

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Control Architecture for Cooperative Autonomous Vehicles Driving in Platoons at Highway Speeds

OVIDIU PAUCA¹, ANCA MAXIM¹, AND CONSTANTIN-FLORIN CARUNTU^{1,2}

¹ Department of Automatic Control and Applied Informatics, "Gheorghe Asachi" Technical University of Iasi, Iasi 700050, Romania; pauca.ovidiu@ac.tuiasi.ro; anca.maxim@ac.tuiasi.ro; caruntuc@ac.tuiasi.ro

² Holistic Engineering and Technologies (he[alt]), Continental Automotive Romania, Iasi 700671, Romania; constantin.caruntu@continental-corporation.com

Corresponding author: Constantin-Florin Caruntu (e-mail: caruntuc@ac.tuiasi.ro).

This work was supported by a publications grant of the TUIASI, project number GI/P7/2021.

ABSTRACT The situation in which a vehicle has to avoid a collision with an obstacle can be difficult to realise in optimum conditions when the roads are crowded. This paper uses the advantages of vehicle grouping and vehicle-to-vehicle (V2V) communication, and proposes a control architecture, which ensures a safe merging between the vehicles from two platoons. The architecture is formed by three layers, with the following tasks: *i*) to analyse the environment and to decide the best action for a certain vehicle, *ii*) to plan the new trajectory, and *iii*) to follow it at an imposed velocity or distance to the vehicle in front. The vehicles are equipped with a trajectory planner designed using two methods: the first one is based on a polynomial equation, and the second one is based on the model predictive control (MPC) algorithm. Each vehicle is also equipped with a trajectory follower, which has a cooperative adaptive cruise control (CACC) functionality based on a distributed model predictive control (DMPC) formulation. Also, the paper proposes a solution to compensate the data-packet-dropouts that are induced by the wireless communication network used to exchange information between vehicles. Moreover, to accommodate various realistic scenarios in the same control framework, the cost function for the DMPC algorithm was designed to take into account different communication topologies. The proposed architecture was tested in a simulation scenario, in which two platoons have to merge in order to avoid a fixed obstacle and the results show its efficiency.

INDEX TERMS cooperative adaptive cruise control, cooperative platoon merging, data-packet-dropouts, path planning, model predictive control, trajectory follower

I. INTRODUCTION

The current studies in the field of autonomous vehicles have the following main directions: *i*) to ensure people's safety by avoiding collisions, *ii*) to reduce costs and pollution by decreasing fuel consumption, *iii*) to improve traffic condition by avoiding congestions, and *iv*) to optimize the space occupied by vehicles by maintaining a small safety distance between them [1]. To accomplish these targets, the researchers proposed various solutions for the following functionalities: lane keeping [2], collision avoidance [3], trajectory planning and tracking [4]–[6], cruise control [7], vehicle platooning and vehicle grouping [8]. The last two functionalities are based on control algorithms that use the adaptive cruise control (ACC) function [9]. In the last years, the idea of connecting the vehicles with the infrastructure using networked communications has determined the development of new strategies based on

the advantages of communications. Using vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication the vehicles can make use of more information about their neighbours and environment. This information is afterward used to improved either: *i*) the performance in traffic (i.e., vehicles can form groups inside of which they negotiate to solve cases like deadlock or traffic congestion [10]) or *ii*) the performance of the existing solution (e.g., the ACC algorithm is turned into a cooperative adaptive cruise control (CACC) algorithm, in which a follower vehicle benefits by receiving the information regarding the velocity of the vehicle in front [11]–[14]).

In the literature, there are many studies that propose solutions for cooperative autonomous vehicles, aiming to solve some realistic situations met in traffic, such as, deadlock position, collision avoidance, platoon merging/splitting, etc.

These solutions are based on different types of models for vehicles and various control strategies developed accordingly. In [15]–[17], the proposed solutions use predictive control algorithms for the lateral and longitudinal dynamics of the vehicles, the strategies being based on the nonlinear bicycle model. To design the CACC strategy, the vehicles are modelled using the point model (neglecting the dimensions of the vehicle) and the control strategies that use this kind of modelling are represented by: classical PID control [18], model based predictive control (MPC) [19]–[21], distributed model based predictive control (DMPC) [13], [22]–[24], control law representing a linear combination of velocity, acceleration and distance errors [25]–[28], and solutions based on the linear quadratic regulator (LQR) [29]. Also, there are solutions for trajectory planning, which use the kinematic/point model to predict the trajectory of vehicles [30]–[35]. Other solutions for path planning are based on computing the trajectory using polynomial equations [24], [26], [36], [37]. Applications of merging vehicles also receive a great interest nowadays. These studies mainly use the point model for the vehicles and the control solutions are based on LQR or MPC algorithms [26], [33], [38]–[40]. The merging manoeuvre often requires a combination between the CACC, trajectory planner and trajectory follower to ensure the required space for vehicles and a safe merging [41]–[45]. The exchange of information between vehicles from a platoon is performed under some specific communication topologies. The vehicles receive and send data only from and to certain vehicles. The most used methodologies which consider such communication topologies when designing the control solutions are based on: *a)* including the vehicle interconnection in the vehicle dynamics model [46], *b)* designing a control law which is defined as a linear combination between the error of the vehicle outputs (distance/velocity) [47], and *c)* considering these topologies in the cost function which has to be minimised to obtain the control command [48]. Note that, the last method is considered in this paper to test the efficiency of different communication topologies.

This paper proposes a control architecture, suitable for vehicles driving in platoons formed by three levels as follows:

- Level I obtains information from sensors and through V2V communication regarding the environment, such as, nearby obstacles, neighbouring vehicles, etc., decides the new actions for the vehicle, e.g., maintain the lane, change the lane, form triangle shapes with the vehicles from neighbouring lanes, change the velocity, etc. Moreover, this level sends information regarding the driving environment, e.g., positions and velocities of neighbour vehicles, positions of the obstacles, and control commands to the next two levels;
- Level II is represented by the trajectory planner, which uses the information and commands received from Level I to plan the lateral trajectory for the vehicle and transmits it to Level III. This paper proposes two methods for designing the trajectory planner: *i)* a polynomial

trajectory planner that requires the initial and target state of the vehicle (lateral position, lateral velocity and lateral acceleration) and computes the parameters of a 5th order polynomial trajectory; *ii)* a MPC algorithm to determine the trajectory of the vehicle. The MPC algorithm uses a simple model for the dynamics of the vehicle to estimate its position and to minimise a cost function in order to lead the vehicle from an initial position to a given target position;

- Level III is composed of a CACC algorithm and a trajectory follower: the CACC is implemented using a DMPC algorithm and its task is to ensure that the vehicles from a platoon are travelling at the imposed velocity and maintain a safe distance between them. Moreover, when Level I commands it, the CACC is used to form triangle shapes between vehicles from two platoons travelling on neighbouring lanes. Moreover, for the CACC algorithm, different types of communication topologies were tested and their performances were evaluated. The trajectory follower is designed using a MPC strategy that computes the steering angle to guide the vehicle to follow the received trajectory from Level II.

The novelty of the proposed architecture with respect to ours previous works [12], [24], [31], [45] and similar works from the literature [33], [40]–[42], [44] is given by the following characteristics:

- the CACC functionality is designed for disturbed communication channels, i.e., subject to frequent data-packets-dropouts;
- different types of communication topologies for the CACC algorithm are considered, and their influence on the platoon performance was evaluated;
- the paper compares two different methods to design the trajectory planner:
 - the first solution for the trajectory planner is based on 5th order polynomials, ensures a smooth path and requires reduced computational power;
 - the second method for the trajectory planner is based on the MPC algorithm, requires more computational power, but it needs less information regarding the target position compared to the first method;
- the trajectory follower is designed using a linear bicycle model, but the vehicles are simulated using a nonlinear bicycle model, which is closer to the dynamical behaviour of a real vehicle;
- the solution combines both lateral and longitudinal dynamics and avoids a collision with a static obstacle by executing a manoeuvre to merge two platoons at highway velocity.

The proposed control architecture was tested in a simulation scenario, in which two platoons have to merge to avoid a collision with a static obstacle. The platoons successfully

executed a safe merging manoeuvre and overpassed the obstacle.

This paper is structured as follows: Section II presents the mathematical models used to describe the longitudinal and lateral dynamics of the vehicles. In Section III, the predictive control algorithms are described, while Section IV gives the solutions for trajectory planning. The proposed control architecture is presented in Section V. The collision avoidance manoeuvre, illustrative results and discussions are presented in Section VI, followed by the conclusion in Section VII.

Notations \mathbb{Z} and \mathbb{Z}_+ are the integer and non-negative integer numbers, \mathbb{R} and \mathbb{R}_+ are the real and non-negative real numbers. We use the notation $\mathbb{Z}_{[c_1, c_2]}$ to denote the set $\{k \in \mathbb{Z}_+ \mid c_1 \leq k \leq c_2\}$, for some $c_1, c_2 \in \mathbb{Z}_+$. $X \succeq 0$ denotes that matrix $X \in \mathbb{R}^{n \times m}$ is semipositive definite and $X \succ 0$ denotes that matrix X is positive definite, $n, m \in \mathbb{Z}_+$. Matrix $I_n \in \mathbb{Z}^{n \times n}$ represents the identity matrix of size $n \in \mathbb{Z}_+$. The symbol $\|X_i\|_\infty$ denotes the infinity norm and represents the maximum of $X_i \in \mathbb{R}^n$.

II. VEHICLE DYNAMICS MODELLING

This section presents the mathematical models which describe the longitudinal and lateral dynamics of the vehicles. The models are used in the design phase of the controller and to simulate the vehicles.

A. LONGITUDINAL DYNAMICS MODEL

The following longitudinal model describes how the velocity of the vehicle varies under the action of the input, longitudinal traction force, and disturbance forces [49]:

$$m\ddot{x}(t) = F_x(t) - mg \sin \alpha - fmg \cos \alpha - 0.5\rho SC_d(\dot{x}(t) - v_w)^2, \quad (1)$$

where t represents the time, \dot{x} represents the longitudinal velocity of the vehicle, F_x is the input, i.e., longitudinal traction force, m represents the mass of the vehicle, g represents the gravitational acceleration, α represents the road slope, ρ represents the air density, S represents the vehicle frontal area, C_d represents the drag coefficient, f represents the rolling resistance, $mg \sin \alpha$ represents the longitudinal component of the gravitational force, $fmg \cos \alpha$ represents the rolling force, $0.5\rho SC_d(\dot{x} - v_w)^2$ represents the air friction force and v_w represents the velocity of the air.

The linearisation of (1) is performed considering a nominal operating point (\dot{x}_0, α_0) , obtaining:

$$\ddot{x}(t) = -\frac{1}{T_x}\dot{x}(t) + \frac{K_x}{T_x}(F_x(t) + F_d), \quad (2)$$

with:

$$\begin{cases} T_x = m/(\rho S C_d(\dot{x}_0 - v_w)) \\ K_x = 1/(\rho S C_d(\dot{x}_0 - v_w)) \\ F_d = mgf \sin \alpha_0 - mg \cos \alpha_0. \end{cases} \quad (3)$$

Model (2) is extended with an integrator state which has as input the error between the imposed longitudinal velocity and the actual velocity of the vehicle:

$$\dot{\chi}_{\dot{x}} = \dot{x}_{ref}(t) - \dot{x}(t), \quad (4)$$

where $\chi_{\dot{x}}$ represents the integrator state and \dot{x}_{ref} is the imposed reference velocity for the vehicle. The integrator state has the advantages that it ensures a zero steady-state error.

Equation (2) mathematically models the velocity of the leader vehicle from the platoon, the following model being used to describe the evolution of the distance between two consecutive vehicles, V_i and V_{i-1} :

$$\begin{bmatrix} \dot{d}_{V_i, V_{i-1}}(t) \\ \ddot{d}_{V_i, V_{i-1}}(t) \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -\frac{1}{T_x} \end{bmatrix} \begin{bmatrix} d_{V_i, V_{i-1}}(t) \\ \dot{d}_{V_i, V_{i-1}}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_x}{T_x} \end{bmatrix} F_{xV_i}(t) + \begin{bmatrix} 0 \\ \frac{K_x}{T_x} \end{bmatrix} F_d + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \dot{x}_{V_{i-1}}(t), \quad (5)$$

where $d_{V_i, V_{i-1}}$ represents the distance between vehicle V_i and the vehicle in front of it, i.e., V_{i-1} , \dot{d}_{V_i} represents the longitudinal velocity of vehicle i , F_{xV_i} represents the input, i.e., the longitudinal traction force, and $\dot{x}_{V_{i-1}}$ represents the velocity of vehicle V_{i-1} .

Considering a follower vehicle, the aim is to maintain a desired distance to the vehicle in front, and for this reason, model (5) is extended with an integrator state, χ_d , which has as input the error between the desired distance, d_{ref} , and the distance between vehicles V_i and V_{i-1} :

$$\begin{cases} \dot{\chi}_d = d_{ref}(t) - d_{V_i, V_{i-1}}(t) \\ d_{ref}(t) = d_0 + \tau \dot{x}_{V_i}, \end{cases} \quad (6)$$

where d_0 represents the parking distance and τ represents the headway time.

B. LATERAL DYNAMICS MODEL

The lateral dynamics model describes how the lateral position and yaw rate of a vehicle vary when the steering angle of the front tire changes.

1) Nonlinear lateral dynamics model

In this subsection, the nonlinear bicycle model is described. The model was obtained considering that the rear and front axles are joined and the vehicle is steered only by the front tire [50]:

$$\begin{cases} m\ddot{y}(t) = -m\dot{x}(t)\dot{\beta}(t) + 2F_{yf}(t) + 2F_{yr}(t) \\ I\ddot{\beta}(t) = 2\ell_f F_{yf}(t) - 2\ell_r F_{yr}(t), \end{cases} \quad (7)$$

where y represents the lateral position, β represents the yaw angle, $\dot{\beta}$ represents the yaw rate, F_{yf} and F_{yr} represent the lateral front and rear forces, ℓ_f represents the distance between the centre of gravity of the vehicle to the front axle, ℓ_r represents the distance between the centre of gravity of the vehicle to the rear axle and I represents the vehicle's rotational inertia. The relation between the lateral forces and steering angles are given by [50]:

$$\begin{cases} F_{yf}(t) = C_f(\delta_f(t) - \gamma_f(t)) \\ F_{yr}(t) = C_r(\delta_r v - \gamma_r(t)), \end{cases} \quad (8)$$

$$\begin{cases} \gamma_f(t) = \arctan(v_{Lf}(t)/v_{lf}(t)) \\ \gamma_r(t) = \arctan(v_{Lr}(t)/v_{lr}(t)), \end{cases} \quad (9)$$

where δ_f and δ_r represent the steering angles of the front and rear tire, respectively, C_f and C_r represent the cornering stiffness coefficients, γ_f and γ_r represent the velocity angles of the tires, and v_{Lf} , v_{Lr} , v_{lf} , v_{lr} represent the lateral and longitudinal velocities of the front and rear tires defined as:

$$\begin{cases} v_{Lf}(t) = (\dot{y}(t) + \ell_f \dot{\beta}(t)) \cos \delta_f(t) - \dot{x}(t) \sin \delta_f(t) \\ v_{Lr}(t) = (\dot{y}(t) - \ell_r \dot{\beta}(t)) \cos \delta_r(t) - \dot{x}(t) \sin \delta_r(t) \\ v_{lf}(t) = \dot{x}(t) \cos \delta_f(t) + (\dot{y}(t) + \ell_f \dot{\beta}(t)) \sin \delta_f(t) \\ v_{lr}(t) = \dot{x}(t) \cos \delta_r(t) + (\dot{y}(t) - \ell_r \dot{\beta}(t)) \sin \delta_r(t). \end{cases} \quad (10)$$

2) Linear lateral dynamics model

To obtain a linear lateral bicycle model, the following assumptions were considered [50]:

- the steering angle of the rear tire, δ_r , is equal to zero, the vehicle being steered by the variation of the front steering angle, δ_f ;
- the steering angle of the front tire, δ_f , has small values ($|\delta_f| \cong 0.1745$ [rad]);
- the vehicle is moving with a constant longitudinal velocity;
- the longitudinal velocity is much higher than the lateral velocity.

Under these assumptions, the linear bicycle model becomes:

$$\dot{\chi}_L(t) = A_L \chi_L(t) + B_L \delta_f(t), \quad (11)$$

where $\chi_L = [y, \dot{y}, \beta, \dot{\beta}]^T$, $B_L = [0, 2\frac{C_f}{m}, 0, 2\frac{\ell_f C_f}{I}]^T$,

$$A_L = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -2\frac{C_f + C_r}{m\dot{x}} & 0 & -\dot{x} - 2\frac{\ell_f C_f - \ell_r C_r}{m\dot{x}} \\ 0 & 0 & 0 & 1 \\ 0 & -2\frac{\ell_f C_f + \ell_r C_r}{I\dot{x}} & 0 & -2\frac{\ell_f^2 C_f + \ell_r^2 C_r}{I\dot{x}} \end{bmatrix}.$$

Model (11) is extended with an integrator state, χ_y , which allows us to impose a nonzero reference for the lateral trajectory of the vehicle. It has as input the error between the imposed lateral trajectory, y_{ref} , and the lateral trajectory of the vehicle:

$$\dot{\chi}_y = y_{ref}(t) - y(t). \quad (12)$$

III. PREDICTIVE CONTROL METHODOLOGY

The MPC strategy has the advantage of its simplicity and can be used to control a variety of processes, no matter how complex, with many inputs and outputs. The method can take into account the limitations of the actuators and allows operation closer to the imposed constraints. The MPC algorithm can compensate the measurable disturbances based on a feed-forward component [51].

A. MODEL BASED PREDICTIVE CONTROL STRATEGY

At each sample time, the MPC algorithm performs the following steps:

- Obtains information regarding the process state and/or output from sensors/estimator;
- Computes a finite optimal control sequence:
 - Minimising a cost function while satisfying all imposed constraints;
 - Predicting the state and output of the system based on its model;
- Implements the first component of the optimal control sequence according to the receding horizon principle [51].

The model used to predict the behaviour of the system can be described as a state-space model:

$$\begin{cases} \chi(k+1) = A_d \chi(k) + B_d u(k) \\ z(k) = C_d \chi(k), \end{cases} \quad (13)$$

where k is the discrete time, $\chi \in \mathbb{R}^n$ is the state of the system, $u \in \mathbb{R}^m$ is the input of the system, $z \in \mathbb{R}^p$ is the output of the system, $A_d \in \mathbb{R}^{n \times n}$ is the state matrix, $B_d \in \mathbb{R}^{m \times n}$ is the input matrix, $C_d \in \mathbb{R}^{p \times n}$ is the output matrix, $n \in \mathbb{Z}_+$ is the number of the system states, $m \in \mathbb{Z}_+$ is the number of the system inputs, $p \in \mathbb{Z}_+$ is the number of the system outputs, the pair (A_d, B_d) is controllable and the pair (C_d, A_d) is detectable. **Problem 1:** Given an initial state $\chi(0)$, at each sample time $k \in \mathbb{Z}_+$, compute a finite horizon optimal input sequence $U^{MPC*}(k) = [u^{MPC*}(k|k), \dots, u^{MPC*}(k + N_{MPC} - 1|k)]^T$ that minimizes the cost function:

$$V_{MPC}(\chi(k), U^{MPC}(k)) = \chi_{N_{MPC}}^T P^{MPC} \chi_{N_{MPC}} + \sum_{i=0}^{N_{MPC}-1} (\chi_i^T Q^{MPC} \chi_i + u_i^{MPC T} R^{MPC} u_i^{MPC}) \quad (14)$$

over U^{MPC} , subject to the constraints:

$$\begin{cases} u_{low} \leq u_i^{MPC} \leq u_{high} \\ z_{low} \leq z_i \leq z_{high}, \end{cases} \quad (15)$$

where N_{MPC} is the prediction horizon and P^{MPC} , Q^{MPC} , R^{MPC} are the weight matrices for the terminal state, the intermediary states and inputs of the system, $\chi_i = \chi(k+i|k)$ is the prediction of χ at time i within the prediction horizon window, computed at discrete time $k \in \mathbb{Z}_+$ and u_i^{MPC} represents the input trajectory, $u^{MPC}(k+i)$.

The prediction equation for the state of the system is determined using (13), and yields as:

$$\zeta^{MPC} = \Phi^{MPC} \chi(k) + \Gamma^{MPC} U^{MPC}, \quad (16)$$

and the matricial form of the cost function can be described as:

$$V_{MPC}(\chi(k), U^{MPC}(k)) = \chi(k)^T Q^{MPC} \chi(k) + \zeta^{MPC T}(k) \Omega^{MPC} \zeta^{MPC}(k) + U^{MPC T} \Psi^{MPC} U^{MPC} \quad (17)$$

where:

$$\bullet \quad \zeta^{MPC} = [\chi_1, \dots, \chi_{N_{MPC}}]^T;$$

- $\Omega^{MPC} = \text{diag}\{Q^{MPC}, \dots, Q^{MPC}, P^{MPC}\};$
- $\Phi^{MPC} = [A_d, A_d^2, \dots, A_d^{N_{MPC}}]^T;$
- $\Psi^{MPC} = \text{diag}\{R^{MPC}, \dots, R^{MPC}\};$
- $\Gamma^{MPC} = \begin{bmatrix} B_d & \dots & 0 \\ A_d B_d & \dots & 0 \\ \dots & \dots & \dots \\ A_d^{N_{MPC}-1} B_d & \dots & B \end{bmatrix};$
- $\Omega^{MPC} \in \mathbb{R}^{nN_{MPC} \times nN_{MPC}};$
- $\Phi^{MPC} \in \mathbb{R}^{mN_{MPC} \times mN_{MPC}};$

The next step is to describe in a matricial form the imposed constraints (15):

$$\Delta^{MPC} \chi(k) + \Upsilon \zeta^{MPC} + E^{MPC} U^{MPC} \leq c^{MPC}, \quad (18)$$

where:

- $\Delta^{MPC} = [M_0^{MPC}, \dots, 0]^T;$
- $\Upsilon = \begin{bmatrix} 0 & \dots & 0 \\ M_1 & \dots & 0 \\ \dots & \dots & \dots \\ 0 & \dots & M_{N_{MPC}} \end{bmatrix};$
- $c^{MPC} = [c_0^{MPC}, \dots, c_{N_{MPC}}^{MPC}]^T;$
- $M_i^{MPC} = [0, 0, -C_d, -C_d]^T;$
- $M_{N_{MPC}}^{MPC} = [-C_d, C_d]^T;$
- $c_i^{MPC} = [-u_{low}, u_{high}, -z_{low}, z_{high}]^T;$
- $E^{MPC} = \begin{bmatrix} E_0^{MPC} & \dots & 0 \\ \dots & \dots & E_{N_{MPC}-1}^{MPC} \\ 0 & \dots & E_{N_{MPC}}^{MPC} \\ 0 & \dots & 0 \end{bmatrix};$
- $E_i^{MPC} = [I_m, -I_m, 0, 0]^T;$
- $M_i^{MPC} \in \mathbb{R}^{(m+p) \times p}, E_i^{MPC} \in \mathbb{R}^{(m+p) \times 1};$
- $i = \overline{0, N_{MPC} - 1}.$

Combing the relations (16)-(18), Problem 1 becomes:

$$\left\{ \begin{array}{l} \min_{U^{MPC}} \frac{1}{2} U^{MPC T} G^{MPC} U^{MPC} + U^{MPC T} F^{MPC} \chi_k \\ \text{subject to:} \\ J^{MPC} U^{MPC} \leq c^{MPC} + W^{MPC} \chi_k, \end{array} \right. \quad (19)$$

where:

$$\left\{ \begin{array}{l} G^{MPC} = 2(\Psi^{MPC} + \Gamma^{MPC T} \Omega^{MPC} \Gamma^{MPC}) \succ 0 \\ F^{MPC} = 2\Gamma^{MPC T} \Omega^{MPC} \Phi^{MPC} \\ J^{MPC} = \Upsilon^{MPC} \Gamma^{MPC} + E^{MPC} \\ W^{MPC} = -\Delta^{MPC} - \Upsilon^{MPC} \Phi^{MPC} \\ \Omega^{MPC} \succeq 0 \\ \Psi^{MPC} \succ 0. \end{array} \right. \quad (20)$$

B. MODELLING FOR DMPC

The DMPC strategy has the advantage that it can be used to control complex systems such as: vehicle groups, chemical plants or wind farms. The algorithm assumes that a complex system can be decomposed in several coupled subsystems, and for each one of them a predictive controller is designed [52]. Each subsystem is described using a simple model and requires a less sophisticated local controller, as opposed to a centralised control solution, designed from a global perspective. The following equation represents the mathematical model for subsystem's j dynamics, including its links with M_j neighbouring subsystems [53]:

$$\chi_j(k+1) = A_{d_{j,j}} \chi_j(k) + B_{d_j} u_j(k) + \sum_{i=1}^{M_j} (A_{d_{i,j}} \chi_i(k) + B_{d_{i,j}} u_i(k)). \quad (21)$$

where $\chi_j \in \mathbb{R}^{n_j}$ represents the state of subsystem j , $u_j \in \mathbb{R}^{m_j}$ represents the input, $u_i \in \mathbb{R}^{m_i}$ represents the input of subsystem i , $\chi_i \in \mathbb{R}^{n_i}$ represents the state of subsystem i , $A_{d_{j,j}} \in \mathbb{R}^{n_j \times n_j}$ represents the state matrix of subsystem j , $A_{d_{i,j}} \in \mathbb{R}^{n_j \times n_i}$, $B_{d_{i,j}} \in \mathbb{R}^{n_j \times m_i}$ represent the coupling matrices, $B_{d_j} \in \mathbb{R}^{n_j \times m_j}$ is the input matrix, $n_i, n_j, m_i, m_j \in \mathbb{Z}_+$.

The dynamics of subsystem (21) depends on both the states and the inputs of its neighbours.

A particular case for this system is represented by a model in which the subsystems are interconnected only by inputs [54]:

$$\chi_j(k+1) = A_{d_{j,j}} \chi_j(k) + B_{d_j} u_j(k) + \sum_{i=1}^{M_j} B_{d_{i,j}} u_i(k). \quad (22)$$

Another particular case of system (21) is represented by a model, in which the subsystems are coupled only through states:

$$\chi_j(k+1) = A_{d_{j,j}} \chi_j(k) + B_{d_j} u_j(k) + \sum_{i=1}^{M_j} A_{d_{i,j}} \chi_i(k). \quad (23)$$

The structure of a vehicle platoon can be described using (23), in which each subsystem is influenced only by its predecessor neighbour [13]:

$$\left\{ \begin{array}{l} \chi_j(k+1) = A_{d_{j,j}} \chi_j(k) + B_{d_j} u_j(k) + A_{d_{j-1,j}} \chi_{j-1}(k-1) \\ z_j(k) = C_{d_j} \chi_j(k), \end{array} \right. \quad (24)$$

where $z_j \in \mathbb{Z}_+^{p_j}$ represents the output and $C_{d_j} \in \mathbb{R}^{p_j \times n_j}$ represents the output matrix, $p_j \in \mathbb{Z}_+$. Note that matrix $A_{d_{0,1}}$ is equal to a null matrix because the subsystems communicate unidirectionally and subsystem 0 does not have a preceding subsystem.

C. COST FUNCTION DEFINITION FOR DMPC

To improve the performances and stability of the vehicle platoons, in literature there are various types of communication topologies. According to the work in [55], [56], the most used communication topologies are: *a)* predecessor following topology ($T1$), which assumes that each vehicle receives information from the vehicle in front, *b)* leader-predecessor following topology ($T2$), which assumes that each follower vehicle uses information from its relative leader and from the leader of the platoon, *c)* bidirectional topology ($T3$), which assumes that each vehicle computes its control command taking into account information from its neighbouring front and rear vehicles and *d)* leader-bidirectional topology ($T4$), which is a combination between $T2$ and $T3$. The influence of the communication topologies on the platoon performances can be modelled through a cost function [48], [55], which includes the interconnections between the vehicles in the platoon.

Problem 2: Given an initial state $\chi_j(0)$, at each sample time $k \in \mathbb{Z}_+$, compute a finite horizon optimal input sequence $U_j^{DMPC*}(k) = [u_j^{DMPC*}(k|k), \dots, u_j^{DMPC*}(k + N_{DMPC} - 1|k)]^T$ that minimizes the cost function:

$$V_{DMPC}(\chi_j(k), U^{DMPC}(k)) = V_{DMPC}^U + V_{DMPC}^Z \quad (25)$$

with:

$$V_{DMPC}^U = \sum_{i=0}^{N_{DMPC}-1} u_j^{DMPC T}(i) R_j^{DMPC} u_j^{DMPC}(i) \quad (26)$$

$$V_{DMPC}^Z = \sum_{i=1}^{N_{DMPC}} [(r_{ref_j}(i) - z_j(i))^2 + \sum_{q \in M_j} \lambda_{j,q} (\chi_q(i) - \chi_j(i))^2] \quad (27)$$

over U_j^{DMPC} , subject to the constraints:

$$U_j^{low} \leq U_j^{DMPC}(k) \leq U_j^{high}, \quad (28)$$

where $u_j(i) = u_j(k + i|k)$, $z_j(i) = z_j(k + i|k)$ are the prediction of the future inputs, respectively outputs of subsystem j at step $k \in \mathbb{Z}_+$, $r_{ref_j}(i) = r_{ref_j}(k + i|k)$ is the predicted target trajectory at step $k \in \mathbb{Z}_+$, R_j^{DMPC} is the input weight matrix, N_{DMPC} is the prediction horizon, $\chi_j(i) = \chi_j(k + i|k)$ and $\chi_q(i) = \chi_q(k + i|k)$ are the prediction of the states of subsystems j and q , λ_{jq} is set 1 if the vehicle j receives information from vehicle q and zero otherwise, M_j represents the set of neighbours of agent j .

Fig. 1 [57] illustrates the four types of communication topologies considered in this paper. To obtain the cost function for each case, the coupling term V_{DMPC}^Z from (25) is particularised as follows:

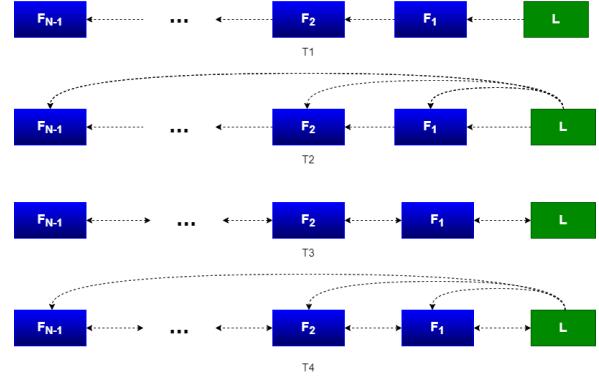


FIGURE 1: Platoon communication topologies

- Topology $T1$ where, each vehicle receives information from the vehicle in front, the cost function is obtained considering all $\lambda_{j,q}$ set to zero:

$$V_{DMPC}^Z = \sum_{i=1}^{N_{DMPC}} [(r_{ref_j}(i) - z_j(i))^2] \quad (29)$$

It has to be mentioned that in this case, the information sent by vehicle $j - 1$ to vehicle j is not considered in the cost function, but in the model, as in (24).

- Topology $T2$ supposes that each agent j will receive information from its relative leader and the leader of the platoon, thus $\lambda_{j,j-1} = \lambda_{j,1} = 1$, $\lambda_{j,q} = 0$, $q \in M_j - \{j-1, 1\}\}$:

$$V_{DMPC}^Z = \sum_{i=1}^{N_{DMPC}} [(r_{ref_j}(i) - z_j(i))^2 + (\chi_{j-1}(i) - \chi_j(i))^2 + (\chi_1(i) - \chi_j(i))^2] \quad (30)$$

- Topology $T3$ assumes that $\lambda_{j,j-1} = \lambda_{j,j+1}$ are set to 1 and the rest of lambda coefficients are set to zero:

$$V_{DMPC}^Z = \sum_{i=1}^{N_{DMPC}} [(r_{ref_j}(i) - z_j(i))^2 + (\chi_{j-1}(i) - \chi_j(i))^2 + (\chi_{j+1}(i) - \chi_j(i))^2] \quad (31)$$

- Topology $T4$ is a combination between $T2$ and $T3$, the cost function being obtained considering $\lambda_{j,j-1} = \lambda_{j,j+1} = \lambda_{j,1} = 1$ and rest of lambda coefficients are set to zero:

$$V_{DMPC}^Z = \sum_{i=1}^{N_{DMPC}} [(r_{ref_j}(i) - z_j(i))^2 + (\chi_{j-1}(i) - \chi_j(i))^2 + (\chi_{j+1}(i) - \chi_j(i))^2 + (\chi_1(i) - \chi_j(i))^2] \quad (32)$$

For all the above mentioned communication topologies, the state predictor for subsystem j is given by:

$$\left\{ \begin{array}{l} \zeta_j^{DMPC} = \Phi_{j,j}^{DMPC} \chi_j(k) + \Gamma_j^{DMPC} U_j^{DMPC} + \\ \quad + \Phi_{j-1,j}^{DMPC} \zeta_{j-1}^{DMPC}(k-1) \\ Z_j(k) = C_j^{DMPC} \zeta_j^{DMPC} \end{array} \right. \quad (33)$$

where $\Phi_{j,j}^{DMPC}$ and Γ_j^{DMPC} are computed in the same manner as Φ^{MPC} and Γ_j^{MPC} , ζ_{j-1}^{DMPC} is received from subsystem $j-1$ and represents the prediction of its state at step $k-1$ and $\Phi_{j-1,j}^{DMPC} = [A_{d_{j-1},j}, A_{d_{j-1},j}^2, \dots, A_{d_{j-1},j}^{N_{DMPC}}]$.

D. DATA-PACKET DROPOUTS COMPENSATION

Subsystem j receives and sends information via a wireless communication network in the presence of communication disturbances, e.g., data-packet-dropouts. So, there exists the possibility that information ζ_q^{DMPC} will not be received by agent j resulting in a misuse within the predictor (33) formulation. In this case, the lost information has to be replaced. This procedure will be accomplished using the last received information. Let us consider that the last received information is $\zeta_{q,j}^{DMPC}(k) = [\chi_q(k+1|k), \dots, \chi_q(k+N_{DMPC}|k)]^T$ sent by subsystem q to subsystem j at step $k+1$. Let us assume that there are $s \leq N_{DMPC}$ consecutive data-packet-dropouts, then the message $\zeta_{q,j}^{DMPC}(k+s)$ will be obtained using [12]:

$$\zeta_{q,j}^{DMPC}(k+s) = [\underbrace{\chi_q(k+s|k), \chi_q(k+s+1|k), \dots,}_{(s+1)\text{times}} \chi_q(k+N_{DMPC}|k), \dots, \chi_q(k+N_{DMPC}|k)]^T. \quad (34)$$

Remark 1: Note that the DMPC strategy offers the possibility to approximate the lost information but, as a disadvantage, the approximation is based on the prediction of subsystem state at least a sample time in the past, which leads to increasing estimation errors due to the differences between the models and the real subsystems.

Remark 2: Another problem which may occur is that the number of consecutive lost data-packets may be greater than the prediction horizon N_{DMPC} . In this case, a possible solution will be a platoon split due to the high number of consecutive lost data, which may occur due to problems in the communication network and may lead to decreased performances.

IV. LATERAL TRAJECTORY PLANNING

In this section, two methods for designing the trajectory planner for the lateral movement of a vehicle are illustrated. The first trajectory planner determines the trajectory of the vehicle using the equation of a 5th order polynomial; the second method is based on the MPC algorithm to estimate the lateral position of a vehicle and minimizes a cost function which depends on the lateral error.

A. 5TH ORDER POLYNOMIAL LATERAL TRAJECTORY PLANNER

The polynomial trajectory planning uses a 5th order polynomial to compute the trajectory of the vehicle. The method requires the initial and the target states of the vehicle and uses them to compute the parameters of the polynomial. In this work, the initial and target states are represented by the lateral position, velocity and acceleration of the vehicle in the initial and target position, respectively. Note that, the advantages of using a 5th order polynomial will be explained after its mathematical description.

Problem 3: Given the initial state $P_{y0} = [y_0, \dot{y}_0, \ddot{y}_0]^T$, $P_{x0} = x_0$ and the target state $P_{yf} = [y_f, \dot{y}_f, \ddot{y}_f]^T$, $P_{xf} = x_f$ of the vehicle, determine the parameters $[a_5, a_4, a_3, a_2, a_1, a_0]$ of a 5th order polynomial so that the following conditions are accomplished:

$$y_{ref}(x) = \sum_{i=0}^5 a_i x^i \quad (35a)$$

$$y_{ref}(x_0) = y_0 \quad (35b)$$

$$\dot{y}_{ref}(x_0) = \dot{y}_0 \quad (35c)$$

$$\ddot{y}_{ref}(x_0) = \ddot{y}_0 \quad (35d)$$

$$y_{ref}(x_f) = y_f \quad (35e)$$

$$\dot{y}_{ref}(x_f) = \dot{y}_f \quad (35f)$$

$$\ddot{y}_{ref}(x_f) = \ddot{y}_f, \quad (35g)$$

where y_{ref} represents the reference, for the lateral trajectory planner, and x represents the longitudinal position of the vehicle.

The conditions (35) represent the values of the lateral position, velocity and acceleration of the vehicle in the initial position and those imposed for the vehicle in the target position. Applying these conditions, the analytical solution of Problem 3 can be determined as:

$$\left\{ \begin{array}{l} a_5 = [12(y_f - y_0) + 6(x_f - x_0)(\dot{y}_0 - \dot{y}_f) + \\ \quad + (x_f - x_0)(\ddot{y}_0 - \ddot{y}_f)]/[2(x_0 - x_f)^5] \\ a_4 = -[2(\dot{y}_0 - \dot{y}_f) + (x_f - x_0)(\ddot{y}_0 - \ddot{y}_f) + \\ \quad + 10a_5(x_0^4 - x_f^4 + 2x_0x_f^3 - 2x_0^3x_f)]/[4(x_0 - x_f)^3] \\ a_3 = [20a_5(x_f^3 - x_0^3) + 12a_4(x_f^2 - x_0^2) + \\ \quad + (\ddot{y}_0 - \ddot{y}_f)]/[6(x_0 - x_f)] \\ a_2 = -10a_5x_f^3 - 6a_4x_f^2 - 3a_3x_f^2 + \ddot{y}_f \\ a_1 = -5a_5x_f^4 - 4a_4x_f^3 - 3a_3x_f^2 - 2a_2x_f + \dot{y}_f \\ a_0 = -a_5x_f^5 - a_4x_f^4 - a_3x_f^3 - a_2x_f^2 - a_1x_f + y_f. \end{array} \right. \quad (36)$$

The conditions (35) can be interpreted in two ways. First, the constraints ensure that the trajectory will lead the vehicle from its initial position to the target position via a smooth path, because the conditions (35b)-(35d) ensure that the lateral trajectory and its first and second derivatives (i.e., $y_{ref}(x)$, $\dot{y}_{ref}(x)$, $\ddot{y}_{ref}(x)$) are continuous in the initial point x_0 , without sudden variations. The conditions (35e)-(35g) ensure that the vehicle approaches the target position, x_f ,

smoothly. The second interpretation analyses the solution from the point of view of vehicle lateral dynamics. The conditions imposed in the initial point (35b)-(35d) will ensure that the lateral position, velocity and acceleration of the vehicle do not have sudden variations in point x_0 . Moreover, the conditions imposed in the target point x_f , ensure a continuity between lane changing and lane keeping manoeuvre if the vehicle has to change a lane and after that has to maintain the lane without having a vehicle in front of it. The solution facilitate the task of following a vehicle if the vehicle has to maintain a desired distance from that vehicle after arriving in the target point x_f . In this last case, the vehicle is required to have the values for the lateral position, velocity and acceleration of the vehicle which will be in front of it. Note that, if the vehicle has to maintain the lane ($y_0 = y_f$), then the trajectory will result constant, $y_{ref}(x) = y_f$.

B. MODEL BASED LATERAL TRAJECTORY PLANNER

This method is based on the MPC algorithm and requires a model for the lateral dynamics of the vehicle to predict its behaviour. To ensure a low computational power, a point model is used:

$$\ddot{y}_p(t) = a_{yp}(t), \quad (37)$$

where y_p represents the lateral position and a_{yp} is the lateral acceleration of the lateral point model, and represents the input of the model.

The lateral trajectory of the vehicle is obtained through the minimisation of the lateral error between the imposed lateral position and the estimated lateral position:

Problem 4: Given the initial state $P_{y0} = [y_0, \dot{y}_0, \ddot{y}_0]^T$, and the target state $P_{yf} = [y_f, \dot{y}_f, \ddot{y}_f]^T$, at each sample time, evaluate the lateral trajectory of the vehicle using (37) and minimize the cost function

$$V_p = \sum_{i=1}^{N_p} [\lambda_{yp}(y_f - y_p(k+i))^2 + \lambda_{a_{yp}}(a_{yp}(k+i) - a_{yp}(k+i-1))^2] \quad (38)$$

over $a_{yp}(k+i)$, subject to:

$$y_{low} \leq y(k+i) \leq y_{high}, \quad (39)$$

where N_p is the prediction horizon, λ_{yp} and $\lambda_{a_{yp}}$ represent tuning parameters.

The solution of Problem 4 is represented by a sequence of N_p lateral positions through which the vehicle has to pass. Due to the modelling errors, between the model used to compute the lateral position of the vehicle and the real vehicle, only the first value will be actually used. At the next sample time, the trajectory planner computes another solution for Problem 4.

Consider that the last imposed position generated by the planner is y_{old} and the new imposed position is y_{new} . Moreover, the working time sample of the planner T_{sp} is greater by M times than the sample time of the trajectory follower, so for the next M time samples, the trajectory follower has

to steer the vehicle to pass through the following M^th lateral positions [31]:

$$y_{ref}(k+j) = y_{ref}(k+j-1) + (y_{new} - y_{old})/M, \quad (40)$$

where $y_{ref}(k+j)$ represents the imposed lateral trajectory in the discrete-time interval $[k+1, k+M]$, $j = \overline{1, M}$.

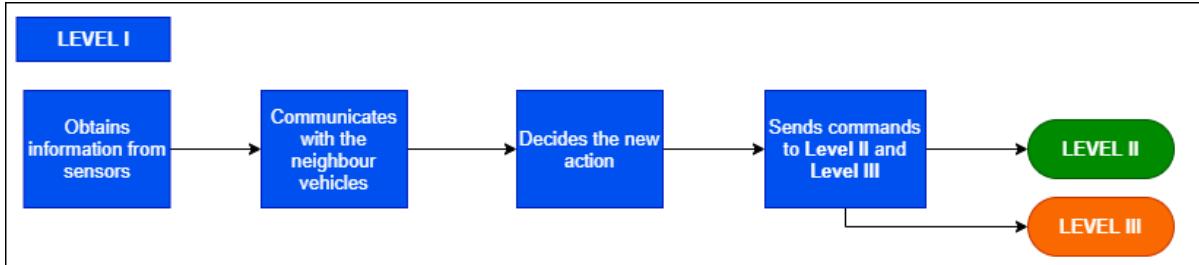
The differences between the two proposed trajectory planners can be pointed out as follows: *i*) the first method, i.e., the one which uses the polynomial trajectory planner (see Section IV-A), requires fewer computational resources and less time, when compared with the second method, MPC-based method (see Section IV-B), which has to compute the position of the vehicle and to solve an optimisation problem; *ii*) regarding the information required by the methods, the first method requires more information, especially in the target position, i.e., lateral position, velocity and acceleration imposed to the vehicle, compared to the second method, which only needs the values of the target lateral position, *iii*) although the first method requires more information regarding the vehicle in its initial and target position, these are used to ensure a smooth and feasible trajectory, without sudden variations compared to the second method, which ensures these properties of the trajectory through an empirical choice of the parameters λ_{yp} and $\lambda_{a_{yp}}$.

V. CONTROL ARCHITECTURE

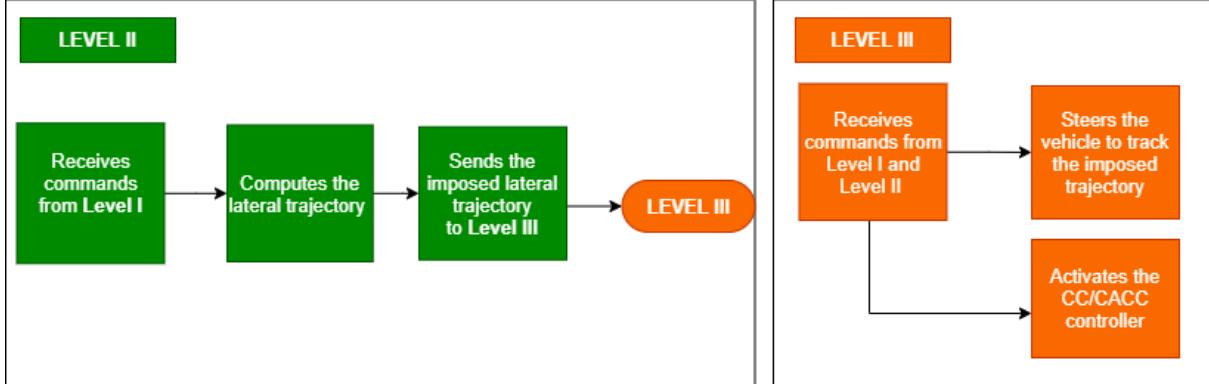
This section describes the proposed control architecture, see Fig. 2, which ensures the longitudinal and the lateral control for a vehicle from a platoon. The architecture is divided into three levels, each of them having several tasks to accomplish. The rules which manage the first level are given by:

Level I

- If the vehicle is the leader then:
 - obtains measurements from the sensors about the vehicle state and environment (information about neighbour vehicles and detected obstacles);
 - communicates with the neighbour vehicle to send and receive data and to negotiate with them;
 - decides the best new action for the vehicle:
 - * switch the platoon leader;
 - * change the lane to avoid an obstacle;
 - * merge with a platoon;
 - * maintain the lane;
 - * reduce/increase the velocity;
 - * form triangles with the neighbour vehicles;
 - sends commands to the follower vehicles;
 - sends commands and required information to Levels II and III:
 - * new imposed travelling velocity;
 - * new target position;



(a) Level I of the control architecture



(b) Levels II and III of the control architecture

FIGURE 2: Vehicle control architecture

- * switch ON/OFF CC/CACC controller.
- If the vehicle is a follower then:
 - obtains measurements from the sensors regarding the vehicle state and environment (information about neighbour vehicles and detected obstacles);
 - communicates with the neighbour vehicles to send and receive data and to negotiate with them;
 - receives commands from the leader vehicle;
 - decides the best new action for the vehicle:
 - * executes the lane changing command received from the leader;
 - * executes the merging manoeuvre command received from the leader;
 - * maintains the lane;
 - * forms triangles with the neighbour vehicles;
 - sends commands and required information to Levels II and III:
 - * new imposed distance between vehicles;
 - * new target position;
 - * switch ON/OFF CC/CACC controller.

Level II is represented by the trajectory planner and has the following tasks:

Level II

- receives the target position from Level I;
- computes the lateral trajectory for the vehicle using one of the proposed strategies;
- sends the obtained lateral trajectory to Level III.

Level III is composed by the trajectory follower and the CC/CACC controller. It controls the velocity and the steering of the vehicle, so that the commands from superior levels are accomplished:

Level III

- receives commands from Level I;
- receives imposed lateral trajectory from Level II;
- steers the vehicle using the MPC algorithm from Section III and uses the lateral dynamics model of the vehicle (11)-(12) to estimate its state;
- controls the velocity/distance of the vehicle/between vehicles using the DMPC algorithm from Section III based on the longitudinal dynamics model (2)-(6).

VI. SIMULATION RESULTS

In this section, the simulation results obtained for the obstacle avoidance manoeuvre using platoons merging are illustrated. Moreover, the analyses of the platoons stability, longitudinal and lateral performances are performed. In the end, a comparison of performances of the CACC algorithm using four

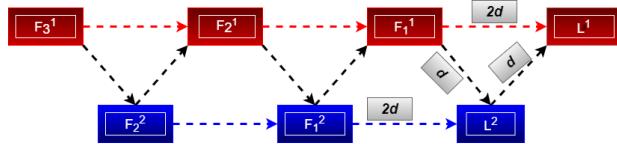


FIGURE 3: Platoons form the triangles

types of communication topologies is given.

A. PLATOONS MERGING MANOEUVRE SIMULATION

The following scenario is proposed. Two platoons are travelling in the same direction on two neighbouring lanes. On the lane of the second platoon, the leader of the second platoon, denoted L^2 detects a fixed obstacle. To avoid it, the leader asks the neighbouring platoon (which is travelling in the same direction) to accept a merging manoeuvre. The leader of the first platoon, denoted L^1 accepts, and the two platoons decrease their velocity to facilitate the execution of the merging manoeuvre. To accomplish a successful merging, the platoons need to ensure enough manoeuvre space, thus the vehicles start to create triangle forms which facilitate a fast merging. Vehicle L^1 will be the leader of the new merged platoon followed by its new follower, L^2 , which starts to maintain a desired distance from its new leader. Vehicle L^2 is followed by its new follower, F_1^1 and so on. This new organisation of the platoon has as result the triangle forms, see Fig.3. When the vehicles are travelling with the imposed velocity and each follower maintains an imposed distance to the vehicle which will be in front of it after the merging manoeuvre, Level I of the leader L^1 commands to Level I of follower vehicles from the old platoon $P2$ to change the lane. After receiving this command, Level I of all vehicles commands to Level II, represented by the trajectory planner, to compute a new path, which leads the vehicle from its initial position to the target position from the platoon. During lane changing, the distance between vehicles is ensured by the CACC functionality (Level III). The trajectory follower receives the new imposed trajectory from Level II and steers the vehicle to follow it. The proposed steps for merging the two platoons, assume the cooperation between longitudinal and lateral control, i.e., the CACC controller ensures enough space between vehicles for a safe merging manoeuvre and the trajectory planners and followers lead the vehicles from their initial lane to the new position in the new platoon. Note that, the results from this subsection are obtained using the first communication topology $T1$, for which the coupling term of the cost function in (25) is given by (29).

The CACC controller is designed using the DMPC algorithm and has the following parameters (see (25)-(29)): the prediction horizon $N_{DMPC} = 50$ samples, the input weight matrix $R_{DMPC} = 5 * 10^{-7}$ and the sample time $T_{s1} = 0.1s$. The trajectory follower controller is designed using the MPC method and has the following parameters (see (14)): the prediction horizon $N_{MPC} = 10$ samples, the weight matrix of the state $Q_{MPC} = 3I_5$, the weight of the

TABLE 1: Parameters of vehicles

| Symbol | Name | Values |
|----------------------|---|------------------------------------|
| m | Vehicle mass | 1000 Kg |
| g | Gravitational acceleration | 9.81 m/s^2 |
| \dot{x}_0 | The initial velocity of the vehicle | 0 m/s |
| α | The road slope | 0 rad |
| α_0 | The initial road slope | 0 rad |
| ρ | The air density | 1.202 kg/m^3 |
| S | The vehicle frontal area | 1.5 m^2 |
| C_d | The drag coefficient | 0.5 |
| f | The rolling resistance | 0.015 |
| v_w | The wind speed | 0 m/s |
| d_0 | The parking distance | 5 m |
| τ | The headway time | 1.6 s |
| F_x^{\min} | Minimum value of the longitudinal traction force | -20000 N |
| F_x^{\max} | Maximum value of the longitudinal traction force | 20000 N |
| C_f | Front tire cornering stiffness coefficient | 63291 N/rad |
| C_r | Rear tire cornering stiffness coefficient | 50041 N/rad |
| ℓ_f | Longitudinal distance from the center of gravity to the front tires | 1.108 m |
| ℓ_r | Longitudinal distance from the center | 1.392 m |
| I | Vehicle's rotational inertia | $1608 \text{ Kg} \cdot \text{m}^2$ |
| δ_f^{\min} | Minimum value of front wheel steering angle | -0.745 rad |
| δ_f^{\max} | Maximum value of front wheel steering angle | 0.745 rad |
| β^{\min} | Minimum value of yaw angle rate | $-\pi/3 \text{ rad}$ |
| $\dot{\beta}^{\max}$ | Maximum value of yaw angle rate | $\pi/3 \text{ rad}$ |
| y^{\min} | Minimum value of the lateral position | -0.5 m |
| y^{\max} | Maximum value of the lateral position | 3.5 m |

input $R_{MPC} = 0.32$, and the sample time $T_{s2} = 0.01s$. The method of designing the trajectory planner using the 5th order polynomials does not have tuning parameters. The inputs of this method are represented by the initial and final state of the vehicle, which are chosen according to the description from Section IV. The initial state is $P_{y0} = [y_0, \dot{y}_0, \ddot{y}_0]^T = [0, 0, 0]^T$ and the imposed final state is chosen to correspond to the lateral state of the vehicle which will be in front, $P_{yf} = [y_f, \dot{y}_f, \ddot{y}_f]^T = [3, 0, 0]^T$. The initial and final values of the lateral velocity and acceleration are considered zero because the vehicles have to maintain the lane, they do not have variations along lateral axis except, the time in which the vehicles change the lane. The target longitudinal position of each vehicle is chosen at 300m ahead. The parameters of the second method used to design the trajectory planner are the following: prediction horizon $N_p = 30$ samples, weight of the lateral error $\lambda_{yp} = 13$, weight of the lateral acceleration variation $\lambda_{a_{yp}} = 300$, and the sample time $T_{sp} = 0.1s$. The parameters of the CACC controller, trajectory follower and trajectory planners were determined heuristically so that for our particular proposed scenario the output (velocity or distance), depending on the position of the vehicle in the platoon, reaches the reference values without overshoot and with the lowest settling time.

Compared to the solution proposed in [44] the advantages of the solution proposed in this paper are represented by: *i*) the solution proposes controllers for both longitudinal and lateral dynamics; *ii*) the vehicles are equipped with a trajectory planner; *iii*) the CACC controller is designed and tested in the presence of data-packet-dropouts, *iv*) the proposed

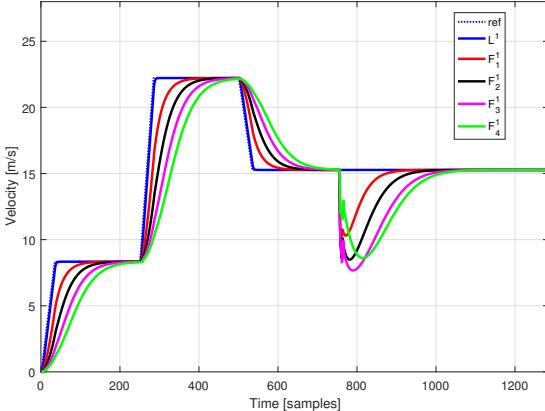


FIGURE 4: Velocities of vehicles from platoon P1

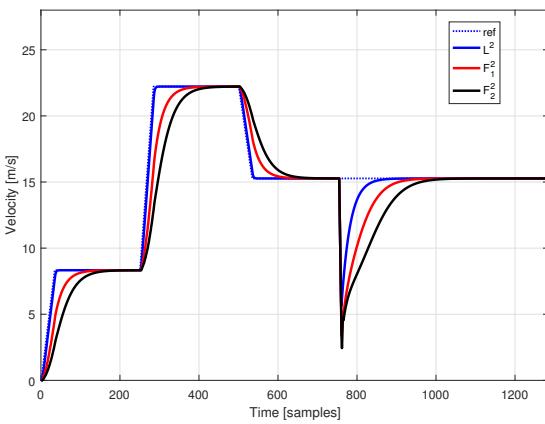


FIGURE 5: Velocities of vehicles from platoon P2

solution is tested in high velocity travel.

The velocities of the vehicles from platoons $P1$ and $P2$ are illustrated in Figs. 4, 5, and the results show that the leaders are moving at the imposed velocity, and the follower vehicles also increase or decrease their velocity to follow the vehicle in front at the desired distance, see Figs. 6, 7, 8, 9. At the beginning, the velocity of vehicles is increased until it reaches a value of $22.22m/s$. When the leader of the second lane, L^2 detects a fixed obstacle on its lane, it starts the merging procedure according to the description provided in the beginning of this section.

To ensure a safe merging, the two platoons are decreasing their travelling velocity to $15.27m/s$ and, after that, the vehicles start to form triangles. The traction forces are illustrated in Figs. 10, 11, and are computed by the CACC for the vehicles to travel at the imposed velocity or to maintain a desired distance. The longitudinal forces are satisfying the imposed constraints, and the sudden variations which appear when the vehicles start to form triangles are due to the sudden variation of the values for the distance errors. At that moment, the vehicles start to maintain an imposed distance

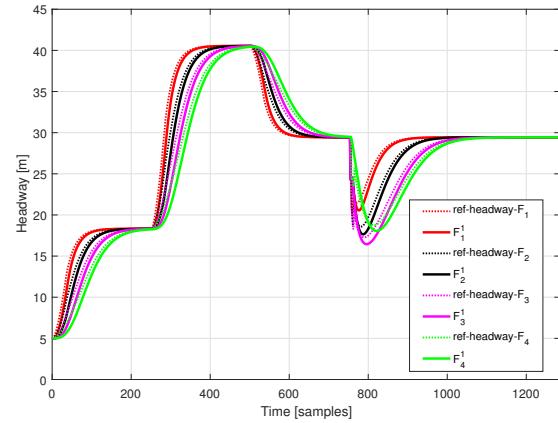


FIGURE 6: Distances between vehicles from platoon P1

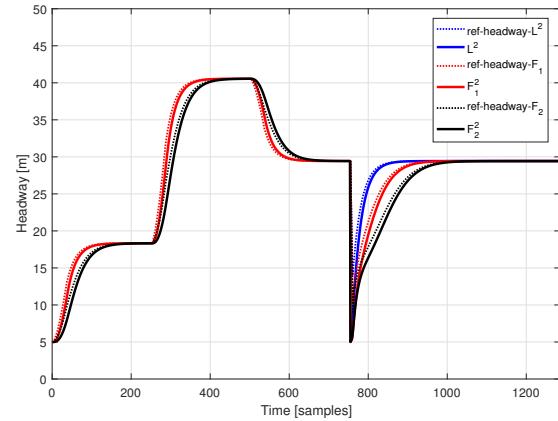


FIGURE 7: Distances between vehicles from platoon P2

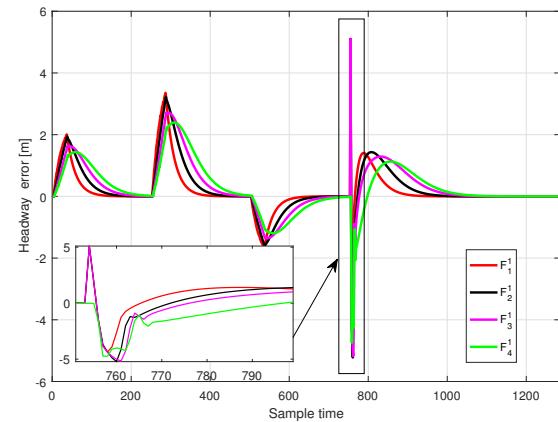


FIGURE 8: Error of distances between vehicles from platoon P1

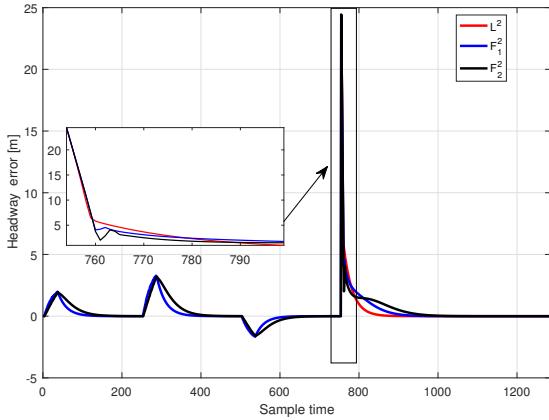


FIGURE 9: Error of distances between vehicles from platoon P2

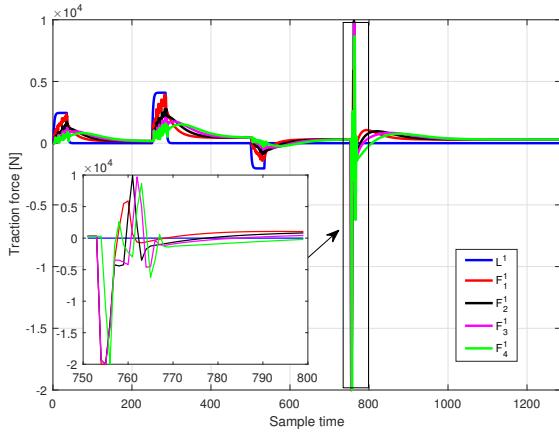


FIGURE 10: Longitudinal traction forces of vehicles from platoon P1

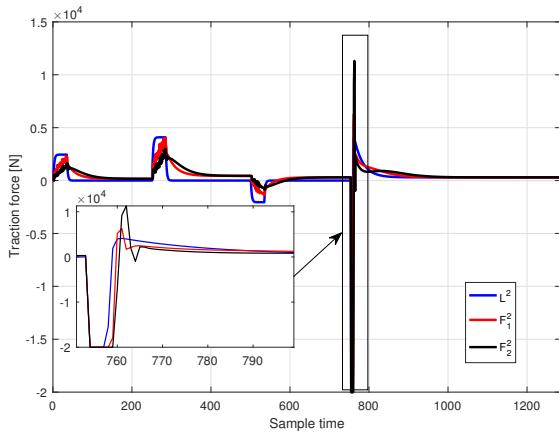


FIGURE 11: Longitudinal traction forces of vehicles from platoon P2

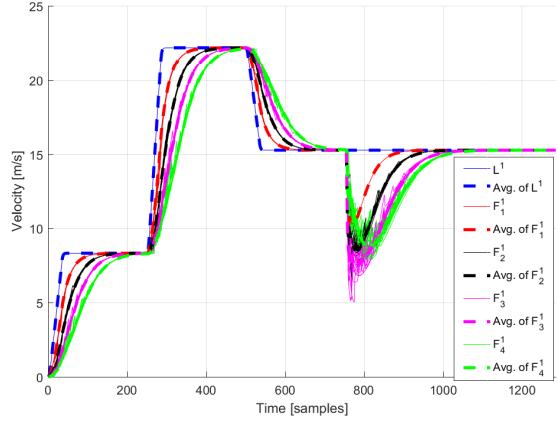


FIGURE 12: Velocities of vehicles from platoon P1 - with data-packet-dropouts

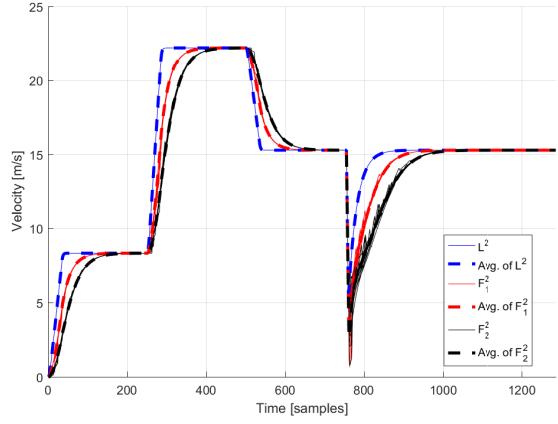


FIGURE 13: Velocities of vehicles from platoon P2 - with data-packet-dropouts

to a new vehicle.

Analysing Figs. 4-9 with the velocity of vehicles and distances between them, it can be affirmed that each vehicle succeeds to maintain a desired distance to its new vehicle in front, thus facilitating the next step, lane changing. The same performances of the CACC controller are obtained even when the communication between vehicles is disturbed by data-packet-dropouts, see Figs. 12, 13 and Figs. 14, 15. In Figs. 12, 13 the results of 55 simulations and the average of velocities of vehicles are illustrated. The time moments at which data losses occur and the number of consecutive lost data-packets are illustrated in Figs. 16, 17.

The data-packet-dropouts are generated randomly with a condition to not have more than 10 consecutive lost data-packets, and the percent of the lost messages to be close to 5%. When the vehicles formed the triangles, see Fig. 20 (a), the Level II of each vehicle starts to generate the path which leads the vehicles of platoon P2 to the neighbour lane, in position from the new platoon formed by merging platoons

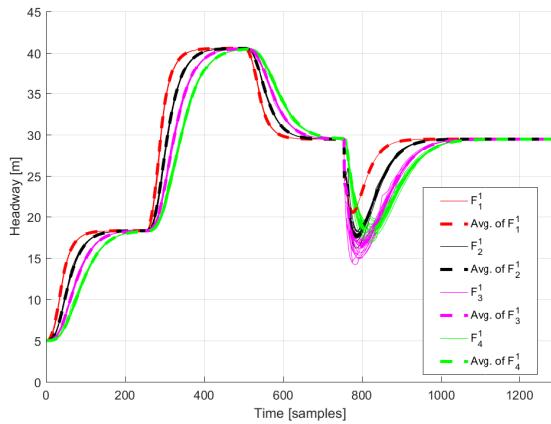


FIGURE 14: Distances between vehicles from platoon P1 - with data-packet-dropouts

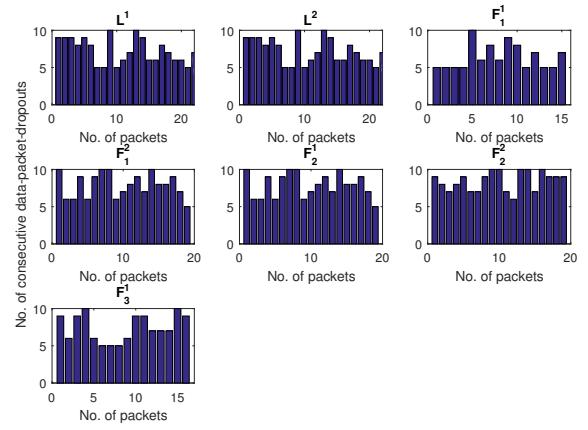


FIGURE 17: Histogram of data-packet-dropouts

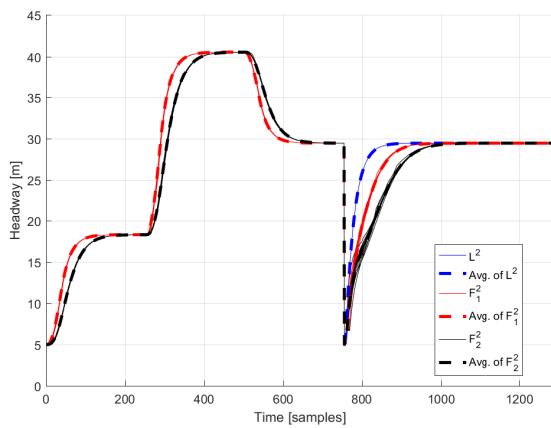


FIGURE 15: Distances between vehicles from platoon P2 - with data-packet-dropouts

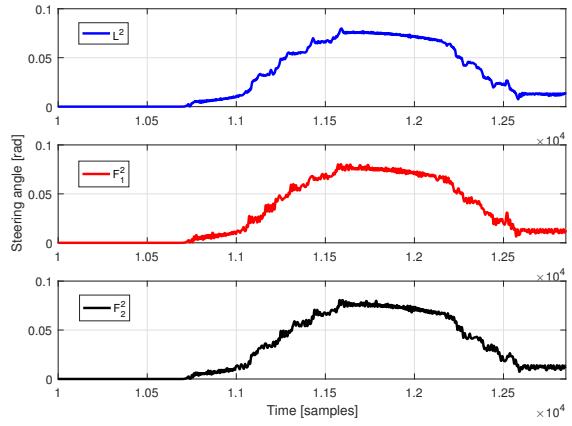


FIGURE 18: Steering angle of the front tire of vehicles from platoon P2 - method from section IV-A

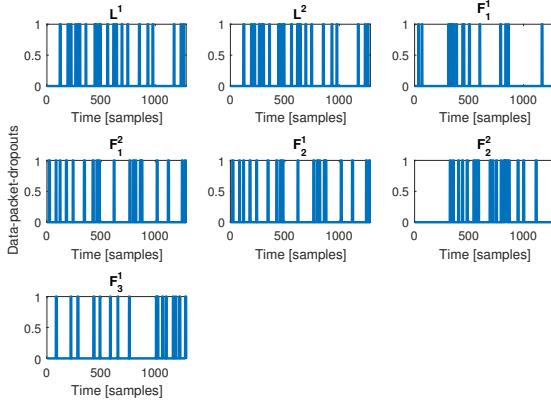


FIGURE 16: Data-packet-dropouts occurrences

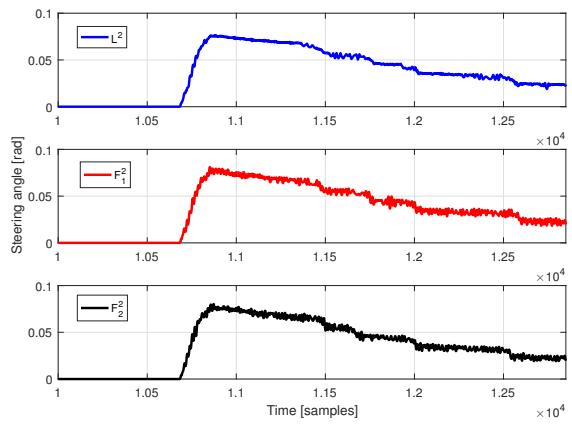


FIGURE 19: Steering angle of the front tire of vehicles from platoon P2 - method from section IV-B

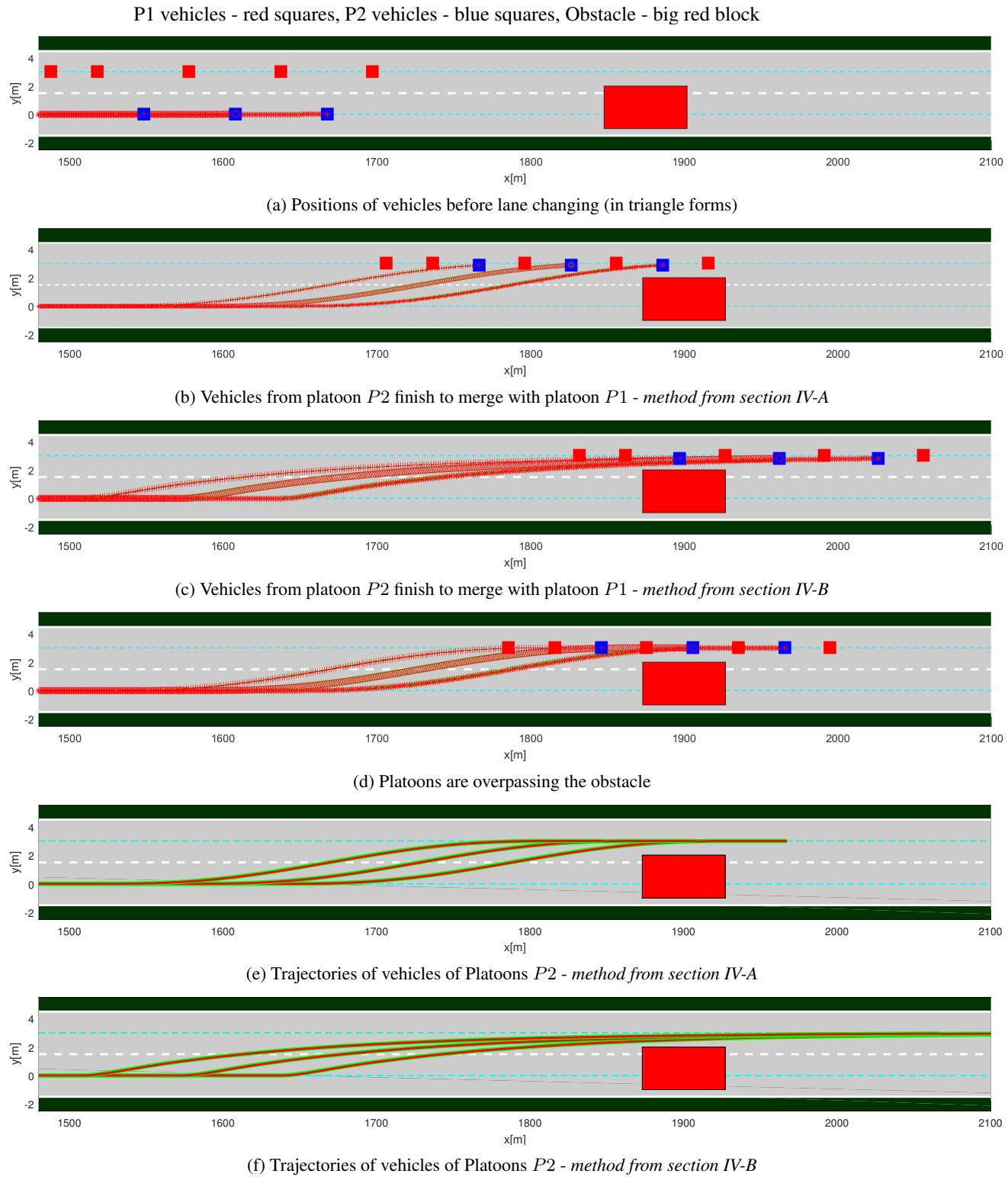


FIGURE 20: Platoons obstacle avoidance through merging manoeuvre

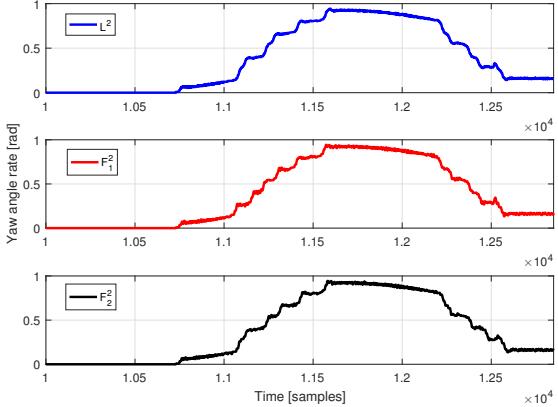


FIGURE 21: Yaw rate of vehicles from platoon P2 - *method from section IV-A*

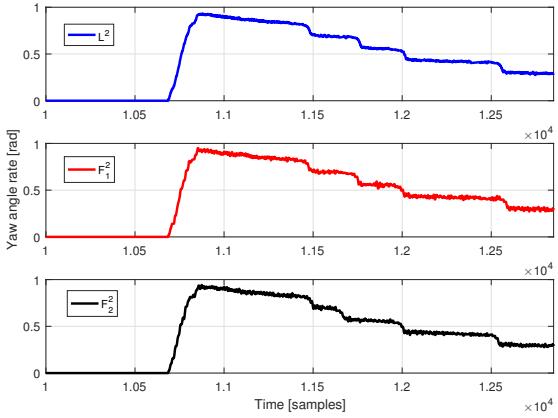


FIGURE 22: Yaw rate of vehicles from platoon P2 - *method from section IV-B*

P1 and *P2*. The trajectories of vehicles are illustrated in Figs. 20 (*b-d*), where the results of both methods, based on polynomial equations and MPC algorithm are illustrated. The trajectory followers of Level III succeed to steer the vehicles so that these follow the imposed trajectories computed by the trajectory planers. Fig. 20 (*e-f*) illustrate with green the paths computed using the trajectory planning strategies described in Section IV and with red colour the paths of actual vehicles. The vehicles from platoon *P2* succeed to change the lane, and to merge with the platoon *P1*, following the imposed path. The steering angle and yaw rate of each vehicle from *P2* are illustrated in Figs. 18, 19, 21, 22. These angles satisfy the imposed constraints and their values and graphical forms are correlated with the trajectories of vehicles from Fig. 20.

B. ANALYSIS OF THE PERFORMANCES FOR THE LATERAL AND LONGITUDINAL DYNAMICS AND THE STRING STABILITY FOR THE PLATOONS

This subsection illustrates the performance of the lateral dynamics and points out the advantages and disadvantages of the two types of trajectory planners. Moreover, an analysis of the string stability and performances of the CACC algorithm is performed. For this use case, the communication topology used to exchange information between the vehicles in the platoon is $T1$.

Analysing the vehicles' trajectories generated by the two methods proposed for path planning (see Fig. 20) it results that the method based on the polynomial equations generates a smoother path w.r.t. the MPC-based method. This relates to the maximum values reached for the lateral acceleration (i.e., $1.87m/s^2$ -method A and $8.27m/s^2$ -method B), the yaw angle rate (see Figs. 21, 22) and the steering angle (see Figs. 18, 19). The computational time required for method A is smaller than the one for the second method ($TC_A = 3.6e-05s < TC_B = 0.29s$). However, it must be noted that the second method has less input variables when compared to method A, as described previously.

Further, the string stability of the platoons is studied. This propriety refers that the platoon has the characteristic to attenuate the distance error or velocity along upstream direction [46], [58]:

$$\|X_i\|_\infty \leq \epsilon_i \|X_{i-1}\|_\infty, \quad (41)$$

where X_i and X_{i-1} represent the distance error or velocity of vehicle i and of its predecessor, vehicle $i-1$ and $\epsilon_i \in (0, 1)$ represents a parameter. When the communication between vehicles is disturbed by the data-packet-dropouts, the study of the string stability is done using the average velocities of vehicles. Analysing the velocities, distances and distance errors of vehicles and their average values in case of presence of data-packet-dropouts results that these states are not amplified along upstream direction and relation (41) is satisfied. *Remark 3:* If the state trajectories X_i , X_{i-1} do not exhibit an overshoot (which is also our case) then, the maximum values of these states are equal and (41) is satisfied as equality condition with $\epsilon_i = 1$.

In Tables 2-5 the maximum, the minimum, the mean and the standard deviation of ϵ_i are illustrated. These values are computed for sets of 50 simulations, with the following characteristics: two values for the headway time, $t_k \in \{0.5s, 1.6s\}$, and two percentage values for the data-packet-dropout, $p \in \{5\%, 10\%\}$, resulting 4 case studies. From the numerical values, it results that $\epsilon_i \in [0.999, 1]$, which proves that the platoon is string stable.

In Tables 6-9 the maximum, the mean the standard deviations for the mean longitudinal accelerations and the maximum distance between vehicles are given. These results are also based on the above mentioned four case studies, and illustrate the fact that the acceleration of the vehicles is decreasing along the upstream direction of the platoon.

TABLE 2: String stability analysis

| $\tau = 0.5s$ | ϵ | | | | | | | |
|---------------|----------------|-----|------|---------|---------------|------|---------|---------|
| $p = 10\%$ | Before merging | | | | After merging | | | |
| VEHICLE | MIN | MAX | MEAN | STD | MIN | MAX | MEAN | STD |
| F_1^1 | 0.99 | 1 | 0.99 | 2.2e-11 | 0.99 | 0.99 | 7.6e-11 | |
| F_1^2 | 0.99 | 1 | 1 | 1.7e-05 | 1 | 1 | 1 | 1.6e-04 |
| F_1^3 | 0.99 | 1 | 1 | 5.3e-05 | 0.99 | 1 | 1 | 2.4e-04 |
| F_1^4 | 1 | 1 | 1 | 1.8e-04 | 0.99 | 1 | 1 | 2e-04 |
| L_1^2 | - | - | - | - | 1 | 1 | 1 | 0 |
| F_2^1 | 0.99 | 1 | 0.99 | 2.2e-11 | 1 | 1 | 1 | 0 |
| F_2^2 | 0.99 | 1 | 1 | 2.6e-06 | 1 | 1 | 1 | 0 |

TABLE 3: String stability analysis

| $\tau = 0.5s$ | ϵ | | | | | | | |
|---------------|----------------|-----|------|---------|---------------|------|------|---------|
| $p = 5\%$ | Before merging | | | | After merging | | | |
| VEHICLE | MIN | MAX | MEAN | STD | MIN | MAX | MEAN | STD |
| F_1^1 | 0.99 | 1 | 0.99 | 3.8e-11 | 0.99 | 0.99 | 0.99 | 4.7e-11 |
| F_1^2 | 0.99 | 1 | 1 | 1.3e-05 | 0.99 | 1 | 1 | 1.3e-04 |
| F_1^3 | 0.99 | 1 | 1 | 9.7e-05 | 0.99 | 1 | 1 | 1.5e-04 |
| F_1^4 | 0.99 | 1 | 1 | 1.1e-04 | 0.99 | 1 | 1 | 1.9e-04 |
| L_1^2 | - | - | - | - | 1 | 1 | 1 | 0 |
| F_2^1 | 0.99 | 1 | 0.99 | 3.8e-11 | 1 | 1 | 1 | 0 |
| F_2^2 | 0.99 | 1 | 1 | 3.6e-06 | 1 | 1 | 1 | 0 |

TABLE 4: String stability analysis

| $\tau = 1.6s$ | ϵ | | | | | | | |
|---------------|----------------|------|------|---------|---------------|------|------|---------|
| $p = 5\%$ | Before merging | | | | After merging | | | |
| VEHICLE | MIN | MAX | MEAN | STD | MIN | MAX | MEAN | STD |
| F_1^1 | 0.99 | 0.99 | 0.99 | 1.1e-06 | 0.99 | 0.99 | 0.99 | 3.8e-06 |
| F_1^2 | 0.99 | 0.99 | 0.99 | 1.4e-06 | 0.99 | 0.99 | 0.99 | 9e-05 |
| F_1^3 | 0.99 | 1 | 0.99 | | 0.99 | 0.99 | 0.99 | 7.1e-04 |
| F_1^4 | 0.99 | 1 | 0.99 | 3e-04 | 0.99 | 0.99 | 0.99 | 9.9e-04 |
| L_1^2 | - | - | - | - | 1 | 1 | 1 | 0 |
| F_2^1 | 0.99 | 1 | 0.99 | 1.1e-05 | 0.99 | 0.99 | 0.99 | 0 |
| F_2^2 | 0.99 | 1 | 0.99 | 2.6e-06 | 1 | 1 | 1 | 0 |

TABLE 5: String stability analysis

| $\tau = 1.6s$ | ϵ | | | | | | | |
|---------------|----------------|------|------|---------|---------------|------|------|---------|
| $p = 10\%$ | Before merging | | | | After merging | | | |
| VEHICLE | MIN | MAX | MEAN | STD | MIN | MAX | MEAN | STD |
| F_1^1 | 0.99 | 0.99 | 0.99 | 1e-06 | 0.99 | 0.99 | 0.99 | 4.5e-06 |
| F_1^2 | 0.99 | 0.99 | 0.99 | 9.8e-06 | 0.99 | 0.99 | 0.99 | 1e-04 |
| F_1^3 | 0.99 | 0.99 | 0.99 | 9.2e-05 | 0.99 | 0.99 | 0.99 | 6.1e-04 |
| F_1^4 | 0.99 | 0.99 | 0.99 | 2.5e-06 | 0.99 | 0.99 | 0.99 | 1e-04 |
| L_1^2 | - | - | - | - | 1 | 1 | 1 | 0 |
| F_2^1 | 0.99 | 0.99 | 0.99 | 1e-06 | 0.99 | 0.99 | 0.99 | 0 |
| F_2^2 | 0.99 | 0.99 | 0.99 | 1.1e-06 | 0.99 | 0.99 | 0.99 | 0 |

TABLE 6: Longitudinal performances analysis

| $\tau = 0.5s$ | Longitudinal Acceleration | | | | | | | |
|---------------|---------------------------|-------------------------|------------------------|------------------------------|------------------------|-------------------------|------------------------|------------------------------|
| $p = 5\%$ | Before merging | | | | After merging | | | |
| VEHICLE | MAX[m/s ²] | MEAN[m/s ²] | STD[m/s ²] | $d_{V_i, V_{i-1}}^{MAX}$ [m] | MAX[m/s ²] | MEAN[m/s ²] | STD[m/s ²] | $d_{V_i, V_{i-1}}^{MAX}$ [m] |
| L^1 | 4.08 | 0.2 | 1.04 | 40.55 | 1.39 | 0 | 0.38 | - |
| F_1^1 | 4.10 | 0.2 | 0.94 | 40.55 | 1.38 | 6.8e-11 | 0.32 | 37 |
| F_1^2 | 3.94 | 0.2 | 0.85 | 40.55 | 1.11 | 5.6e-06 | 0.27 | 37 |
| F_1^3 | 3.55 | 0.2 | 0.79 | 40.55 | 0.87 | 1.1e-05 | 0.24 | 37 |
| F_1^4 | 3.16 | 0.2 | 0.73 | 40.55 | 0.76 | 8.1e-06 | 0.23 | 37 |
| L_1^2 | 4.08 | 0.2 | 1.04 | - | 1.57 | 3.6e-12 | 0.35 | 37 |
| F_2^1 | 4.16 | 0.2 | 0.94 | 40.55 | 1.25 | 5.1e-07 | 0.29 | 37 |
| F_2^2 | 4.10 | 0.2 | 0.86 | 40.55 | 0.99 | 9.2e-06 | 0.26 | 37 |

TABLE 7: Longitudinal performances analysis

| $\tau = 0.5s$ | Longitudinal Acceleration | | | | | | | |
|---------------|---------------------------|-------------------------|------------------------|------------------------------|------------------------|-------------------------|------------------------|------------------------------|
| $p = 10\%$ | Before merging | | | | After merging | | | |
| VEHICLE | MAX[m/s ²] | MEAN[m/s ²] | STD[m/s ²] | $d_{V_i, V_{i-1}}^{MAX}$ [m] | MAX[m/s ²] | MEAN[m/s ²] | STD[m/s ²] | $d_{V_i, V_{i-1}}^{MAX}$ [m] |
| L^1 | 4.08 | 0.2 | 1.04 | - | 1.39 | 0 | 0.38 | - |
| F_1^1 | 4.24 | 0.2 | 0.94 | 16.11 | 1.38 | 7.6e-11 | 0.32 | 15 |
| F_1^1 | 3.98 | 0.2 | 0.96 | 16.11 | 1.21 | 2.2e-06 | 0.27 | 15 |
| F_1^1 | 3.57 | 0.2 | 0.79 | 16.11 | 0.97 | 7.6e-06 | 0.24 | 15 |
| F_1^1 | 3.21 | 0.2 | 0.73 | 16.11 | 0.86 | 0.2 | 0.23 | 15 |
| L^2 | 4.08 | 0.2 | 1.04 | - | 4.08 | 0.2 | 1.04 | 15 |
| F_1^2 | 4.26 | 0.2 | 0.94 | 16.11 | 4.26 | 0.2 | 0.94 | 15 |
| F_1^2 | 3.98 | 0.2 | 0.85 | 16.11 | 3.98 | 0.2 | 0.85 | 15 |

TABLE 8: Longitudinal performances analysis

| $\tau = 1.6s$ | Longitudinal Acceleration | | | | | | | |
|---------------|---------------------------|-------------------------|------------------------|------------------------------|------------------------|-------------------------|------------------------|------------------------------|
| $p = 5\%$ | Before merging | | | | After merging | | | |
| VEHICLE | MAX[m/s ²] | MEAN[m/s ²] | STD[m/s ²] | $d_{V_i, V_{i-1}}^{MAX}$ [m] | MAX[m/s ²] | MEAN[m/s ²] | STD[m/s ²] | $d_{V_i, V_{i-1}}^{MAX}$ [m] |
| L^1 | 4.08 | 0.2 | 1.04 | 40.55 | 1.39 | 0 | 0.30 | - |
| F_1^1 | 3.61 | 0.2 | 7.6e-01 | 40.55 | 0.86 | 3.6e-05 | 0.23 | 37 |
| F_1^1 | 2.66 | 0.2 | 6.1e-01 | 40.55 | 0.52 | 6.1e-04 | 0.18 | 37 |
| F_1^1 | 1.90 | 0.2 | 5.3e-01 | 40.55 | 0.45 | 3e-03 | 0.16 | 37 |
| F_1^1 | 1.47 | 0.2 | 4.7e-01 | 40.55 | 0.44 | 7e-03 | 0.16 | 37 |
| L^2 | 4.08 | 0.2 | 1.04 | - | 4.08 | 0.2 | 1.00 | 37 |
| F_1^2 | 3.59 | 0.2 | 7.6e-01 | 40.55 | 3.59 | 0.2 | 0.76 | 37 |
| F_1^2 | 2.63 | 0.2 | 6.1e-01 | 40.55 | 2.63 | 0.2 | 0.61 | 37 |

TABLE 9: Longitudinal performances analysis

| $\tau = 1.6s$ | Longitudinal Acceleration | | | | | | | |
|---------------|---------------------------|-------------------------|------------------------|------------------------------|------------------------|-------------------------|------------------------|------------------------------|
| $p = 10\%$ | Before merging | | | | After merging | | | |
| VEHICLE | MAX[m/s ²] | MEAN[m/s ²] | STD[m/s ²] | $d_{V_i, V_{i-1}}^{MAX}$ [m] | MAX[m/s ²] | MEAN[m/s ²] | STD[m/s ²] | $d_{V_i, V_{i-1}}^{MAX}$ [m] |
| L^1 | 4.08 | 0.2 | 1.04 | 40.55 | 1.39 | 0 | 0.38 | - |
| F_1^1 | 3.67 | 0.2 | 7.6e-01 | 40.55 | 0.86 | 3.6e-05 | 0.23 | 37 |
| F_1^1 | 2.72 | 0.2 | 6.1e-01 | 40.55 | 0.56 | 6.3e-04 | 0.19 | 37 |
| F_1^1 | 2.02 | 0.2 | 5.3e-01 | 40.55 | 0.43 | 3e-03 | 0.16 | 37 |
| F_1^1 | 1.44 | 0.2 | 4.8e-01 | 40.55 | 0.41 | 0 | 0.16 | 37 |
| L^2 | 4.08 | 0.2 | 1.04 | - | 4.08 | 0.2 | 1.04 | 37 |
| F_1^2 | 3.75 | 0.2 | 7.6e-01 | 40.55 | 3.74 | 0.2 | 0.76 | 37 |
| F_1^2 | 2.18 | 0.2 | 6.1e-01 | 40.55 | 2.81 | 0.2 | 0.61 | 37 |

The comfort of the passengers is influenced by the value of the headway time, which relates to the distances between the vehicles and their acceleration. A small headway time will result in a short inter-vehicle distance, i.e., reducing the platoon length on the highway. However, as previously mentioned, the high values of the longitudinal acceleration, because of the small headway time, will negatively influence the passengers' comfort.

Remark 4: To compute the numerical values from Tables 2-9 the entire simulation test was split into two parts: *i)* part 1 - initial phase, in which the two platoons travel independently on two separate lanes, and part 2 - final phase, after the merging procedure is completed, obtaining a single platoon. In the latter, to test the performances for the newly formed platoon, a velocity reference change for the leader vehicle was also performed (i.e., at time 131 seconds, the velocity is increased from 15.27m/s to 20m/s and at time 157 seconds the velocity is decreased from 20m/s to 15.27m/s).

C. COMPARISON OF THE PERFORMANCES USING DIFFERENT COMMUNICATION TOPOLOGIES IN CACC ALGORITHM

To analyse the performances obtained using the communication topologies $T1-T4$, 50 simulations were performed in presence of data-packets-dropouts for each topology with the following characteristics: the headway time, $t_k = 1.6s$, and percentage value for the data-packet-dropout, $p = 5\%$. The reference velocities of the leaders of the two platoons were imposed from 0m/s to 22.22ms/s and after decreased to 15.27m/s, as it is illustrated in the first 700 sample times in Figs. 12- 13. The model used to simulate the longitudinal dynamics of platoons is the same for all topologies (2)-(6). The influences of communication topologies were considered in the cost function, for each case having a particular one, (29)-(32). The performances of the platoons were illustrated in Table 10 (maximum acceleration and standard deviation) and in Table 11 (mean squared velocity error). From these numerical results it can be observed that the smallest values for the maximum acceleration were obtained using topologies $T3$ and $T4$, which means that the passengers will have a more

TABLE 10: Topologies performances analysis 1

| $\tau = 1.6s$ $p = 5\%$ | Longitudinal Acceleration | | | | | | | |
|----------------------------|---------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|
| | T1 | | T2 | | T3 | | T4 | |
| | VEHICLE | MAX[m/s ²] | STD[m/s ²] | MAX[m/s ²] | STD[m/s ²] | MAX[m/s ²] | STD[m/s ²] | MAX[m/s ²] |
| L^1 | 4.08 | 1.05 | 4.08 | 1.05 | 3.66 | 0.88 | 3.52 | 0.8 |
| F_1^1 | 3.61 | 0.76 | 3.72 | 0.76 | 3.2 | 0.62 | 2.82 | 0.62 |
| F_2^1 | 2.66 | 0.61 | 2.84 | 0.61 | 2.45 | 0.52 | 2.13 | 0.52 |
| F_3^1 | 1.90 | 0.53 | 2.44 | 0.51 | 1.77 | 0.46 | 1.75 | 0.45 |
| F_4^1 | 1.47 | 0.48 | 1.82 | 0.45 | 1.31 | 0.42 | 1.63 | 0.41 |
| L^2 | 4.08 | 1.6 | 4.08 | 1.05 | 3.56 | 0.8 | 3.43 | 0.8 |
| F_2^2 | 3.59 | 1.76 | 3.61 | 0.75 | 3.19 | 0.64 | 2.97 | 0.63 |
| F_2^2 | 2.63 | 0.61 | 2.84 | 0.61 | 2.42 | 0.54 | 2.62 | 0.54 |

TABLE 11: Topologies performances analysis 2

| $\tau = 1.6s$ $p = 5\%$ | Mean squared velocity error | | | | | | | |
|----------------------------|-----------------------------|-----------------|-------------------|-------------------|-------------------|-------------------|-----------------|-------------------|
| | Topology | $L^1[ms^2/s^2]$ | $F_1^1[ms^2/s^2]$ | $F_2^1[ms^2/s^2]$ | $F_3^1[ms^2/s^2]$ | $F_4^1[ms^2/s^2]$ | $L^2[ms^2/s^2]$ | $F_1^2[ms^2/s^2]$ |
| T1 | 0.088 | 1.67 | 1.1 | 0.84 | 0.66 | 0.088 | 1.68 | 1.11 |
| T2 | 0.088 | 1.47 | 0.9 | 0.68 | 0.55 | 0.088 | 1.53 | 0.88 |
| T3 | 1.27 | 1.04 | 0.78 | 0.62 | 0.5 | 1.3 | 1.04 | 0.75 |
| T4 | 1.33 | 1.03 | 0.72 | 0.57 | 0.47 | 1.34 | 1.03 | 0.69 |

conformable travel and the fuel consumption will be reduced. Moreover, from Table 10 it can be noticed that smallest velocity errors are obtained using the same topologies $T3$ and $T4$ in which the vehicles communicate with their neighbouring vehicles from front and rear. From the point of view of string stability, all topologies ensure it for the platoons, the velocity error being attenuated along upstream direction.

Remark 5: For topology $T1$ and $T2$ the controllers of the leader vehicles have to control only their velocity so, the stability is analysed from the first follower to the last one. In case of bidirectional topologies $T3$ and $T4$, because the leaders receive the prediction of the velocity of their first followers, they will use their controllers to control the velocities in order to reach the imposed values but, also, to minimise the velocity error between them and the vehicles behind them. This means that, the leader vehicles can decrease their velocity to ensure that the first followers are not too far away. Thus, the leaders will have also the role of maintaining a target distance to the first followers and for this reason, the string stability can be analysed starting from the leader vehicle for these two topologies $T3$ and $T4$.

Remark 6: The advantage of the first topology $T1$ is related to its simplicity, requiring only the exchange of information from a certain vehicle to its follower, but with higher values w.r.t. to the analysed parameters (maximum longitudinal acceleration and mean squared velocity error). The last topology $T4$ has as disadvantage an increased volume of exchanged information between the vehicles in the platoon, but the obtained performances are increased as detailed previously.

VII. CONCLUSION

This paper proposes a control architecture formed by three levels, which can be used by the vehicles driving in platoons to ensure a safe travelling. Level I is used to obtain information about traffic and to decide the future actions. Levels

II and III receive commands from Level I and accomplish them. The second level is represented by a trajectory planner for which two methods were proposed: the first method is based on polynomial equations, and the second method is designed using a MPC strategy. Level III is represented by a CACC controller and a trajectory follower having the tasks to maintain a desired distance between vehicles, and to steer them to follow an imposed lateral trajectory. Moreover, the CACC controller was designed to consider different types of communication topologies, resulting in a framework that is beneficial to be used for various realistic scenarios. The proposed control architecture was tested in a simulation scenario in which two platoons had to merge to avoid the collision with a fixed obstacle. The results prove the efficiency of the proposed solution and the string stability of platoons. The main contribution of this paper with respect to similar works is given by the following characteristics: the solution was tested in the presence of the disturbed communications, the trajectory planner was implemented using two different approaches, the lateral dynamics of the vehicles was simulated using the nonlinear bicycle model, the solution combines both lateral and longitudinal dynamics control and the CACC algorithm was tested for different communication topologies to illustrate their advantages and disadvantages.

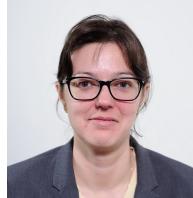
Future work will focus on extending the proposed control architecture to ensure a safe travel for vehicles in presence of moving obstacles.

REFERENCES

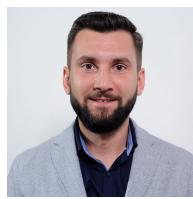
- [1] C. Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T. Paixao, F. Mutz, L. Veronese, T. Oliveira-Santos, and A. F. D. Souza, "Self-driving cars: A survey," *CoRR*, 2019.
- [2] Y. Bian, J. Ding, M. Hu, Q. Xu, J. Wang, and K. Li, "An advanced lane-keeping assistance system with switchable assistance modes," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 1, pp. 385–396, 2020.
- [3] K. Lee and D. Kum, "Collision avoidance/mitigation system: Motion

- planning of autonomous vehicle via predictive occupancy map,” *IEEE Access*, vol. 7, pp. 52 846–52 857, 2019.
- [4] F. Lin, Y. Zhang, Y. Zhao, G. Yin, H. Zhang, and K. Wang, “Trajectory tracking of autonomous vehicle with the fusion of DYC and longitudinal-lateral control,” *Chinese Journal of Mechanical Engineering*, vol. 32, no. 1, 2019.
- [5] R. Lattarulo, M. Marcano, and J. Pérez, “Overtaking maneuver for automated driving using virtual environments,” in *Computer Aided Systems Theory – EUROCAST 2017*. Cham: Springer International Publishing, 2018, pp. 446–453.
- [6] S. Li, Z. Li, Z. Yu, B. Zhang, and N. Zhang, “Dynamic trajectory planning and tracking for autonomous vehicle with obstacle avoidance based on model predictive control,” *IEEE Access*, vol. 7, pp. 132 074–132 086, 2019.
- [7] P. Ioannou and C. Chien, “Autonomous intelligent cruise control,” *IEEE Transactions on Vehicular Technology*, vol. 42, no. 4, pp. 657–672, 1993.
- [8] Q. Wang and B. Ayalew, “A probabilistic framework for tracking the formation and evolution of multi-vehicle groups in public traffic in the presence of observation uncertainties,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, pp. 560–571, 2018.
- [9] D. Corona and B. De Schutter, “Adaptive cruise control for a smart car: A comparison benchmark for MPC-PWA control methods,” *IEEE Transactions on Control Systems Technology*, vol. 16, no. 2, pp. 365–372, 2008.
- [10] N. Wang, X. Wang, P. Palacharla, and T. Ikeuchi, “Cooperative autonomous driving for traffic congestion avoidance through vehicle-to-vehicle communications,” in *IEEE Vehicular Networking Conference*, Nov 2017, pp. 327–330.
- [11] H. Min, Y. Yang, Y. Fang, P. Sun, and X. Zhao, “Constrained optimization and distributed model predictive control-based merging strategies for adjacent connected autonomous vehicle platoons,” *IEEE Access*, vol. 7, pp. 163 085–163 096, 2019.
- [12] O. Pauca, A. Maxim, and C. F. Caruntu, “DMPC-based data-packet dropout compensation in vehicle platooning applications using V2V communications,” in *European Control Conference*, Rotterdam, Netherlands, 2021, p. accepted.
- [13] A. Maxim, C. Lazar, and C. F. Caruntu, “Distributed model predictive control algorithm with communication delays for a cooperative adaptive cruise control vehicle platoon,” in *28th Mediterranean Conference on Control and Automation*, 2020, pp. 909–914.
- [14] T. Zhang, Y. Zou, X. Zhang, N. Guo, and W. Wang, “A cruise control method for connected vehicle systems considering side vehicles merging behavior,” *IEEE Access*, vol. 7, pp. 6922–6936, 2019.
- [15] R. Attia, R. Orjuela, and M. Basset, “Combined longitudinal and lateral control for automated vehicle guidance,” *Vehicle System Dynamics*, vol. 52, no. 2, pp. 261–279, 2014.
- [16] P. Falcone, H. E. Tseng, F. Borrelli, J. Asgari, and D. Hrovat, “MPC-based yaw and lateral stabilisation via active front steering and braking,” *Vehicle System Dynamics*, vol. 46, no. sup1, pp. 611–628, 2008.
- [17] O. Pauca, C. F. Caruntu, and C. Lazar, “Predictive control for the lateral and longitudinal dynamics in automated vehicles,” in *23rd International Conference on System Theory, Control and Computing*, Sinaia, Romania, 2019, pp. 797–802.
- [18] B. Schaeufele, O. Sawade, D. Pfahl, K. Massow, S. Bunk, B. Henke, and I. Radusch, “Forward-looking automated cooperative longitudinal control: Extending cooperative adaptive cruise control (CACC) with column-wide reach and automated network quality assessment,” in *IEEE 20th International Conference on Intelligent Transportation Systems*, 2017, pp. 1–6.
- [19] H. Kazemi, H. N. Mahjoub, A. Tahmasbi-Sarvestani, and Y. P. Fallah, “A learning-based stochastic MPC design for cooperative adaptive cruise control to handle interfering vehicles,” *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 3, pp. 266–275, 2018.
- [20] H. Ma, L. Chu, J. Guo, J. Wang, and C. Guo, “Cooperative adaptive cruise control strategy optimization for electric vehicles based on SA-PSO with model predictive control,” *IEEE Access*, vol. 8, pp. 225 745–225 756, 2020.
- [21] Y. Sun, P. Wang, and F. Liu, “Design and implementation of cooperative adaptive cruise control based on V2X,” in *IEEE MTT-S International Wireless Symposium*, 2020, pp. 1–3.
- [22] T. Tapli and M. Akar, “Cooperative adaptive cruise control algorithms for vehicular platoons based on distributed model predictive control,” in *IEEE 16th International Workshop on Advanced Motion Control*, 2020, pp. 305–310.
- [23] A. Maxim, O. Pauca, C. F. Caruntu, and C. Lazar, “Distributed model predictive control algorithm with time-varying communication delays for a CACC vehicle platoon,” in *24th International Conference on System Theory, Control and Computing*, 2020, pp. 775–780.
- [24] O. Pauca, C. F. Caruntu, and A. Maxim, “Trajectory planner based on third-order polynomials applied for platoon merging and splitting,” in *29th Mediterranean Conference on Control and Automation*, Bari, Italy, 2021, pp. 83–88.
- [25] L. Zhang, “Cooperative adaptive cruise control in mixed traffic with selective use of vehicle-to-vehicle communication,” *IET Intelligent Transport Systems*, vol. 12, no. 10, pp. 1243–1254, 2018.
- [26] Y. Zhou, H. Zhu, M. Guo, and J. Zhou, “Impact of CACC vehicles’ cooperative driving strategy on mixed four-lane highway traffic flow,” *Physica A: Statistical Mechanics and its Applications*, vol. 540, p. 122721, 2020.
- [27] A. S. Valente, S. Santini, and P. A. Palladino, “Cooperative driving in inter-vehicular communication network,” Ph.D. dissertation, University of Naples, Federico II-Italy, 2015.
- [28] G. Orosz, “Connected automated vehicle design among human-driven vehicles,” *IFAC-PapersOnLine*, vol. 51, no. 34, pp. 403 – 406, 2019, 2nd IFAC Conference on Cyber-Physical and Human Systems.
- [29] H. Liu, W. Zhuang, G. Yin, Z. Tang, and L. Xu, “Strategy for heterogeneous vehicular platoons merging in automated highway system,” in *Chinese Control and Decision Conference*, Shenyang, China, 2018, pp. 2736–2740.
- [30] O. Pauca, C. F. Caruntu, A. Maxim, and C. Lazar, “Control architecture for automated vehicles to ensure obstacle avoidance,” in *Buletinul Institutului Politehnic din Iasi*, Iasi, Romania, 2021, p. accepted.
- [31] O. Pauca, C. F. Caruntu, and A. Maxim, “Trajectory planning and tracking for cooperative automated vehicles in a platoon,” in *24th International Conference on System Theory, Control and Computing*, Sinaia, Romania, 2020, pp. 769–774.
- [32] I. B. Viana, H. Kanchwala, and N. Aouf, “Cooperative trajectory planning for autonomous driving using nonlinear model predictive control,” in *IEEE International Conference on Connected Vehicles and Expo*, 2019, pp. 1–6.
- [33] J. Rios-Torres and A. A. Malikopoulos, “Automated and cooperative vehicle merging at highway on-ramps,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 4, pp. 780–789, April 2017.
- [34] Y. Gao, “Model predictive control for autonomous and semiautonomous vehicles,” Ph.D. dissertation, UC Berkeley, 2014.
- [35] R. Zhou, Z. Li, and Z. Yang, “Optimal lane change analysis for vehicle platooning based on lateral and longitudinal control,” in *39th Chinese Control Conference*, 2020, pp. 5534–5539.
- [36] A. Wang, A. Jasour, and B. C. Williams, “Non-gaussian chance-constrained trajectory planning for autonomous vehicles under agent uncertainty,” *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6041–6048, 2020.
- [37] J. P. Talamino and A. Sanfeliu, “Anticipatory kinodynamic motion planner for computing the best path and velocity trajectory in autonomous driving,” *Robotics and Autonomous Systems*, vol. 114, pp. 93 – 105, 2019.
- [38] A. Duret, M. Wang, and A. Ladino, “A hierarchical approach for splitting truck platoons near network discontinuities,” *Transportation Research Part B: Methodological*, vol. 132, pp. 285 – 302, 2020, 23rd International Symposium on Transportation and Traffic Theory.
- [39] Q. Xin, R. Fu, S. V. Ukkusuri, S. Yu, and R. Jiang, “Modeling and impact analysis of connected vehicle merging accounting for mainline random length tight-platoon,” *Physica A: Statistical Mechanics and its Applications*, vol. 563, p. 125452, 2021.
- [40] M. Cicic, K.-Y. Liang, and K. Henrik Johansson, “Platoon merging distance prediction using a neural network vehicle speed model,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3720 – 3725, 2017.
- [41] A. Schwab and J. Lunze, “Cooperative vehicle merging with guaranteed collision avoidance,” *IFAC-PapersOnLine*, vol. 52, no. 6, pp. 7 – 12, 2019, 1st IFAC Workshop on Control of Transportation Systems.
- [42] ———, “Cooperative vehicle merging with guaranteed collision avoidance,” *IFAC-PapersOnLine*, vol. 52, no. 6, pp. 7 – 12, 2019, 1st IFAC Workshop on Control of Transportation Systems.
- [43] D. Chen, M. Zhao, D. Sun, L. Zheng, S. Jin, and J. Chen, “Robust h control of cooperative driving system with external disturbances and communication delays in the vicinity of traffic signals,” *Physica A: Statistical Mechanics and its Applications*, p. 123385, 2019.
- [44] C. Hidalgo, R. Lattarulo, C. Flores, and J. Perez Rastelli, “Platoon merging approach based on hybrid trajectory planning and CACC strategies,” *Sensors*, vol. 21, no. 8, 2021.

- [45] O. Pauca, A. Maxim, and C. F. Caruntu, "Cooperative platoons merging for obstacle avoidance on highways," in *25th International Conference on System Theory, Control and Computing*, Iasi, Romania, 2021, pp. –.
- [46] Z. Wang, G. Wu, and M. J. Barth, "A review on cooperative adaptive cruise control (CACC) systems: Architectures, controls, and applications," in *21st International Conference on Intelligent Transportation Systems*, 2018, pp. 2884–2891.
- [47] Y. Zheng, S. Eben Li, J. Wang, D. Cao, and K. Li, "Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 1, pp. 14–26, 2016.
- [48] M. H. Basiri, B. Ghojogh, N. L. Azad, S. Fischmeister, F. Karray, and M. Crowley, "Distributed nonlinear model predictive control and metric learning for heterogeneous vehicle platooning with cut-in/cut-out maneuvers," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 2849–2856.
- [49] A. Ulsoy, H. Peng, and M. Çakmacı, *Automotive control systems*. Cambridge University Press, 05 2012.
- [50] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.
- [51] E. F. Camacho and C. Bordons, *Introduction to Model Based Predictive Control*. London: Springer London, 1999, pp. 1–11.
- [52] I. Necoara, V. Nedelcu, and I. Dumitrasche, "Parallel and distributed optimization methods for estimation and control in networks," *Journal of Process Control*, vol. 21, no. 5, pp. 756–766, 2011.
- [53] I. Necoara, *Rate Analysis of Inexact Dual Fast Gradient Method for Distributed MPC*. Dordrecht: Springer Netherlands, 2014, pp. 163–178.
- [54] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia, "Cooperative distributed model predictive control," *Systems and Control Letters*, vol. 59, no. 8, pp. 460–469, 2010.
- [55] B. Caiazzo, A. Coppola, A. Petrillo, and S. Santini, "Distributed nonlinear model predictive control for connected autonomous electric vehicles platoon with distance-dependent air drag formulation," *Energies*, vol. 14, no. 16, 2021.
- [56] Y. Wu, S. E. Li, Y. Zheng, and J. K. Hedrick, "Distributed sliding mode control for multi-vehicle systems with positive definite topologies," in *IEEE 55th Conference on Decision and Control*, 2016, pp. 5213–5219.
- [57] Y. Wu, S. E. Li, J. Cortes, and K. Poolla, "Distributed sliding mode control for nonlinear heterogeneous platoon systems with positive definite topologies," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 4, pp. 1272–1283, 2020.
- [58] W. B. Dunbar and D. S. Caveney, "Distributed receding horizon control of vehicle platoons: Stability and string stability," *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 620–633, 2012.



ANCA MAXIM received a joint-Ph.D. degree in Systems Engineering from the Gheorghe Asachi Technical University of Iasi (TUIasi), Romania, and in Engineering from the Ghent University (UGent), Belgium in June 2019. She is currently assistant professor in the Department of Automatic Control and Applied Informatics at Gheorghe Asachi Technical University of Iasi, Faculty of Automatic Control and Computer Engineering. She is author/co-author of about 30 publications in journals, books, conference proceedings and participated in more than 4 collaborative projects. Her current research interests include distributed model predictive control, coalitional control systems and automotive control systems.



CONSTANTIN FLORIN CARUNTU received his Ph.D. degree in Systems Engineering from the Gheorghe Asachi Technical University of Iasi (TUIasi), Romania, November 2011. He is currently full Professor in the Department of Automatic Control and Applied Informatics at the same university, Faculty of Automatic Control and Computer Engineering. He is author/co-author of about 100 publications in journals, books, conference proceedings and participated in more than 10 collaborative projects. His current research interests include model predictive control, networked control systems, automotive control systems and cooperative driving.



OVIDIU PAUCA received his M.S. degree in Systems and Control from Gheorghe Asachi Technical University of Iasi, Romania, in June 2019. He is currently pursuing Ph.D. degree in the field of cooperative automotous vehicles at the Gheorghe Asachi Technical University of Iasi. His current research interests include vehicle dynamics control, cooperative vehicles, model based predictive control, networked control systems.