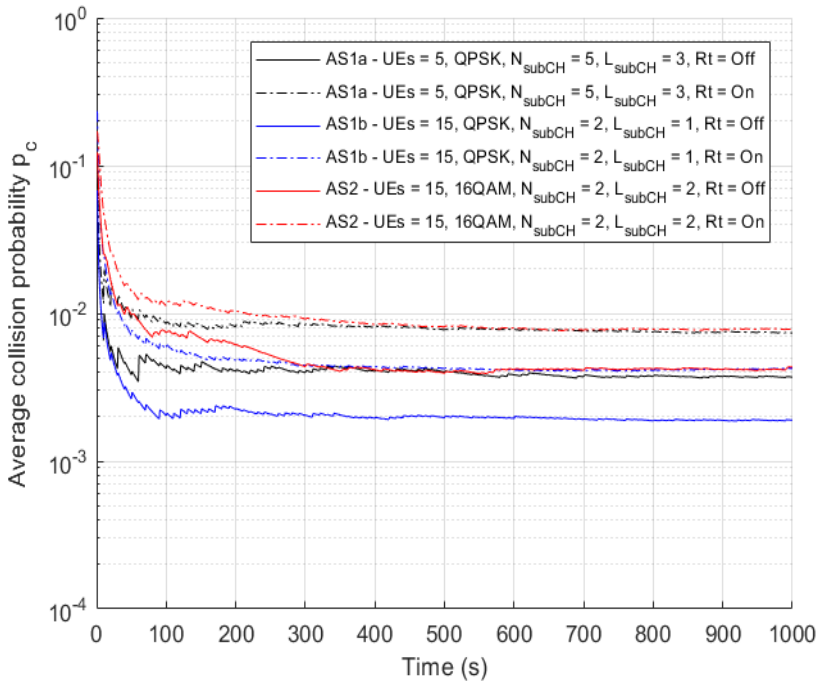

Figure 26 - Average collision probability for different Application Scenarios (ASs).

3.3.2 3GPP implementation details

Figure **27** depicts the structure of the SCI format 1 message, transmitted in modes 3 and 4 in PSCCH and used for the scheduling of PSSCH [57]. The SCI format 1 consists of the fields shown in Fig.10, where "RIV", which stands for Resource Indication Value, represents the "Frequency resource location of initial transmission and retransmission" field, "Time gap" represents the "Time gap between initial transmission and retransmission" field, and "Rt. idx" represents the "Retransmission Index" field. According to [57], all the SCI format 1 fields but "RIV" occupy a fixed number of bits, which sum 17 bits in total. The number of bits required for "RIV" depends on the number of subchannels in the resource grid and it might range from 0 to 8 bits. Reserved information bits are added (and set to zero) until the size of SCI format 1 is equal to 32 bits. Therefore, at least 7 bits are not used (set to zero) in each SCI format 1 transmission. In our proposal, we assume that the UEs transmit their current counter value in each transmission and that the maximum counter value is 63, i.e., it can be represented in 6 bits. Therefore, we propose for the counter to be transmitted in each SCI format 1 transmission in place of some of the reserved information bits. In this way, the UEs are able to know the counter value of their surrounding UEs by just decoding the SCI.

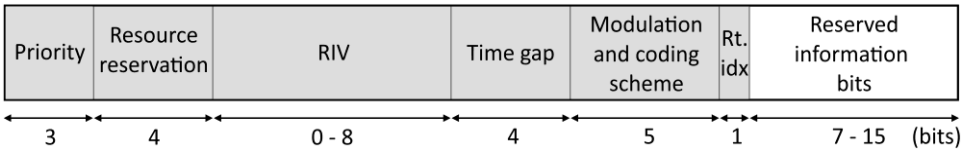

Figure 27 - SCI format 1 fields.

3.4 Ultra-reliable distributed control for cooperative vehicular CPS

Technology Description Table – TDT22
Title: Ultra-reliable distributed control for cooperative vehicular CPS
Property: Safety
Type: Protocol
Description: A communication protocol that can allocate the resources, i.e. channels, to each vCPS with high reliability and low latency.
Provider: DTU
Provided as SafeCOP Technology Brick: NO
Readiness: N/A
Integration Status: N/A
Additional Details: N/A

A vehicular CPS (vCPS) is considered in our study, where all vCPS communicate through wireless communications and machine-to-machine (M2M) type of communications that are driven using the required pattern. Such a pattern is implemented through cooperative awareness and distributed control. This is considered a complex modeling and thus, we propose a distributed control plane (DCP) using a layered approach, which aims to provide the following functionality to support the cooperation among the vCPS [58]:

- Providing point-to-point connectivity information in order to ensure end-to-end network integrity.
- Mapping of vCPS information into particular local control tasks supporting the cooperative global task.
- Actuating the cooperation by controlling the cooperative task locally and globally.

Such a layered architecture also guarantees the interoperability (i.e. different type of communication protocols and distributed control) required for successful implementations.

The proposed DCP integrates three different levels of functionalities, formulating a type of protocol suite. On top of the protocol stack, a cooperative awareness message (CAM) application protocol is situated that is responsible to transmit and receive the messages to each vCPS. CAM application protocol conveys useful information related to the cooperative task. The CAM application protocol is implemented on top of a communication protocol that is a machine-to-machine (M2M) protocol providing a single-hop communication (similar to V2V communications) [59].

At the bottom of the architecture, a distribute control protocol is situated, which is responsible for mapping the CAM information exchanged among the vCPSs to a particular control functionality. The distributed control functionality should be divided into local and global in order to accomplish the cooperative task. The specification of the cooperative task is required and the corresponding distributed control protocol that consists of the local actuators and the global ones as it is discussed later in this paper.

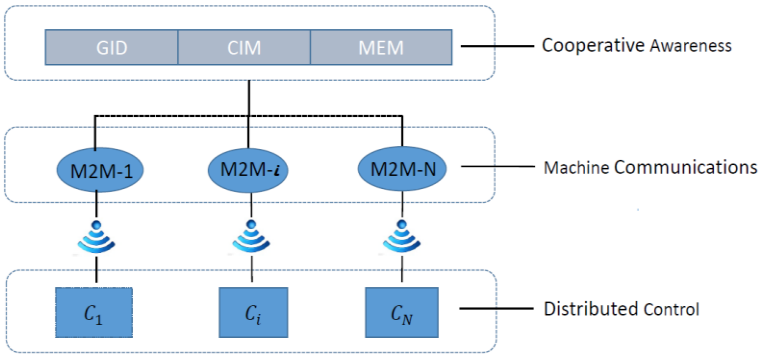

Figure 28- Distributed control plane functional architecture.

To design a distributed control system for vCPS, the following are required:

- to provide distributed control techniques for a linear control system
- to provide a communication control channel to send critical control messages.

To this end, we first describe below the distributed control and next, the communication control channel details.

We assume an adaptive cruise control (ACC) system in order to model a leader-follow formation control for our vCPS use case. Leader-follower formation control for mobile robots using model predictive control is recently proposed in [60]. It is known that an ACC system uses its two modes: a) speed control mode and b) space control mode, where the first regulates the vehicle speed at a driver-defined setting and the second to avoid a collision with the leader vehicle. Space control can be implemented based on constant spacing or on constant time gap. Moreover, the space control should be implemented with a particular car-following policy.

For testing controller behavior when the driver chooses to change the gap setting, only two vehicles were used, one of them acting as the leading vehicle and the other one running the ACC controller. The vehicle dynamics are considered according to the following open-looped cruise control transfer function:

$$H(s) = \frac{1}{s(0.5s+1)},$$

which approximates the dynamics of the throttle body and vehicle inertia. The vehicle dynamics block has connection with the MPC system, where the former gives the actual velocity y_a as an input to the latter. The MPC gives always its own acceleration value a_i to the vehicle dynamics. Finally, the vehicle dynamics of the leader sends out through wireless communication the pair of velocity and position $\{x_0, y_0\}$. Fig.3 depicts also the specified safe distance d_s and the actual distance d_a that the overall ACC system must retain by interchanging between speed and headway modes of control.

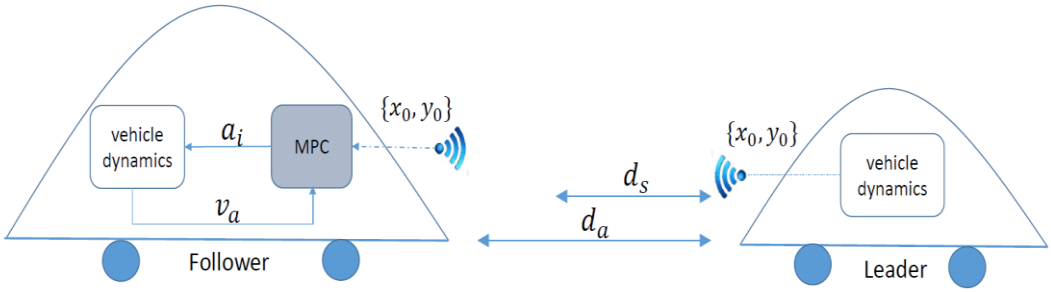

Figure 29 - ACC enabled by MPC.

The cooperative task is provided through wireless communications that enables eventually the distributed control among different vCPS. The most known standard for vehicular type of communications is considered the dedicated short-range communications (DSRC) standard [61].

DSRC uses the layered protocol architecture based on the IEEE 802.11p and IEEE 1609 family standards. DSRC protocol above layer 2 divides into two stacks. General application and service can use both traditional protocols (UDP, TCP and IPv6) and new protocol (WAVE Short Message Protocol). WSMP is simple, efficient and specifically customized in the vehicular environment instead of layer 3 and 4 Internet protocols. The logical link control (LLC) sub-layer protocol stack uses IEEE 802.2 standard. LLC protocol frequently uses Subnetwork Access Protocol (SNAP) to identify the protocol associated with the payload of upper layer. DSRC MAC and PHY are defined in IEEE 802.11p. Especially, Multi-Channel Operation protocol is newly added in MAC to support multiple channels. This multi-channel operation concept is defined in IEEE 1609.4 standard. DSRC device in VANETs should encapsulate the safety message from application layer to PHY layer. Each layer of WAVE protocol architecture attaches the protocol header in order to compliance with the current layer protocol. Application selects the message set, data frames and data elements for its purpose from SAE J2735 DSRC message dictionary. Those messages are composed and encoded as WAVE Short Message (WSM) format and become the WSMP payload.

We tried to summarize the analysis above into the frame slot as encapsulated from the MAC to PHY layer in order to transmit from one vCPS to another, i.e. through M2M type of communications. According to DSRC, the full MAC frame format is passed to the PHY layer through the Physical Layer Service Data Unit (PSDU). Depending on the specified target rate, e.g. 12Mbps, the PHY layer is being processed with a total of 55.3μs time period, where 32μs is for the PLCP preamble, 8 μs for the signal field and 15.3 μs that provides the service field, the PSDU, the tail and the PAD [62]. Fig. F3 depicts the generic format of the time slot which is divided into the PDU part with control information denoted as (*n-l)* , e.g. 55.3 μs and the CAM part that conveys the actual data information denoted as l while the overall time slot is denoted as *n*. This notation is used also below to provide the ultra-reliable and low latency solution for cooperative vCPS.

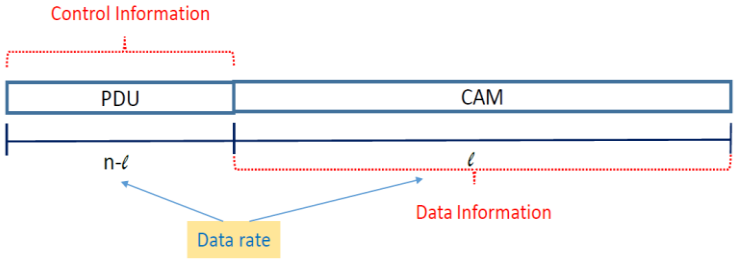

Figure 30 - Time slot for M2M communications.

Wireless communications in networked robotics is still an open challenge. Although robots communications can be assumed as an ad hoc network application, conventional mechanisms can not be used [63]. Cooperation with reliability and low latency is more important than having higher data rates. An interesting application could be considered using wireless communications with short packets like an internet-of-thing (IoT) application. Such a design should provide a type of communication protocol with short packets. Key design factors of such a protocol are the number of information bits *k* and the number of the overall packet size *n*, where (*n*-*k*) is considered the number of the control bits. The rate approximation for a particular packet size *n* and information bits *k* for a specific packet error probability ϵ is given as follows [64]:

Using the analysis above, we are going to derive reliability $1-\epsilon(l,n)$ and spectral efficiency S_e results for different (1/n) values. The (1/n) values could vary from 1/6 for low data rates to 2/3 for higher data rates according to the performance analysis of the WAVE control channels [62]. Thus, we obtained results and Fig.4 below depicts the reliability in % versus SNR values in dB for different number of n values and ratios 1/n. It is observed that a higher number of packet length n gives a higher reliability. This is due to the short packet design requirement as pointed out in [64]. Lower number of information bits that means higher number of control bits will result in an additional higher reliability.

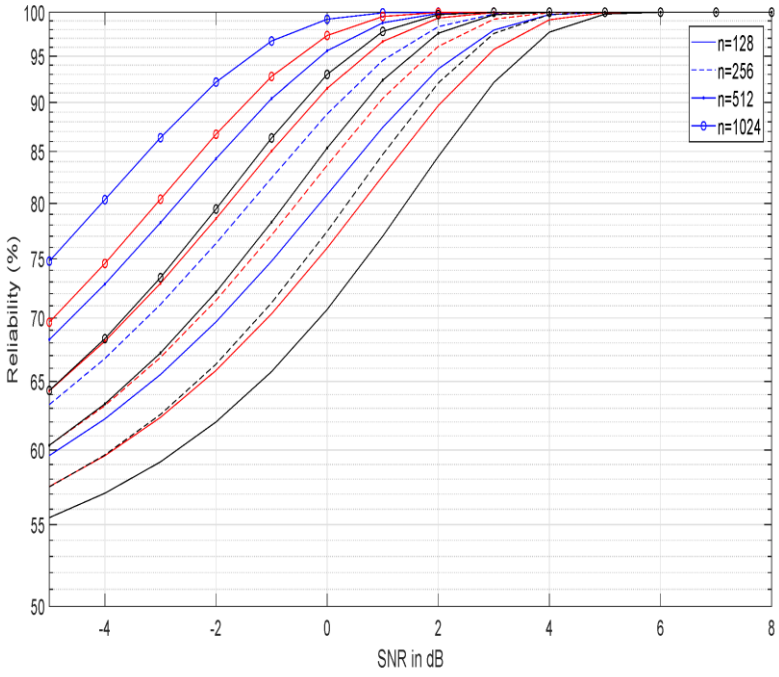

Figure 31 - : Reliability vs. SNR in dB (blue line l = n/3, red line l=n/2 and black line l=2n/3)

Moreover, Figure 32 below depicts the spectral efficiency, where spectral efficiency is calculated as follows $S_e = \frac{1}{n}R$ achieved for different n packet size values and ratio 1/n. It is observed that the higher the information bits k, the higher the achievable spectral efficiency. A high packet size results also in higher spectral efficiency as expected.

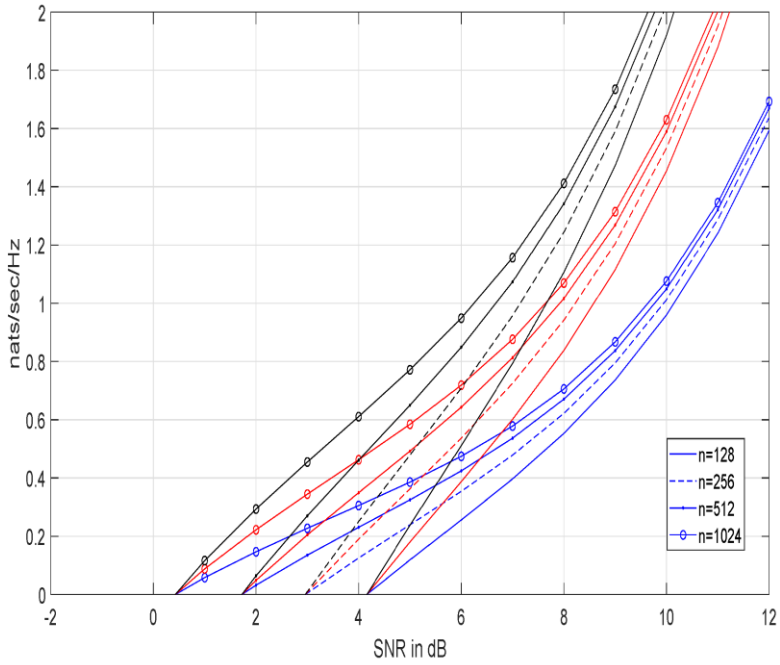

Figure 32 -Spectral efficiency vs. SNR in dB (blue line l = n/3, red line l=n/2 and black line l=2n/3).

Therefore, we aim to design a distributed solution, i.e. a communication protocol that can allocate the resources, i.e. channels, to each vCPS with high reliability and low latency. Ideas from both [64] and [65] will be taken into account in order to conclude to our solution described below.

We would like to design a distribute solution that can provide decisions about the overall frame structure in an adaptive fashion. In particular, the frame can have different sizes per vCPS use case instant retaining the overall delay at a specified level providing the reliability in parallel too. To this end, we formulate the problem below, where the spectral efficiency maximization is considered as the objective function subject to the overall frame to not exist N_0, the overall delay D_{tot} to not exist D_0 , where delay is considered the latency requirement that is equal to overall adaptive frame T format, and the per_i per vCPS to be always below per_0 to retain the required reliability. The final problem formulation is as follows:

$$\max_{l_i, n_i} Se_i(l_i, n_i)$$
$$s.t. \Sigma_i n_i \leq N_0$$
$$D_{tot} < D_0$$
$$per^i \leq per_0.$$

In order to solve such a problem, an heuristic algorithm that combines also the Hungarian method is devised. Such heuristic algorithms are considered practical to many wireless communications use cases such as device-to-device (D2D) [66].

The proposed algorithm is mainly devised to reduce high computational complexity guaranteeing however the reliability and latency constraints. The assignment problem is a linear program, where in our case the number of sources (channels) equals the number of designations (vCPS), all number of N.

The proposed algorithm is mainly devised to reduce high computational complexity guaranteeing however the reliability and latency constraints. The assignment problem is a linear program, where in our case the number of sources (channels) equals the number of designations (vCPS), all number of N. The algorithm can be found below (Algorithm 2), where first the frame T is equally divided into time slots $\tau_i = \frac{T}{N}, \forall\, i \epsilon N$ vCPS. Next, checking out the total latency requirement $D_{tot} < D_0$ is guaranteed. In the sequel, the SNR values γ_i γ_i are calculated per vCPS and the HM is activated to assign new τ_i for $per_i < per_0$, $\forall\, i \epsilon N$ vCPS. At the second level, the HM is activated to keep the 1/n values over a set of M. The set M is the number of 1/n values that are considered to each application scenario, which maximizes the spectral efficiency by selecting the $l - n$ value over the n packet size. Finally, the complexity the proposed algorithm is equal to $O[\max(N, M)^3]$ where when M=N the complexity is getting lower and equal to $O[\max(N)^3]$ [66]. The proposed algorithm is named heuristic adaptive resource allocation (HARA) algorithm that can be implemented both in centralized and decentralized fashions.

Algorithm 1 - Heuristic Adaptive Resoruce Allocation (HARA)

$\forall i \in N$ vCPS, initialize:
Frame T with time slots $\tau_i = T/N, \forall i \in N$
Matrix $H[N, M]$, where n is the number of vCPS and M the number of l/n values.
procedure:
while $D_{tot} < D_0$ **do**
  Calculate SNR $\gamma_i, \forall i \in N$
  Activate HM: assign τ_i for $per_i \leq per_0$
  if $\Sigma(n_i) > N_0$ **then**
    Adapt the frame length
  else if $\Sigma(n_i) \leq N_0$ **then**
    Activate HM: assign $maxSe_i, \forall i \in N, \forall j \in M$
  end if
end while
end procedure

Regarding the leader-follower use case, we assume that the speed of a vCPS could be retained among 15km/h to 30km/h, which is considers a regular speed for mobile robot applications. Thus, our simulation results are carried out with a value of 5m/s to 10m/s. With such speed specification, the safe distance between two vCPS can be kept below 10m, where safe distance is the difference of the actual distance from the specified one. To this end, the following mode switching are provided to the system:

- Speed control mode in case of maintaining the target speed.
- Gap control mode in case of maintaining the target space gap.

The mode switching is enabled by MPC and controlled by the input information $\{x_0, y_0\}$ sent out by the leader to follower through wireless communication. Such control information can be transmitted through wireless communications, e.g. a V2V service. We assume that the update time *k* of the MPC presented in Sec.III.A is the slot time τ by which a vCPS receives message payload.

Under the assumptions above, the requirement is to keep the reliability and the latency high and low respectively for a particular payload message size. To this end, we provide simulation results obtained using our simulation setup. Fig.7 depicts the safe gap in meters between the leader and the follower for different reliability constraints. The latency is considered T=50ms, where is adequate time to process 5 vCPS of $\tau_i = 5ms$. This value should be extended to higher latency requirements in order to allocate more vCPS. In case now of ultra reliability, i.e. 99.99%, the gap is below 10m even in case of high acceleration. However, the lower reliability makes the situation worst exceeding the threshold of 10m even in case of a moderate reliability equal to 90%. We have also plotted results for different acceleration

pattern denoted as low and high, where the higher acceleration is possible to be managed over the time. We also highlight the points that Join/Leave maneuvers can be carried out over time, where in low acceleration the things are much doable to retain ultra-reliability even at the beginning.

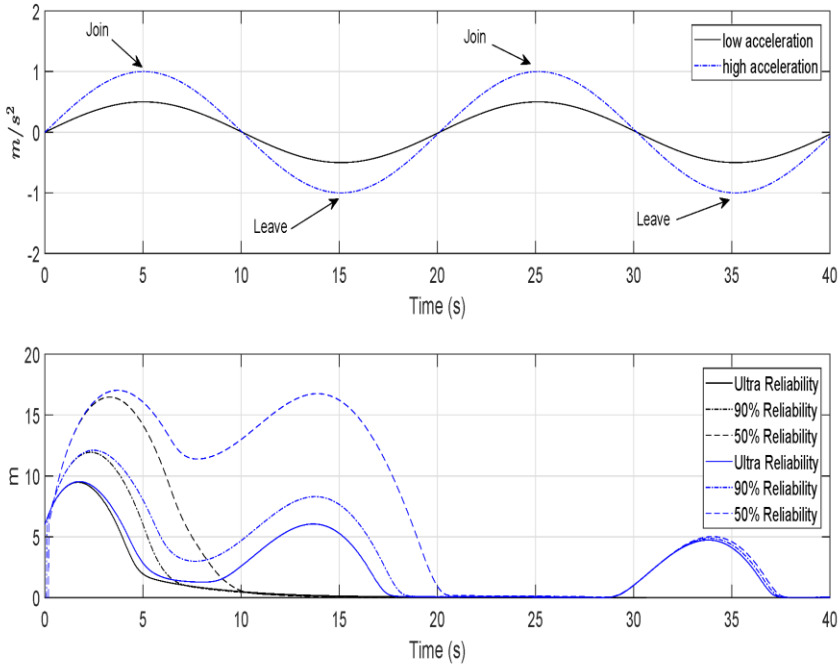

Figure 33 - Gap in meters over time for different reliability constraints in case of space gap (solid lines) and leader speed (dashed lines) outdated information.

3.5 Machine Learning techniques applied to platooning

Technology Description Table – TDT23
Title: Machine Learning techniques applied to platooning
Property: Safety
Type: Method and a tool
Description: Machine learning techniques to identify the best allocation of control parameters in platooning settings. The stability of the platoon is taken as a reference.
Provider: CNR and IMPARA
Provided as SafeCOP Technology Brick: YES (TB035)
Readiness: Complete
Integration Status: In progress integration with UC3
Additional Details: N/A

This contribution outlines how machine learning may be used to identify the best allocation of control parameters in platooning settings. The stability of the platoon is taken as a reference.

String stability (SS) in platooning means that speed and acceleration fluctuations are attenuated downstream the string of vehicles of the platoon. [67] and [69] are two good examples of how setting the control parameters to achieve stability may be a hard task.

Being $\boldsymbol{v}$ and $\boldsymbol{d}$ the vectors of speed and reciprocal distance of the vehicles, we consider here the following system dynamics:

$$\dot{v}_i = {}^1/_{m_i}\left(F_i - \left(a_i + b_i \cdot v_i^2\right)\right);\ \dot{d}_i = v_{i-1} - v_i; i = 0, \dots N-1$$

and control law (Figure 34) of [70]:

$$F_i = g(d_i), i \geq 1;\ g(d) = \max\{50(d-27) + 4(d-27)^3, -10000\}$$

The following setting is applied: $N = 2$ (3 cars in the platoon), $a_i = 0$ (tire/road rolling resistance), $b_i = 0.43$ (aerodynamic drag), $m_i = 1050$ Kg $i = 0, \dots N-1$ (weight of the cars). The breaking force applied by the leader is: $F_0 = -500 \cdot sin\ (0.2 \cdot t)$. Variable communication delays (*del*) between the vehicles are considered as well. For the sake of simplicity, only fixed delays are applied.

The performance metric to measure SS is $J = \Sigma_h |v_2(h) - v_0(h)|$, being h the sampling index over time, $h = 0, \dots H$, being H the time index at the end of the observation horizon (300 s).

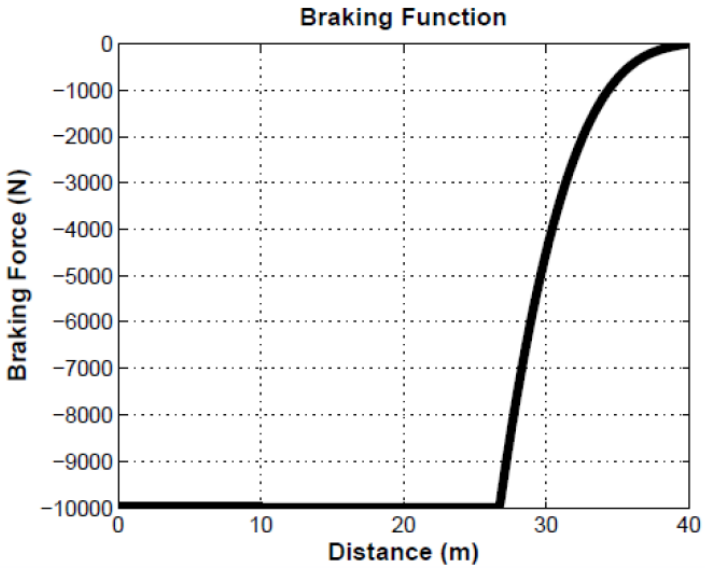

Figure 34 - Breaking function with k=1.

A family of control laws of the type above are considered by introducing the *k* parameter that multiplies the cubic part of the control law, namely: $g(d) = max\{k \cdot [50\,(d-27) + 4\,(d-27)^3], -10000\}$. The *k* parameter is studied here to achieve SS. Figures 14-17 show the effect of setting *k*=1 and *k*=30 with: *del*= 5 ms, $d_i(0) = 40$ m and $v_i(0) = 110$ Km/h, $i = 0, \ldots, N-1$ (the last two are the initial distance and speed in the platoon, respectively). The figures suggest that $J < 1$ values are associated to SS. This was validated by further experiments, not reported here.

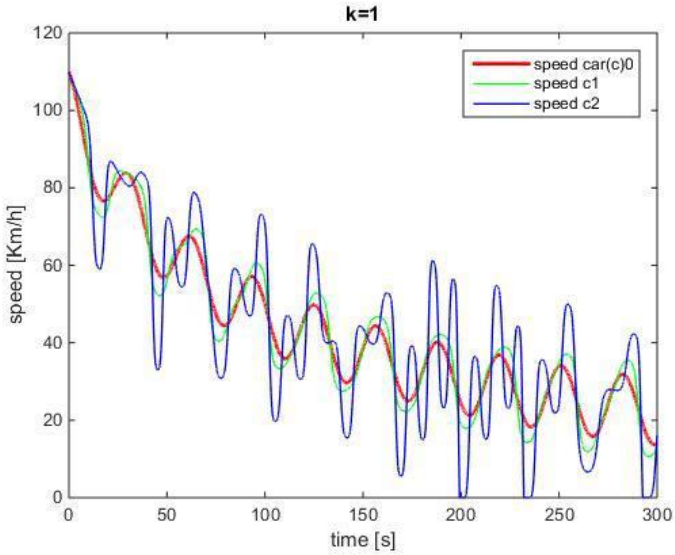

Figure 35 - Sinusoidal decrease of leader (car=0). (0)=40 m; (0)=110 Km/h. Vehicles speed. k=1. Large oscillations, J=2.744.

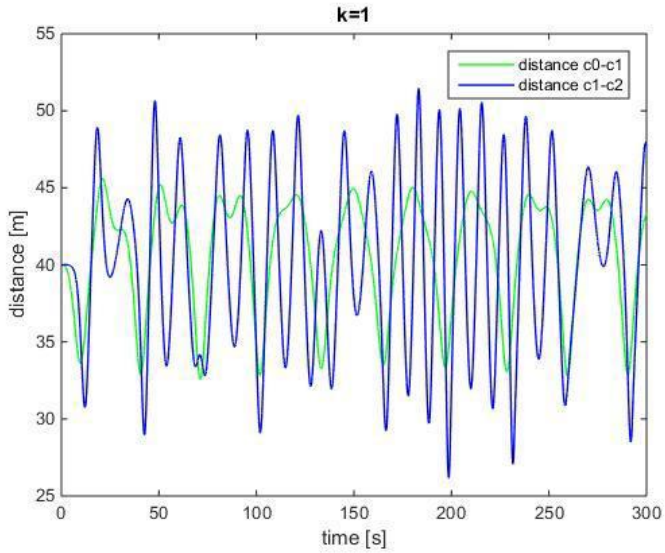

Figure 36 - Sinusoidal decrease of leader (car=0). (0)=40 m; (0)=110 Km/h. Vehicles distance. k=1. Large oscillations, J=2.744.

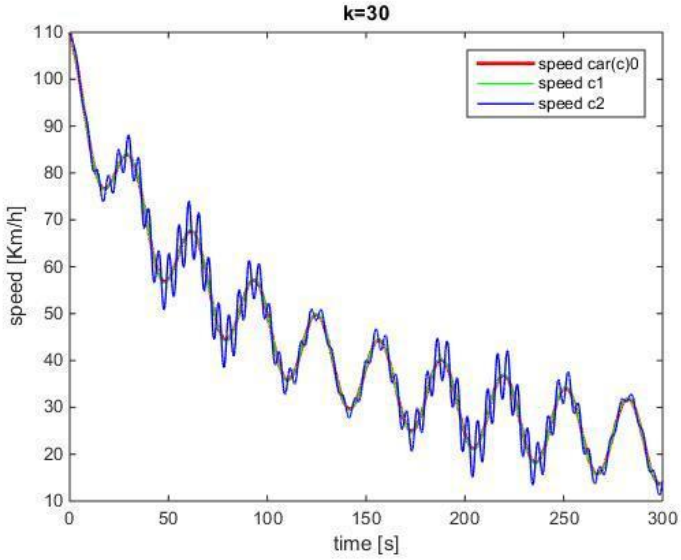

Figure 37 - Sinusoidal decrease of leader (car=0). (0)=40 m; (0)=110 Km/h. Vehicles speed. k=30. Small oscillations, J=0.653.

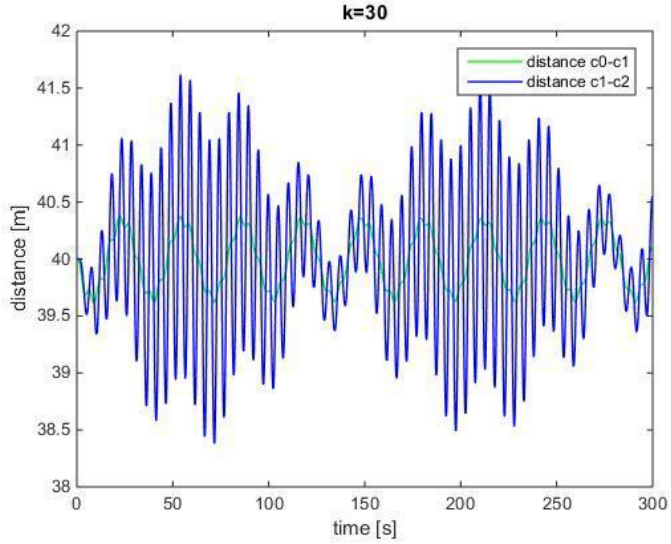

Figure 38 - Sinusoidal decrease of leader (car=0). (0)=40 m; (0)=110 Km/h. Vehicles distance. k=30. Small oscillations, J=0.653.

3.5.1 Minimum breaking force

We want to discover the minimum k (i.e., minimum breaking force) to guarantee SS under the following conditions: $del \in [1,15]$ ms, $d(0) \in [25,40]$ m, $v(0) \in [90,110]$ Km/h.

The non-linearity of both dynamics and control law makes the analytical derivation of stability conditions intractable. Machine learning may be applied to derive a model for SS only on the basis of samples, acquired during the system evolution. A dataset is built by simulating the system under the chosen conditions and registering the $J2$ variable in correspondence of registered stable and unstable behaviours of the platoon: $J2 = \{\text{“}stable\text{”}\ if\ J \leq 1, else\ \text{“}unstable\text{”}\}$.

Two thousand (2000) samples of system conditions are derived with a simulation time of less than 10’ over an IntelCore i7 @2.4 Ghz. The Logic Learning Machine (LLM) [68] is then applied to map the variables into the $J2$ output. The mapping is expressed by the boolean rules reported below (only rules with high covering for the “stable” class are reported).

Table 3 - Rules for "stable" class with: $del \in [1,15]$ ms, $d(0) \in [25,40]$ m, $v(0) \in [90,110]$ Km/h.

RULE 1: IF *del* <= 11 AND *d(0)* > 28 AND *k* > 7.845 THEN SS = stable, COVERING: 0.910547, ERROR: 0.039169

RULE 2: IF *del* <= 10 AND *d(0)* > 27 AND *k* > 8.145 THEN SS = stable, COVERING: 0.885180, ERROR: 0.033573

RULE 3: IF *del* <= 10 AND *d(0)* > 28 AND *k* > 7.575 THEN SS = stable, COVERING: 0.879840, ERROR: 0.020783

RULE 4: IF *del* <= 8 AND *d(0)* > 27 AND *k* > 7.255 THEN SS = stable, COVERING: 0.734312, ERROR: 0.042366

RULE 5: IF *del* <= 13 AND *d(0)* > 30 AND 8.415 < *k* <= 17.69 THEN SS = stable, COVERING: 0.635514, ERROR: 0.042366

The covering metric measures how many points of a class are synthetized by a rule (e.g., for rule 1: 91% of "stable" points satisfy rule 1); the error measures how many points are misclassified by that rule on the other classes (e.g., for rule 1: 3.9% of "unstable" points are misclassified, i.e., assigned to "stable" class by rule 1).

By introducing safe margins over the thresholds presented in the first 4 rules, it is intuitive to state the following conditions for stability with $k = 8$: $del \in [1,9]$ and $d(0) \in [31,40]$ (independentlyto $v(0) \in [90,110]$). This was validated by further 1000 test samples as well.

Rule 5 also suggests another setting of k with $del \in [10,13]$. Together with rules 1-4, it appears that the $[13,15]$ range for delay is related to unstable behaviours.

This is confirmed by the delay histogram reported below. Values of the histogram below 10 ms are mapped onto the two classes with an equivalent proportion of samples; this suggests the use of the other parameters to discriminate between the two classes (as done by the presented rules).

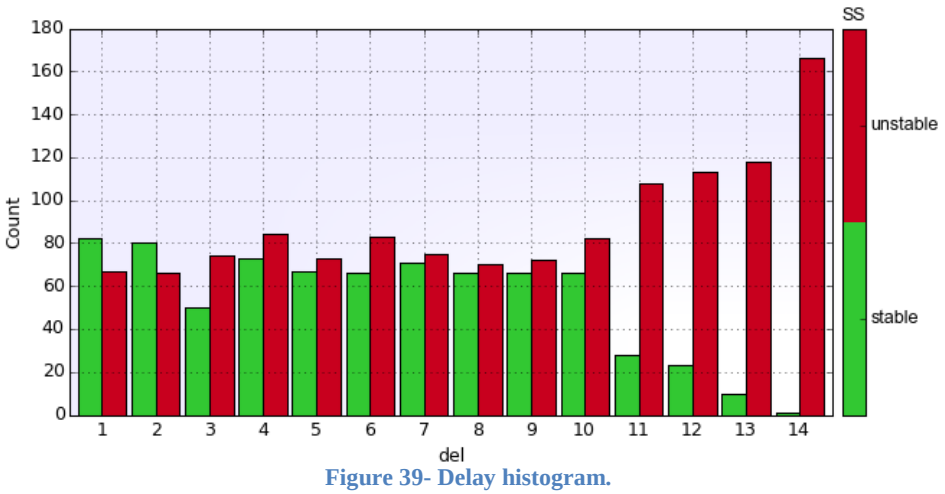

Figure 39- Delay histogram.

3.5.2 Optimal allocation of thresholds

In general, one may want to find the relation among: minimum distance, minimum breaking force and maximum delay, still achieving stability. In order to understand how tricky the problem is, we firstly start from data visualization.

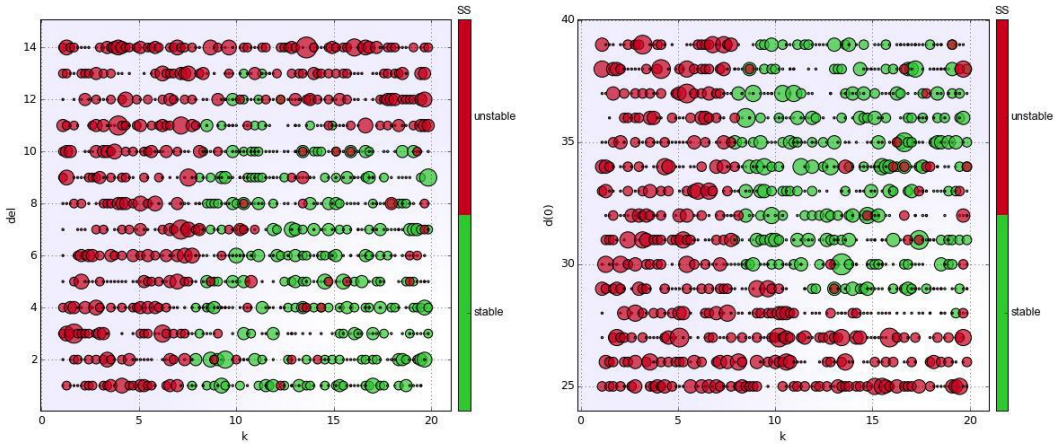

Figure 40 - Scatter plots of k-del and k-(0).

Figure 40 reports the scatter plots of the three variables. The empirical rule found above by further filtering rules 1-4 of Table 3 (i.e., $k = 8, del \in [1,9]$ and $(0) \in [31,40]$), actually identifies stable areas over ranges of delay and distance under minimum breaking force.

In order to find minimum delay as well, we start from the k histogram (Fig. 8) and set $k \geq 9$ in the following analysis. The information with $k<9$ is not considered anymore, being $k=9$ on the border of stable/unstable classes. We then apply the LLM over the subset of data corresponding to $d(0) \in [30,40]$. The rules obtained are reported in table 2. Rules 1 and rules for the unstable class help easily identify $del=10$ as the minimum delay with $k=9$ as stability conditions in $d\ 0 \in 30,40$. This was validated over another test set of 1000 points.

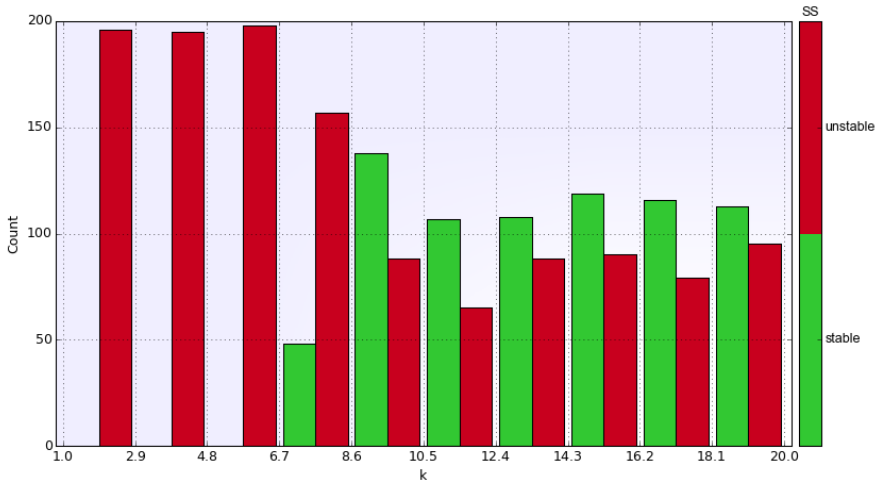

Figure 41 - k histogram.

The situation with $d(0) \in [25,30]$ is much more complicated as outlined by the $d(0)$ histogram (Figure 42): only instability seems arising over that subset of data. Is there any setting of the other parameters to still achieve stability?

The analysis on the new subset: $k \geq 9$, $0\ \in [25,30]$, over a new training set with 1000 points, leads to an unbalanced proportion of registered unstable outputs (Figure 42), while a balanced proportion between the two classes appears in the previous analysis. Moreover, the $d(0) \leq 27$ rule has been obtained by the LLM in the 69% of the unstable cases. These two facts give a clear indication that the $d(0) \in [25,30]$ interval should be avoided or restricted even more: $(0) \in [28,30]$. By further restricting the analysis on that interval, several rules (similar to the ones already reported) are found with a coverage lower than 40%. In virtue of such reduced coverage, the most restricting thresholds are considered and

successfully validated over a new test set of 1000 points: del=2, k=11.

Table 4 - Rules on SS with: k ≥ 9, d 0 ∈ 30, 40 (del ∈ [1, 15], v(0) ∈ [90,110]).

RULE 1: IF *del* <= 10 THEN SS = **stable**, COVERING: 0.903172, ERROR: 0.000000

RULE 2: IF *del* <= 11 AND *k* <= 16.58 THEN SS = stable, COVERING: 0.649416, ERROR: 0.024242

RULE 3: IF *del* <= 12 AND *k* <= 14.775 THEN SS = stable, COVERING: 0.522538, ERROR: 0.048485

RULE 4: IF del <= 13 AND 9.005 < *k* <= 13.26 THEN SS = stable, COVERING: 0.410684, ERROR: 0.048485

RULE 5: IF *k* <= 10.195 THEN SS = stable, COVERING: 0.138564, ERROR: 0.042424

RULE 6: IF *del* > 11 AND *k* > 10.365 THEN SS = **unstable**, COVERING: 0.800000, ERROR: 0.031720

RULE 7: IF *del* > 10 AND *k* > 14.64 THEN SS = unstable, COVERING: 0.666667, ERROR: 0.006678

RULE 8: IF *del* > 12 THEN SS = unstable, COVERING: 0.624242, ERROR: 0.016694

RULE 9: IF *del* > 11 AND *d(0)* > 32 THEN SS = unstable, COVERING: 0.587879, ERROR: 0.030050

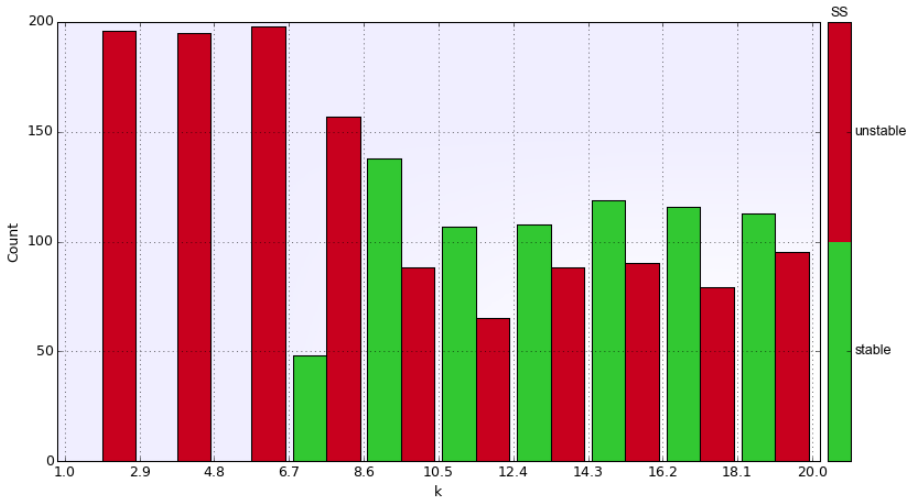

Figure 42 - d(0) histogram.

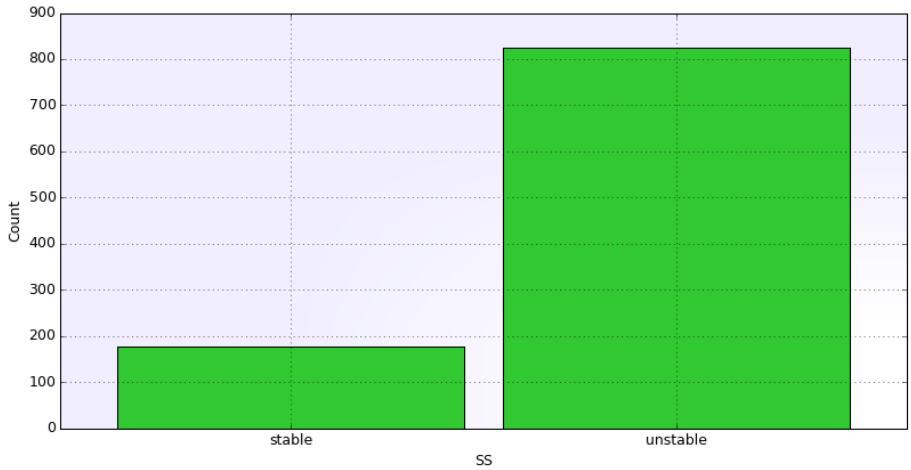

Figure 43 - SS histogram with: $k \geq 9$, d 0 ∈ 25,30 ($del \in [1,15]$, $v(0) \in [90,110]$).

4 Final Remarks

4.1 Overview of contributions

Overall, technical contributions presented in this report are quite complete, several with supporting publications, and proven results. They begin with a platooning control model and a Webots simulation model, which is being currently used in UC3 to test, evaluate, and validate the platooning robotic testbed. Then, we delve into a cooperative resource allocation problem for enabling V2X in 5G networks, a hot and important topic. Then, we shift into IEEE 802.11p-based V2V networks, and propose a cooperative reliable allocation mechanism that can allocate the channels, to each vCPS with high reliability and low latency. The proposed algorithm is mainly devised to reduce the high computational complexity guaranteeing however the reliability and latency constraints.

Finally, we look into the application of machine learning techniques to platooning, in particular by finding the relation among minimum distance, minimum breaking force and maximum delay, while guaranteeing platooning stability and ensuring safety. The table below, provides an overview of the contributions.

Table 5 - Summary of technologies

TECHNOLOGY	PROPERTY ADDRESSED	TYPE	PARTNER	SAFECOP TECHNOLOGY BRICK	SECTION
DUAL MODE VEHICULAR PLATOONING CONTROL MODEL	Safety	Control Model	ISEP	TB019	3.1
VEHICLE PLATOONING SIMULATOR IN WEBOTS	Safety	Method	ISEP	TB020	3.2
COOPERATIVE RESOURCE ALLOCATION AND SCHEDULING FOR 5G EV2X SERVICES	Safety	Protocol	DTU	N/A	3.3
ULTRA-RELIABLE DISTRIBUTED CONTROL FOR COOPERATIVE VEHICULAR CPS	Safety	Protocol	DTU	N/A	3.4
MACHINE LEARNING TECHNIQUES APPLIED TO PLATOONING	Safety	Method and tool	CNR and IMPARA	TB035	3.5

4.2 Conclusions

This report provides a total of 5 important pieces of technology, most of them already integrated in the SafeCOP platooning Use Case (UC3) or under integration. Importantly, most of the contributions focus heavily on improving safety of the cooperative processes either by encompassing communication shortage in the control models or by increasing the reliability of the communications protocols. Notably, in addition to methods and protocols, a set of tools was also developed e.g., the Webots simulator model and the machine learning software framework, which greatly eases the process of UC development. Many of the proposals are still under integration in the respective UCs which means, despite the preliminary results obtained either analytically, in simulations or in testbeds, the validation in the UC scenario is to be carried out in the long run, and should be ready be the end of the project.

5 References

[1] C. Bonnet and H. Fritz, “Fuel consumption reduction in a platoon: Experimental results with two electronically coupled trucks at close spacing,” SAE International, 2000.

[2] J. Axelsson, "Safety in Vehicle Platooning: A Systematic Literature Review," in IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 5, pp. 1033-1045, May 2017.

[3] R. Rajamani, Vehicle Dynamics and Control, 2nd ed. Springer, 2012.

[4] Oussama Karoui, Mohamed Khalgui, Anis Koubâa, Emna Guerfala, Zhiwu Li, Eduardo Tovar, “Dual Mode for Vehicular Platoon Safety: Simulation and Formal Verification” in Information Sciences, Elsevier. Volume 402, pp 216-232, Sep 2017.

[5] J. Fink, A. Ribeiro and V. Kumar, Robust Control for Mobility and Wireless Communication in Cyber–Physical Systems With Application to Robot Teams, Proceedings of the IEEE, vol. 100, no.1, pp. 154-178, Jan. 2012.

[6] Z. Liu, W. Chen, J. Lu, H. Wang and J. Wang, “Formation Control of Mobile Robots Using Distributed Controller With Sampled-Data and Communication Delays”, IEEE Trans. On Control Systems Techn., vol. 24, no.6, pp. 2125-2132, Nov. 2016.

[7] T. Dierks, B. Brenner and J. Jagannathan, Neural Network-Based Optimal Control of Mobile Robot Formations With Reduced Information Exchange, IEEE Trans. On Control Systems Techn., vol. 21, no.4, pp. 1407-1415, Jul. 2013.

[8] T. Dierks and J. Jagannathan, Neural Network Control of Mobile Robot Formations Using RISE Feedback, IEEE Trans. On Systems, Man and Cybernetics, vol. 39, no.2, pp. 332-347, Apr. 2009.

[9] M. Guinaldo, G. Farias, E. Fabregas, J. Sánchez, S. Dormido-Canto and S. Dormido, An Interactive Simulator for Networked Mobile Robots, IEEE Network, vol. 26, no. 3, May/June 2012.

[10] B. Ning, J. Jin, B. Krishnamachari, J. Zheng and Z. Man, Two-Stage Deployment Strategy for Wireless Robotic Networks via a Class of Interaction Models, IEEE Trans. On Systems, Man and Cybernetics, vol. 47, no. 7, pp. 1510-1521, Jul. 2017.

[11] K. Derr and M. Manic, Adaptive Control Parameters for Dispersal of Multi-Agent Mobile Ad Hoc Network (MANET) Swarms, IEEE Industrial Informatics, vol. 9, no. 4, pp. 1900-1911, Nov. 2013.

[12] H. Hur and H-S. Ahn, Discrete-Time H∞ Filtering for Mobile Robot Localization Using Wireless Sensor Network, IEEE Systems Journal, vol. 13, no. 1, pp. 245-252, Jan. 2013.

[13] J. Suh, S. You, S. Choi and S. Oh, Vision-Based Coordinated Localization for Mobile Sensor Networks, vol. 13, no.2, pp. 611-620, Apr. 2016.

[14] J-H. Son and H-S. Ahn, Formation Coordination for the Propagation of a Group of Mobile Agents via Self-Mobile Localization, IEEE Systems Journal, vol. 9, no. 4, pp. 1285-1298, Dec. 2015.

[15] M.O. Ben Salem, O. Mosbahi, M. Khalgui, Z. Jlalia, G. Frey, M. Smida, BROMETH: Methodology to design safe reconfigurable medical robotic systems, Int. J. Med. Robot. DOI: 10.1002/rcs.1786 .

[16] M. Gasmi, O. Mosbahi, M. Khalgui, L. Gomes, and Z. Li, R-Node: New pipelined approach for an effective reconfigurable wireless sensor node, IEEE Trans. Syst., Man, Cybern., Syst., to be published. Doi: 10.1109/TSMC.2016.2625817.

[17] H. Grichi , O. Mosbahi , M. Khalgui , Z. Li , RWiN: New methodology for the development of reconfigurable WSN, IEEE Trans. Autom. Sci. Eng. 14 (1) (2017) 109–125 .

[18] H. Grichi, O. Mosbahi, M. Khalgui, and Z. Li, New power-oriented methodology for dynamic resizing and mobility of reconfigurable wireless sensor networks, IEEE Trans. Syst. Man. Cybern. Syst., to be published.

[19] S. Aradi , G. Rödönyi , P. Gáspár , Z. Hankovszki , R. Kovács , Experimental verification of vehicle platoon control algorithms, Period. Polytech. Transp. Eng. 41 (1) (2013) 39–43 .

[20] C. Bergenhem , R. Johansson , E. Coelingh , Measurements on v2v communication quality in a vehicle platooning application, in: M. Jonsson, A. Vinel, B. Bellalta, E. Belyaev (Eds.), Multiple Access Communications, Vol. 8715 of Lecture Notes in Computer Science, Springer International Publishing, 2014, pp. 35–48 .

[21] A. Casimiro , J. Kaiser , E. Schiller , P. Costa , J. Parizi , R. Johansson , R. Librino , The karyon

project: predictable and safe coordination in cooperative vehicular systems, in: Proceedings of the 43rd Annual IEEE/IFIP Conference on Dependable Systems and Networks Workshop (DSN-W), 2013, pp. 1–12 .
[22] Y.F. Chen , Z.W. Li , K. Barkaoui , N.Q. Wu , M.C. Zhou , Compact supervisory control of discrete event systems by petri nets with data inhibitor arcs, IEEE Trans. Syst. Man Cybern. (47) (2017) 364–379 .
[23] Y.F. Chen , Z.W. Li , A. Al-Ahmari , N.Q. Wu , T. Qu , Deadlock recovery for flexible manufacturing systems modeled with petri nets, Inf. Sci. (381) (2017) 290–303 .
[24] V. Chevrier, F. Gechter, F. Charpillet, A reactive agent-based solving model: application to localization and tracking, ACM Trans. Auton. Adapt. Syst. URL https://hal.inria.fr/inria-00104498 .
[25] S. Colin , A. Lanoix , O. Kouchnarenko , J. Souquières , Using CSP || b components: application to a platoon of vehicles, in: International Workshop on Formal Methods for Industrial Critical Systems, Springer, Springer-Verlag, Italy, 2008, pp. 103–118 .
[26] J. Contet, F. Gechter, P. Gruer, A. Koukam, Reactive multi-agent approach to local platoon control; stability analysis and experimentation, Int. J. Intell. Syst. Technol. Appl. 10 (3) (2011) 231–249, doi: 10.1504/IJISTA .
[27] A. Costas , M. Cerdeira-Corujo , A. Barreiro , E. Delgado , A. Baños , Car platooning reconfiguration applying reset control techniques, in: IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, Berlin, 2016, pp. 1–8 .
[28] M. El-Zaher, J.M. Contet, F. Gechter, P. Gruer, Reconfigurable and adaptable urban transportation systems: the platoon solution, in: International Conference on Smart and Sustainable City (ICSSC), IET, Shanghai, 2011, pp. 1–5, doi: 10.1049/cp.2011.0286 .
[29] T. Fang, D. Cho, J. Choi, Optimal scheduling of a communication channel for the centralized control of a platoon of vehicles, Int. J. Control Autom. Syst. 11 (4) (2013) 752–760, doi: 10.1007/s12555-012- 0266- z .
[30] P. Fernandes, U. Nunes, Multiplatooning leaders positioning and cooperative behavior algorithms of communicant automated vehicles for high traffic capacity 16(3) (2015) 1172–1187. 10.1109/TITS.2014.2352858
[31] C. Garcia-Costa, E. Egea-Lopez, J. Tomas-Gabarron, J. Garcia-Haro, Z. Haas, A stochastic model for chain collisions of vehicles equipped with vehicular communications 13(2) (2012) 503–518. 10.1109/TITS.2011. 2171336.
[32] E. Guerfala , M. Khalgui , A. Koubaa , O. Karoui , Modeling and formal verification of reconfigurable vehicular platoons, in: Proceedings of the 29th Conference on European Simulation and Modeling (ESM'15), 2015 .
[33] G. Guo, W. Yue, Hierarchical platoon control with heterogeneous information feedback, Control Theory Appl. IET 5 (15) (2011) 1766–1781, doi: 10.1049/iet-cta.2010.0765 .
[34] H.C.-H. Hsu , A. Liu , Kinematic design for platoon-lane-change maneuvers, IEEE Trans. Intell. Transp. Syst. 9 (1) (2008) 185–190.
[35] M. Kamali, L.A. Dennis, O. McAree, M. Fisher, S.M. Veres, Formal verification of autonomous vehicle platooning,(2016) ArXiv preprint arXiv:1602.01718.
[36] R. Kianfar, B. Augusto, A. Ebadighajari, U. Hakeem, J. Nilsson, A. Raza, R. Tabar, N. Irukulapati, C. Englund, P. Falcone, S. Papanastasiou, L. Svensson, H. Wymeersch, Design and experimental validation of a cooperative driving system in the grand cooperative driving challenge 13(3) (2012) 994–1007.10.1109/TITS.2012.2186513.
[37] J. Kolodko , L. Vlacic , Cooperative autonomous driving at the intelligent control systems laboratory, IEEE Intell. Syst. 18 (4) (2003) 8–11.
[38] B.W. Kolosz, S.M. Grant-Muller, K. Djemame, A macroscopic forecasting framework for estimating socioeconomic and environmental performance of intelligent transport highways 15(2) (2014) 723–736. 10.1109/TITS.2013.2284638.
[39] S. Konduri, P. Pagilla, S. Darbha, Vehicle formations using directed information flow graphs, in: Proceedings of the American Control Conference (ACC), 2013, pp. 3045–3050, doi: 10.1109/ACC.2013.6580298 .
[40] P. Kumar , R. Merzouki , B. Conrard , B. Ould-Bouamama , Multilevel reconfiguration strategy

for the system of systems engineering: application to platoon of vehicles, in: IFAC Proceedings Volumes, 47(3), 2014, pp. 8103–8109 .
[41] C. Lei , E.M. van Eenennaam , W.K. Wolterink , G. Karagiannis , G. Heijenk , J. Ploeg , Impact of packet loss on cacc string stability performance, in: 11th International Conference on ITS Telecommunications (ITST), IEEE, 2011, pp. 381–386 .
[42] H.C. Liu , J.X. You , Z.W. Li , G. Tian , Fuzzy petri nets for knowledge representation and reasoning: a literature review, Eng. Appl. Artif. Intell. (60) (2017) m45–56 .
[43] J. Lygeros , D.N. Godbole , M. Broucke , A fault tolerant control architecture for automated highway systems, IEEE Trans. Control Syst. Technol. 8 (2) (20 0 0) 205–219 .
[44] R. Merzouki , B. Conrard , P. Kumar , V. Coelen , Model based tracking control using jerky behavior in platoon of vehicles, in: Proceedings of the 12th European Control Conference (ECC), IEEE, Zurich, Switzerland, 2013, pp. 3488–3493 .
[45] B. Nemeth, P. Gaspar, Optimised speed profile design of a vehicle platoon considering road inclinations, Intell. Transp. Syst. IET 8 (3) (2014) 200–208, doi: 10.1049/iet-its.2012.0093 . 720.
[46] H. Paynter , Analysis and Design of Engineering Systems, MIT press, Cambridge, 1961 .
[47] J. Ploeg, E. Semsar-Kazerooni, G. Lijster, N. van de Wouw, H. Nijmeijer, Graceful degradation of cooperative adaptive cruise control 16(1) (2015) 488–497. 10.1109/TITS.2014.2349498.
[48] S. Stankovi , M. Stanojevi , D. iljak , Stochastic decentralized control of a platoon of vehicles based on the inclusion principle, in: A. Migdalas, A. Sifaleras, C.K. Georgiadis, J. Papathanasiou, E. Stiakakis (Eds.), Optimization Theory, Decision Making, and Operations Research Applications, Vol. 31 of Springer Proceedings in Mathematics & Statistics, Springer New York, 2013, pp. 223–239 .
[49] L. Stranden , A framework for complex scenarios for vehicle platoons, in: 18th IEEE International Conference on Intelligent Transportation Systems, IEEE, Las Palmas, 2015, pp. 2186–2193 .
[50] C. Toy , K. Leung , L. Alvarez , R. Horowitz , Emergency vehicle maneuvers and control laws for automated highway systems, IEEE Trans. Intell. Transp. Syst. 3 (2) (2002) 109–119 .
[51] W. Wang , The safety and comfort control of vehicles by the separation principle of pid controller tuning, in: IEEE International Conference on Industrial Technology (ICIT), IEEE, Vi a del Mar, 2010, pp. 145–150 .
[52] L. Xiao, F. Gao, Practical string stability of platoon of adaptive cruise control vehicles 12(4) (2011) 1184–1194. 10.1109/TITS.2011.2143407
[53] Y. Ying, T. Mei, Y. Song, Y. Liu, A sliding mode control approach to longitudinal control of vehicles in a platoon, in: Proceedings of the IEEE International Conference on Mechatronics and Automation (ICMA), 2014, pp. 1509–1514, doi: 10.1109/ICMA.2014.6885923 .
[54] Y. Zhang, G. Cao, V-pada: vehicle-platoon-aware data access in vanets 60(5) (2011) 2326–2339. 10.1109/TVT.2011.2148202.
[55] 3GPP TR 22.886, ”Study on enhancement of 3GPP Support for 5G V2X Services”, v15.1.0, Mar. 2017.
[56] 3GPP, “Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification (Release 14)”, TS 36.331 V14.4.0, Sept. 2017.
[57] 3GPP, “Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding (Release 14)”, TS 36.212 V14.4.0, Sept. 2017.
[58] J. Fink, A. Ribeiro, V. Kumar, Robust Control for Mobility and Wireless Communication in Cyber Physical Systems With Application to Robot Teams, Proceedings of the IEEE, vol. 100, no. 1, pp. 164-178, Jan. 2012.
[59] V. Gazis, A Survey of Standards for Machine-to-Machine and the Internet of Things, IEEE Commun. Surveys & Tutorials, vol. 19, no. 1, pp. 482-511, 1st Quarter, 2017.
[60] H. Xiao, Z. Li and C. L. P. Chen, Formation Control of Leader Follower Mobile Robots Systems Using Model Predictive Control Based on Neural- Dynamic Optimization, IEEE Trans. on Ind. Electr., vol.63, no. 9, pp. 5752-5762, Sept. 2016.
[61] J. B. Kenney, Dedicated Short-Range Communications (DSRC) Standards in the United States, Proceedings of the IEEE, vol. 99, no. 7, pp. 1162-1182, Jul. 2011.

[62] J-M. Lee, M-S. Woo and S-G. Min, Performance Analysis of WAVE Control Channels for Public Safety Services in VANETs, Int. J. Comp. and Communi. Engin, vol. 2, no. 5, Sept. 2013.
[63] S-L. Kim, W. Burgard and D. Kim, Wireless Communications in Networked Robotics, IEEE Wireless Communications, vol. 16, no. 1, pp. 4-5, Feb. 2009.
[64] G. Durisi, T. Koch and P. Popovski, Toward Massive, Ultrareliable, and Low-Latency Wireless Communication With Short Packets, IEEE Proceedings, vol. 104, no.9, pp. 1711-1726, Sept. 2016.
[65] M. Hadded, P. Muhlethaler, A. Laouiti, R. Zagrouba and L. A. Saidane, TDMA-Based MAC Protocols for Vehicular Ad Hoc Networks: A Survey, Qualitative Analysis, and Open Research Issues, IEEE Commun. Survey $\&$ Tutorials, vol. 17, no. 4, pp. 2461-2492, 4th Quarter 2015.
[66] J. Gu, S. J. Bae, S. F. Hasan and M. Y. Chung, Heuristic Algorithm for Proportional Fair Scheduling in D2D-Cellular Systems, IEEE Trans. Wireless Communi., vol. 15, no. 1, pp. 769-780, Jan. 2016.
[67] P. Fernandes and U. Nunes, "Platooning With IVC-Enabled Autonomous Vehicles: Strategies to Mitigate Communication Delays, Improve Safety and Traffic Flow," in IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 1, pp. 91-106, March 2012.
[68] The explainable AI platform using logic-based predictive analytics www.rulex.ai.
[69] S. Santini, A. Salvi, A. S. Valente, A. Pescapè, M. Segata and R. L. Cigno, "A consensus-based approach for platooning with inter-vehicular communications," 2015 IEEE Conference on Computer Communications (INFOCOM), Kowloon, 2015, pp. 1158-1166.
[70] L. Xu, L. Y. Wang, G. Yin and H. Zhang, "Communication Information Structures and Contents for Enhanced Safety of Highway Vehicle Platoons," in IEEE Transactions on Vehicular Technology, vol. 63, no. 9, pp. 4206-4220, Nov. 2014.

