Conference Paper

Reconfiguring TDMA Communications for Dynamic Formation of Vehicle Platoons

Aqsa Aslam
Frederico Santos
Luís Almeida*

*CISTER Research Centre

CISTER-TR-201007

2020/09/08

Reconfiguring TDMA Communications for Dynamic Formation of Vehicle Platoons

Aqsa Aslam, Frederico Santos, Luís Almeida*

*CISTER Research Centre

Faculdade de Engenharia Universidade do Porto (FEUP)

Rua Dr. António Bernardino de Almeida, 431

4200-072 Porto

Portugal

Tel.: +351.22.8340509, Fax: +351.22.8321159

E-mail: lda@fe.up.pt

https://www.cister-labs.pt

Abstract

Platooning is a promising concept used within the Intelligent Transportation System to increase efficiency and safety of road transportation. It is based on periodically sharing the kinematic status of the platoon members to allow reducing inter-vehicle distances in a safe way. This coordination is automatic and depends heavily on the wireless channel. A common technique to improve the channel properties is to use Time-Division Multiple Access (TDMA) that organizes the access to the wireless medium in slots assigned exclusively to each vehicle. However, while platoons are physical and dynamic, the corresponding dynamic reconfiguration of a logical TDMA frame is non-trivial. In this paper we address this Cyber-Physical problem resorting to the RA-TDMAp protocol to track the dynamics of a platoon, specifically joining, merging and leaving. In our solution, we include an adequate admission control block, to verify whether joining or merging can be accepted, and we present the state-machine that handles the reconfiguration process. We validate our TDMA reconfiguration mechanism with simulations using the Plexe/Vein/OMNeT++ framework. We show the effectiveness of the proposed mechanisms which ensures a synchronized start of the platoon control with the TDMA frame reconfiguration.

Reconfiguring TDMA Communications for Dynamic Formation of Vehicle Platoons

Aqsa Aslam
FEUP / Instituto de Telecomunicações
Universidade do Porto
Porto, Portugal

Frederico Santos
ISEC – DEE / Instituto de Telecomunicações
Instituto Politécnico de Coimbra
Coimbra, Portugal

Luis Almeida
FEUP / CISTER
Universidade do Porto
Porto, Portugal

***Abstract*—Platooning is a promising concept used within the Intelligent Transportation System to increase efficiency and safety of road transportation. It is based on periodically sharing the kinematic status of the platoon members to allow reducing inter-vehicle distances in a safe way. This coordination is automatic and depends heavily on the wireless channel. A common technique to improve the channel properties is to use Time-Division Multiple Access (TDMA) that organizes the access to the wireless medium in slots assigned exclusively to each vehicle. However, while platoons are physical and dynamic, the corresponding dynamic reconfiguration of a logical TDMA frame is non-trivial. In this paper we address this Cyber-Physical problem resorting to the RA-TDMAp protocol to track the dynamics of a platoon, specifically joining, merging and leaving. In our solution, we include an adequate admission control block, to verify whether joining or merging can be accepted, and we present the state-machine that handles the reconfiguration process. We validate our TDMA reconfiguration mechanism with simulations using the Plexe/Vein/OMNeT++ framework. We show the effectiveness of the proposed mechanisms which ensures a synchronized start of the platoon control with the TDMA frame reconfiguration.**

***Keywords*—VANET, MAC, TDMA, CSMA/CA, IEEE 802.11p, Vehicles platooning.**

I. INTRODUCTION

Vehicle platooning is a technique that organizes traffic in groups of close-following vehicles called *platoons*. It has been attracting much attention for over two decades due to its ability to improve traffic efficiency and safety within the so-called Intelligent Transportation System (ITS) [1][2]. A vehicle platoon is an example of a Cyber-Physical System, in which the vehicles (the physical part) execute a car-following application (the cyber part) engaging both the execution of feedback-control with local sensors and the wireless communication of vehicles' kinematic status [3].

The most widely studied platoon configuration in transportation is the column, also known as *road train* [4]. This configuration considers longitudinal operations of the platoon, only, with vehicles following exactly the dynamics of the platoon leader generally using some kind of cruise control, either adaptive (ACC) or cooperative and adaptive (CACC) [5][6]. Specifically, the latter relies on inter-vehicle communications and must consider inherent communication delays and losses [2]. The platoon control may also benefit from specific structuring of the communications [7] that improves the channel quality. Further channel improvements can be achieved enforcing a Time-Division Multiple Access (TDMA) scheme to reduce collisions in the access to the communication medium [8][9].

Beyond the essential function of car-following, there are a number of practical aspects related to the management of platoons that must be handled adequately, such as vehicles joining and leaving, as well as platoons splitting and merging, or even change of platoon leader. These maneuvers have been addressed at the control application level [4]. However, when using structured communications, such as platoon-oriented TDMA frameworks, the reconfiguration of both platoon control and TDMA frame must be synchronized. Curiously, few works, only, have addressed this problem in the past, e.g. [8].

In this paper, we address the case of a recently proposed protocol for Automated Highway Systems (AHS), namely Reconfigurable and Adaptive TDMA for platooning (RA-TDMAp) [9], and we propose a dynamic frame reconfiguration mechanism that tracks the current platoon configuration. RA-TDMAp is a fully distributed TDMA overlay protocol on top of IEEE 802.11p, which uses Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) medium access control. Thus, RA-TDMAp combines the benefits of both TDMA and CSMA/CA paradigms, namely collision reductions with efficient bandwidth use. This efficiency results from the adaptive feature of the protocol that enforce the co-existence of frames of neighboring platoons with minimum separation.

The dynamic reconfiguration feature of RA-TDMAp is novel and constitutes the main contribution of this work. We discuss it in the context of platoon formation and splitting maneuvers and we show its state machine together with validating simulations on Plexe/Veins/OMNeT++.

This paper is organized as follows. In Section II we present related work focused on platoon formation techniques. Section III presents the proposed TDMA reconfiguration approach while Section IV presents the validation of the proposed approach through simultions in different highway scenarios. Finally, Section V concludes the paper.

II. COMMUNICATIONS AND PLATOON FORMATION

Platoon formation, i.e., building up the physical platoon configuration, has been extensively addressed in the literature, but mostly from the control application perspective. Here

978-1-7281-8956-7/20/$31.00 ©2020 IEEE

we briefly revisit some representative works that consider the platoon communications jointly with the platoon control during platoon formation.

In this topic, the work in [10] deserves a particular reference since it proposed an integrated simulator, Plexe, for studying joint control and communication strategies for longitudinal platoons operation, which had a significant impact in the research community. It also proposed a specific coordination of the communications inside each platoon, synchronizing with offsets the CAM messages of IEEE 802.11p transmitted by the platoon vehicles. Extensive comparisons between this communication protocol and RA-TDMAp were presented in [11] showing a significant improvement in channel metrics in favor of the latter. Moreover, concerning platoon formation, the work addressed the join maneuver, only, and from a control application perspective, not clarifying the dynamic reconfiguration of lower-layer communications.

In [12], the authors proposed a high-level platoon management protocol that supported three basic maneuvers, namely platoon merge, split and lane change. This protocol uses CACC as control strategy and defines several specific IEEE 802.11p messages to negotiate the platoon reconfiguration. However, no specific reconfiguration of the communications used in CACC operation is done. Similarly, the work in [13] proposed a protocol specifically for platoon merging in highways, when two platoons that run side by side have to merge due to road works on one of the lanes. The authors introduced new messages to support the merging negotiation and maneuvering scenarios, but without reconfiguring the messages used for platoon control.

The work in [8] proposes a TDMA-based communication protocol, VeSOMAC, that supports platooning and self-reconfigures the TDMA frame to track dynamically the current platoon topology. It is based on in-band signaling using the vehicles beacons to carry information about allocated slots, thus allowing fast slot reconfiguration upon topology changes such as when a vehicle joins the platoon. Despite considering control applications as a motivation for the reconfigurable TDMA approach, the work does not clearly synchronize the platoon controller activity with the communication protocol states and focuses on the communications aspect.

III. DYNAMIC RECONFIGURATION OF RA-TDMAP

RA-TDMAp was proposed in [9] and later extensively studied in simulation in [11] but its reconfiguration mechanism was not defined until this work. Previous RA-TDMA protocols [14][15] addressed cases of teams of autonomous robots and had their own dynamic reconfiguration mechanism to cope with run time joining and leaving of robots. However, the specific power control used in RA-TDMAp and the position constraints imposed by the platoon maneuvers and operation make the reconfiguration mechanism of the previous protocols non-applicable to the current case.

Therefore, in this section we propose a specific reconfiguration mechanism for RA-TDMAp that is suitable for operation in realistic highway scenarios and to support the platoon maneuvers that are expected therein. As referred before, our proposal enforces the synchronization between the platoon control and the platoon TDMA frame so that topological changes in the platoon are reflected in both domains, simultaneously. This is achieved with a single state machine that runs in all vehicles, without global information across platoons.

A. RA-TDMAp Basics and System Model

In this work we consider a set of vehicles that move in a highway scenario and which are engaged in a platooning application, exploiting opportunities to build up platoons. Each vehicle has a unique vehicle identifier and each platoon is composed by a leader and $n-1$ followers, where n is the current number of platoon members. Note that $1 \leq n \leq N$, where N is a limit to the platoon size that is specific to each platoon and may be determined by multiple criteria such as control algorithm, communication range and weather or road conditions. Platoons have a unique platoon identifier, too, which can be derived from the respective leader vehicle identifier. When a vehicle activates the platooning application, it immediately starts as leader without followers ($n = 1$).

In RA-TDMAp, all members of a platoon transmit their IEEE 802.11p beacons (CAM) organized in a periodic TDMA frame with duration T_{tup}, typically 100ms. This frame is divided in n regular slots of T_{tup}/n duration. Each platoon member transmits in the beginning of a specific slot using an adequate offset. The beginning of the TDMA frame, and thus offset 0, is marked by the leader transmission, which is carried out using high power, typically 100mW, so that it reaches all platoon members. Conversely, the transmissions of the followers are carried out with low power, typically 1mW, to decrease the medium usage. Consequently, they may not reach all members in the platoon and a multi-hop scheme is needed to convey the information through the platoon from the last vehicle to the leader. This operation is optimized by using a reverse transmissions order with respect to the position in the platoon. Immediately after the leader, the first vehicle to transmit, with offset T_{tup}/n, is the last vehicle. The second vehicle to transmit, with offset $2{*}T_{tup}/n$, is the one before the last, and so on until the first follower, which transmits in the last slot with offset $(n-1) * T_{tup}/n$. This scheme guarantees that the leader is able to collect information from all followers in a single TDMA round.

One fundamental information that is collected every TDMA round and transmitted in multi-hop to the leader is the delay with which every follower receives the beacons from the followers behind. These delays are caused by the IEEE 802.11p CSMA/CA medium access control when there is interference due to other concurrent transmissions, such as from neighboring platoons. These delays are then used by the leader to shift the phase of the next TDMA frame, i.e., delay its start, so that the following platoon transmissions will likely avoid the referred interference. This is the adaptive feature of

RA-TDMA protocols that allows the seamless coexistence of multiple interleaved TDMA frames [11].

Since the leader gathers information from the whole platoon every TDMA round, it is in a privileged position to enforce consistency in the platoon. For this purpose, the leader announces in its beacon the current TDMA frame structure, namely the current number of slots, which equals the number of platoon members, and a vector with the current platoon formation. This is used by all followers to determine their slot and compute the respective offset.

Finally, the proper operation of RA-TDMAp, as common in ITS protocols, requires the vehicles to know their positions. Thus, we assume that all vehicles are equipped with a Global Navigation Satellite System (GNSS) such as GPS.

The structure of RA-TDMAp regular beacons (Type-0) of both leader and followers is shown in Table I.

B. RA-TDMAp Reconfiguration Mechanism

As a specific goal for our work we consider the following concrete platoon maneuvers that we wish to support:

1) **Joining**: Independent vehicles traveling in a highway find one or more platoons ahead of them, select one and request joining;
2) **Merging**: A platoon finds another platoon ahead of it, in the same lane, and issues a merging request;
3) **Follower leaving**: A follower vehicle in the platoon announces it wishes to leave;
4) **Leader leaving**: The platoon leader announces its intention to leave;

The first two maneuvers involve different entities, either independent vehicles or platoons, that will merge in a single platoon. The last two maneuvers start from a single platoon in which one of its vehicles wishes to leave. We consider that multiple vehicles can leave a platoon, but one at a time, only. Direct splitting of one platoon into several with more than one vehicle is not currently supported. For simplicity of terminology, we refer to leaving announcements as requests.

Each maneuver is coordinated by a structured exchange of specific IEEE 802.11p CAM messages (beacons), piggybacked with necessary information. The regular RA-TDMAp beacons are used for certain implicit confirmation actions, too. The additional beacons are transmitted periodically, with the same T_{tup} period, but during a short interval of time, until proper acknowledgement is received. These beacons are transmitted asynchronously with respect to the RA-TDMAp frame and handled by the CSMA/CA medium access mechanism of IEEE 802.11p. They are classified in two groups, namely request and response. All the beacons implied in the RA-TDMAp dynamic reconfiguration are displayed in Table I. We use request/response beacons for convenience of implementation and not to tamper the RA-TDMAp regular beacons. Alternatively, we could use just the RA-TDMAp beacons with changing types, but this would require frequent changes of the beacon semantics and structure, which we opted to avoid.

During steady operation, all vehicles engaged in platooning are in one of two states, *Leader* or *Follower*. During reconfiguration, some of the involved vehicles have to temporarily move to transient maneuvering states. In order to support the maneuvers specified before, we use three extra states, two related to the process of joining or merging, and the third one related to the process of leader leaving, which requires designating a new leader. All the states are listed here:

- **Leader**: Platoon head, or single vehicle looking for platooning opportunities (high power beacons);
- **Follower**: All the trailing vehicles in a platoon after the platoon head (low power beacons);
- **Decide platoon**: A leader (single vehicle or platoon) that detects one of more platoons ahead of it that it can join, deciding which to join and moving to a *joining position* (high power beacons);
- **Waiting for response**: A joining leader that issued a join request to a specific platoon, and is in the joining position waiting for the respective response (high power beacons);
- **Leader election**: First follower of a leader that issued a leave request (low power beacons).

While in the *Leader* state, a vehicle has to carry out three main functions. It has to execute the TDMA frame phase adjustment, announce the current platoon presence and composition, and handle any joining requests that may appear. However, when a leader enters a joining process and moves to the maneuvering states, it still does the phase adjustment and platoon announcement, but blocks joining requests. This avoids potentially inconsistent states of a leader simultaneously asking to join a platoon while receiving joining vehicles itself. From the maneuvering states, a leader will eventually become a follower or return to its original *Leader* state.

On the other hand, the *Leader election* state is a transient state in which the first follower acknowledges that it is ready to take over, so that the leaving leader can exit the platoon.

While in the *Follower* state, the vehicles execute an adequate vehicle-following controller such as CACC. Note that followers do not seek other platoons. They remain in the same platoon until explicitly signaling intention to leave or disengaging the platooning application.

The state machine that governs the reconfiguration process using the states referred above is shown in Figure 1. This state machine is the same for all vehicles engaged in a platooning application with RA-TDMAp. Note that, initially, when the platooning application is turned on in any vehicle, it immediately starts as a leader. By detecting neighboring leaders in adequate relative positions, the vehicles use the joining maneuver and start building up platoons. In the remainder of this section we go through each maneuver scenario to observe how the state machine operates.

1) Joining: The joining maneuver is illustrated in Figure 2. In this scenario we have a platoon travelling in the highway and, for now, let us assume there is one independent vehicle approaching the platoon from the tail. Both independent vehicle and platoon leader are in the *Leader* state and transmitting

TABLE I
RA-TDMAp BEACON TYPES AND FORMATS

Regular	Type-0	Platoon	Source	Speed	Position	LaneID	Numberofslots	PlatoonFormation[]	DelayList[]
Request	Type-1	PlatoonID	SourceID	Position	LaneID	Speed	TargatedPlatoonID		
	Type-2	PlatoonID	SourceID	LeaveInform					
Response	Type-3	PlatoonID	SourceID	Distance	Lane	Capacity			
	Type-4	PlatoonID	SourceID	Ack					

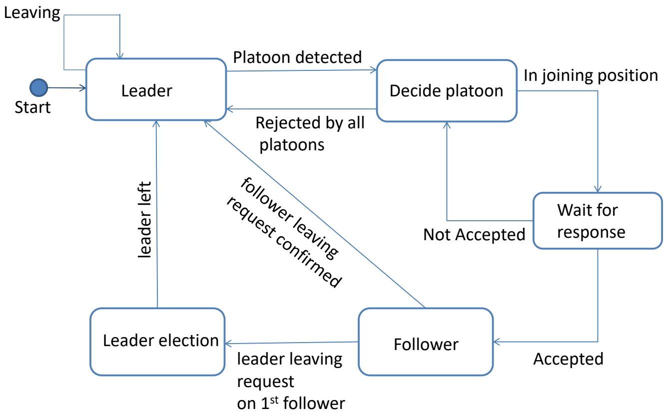

Fig. 1. RA-TDMAp reconfiguration state machine

Type-0 beacons with high power, asynchronously to each other. When the vehicle starts receiving the platoon beacons, it checks whether the platoon is in a compatible position, e.g., sufficiently ahead, and in that case moves to the *Decide platoon* state. In this state the vehicle checks whether there are other potential platoons it can join in different lanes, decides for one based on convenient criteria, e.g., lane, distance, speed, and moves to a joining position, which is at a certain distance d behind the chosen platoon tail in the same lane.

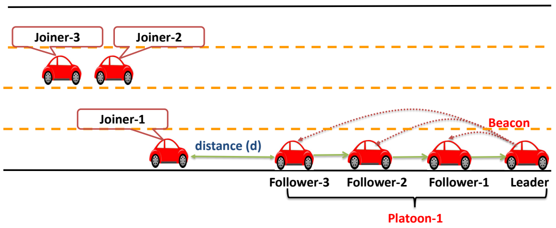

Fig. 2. Joining scenario with multiple independent vehicles reaching up to a platoon and requesting to join

Once the joining vehicle gets to that position, it moves to the *Wait for response* state and starts emitting *Type-1* join request beacons indicating the ID of the platoon to join. These beacons are transmitted with high power to guarantee reaching the leader. The chosen platoon leader receives these beacons and triggers an admission control checking three parameters: (i) if the joiner is in the same lane of the platoon; (ii) if the current number of platoon members n plus the joiner does not override its limit N; and (iii) if the joiner is within a certain range around distance d from the platoon tail. If these conditions are met, the platoon leader integrates the joiner incrementing n and adding its ID to the platoon formation vector, and starts sending *Type-3* response beacons. Upon receiving these beacons, the joiner moves to *Follower* state and engages the platoon vehicle-following controller. Once the leader verifies that the joiner is integrated among its followers, it stops the response beacons and concludes the admission process.

If multiple independent vehicles that do not engage in joining among themselves detect the same platoon they may all start concurrent joining processes. However, they remain in the *Wait for response* state (or in the *Decide platoon* state until reaching the joining position) while the platoon handles the join requests in sequence, one at a time. If the independent joiners engage in a joining process among themselves, then they will first conclude the creation of a new platoon before attempting a merge with the other platoon. Remember that a leader is either handling joining processes or issuing joining requests, but not both at the same time.

2) Merging: In this scenario, currently limited to *concatenation*, one platoon with $n > 1$ (the rear platoon) is traveling in the highway and joins another platoon ahead of it (the front platoon). The leader of the rear platoon follows through the maneuvering states. In the *Decide platoon* state it decides whether to try joining the front platoon. Due to control complexity, currently we do not consider merging of platoons in different lanes. In that case, the rear leader returns to the *Leader* state marking the front platoon as non-suitable for joining for a certain time, after which it may try again. If the front platoon is in the same lane, the rear platoon leader may request merging. In this case, it moves, dragging its followers, to a joining position at distance d behind the tail of the front platoon. Once there, it moves to the *Wait for response* state and starts emitting the *Type-1* request beacon, which triggers a request for joining in the leader of the front platoon.

Comparing to the previous case, the request beacon now indicates that the joining vehicle is a platoon leader with $n_{rear} - 1$ followers. The request also triggers the admission control in the front leader, which now considers all the n_{rear} vehicles of the rear platoon. If all can be accommodated in the front platoon, the front leader starts emitting *Type-3* response beacons with a positive response and changes its *Type-0* regular beacon with the updated information of the merged platoon (number of vehicles $n_{front} + n_{rear}$ and platoon formation). When the *Type-3* response beacons are received by the rear platoon vehicles they all move to the TDMA frame of the enlarged front platoon. The rear leader, stops emitting the request beacon, changes its state to *Follower* and switches on the platoon vehicle-following controller that will close the gap to the front platoon.

If the admission control in the front leader fails, the *Type-3* response beacons will indicate this condition. The rear leader also stops emitting *Type-1* requests and returns to the *Leader* state (through the *Decide platoon* state). Its TDMA frame will remain unchanged and its followers will remain with it.

3) Follower leaving: Followers remain in this state until they disengage the platooning application or issue an explicit request to leave. In the former case, by disengaging platooning a vehicle will no longer participate in any platoon and stops executing the RA-TDMAp state machine and sending the respective beacons. The leader will detect the omissions of that follower and after a pre-defined number of consecutive TDMA rounds with omissions, the leader reconfigures the platoon and the TDMA frame, excluding the vehicle.

Alternatively, a follower willing to leave may start emitting *Type-2* requests that cause the leader to reconfigure the platoon and TDMA frame to remove the requesting follower. When this follower detects it is no longer in the platoon formation announced by the leader in its regular *Type-0* beacons, it considers itself out of the platoon, stops emitting request beacons and moves to the *Leader* state.

In both cases, the leaving vehicle may stay amidst the platoon for some time, until it physically leaves, e.g., by taking an out ramp (Figure 3). This does not compromise the platoon operation. While in the middle of the platoon, the platooning controllers, e.g., CACC, will keep a safe distance to that vehicle. As soon as the vehicle leaves its position, the platooning controller in the following followers will make them close the gap to the previous followers in the platoon.

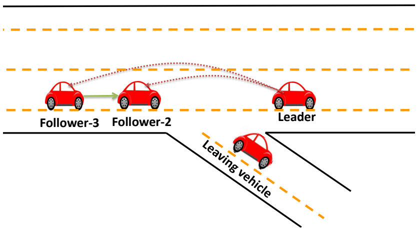

Fig. 3. Follower leaving a platoon

4) Leader leaving: In this case, the first follower is promoted to leader, taking over the role of synchronizing the platoon, rearranging the TDMA frame and platoon formation accordingly. If the leader simply disengages the platooning application, its absence will be detected by omission after some time (pre-defined number of consecutive omissions). This will be a period without synchronization and TDMA frame adaptation. During this time the first follower keeps its speed constant to avoid following an absent or unreliable leader. The leader may also request leaving by emitting *Type-2* requests. These will be detected by the first follower that moves to the *Leader election* state. Once in this state, it emits *Type-4* response beacons until the leader leaves. This is signalled by the leaving leader stopping emitting the request beacons and changing its platoon ID to a different value and its formation to itself alone. In the original platoon, the first follower then moves to the Leader state, reconfigures the TDMA frame and platoon formation and resumes regular platoon operation. The new platoon with the leaving leader does not accept join requests until it is sufficiently away from the original one.

IV. VALIDATION

In this section we show a validation of the proposed RA-TDMAp dynamic reconfiguration mechanism in the maneuvering scenarios that were referred before. We will use simulation for our validation purposes, relying on the Plexe/Veins/OMNeT++ framework [10][16]. We first present the simulation setup and then simulation results that confirm the proper operation of the proposed mechanism.

A. Simulation Setup

Plexe is the current state-of-the-art system level platooning simulator, incorporating mobility tightly-coupled with automatic control and communications. It allows defining platooning applications on highway scenarios, including the full stack of IEEE 802.11p/ IEEE 1609.4 network standards and analyzing network metrics such as collisions and packet delivery ratio. We use these features to validate the proposed reconfigurable mechanism of RA-TDMAp in a straight four-lanes highway scenario.

1) Communication model: We use the PHY and MAC models of IEEE 802.11p proposed in [17], with a bitrate of 6 Mbit/s, which is suited for demanding safety related applications [18]. The leader transmission power is set to 100 mW (high power) and the followers to 1 mW (low power). We employed the common free-space propagation model with $\alpha = 2.0$. Furthermore, we used the Control Channel (CCH), only, without the Service Channel (SCH), and all beacons used the same Access Category, namely (AC_VI). Table II summarizes all communication related parameters.

TABLE II
PHY AND MAC PARAMETERS

Parameter	Values
PHY/MAC model	IEEE 802.11p/1609.4, CCH only
Path loss model	Free space ($\alpha = 2.0$)
Channel	5.89 GHz
Bitrate	6 Mbit/s
MSDU size	200B
Access category	(AC_VI)
Leader Tx power	100mW
Followers Tx power	1mW

B. Validating the reconfiguration mechanism

To validate the reconfiguration mechanism we carried out simulations recreating the previous maneuvering scenarios. Each simulation trace is 100s long and we logged all the transmissions in that interval. Target speeds are set to 100km/h and RA-TDMAp round periods are set to 100ms with a small random variation. The platooning application uses CACC with a target distance between vehicles of 5m and the joining position d is 17m behind the platoon tail.

1) Joining: To validate the joining maneuver we consider one independent vehicle (ID=4) and one platoon with four vehicles (ID=0 -leader; IDs=1,2,3 -followers). Figure 4 shows the TDMA frame reconfiguration process in time for $0 \leq t \leq 70$s (the remaining part of the trace has no relevant information). The top plot shows the offsets of the vehicles transmissions relative to the transmission of the leader (offset 0) in each TDMA round. A vertical cut would show the actual offsets in a particular TDMA frame. The top line is the offset of the next leader transmission relative to the previous one, i.e., the actual TDMA round period. Thus, from top to bottom we have the offsets of vehicles 0 through 3. These offsets are constant due to the synchronization enforced by RA-TDMAp and the absence of interference. The lower plot in the figure shows the inter-vehicle distances. The upper line is the distance between the joining vehicle and the platoon tail and the lower line is the distance between the other vehicles (overlapped), which is constant at 5m as enforced by the CACC.

The joining vehicle was inserted in the simulation at $t = 5$s at a distance of 100m to the platoon tail and traveling at the same speed, with the platooning application switched off. At $t = 15$s it engages the platooning application and enter the *Leader* state, becoming an independent leader, and starts transmitting its own beacon (black line). Since they are not synchronized and its effective T_{tup} is slightly shorter than that of the platoon, its offsets (phases) relative to the platoon leader keep changing linearly, thus the diagonal traces.

Soon after starting the platooning application it detects the beacons of the platoon leader and moves to *Decide platoon* state. Since this is the only platoon in range and it is sufficiently ahead of it (more than distance d), it chooses to try joining this platoon and moves to the joining position (17m behind the tail). It reaches this position at $t = 49s$, moving to the *Wait for response* state and sending the join request beacon (Type-1). The platoon leader receives the request, runs the admission control and sends the joiner a response beacon (Type-3) informing acceptance. In this case, just one beacon of each type was sent. If one of them is not received, the protocol waits for the next beacon, since they are transmitted periodically until successfully received.

At this point the leader reconfigures the TDMA frame updating the number of platoon vehicles to $n = 5$ and the formation vector to include vehicle ID=4. All followers start using the offsets of the new frame. The joiner changes to the *Follower* state, starts transmitting in the right offset in the platoon frame and switches on the CACC. The CACC then controls the speed of the joiner to bring it to a distance of 5m to the next follower, which it achieves at $t = 70$s.

2) Merging: This scenario is illustrated in Figure 5, where the two plots have similar semantics as in the previous case. We have two platoons with $n = 4$, entering the simulation at the same speed. Platoon 0 (in blue) enters first and contains vehicles 0 (leader), 1, 2 and 3. Platoon 1 (in red) comes after and contains vehicles 4 (leader), 5, 6 and 7. The upper plot shows the offsets of the transmissions of both platoons with respect to the leader of platoon 0 (blue). Again, while the platoons are separated, their transmissions are synchronized internally but not between platoons. Thus, the offsets of platoon 1 (red) appear in diagonal when referred to the transmissions of platoon 0 (blue), but showing parallel lines, i.e., they are internally synchronized.

The lower plot in Figure 5 shows the inter vehicle distances. Initially, all distances inside each platoon are at 5m and we insert platoon 1 at 33m behind the tail of platoon 0. We give some initial time for platoon 1 to be fully inserted in the simulation before allowing its state machine to evolve. For this reason, only at $t = 10$s platoon 1 reacts to the detection of platoon 0 in a compatible position and its leader (ID=4) transitions to the *Decide platoon* state. In that state it decides to merge with platoon 0 and thus brings its platoon to the joining position (17m). The leader gets there at $t = 34$s. At that point, it transitions to *Wait for response* state and starts emitting joining/merging request beacons (Type-1) informing the leader of platoon 0 that it wishes to merge with its 4 vehicles altogether.

The platoon 0 leader runs the admission control and, given the positive outcome, it reconfigures the TDMA frame accordingly, i.e., increases its n to 8 and adds vehicles ID 4 through 7 to its formation vector. Then, it starts emitting response beacons (Type-3) allowing the merging. At that point the vehicles of platoon 1 start engaging the platoon 0 with their corresponding offsets. The leader of platoon 1 moves to the *Follower* state, starts transmitting with low power in the right offset of the TDMA frame of platoon 0 and activates the CACC. As soon as the leader detects that all vehicles of the joining platoon are merged, it stops emitting the response beacons. The activation of the CACC in the follower that was the previous leader closes the gap between both platoons, and all inter-vehicle distances converge to 5m at $t = 70$s.

3) Follower leaving: Figure 6 shows the reconfiguration process when a follower leaves the platoon. We consider platoon 0 traveling in a highway with 4 vehicles (IDs 0-leader, 1, 2 and 3). In the upper plot (offsets) we can see that, at $t = 21$s, the first follower announces it is leaving the platoon. For that purpose, still in the *Follower* state it starts emitting the a request leaving beacon. The platoon leader receives this beacon and reconfigures the platoon and the TDMA frame, removing the vehicle from the formation vector and decrementing n. When the leaving follower receives a regular beacon from the platoon leader showing it is no longer in the platoon, it changes to *Leader* state. This is clearly visible in the offsets plot. The leaving vehicle is now transmitting asynchronously with respect to the platoon.

In this experiment, we make the leader move away from the platoon, driving to a different lane. This is visible in the lower plot, where we can see the sudden increase in the distance between the previous second follower, now promoted to first

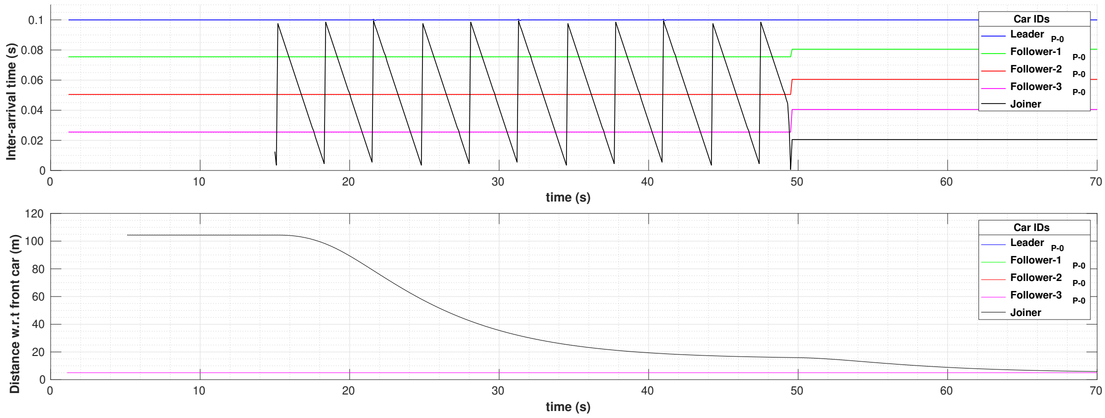

Fig. 4. One independent vehicle joining a platoon from the tail

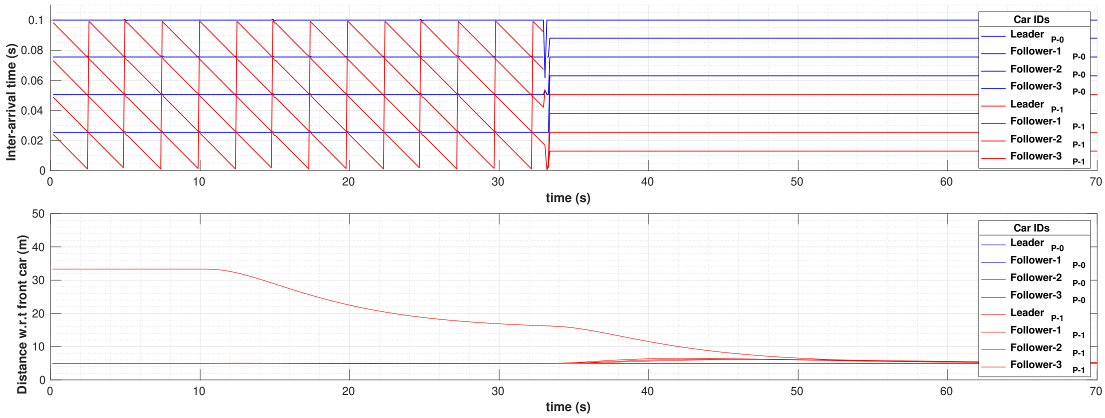

Fig. 5. Scenario 2: Merge of two platoons, traveling on same lane

follower, and the platoon leader. As the leaving car moves to another lane, the CACC of the followers brings them to the target 5m inter-vehicle distance, converging at $t = 40$s.

4) Leader leaving: Figure 7 shows the RA-TDMAp reconfiguration in the case a leader explicitly leaves a platoon. In this case, the inter-vehicle distances are not relevant and are thus omitted. We use the same scenario as in the previous case, with one platoon of four vehicles. However, in this case it is the leader that announces its willingness to abandon the platoon at $t = 20$s. It starts emitting the leave request beacon (Type-2), which is captured by the first follower causing it to transition to the *Leader election* state. In this state, the first follower starts emitting the response beacon (Type-4) acknowledging that it is ready to take over the leader role. Upon receiving this beacon, the current leader creates a new platoon just with itself and leaves the previous platoon. The first follower waits for one omission of the regular beacon (Type-0) of the previous leader and changes to *Leader* state, effectively taking over the platoon. This occurs at the time it would do its regular beacon transmission. The figure shows the offsets referred to the old leader (blue). The reconfigured platoon, with one vehicle less, maintains its internal synchronization, despite being asynchronous to the old leader.

C. *Global Comments*

The results in this section validate the correct behavior of the reconfiguration mechanism and its capacity to keep the platoon controller synchronized with the corresponding state of the communication protocol. The transition between configurations was swift, even under distributed operation, reducing the interval during which synchronization ambiguities can occur. Our simulations, under low traffic, revealed reconfiguration times below 2.5 TDMA rounds. Losses of signalling beacons extend this interval an integer number of TDMA rounds. The impact of high traffic is left for future work.

V. CONCLUSION

Platooning of vehicles in highways is an important Cyber-Physical application within the ITS. It requires regularly sharing vehicles kinematic states for effective and safe speed control. Practical platooning requires handling reconfiguration scenarios, to allow vehicles joining or leaving. These scenarios have been addressed mostly from the control point of view, disregarding the communications reconfiguration. However, this is mandatory when using TDMA-based protocols. This paper proposed a novel reconfiguration mechanism for RA-TDMAp, a protocol that has shown superior collision reduction ability in previous work [11]. The paper presented the

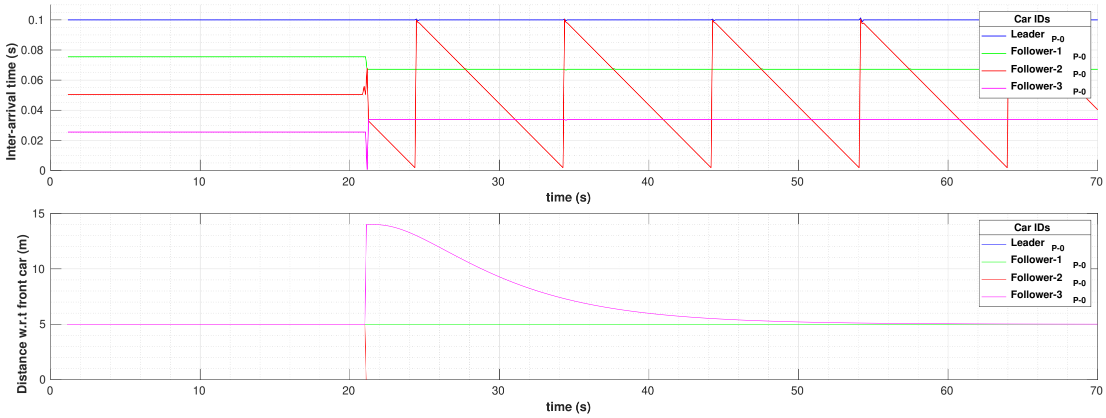

Fig. 6. Follower leaving with explicit announcement

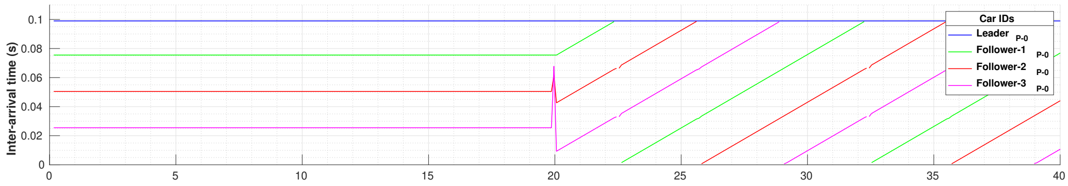

Fig. 7. Leader leaving the platoon

state machine of the reconfiguration mechanism, which is the same for all vehicles, and validated it with simulation using Plexe/Veins/OMNeT++ and realistic joining, merging and leaving scenarios in highways. Future work will consider more reconfiguration scenarios, including with high traffic density and non-longitudinal platoon operation (e.g, zipper merging [19]), and validation with IEEE 802.11p equipment.

REFERENCES

[1] M. Michaelian and F. Browand, "Field experiments demonstrate fuel savings for close-following," *UC Berkeley: California Partners for Advanced Transportation Technology*, 2000.

[2] P. Fernandes and U. Nunes, "Platooning with IVC-enabled autonomous vehicles: Strategies to mitigate communication delays, improve safety and traffic flow," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 91–106, 2012.

[3] J. Ploeg and *etal*, "Cooperative Automated Maneuvering at the 2016 Grand Cooperative Driving Challenge," *IEEE Trans. on Intelligent Transportation Systems*, vol. 19, no. 4, pp. 1213–1226, April 2018.

[4] E.-Z. Madeleine, B. Dafflon, F. Gechter, and J.-M. Contet, "Vehicle platoon control with multi-configuration ability," *Procedia Computer Science*, vol. 9, pp. 1503–1512, 2012.

[5] Z. Wang, G. Wu, and M. J. Barth, "A Review on Cooperative Adaptive Cruise Control (CACC) Systems: Architectures, Controls, and Applications," in *2018 21st Int. Conf. on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2884–2891.

[6] Z. Wang, G. Wu, P. Hao, K. Boriboonsomsin, and M. Barth, "Developing a platoon-wide eco-cooperative adaptive cruise control (CACC) system," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 1256–1261.

[7] M. Segata, B. Bloessl, S. Joerer, F. Dressler, and R. L. Cigno, "Supporting platooning maneuvers through IVC: An initial protocol analysis for the join maneuver," in *Wireless On-demand Network Systems and Services (WONS), 2014 Annual Conf. on*. IEEE, 2014, pp. 130–137.

[8] F. Yu and S. Biswas, "Self-configuring TDMA protocols for enhancing vehicle safety with DSRC based vehicle-to-vehicle communications," *IEEE journal on selected areas in communications*, vol. 25, no. 8, 2007.

[9] A. Aslam, L. Almeida, and F. Santos, "A flexible TDMA overlay protocol for vehicles platooning," in *Communication Technologies For Vehicles, Int. Workshop, Net4Cars 2018*. Springer, 2018, pp. 169–180.

[10] M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler, and R. Lo Cigno, "PLEXE: A Platooning Extension for Veins," in *6th IEEE Vehicular Networking Conference (VNC 2014)*. Paderborn, Germany: IEEE, December 2014, pp. 53–60.

[11] A. Aslam, P. Santos, F. Santos, and L. Almeida, "Empirical Performance Models of MAC Protocols for Cooperative Platooning Applications," *Electronics*, vol. 8, no. 1334, 2019.

[12] M. Amoozadeh, H. Deng, C.-N. Chuah, H. M. Zhang, and D. Ghosal, "Platoon management with cooperative adaptive cruise control enabled by VANET," *Vehicular Communic.*, vol. 2, no. 2, pp. 110–123, 2015.

[13] H. H. Bengtsson, L. Chen, A. Voronov, and C. Englund, "Interaction protocol for highway platoon merge," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. IEEE, 2015.

[14] F. Santos, L. Almeida, and L. S. Lopes, "Self-configuration of an adaptive TDMA wireless communication protocol for teams of mobile robots," in *Proceedings of the 2008 IEEE International Conference on Emerging Technologies for Factory Automation. ETFA*. IEEE, 2008.

[15] L. Oliveira, L. Almeida, and D. Mosse, "A Clockless Synchronisation Framework for Cooperating Mobile Robots," in *Proceedings of the 2018 IEEE Real-Time and Embedded Technology and Applications Symposium. RTAS*. IEEE, 2018, pp. 305–315.

[16] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Trans. on Mobile Computing*, vol. 10, no. 1, pp. 3–15, January 2011.

[17] D. Eckhoff and C. Sommer, "A Multi-Channel IEEE 1609.4 and 802.11p EDCA Model for the Veins Framework," in *5th ACM/ICST International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools 2012)*. Desenzano, Italy: ACM, March 2012.

[18] D. Jiang, Q. Chen, and L. Delgrossi, "Optimal data rate selection for vehicle safety communications," in *5th ACM Int. Workshop on VehiculAr Inter-NETworking*. ACM, 2008, pp. 30–38.

[19] B. R. J. Batsuuri, T. and J. J. Lukkien, "Model, analysis, and improvements for inter-vehicle communication using one-hop periodic broadcasting based on the 802.11p protocol," in *Wireless Sensor Networks: Vehicle and Space Applications*, D. Benhaddou and A. Al-Fuqaha, Eds. New York: Springer, 2015, pp. 185–215.