A H∞ LPV State Feedback Controller for Velocity Dependent Time Headway in Heterogeneous Platoon -Application to Microscale Platoon

Maëlle Haudrechy, Olivier Sename, Péter Gáspár

A $\mathscr{H}_\infty$ LPV State Feedback Controller for Velocity Dependent Time Headway in Heterogeneous Platoon - Application to Microscale Platoon ⋆

Maëlle Haudrechy * **Olivier Sename** * **Péter Gáspár** **

* *Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, 38000 Grenoble, France (e-mail: maelle.haudrechy@grenoble-inp.fr, olivier.sename@grenoble-inp.fr).*
** *Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (ELKH), H-1111, Budapest, Hungary (e-mail: gaspar@sztaki.hun-ren.hu)*

Abstract: This paper presents the design of a Cooperative Adaptive Cruise Control (CACC) state feedback controller for a heterogeneous vehicle platoon with varying driveline dynamics. The proposed architecture adopts a Predecessor-Follower (PF) communication topology and a velocity-dependent time headway spacing policy. The control problem is formulated within the $\mathscr{H}_\infty$ linear parameter-varying (LPV) framework and solved using polytopic and grid-based LPV methods. The methods are applied to a microscale platoon to illustrate and validate the proposed controller through both frequency and time-domain simulations using a dedicated microcar platoon simulator.

Keywords: Cooperative Adaptive Cruise Control, Heterogeneous platoon, Variable time headway, Linear parameter-varying, $\mathscr{H}_\infty$ control, microscale platoon

1. INTRODUCTION

The concept of Cooperative Adaptive Cruise Control (CACC), reviewed in Wang et al. (2018), extends Adaptive Cruise Control by incorporating vehicle-to-vehicle communication. CACC systems enable cooperative behavior among vehicles and ensure string stability, defined as the ability to attenuate disturbances along a platoon of vehicles, even at very close inter vehicle distances. By reducing inter-vehicle distances, CACC systems represent an interesting solution to increase traffic flow and safety on highways while reducing fuel consumption and emissions due to the reduction of aerodynamic drag.

This paper focuses on CACC controller design in which vehicle longitudinal dynamics modeling plays a key role. A simple first-order model, having the desired acceleration as input and the actual acceleration as output, is commonly adopted to model the longitudinal control of the vehicle as described in Rajamani (2012). In practical applications, the platoon may involve vehicles with different driveline dynamics, thus forming a heterogeneous platoon. Since the driveline characteristics of other vehicles are typically unknown, studies have addressed this challenge either by treating heterogeneity as model uncertainty as in Gao et al. (2016) or by designing controllers that do not require explicit knowledge of other vehicles' dynamics as in Lefeber et al. (2020), which is also the approach adopted in this paper.

In the platoon, each vehicle follows its predecessor according to a predefined spacing policy. Common strategies include Constant Distance, Constant Time Headway, Variable Time Headway (VTH), and other nonlinear spacing strategies as summarized in Flores et al. (2017). Time-headway policies are generally preferred for platooning, with VTH offering enhanced adaptability by dynamically adjusting to vehicle speed or road conditions, improving safety, energy efficiency, and platoon stability in complex driving environments. Several VTH-based policies have been proposed: Flores et al. (2017) and Zhou and Peng (2005) aim to optimize traffic flow, safety, and stability constraints, while Zhuang et al. (2024), and Wang et al. (2025) incorporate relative velocity and acceleration of the preceding vehicle to further enhance string stability and robustness.

The novelty of this work lies in addressing the nonlinearity introduced by the use of a velocity-dependent time headway policy within an LPV framework. This approach enables the design of an LPV state-feedback controller using both polytopic and grid-based methods. The controller synthesis relies on an $\mathscr{H}_\infty$ approach, as in Gao et al. (2016) and Ploeg et al. (2014a), which is particularly well suited to express performance specifications in the frequency domain such as string stability. The proposed methodology is applied to a microscale platoon and its effectiveness is demonstrated through both frequency-domain analyzes and time-domain simulations using a dedicated simulation environment.

The remainder of this paper is organized as follows. Section 2 presents the problem formulation and the quasi-LPV (qLPV) modeling of an individual vehicle within the platoon. Section 3 details the $\mathscr{H}_\infty$ LPV state-feedback controller design, including

⋆ This work was partially supported by I-TireLab (2025– 2029), a joint public-private laboratory focused on modeling, estimation, and control for connected tires and rubber manufacturing processes. The laboratory is supported by Michelin, Université de Poitiers, Univ. Grenoble Alpes, Grenoble INP, and CNRS. The research was also partially supported by the National Research, Development and Innovation Office (NKFIH) through the project "Design of high performance safe autonomous vehicle systems via integrated robust control and learning-based methods" (2021-1.2.4-TÉT-2022-00065).

polytopic and grid-based approaches. Section 4 describes the microscale vehicle model and simulation environment. Section 5 applies the controller to a microscale platoon with a specific spacing policy, supported by frequency-domain analysis. Section 6 presents time-domain simulation results. Finally, Section 7 concludes the paper.

2. PROBLEM FORMULATION AND QLPV MODELING

2.1 Platoon System Description

A platoon of $N+1$ ($N \in \mathbb{N}$) heterogeneous vehicles is considered, as illustrated in Fig.1. Adopting the PF topology, it is assumed that the acceleration of the predecessor vehicle, a_{i-1}, is communicated through wireless communication. Each vehicle $i \in [1,N]$ aims to follow its predecessor at a desired inter-vehicle distance d_i^{ref} while the leader ($i=0$) is assumed to be velocity controlled.

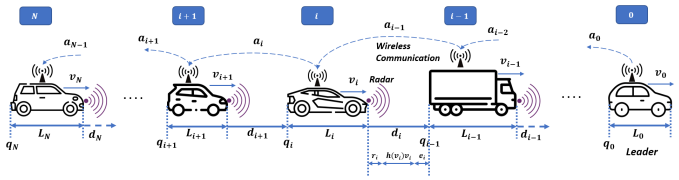

Fig. 1. CACC-equipped heterogeneous platoon with PF topology

Considering a spacing policy based on a velocity-dependent time headway, the desired intervehicle distance is defined as

$$d_i^{ref}(t) = r_i + h(v_i(t)) \cdot v_i(t), \quad i \in [1,N], \tag{1}$$

where r_i denotes the standstill distance associated with vehicle i, and h is the time-headway function, which depends on the velocity of the vehicle v_i. It is assumed that the time-headway function $h : [v_{min}, v_{max}] \subset \mathbb{R}_+ \to \mathbb{R}_+^*$ satisfies the following properties:

(i) $h(v) > 0, \quad \forall v \in [v_{min}, v_{max}]$,

(ii) $h \in \mathscr{C}^1([v_{min}, v_{max}])$,

(iii) $\sup_{v \in [v_{min}, v_{max}]} |h(v)| < +\infty$ and $\sup_{v \in [v_{min}, v_{max}]} |h'(v)| < +\infty$.

The notation h' refers to the first-order derivative of h with respect to v.

Denoting d_i the actual distance between vehicle i and its predecessor $i-1$, the distance error, $e_i = d_i - d_i^{ref}$, is formulated, for $i \in [1,N]$, as

$$e_i(t) = (q_{i-1}(t) - q_i(t) - L_i) - (r_i + h(v_i(t)) \cdot v_i(t)), \tag{2}$$

where q_i denotes the rear-bumper position of vehicle i and L_i its length. Furthermore, denoting a_i the actual acceleration of vehicle i, the dynamics of vehicle i, can be described by

$$\begin{cases} \dot{q}_i(t) = v_i(t) \\ \dot{v}_i(t) = a_i(t) \\ \dot{a}_i(t) = -\dfrac{1}{\tau_i} \cdot a_i(t) + \dfrac{1}{\tau_i} \cdot u_i(t) \end{cases}, \quad i \in [0,N], \tag{3}$$

where u_i denotes the desired acceleration of vehicle i and τ_i is the time constant reflecting its engine dynamics.

The platoon is assumed to be heterogeneous with respect to driveline dynamics, which is reflected by $\tau_i \neq \tau_{i-1}$, and handled by adopting the following nonlinear input transformation as proposed in Lefeber et al. (2020) (here adapted for the VTH case):

$$\begin{aligned} u_i(t) = &\frac{\tau_i}{h(v_i(t))} \cdot (\xi_i(t) + a_{i-1}^*(t)) \\ &+ (1 - \frac{\tau_i}{h(v_i(t))}) \cdot a_i(t) \end{aligned}, \quad i \in [1,N], \tag{4}$$

where ξ_i is the new control input and $a_{i-1}^*(t) = a_{i-1}(t - \theta_i)$ is the actual acceleration of the predecessor vehicle $i-1$ communicated to vehicle i with a delay θ_i. Here, a_{i-1}, considered as a disturbance for vehicle i, is handled in a feedforward term to allow anticipation and coordination within the platoon. Combining (3) with (4), the derivative of a_i, for $i \in [1,N]$, is now given by

$$\dot{a}_i(t) = \frac{1}{h(v_i(t))} \cdot (\xi_i(t) + a_{i-1}^*(t)) - \frac{1}{h(v_i(t))} \cdot a_i(t). \tag{5}$$

It can be noticed that (5) is independent of the time constant τ_i.

2.2 A qLPV model of vehicle i

To achieve the control objectives outlined in Section 3, and as is common in platooning control, the distance tracking error e_i is used as a state variable instead of the absolute position q_i in (3). Accordingly, for each vehicle $i \in [1,N]$, the state vector is defined as $x_i = (e_i, v_i, a_i)^T$. According to (2) and (3), the derivative of the spacing error, $\dot{e}_i$, is computed as

$$\begin{aligned} \dot{e}_i(t) = &v_{i-1}(t) - v_i(t) \\ &- a_i(t) \cdot \big(h(v_i(t)) + h'(v_i(t)) \cdot v_i(t)\big) \end{aligned}, \quad i \in [1,N]. \tag{6}$$

Vehicle i can therefore be described by the following nonlinear state-space representation

$$\dot{x}_i(t) = f(x_i(t), v_{i-1}(t), a_{i-1}^*(t), \xi_i(t)), \quad i \in [1,N], \tag{7}$$

where f is a nonlinear function derived from (3), (5), and (6).

To enable control design using linear tools while accounting for system nonlinearities, the LPV framework is employed. In this framework, the nonlinearities in f are embedded into time-varying parameters, resulting in a system that retains a linear structure but whose state-space matrices depend explicitly on these scheduling parameters. To this end, (7) is reformulated as the following qLPV form, for $i \in [1,N]$:

$$\begin{aligned} \dot{x}_i(t) = &A(v_i(t))x_i(t) + B_{v_{i-1}} v_{i-1}(t) \\ &+ B_{a_{i-1}^*}(v_i(t)) a_{i-1}^*(t) + B_{\xi_i}(v_i(t))\xi_i(t) \end{aligned}, \tag{8}$$

where

$$A(v_i(t)) = \begin{pmatrix} 0 & -1 & -\big(h(v_i(t)) + h'(v_i(t))\big) \\ 0 & 0 & 1 \\ 0 & 0 & -\dfrac{1}{h(v_i(t))} \end{pmatrix}, \quad B_{v_{i-1}} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix},$$

$$B_{a_{i-1}^*}(v_i(t)) = \begin{pmatrix} 0 \\ 0 \\ \dfrac{1}{h(v_i(t))} \end{pmatrix} \quad B_{\xi_i}(v_i(t)) = \begin{pmatrix} 0 \\ 0 \\ \dfrac{1}{h(v_i(t))} \end{pmatrix}.$$

Remark: v_{i-1} and a_{i-1} are assumed to be uncoupled in the control design step.

3. $\mathscr{H}_\infty$ LPV STATE-FEEDBACK CONTROLLER DESIGN

The full availability of the state vector x_i for all $i \in [1,N]$ motivates the use of a state-feedback controller in the upper-

level control layer, which computes the corresponding lower-level input u_i for each vehicle $i \in [1,N]$. This section aims to design the CACC state-feedback controller within a $\mathscr{H}_\infty$ LPV framework, where weighting functions are selected to meet the control objectives. Two LPV methods are proposed: a polytopic approach and a grid-based method, both adapted to the qLPV system structure.

3.1 Control Objectives

The platoon control problem involves two fundamental objectives: minimizing the distance error and ensuring string stability. String stability, which prevents the amplification of disturbances along the vehicle string, is crucial for maintaining smooth traffic flow and avoiding rear-end collisions. A third objective is defined to handle the constraints on the actuator.

Objective 1: Inter-vehicle distance tracking. In the steady state, the distance error must converge to zero. Specifically, when the lead vehicle maintains a constant velocity (i.e., $a_0 = 0$), the following condition should hold:

$$a_0 = 0 \Longrightarrow \lim_{t\to\infty} e_i(t) = 0, \quad i \in [1,N]. \tag{9}$$

Objective 2: String stability. For linear platoon systems with a PF topology, the system is characterized in the frequency domain as strongly strictly stable (according to Feng et al. (2019)) or strictly $\mathscr{L}_2$ string stable (according to Ploeg et al. (2014b)) if the $\mathscr{H}_\infty$ norm of the transfer function from the output of vehicle $i-1$ to the output of vehicle i is less than or equal to 1. Considering a_i as the output of interest for vehicle i, string stability is ensured if the following condition on the induced $\mathscr{L}_2$ norm (equivalent to the $\mathscr{H}_\infty$ norm in the linear time-invariant (LTI) case) holds:

$$\sup_{\|a_{i-1}\|_2 \neq 0} \frac{\|a_i\|_2}{\|a_{i-1}\|_2} \leq 1, \quad i \in [1,N], \tag{10}$$

where $\|\cdot\|_2$ denotes the norm in the space of square-integrable signals $\mathscr{L}_2$.

Objective 3: Actuator Limitations. The desired acceleration input u_i should respect the physical limitations of the actuators to avoid saturation. Specifically, it must not exceed the vehicle's maximum acceleration capability.

3.2 Problem Formulation in the $\mathscr{H}_\infty$ Framework

The proposed $\mathscr{H}_\infty$ control approach is based on the generalized plant $\mathscr{P}_i$ shown in Fig. 2, formed by the previous vehicle model (8) augmented with weighting functions, namely W_{e_i}, W_{a_i}, and W_{ξ_i} chosen to handle the three given performance objectives. As shown, $\mathscr{P}_i$ therefore has two external inputs, namely v_{i-1}, and a_{i-1}, the new control input ξ_i and three controlled outputs, namely $z_{i_1} = W_{e_i} e_i$, $z_{i_2} = W_{a_i} a_i$, and $z_{i_3} = W_{\xi_i} \xi_i$.

Remark: the actual acceleration a_{i-1} is used instead of a^*_{i-1}, since the controller is designed within the $\mathscr{H}_\infty$ framework and $||a_{i-1}||_2 = ||a^*_{i-1}||_2$.

The weighting functions are selected as follows.

- With respect to objective 1 :

$$W_{e_i} = \frac{s/M_{ei} + \omega_{ei}}{s + \omega_{ei} \cdot \varepsilon_{ei}}, \quad i \in [1,N], \tag{11}$$

where M_{ei}, ω_{ei}, and ε_{ei} represent, respectively, the maximum permitted overshoot value for the error, the desired closed-loop tracking error response cut-off frequency, and the maximum permitted steady-state error.

- With respect to objective 2:

$$W_{a_i} = 1, \quad i \in [1,N]. \tag{12}$$

- With respect to objective 3 :

$$W_{\xi_i} = \frac{s + \omega_{\xi_i}/M_{\xi_i}}{\varepsilon_{\xi_i} \cdot s + \omega_{\xi_i}}, \quad i \in [1,N], \tag{13}$$

where M_{ξ_i} depends on saturation constraints, ω_{ξ_i} and ε_{ξ_i} are chosen enough small to limit the effect of measurement noises and numerical issues.

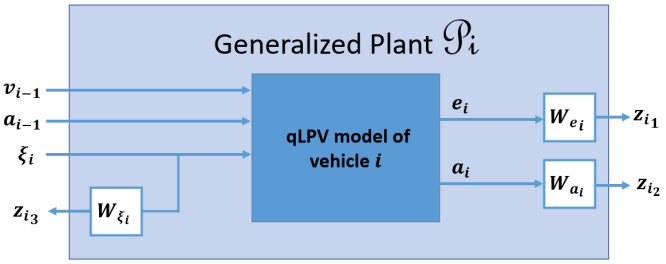

Fig. 2. Generalized plant $\mathscr{P}_i$ associated with vehicle $i \in [1,N]$

Control problem formulation : Given the generalized plant in Fig.2 and a scalar $\gamma_\infty > 0$, the control problem is to find a state feedback control law $\xi_i = F(v_i) x_{\mathscr{P}_i}$ such that the closed-loop system is stable and

$$\sup_{v_i \in [v_{min}, v_{max}]} \sup_{||w_i||_2 \neq 0} \frac{||z_i||_2}{||w_i||_2} < \gamma_\infty, \quad i \in [1,N], \tag{14}$$

where $x_{\mathscr{P}_i} = (x_i, x_{W_{e_i}}, x_{W_{\xi_i}})^T$ is the state of the generalized plant $\mathscr{P}_i$, with $x_{W_{e_i}}$ and $x_{W_{\xi_i}}$, respectively state of W_{e_i} and W_{xi_i}; $z_i = (z_{i_1}, z_{i_2}, z_{i_3})^T$, and $w_i = (v_{i-1}, a_{i-1})^T$ are, respectively, the vector of controlled outputs and external inputs. To obtain an optimal solution γ_∞ must be minimized.

3.3 Polytopic Approach

In this approach, the parameters are defined so that $\mathscr{P}_i$ exhibits an affine dependence on the scheduling vector. Specifically, this vector is constructed, for $i \in [1,N]$, as:

$$\rho_i := \begin{pmatrix} \rho_{i_1}(v_i(t)) \\ \rho_{i_2}(v_i(t)) \end{pmatrix} = \begin{pmatrix} h(v_i(t)) + h'(v_i(t)) \cdot v_i(t) \\ \dfrac{1}{h(v_i(t))} \end{pmatrix}. \tag{15}$$

Since ρ_{i_1} and ρ_{i_2} are bounded for all $v_i \in [v_{\min}, v_{\max}]$, the parameter vector ρ_i evolves within a four-vertex polytope defined by the extreme values of the parameters, i.e., their evaluations at $v_{\min}$ and $v_{\max}$. These extreme values do not depend on the index i, since it is assumed that all vehicles follow the same spacing policy and the controller is designed for a common speed range.

In this context, for $i \in [1,N]$, (15) can therefore be rewritten as the convex combination:

$$\rho_i = \sum_{j=1}^{4} \alpha_{i_j} \cdot \omega_j, \quad \text{with } \alpha_{i_j} \geq 0, \sum_{j=1}^{4} \alpha_{i_j} = 1, \tag{16}$$

and the generalized plant $\mathscr{P}_i$ can be rewritten as

$$\sum_{j=1}^{4} \alpha_{i_j}(\rho_i) \begin{bmatrix} A(\omega_j) & B(\omega_j) \\ C(\omega_j) & D(\omega_j) \end{bmatrix}, \quad \text{with } \alpha_{i_j}(\rho_i) \geq 0, \sum_{j=1}^{4} \alpha_{i_j}(\rho_i) = 1, \tag{17}$$

where $\begin{bmatrix} A(\omega_j) & B(\omega_j) \\ C(\omega_j) & D(\omega_j) \end{bmatrix}$ are the state matrices of $\mathscr{P}_i$ frozen at the vertex ω_j, $j \in [1,4]$.

Offline: the optimal LPV state feedback control law $\xi_i = F(\rho_i)x_{\mathscr{P}_i}$ is computed by solving a convex optimization problem based on Linear Matrix Inequalities (LMIs), evaluated at all vertices of the polytopic model as described in Sename (2025) and Gáspár et al. (2017).

A single, common quadratic Lyapunov function is employed for the entire polytope, which ensures the quadratic stability of the closed-loop system. Since the stability conditions—such as those derived from the Bounded Real Lemma—are convex with respect to the system matrices, verifying them at each vertex of the polytope is sufficient to guarantee that they hold for any convex combination of these vertices. As a result, $\mathscr{L}_2$ stability is ensured for the entire parameter-varying system, under all admissible parameter trajectories.

Online: the controller is computed as a convex combination of the linear state-feedback gains obtained at the vertices of the polytope, according to

$$F(\rho_i) = \sum_{j=1}^{4} \alpha_{i_j}(\rho_i) F_j, \quad i \in [1,N], \tag{18}$$

where $F_j = Y_j P^{-1}$ denotes the state-feedback gain computed at vertex ω_j, $j \in [1,4]$. Here, P is a constant Lyapunov matrix, and Y_j are constant matrices introduced through a change of variables in the LMI formulation used to compute the control gains. The coefficients $\alpha_{i_j}(\rho_i)$ are computed in real time based on the current value of the scheduling parameter $\rho_i(t)$, as detailed in Sename (2025).

Finally, to meet the requirements of the polytopic approach, the control input matrix B_{ξ_i} must be independent of the parameters, which is not the case in (8). Then a strictly proper filter is added to the controller output ξ_i. This filtering step as well as the polytopic approach are described in Apkarian et al. (1995). Note that in this case, the state of the generalized plant $\mathscr{P}_i$ becomes $x_{\mathscr{P}_i} = (x_i, x_{W_{e_i}}, x_{W_{\xi_i}}, x_{f_i})^T$, where x_{f_i} is the state of the filter.

Remark: The parameters ρ_{i_1} and ρ_{i_2} depend exclusively on v_i, which intrinsically links them. Consequently, as will be illustrated in section 6, reducing the size of the polytope can often be advantageous to mitigate the conservatism typically associated with the polytopic approach and contributes to improved closed-loop performance (see Kapsalis et al. (2022)).

3.4 Grid-Based Approach

Unlike the polytopic representation, the grid-based approach does not require an affine dependence of the system matrices on the scheduling parameters. This is an advantage here since a single scheduling variable is needed: $v_i(t)$ for all $i \in [1,N]$. The parameter space $[v_{\min}, v_{\max}]$ is discretized into a finite set of Z grid points, resulting in a collection of frozen LTI models extracted from $\mathscr{P}_i$, each corresponding to a fixed value of the scheduling parameter.

Offline: at each grid point, a local state-feedback control law is obtained by solving a convex optimization problem based on LMIs as described in Sename (2025) and Gáspár et al. (2017).

In this framework, the system is considered exponentially stable if there exists a Lyapunov function that depends on both the system state and the scheduling parameter, and that decreases along the system trajectories. This condition explicitly accounts for the time variation of the scheduling parameter, which is assumed to be bounded. The solution of the LMI problem yields $F(v_i) = Y(v_i)P(v_i)^{-1}$, where $P(v_i)$ is a parameter-dependent Lyapunov matrix and $Y(v_i)$ is a parameter-dependent matrix introduced through a change of variables in the LMI formulation. To solve the LMI problem over the entire gridded domain, the matrices $P(v_i)$ and $Y(v_i)$ are expressed as functions of v_i in order to compute an approximate solution over the infinite-dimensional parameter space.

Online: the complete LPV state-feedback controller is implemented through continuous interpolation of the controller gains between the grid points.

A key distinction from the polytopic approach lies in the nature of the stability guarantees. While the polytopic method ensures quadratic stability over the entire parameter space by verifying the conditions at the vertices of the polytope, the grid-based approach only guarantees stability at the selected grid points with, a priori, no formal guarantees for intermediate parameter values.

4. MICROSCALE VEHICLE MODELING AND SIMULATION ENVIRONMENT

This section presents a simulation environment for a scaled automated vehicle (SAV), inspired by the physical platform developed at GIPSA-Lab, illustrated in Fig. 3, which is a 1:12-scale experimental setup designed for testing control and motion-planning algorithms for autonomous driving applications.

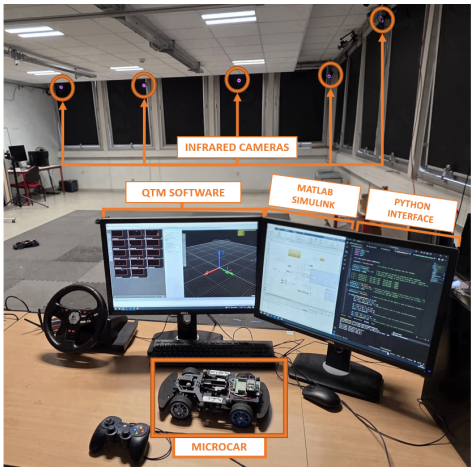

Fig. 3. Microscale platform overview

The platform employs a Qualisys motion capture system composed of 14 infrared cameras operating at 100 Hz, providing real-time position and orientation measurements of the SAVs with sub-millimeter accuracy. Data acquisition is handled by the Qualisys Track Manager (QTM) software, ensuring precise and low-noise tracking of each vehicle's trajectory.

The communication and control loop of the platform are fully managed within MATLAB/Simulink. The control algorithms are executed in real time on a remote PC, which transmits commands to the vehicles via WiFi using the UDP protocol. In return, onboard sensor and motion-capture data from the SAVs are received for feedback. A dedicated Python interface processes the Qualisys motion-capture data and converts it into comprehensive position and orientation information for MATLAB/Simulink.

The vehicle is a modified remote-controlled (RC) car equipped with two brushed DC motors that provide longitudinal traction to the rear wheels, and a servo motor that actuates the steering mechanism of the front wheels. Since the front wheels are not driven, the longitudinal tire force at the front is assumed to be zero. The total longitudinal tire force at the rear is modeled as the sum of the forces generated by the left and right rear wheels, denoted F_{xrl} and F_{xrr}, respectively.

Given that the microcar operates on a nearly horizontal surface, gravitational effects are neglected. Additionally, the rolling resistance forces acting on both the front and rear wheels are lumped into a single equivalent resistance force, denoted R_x. Aerodynamic drag is also neglected, which is a reasonable assumption due to the vehicle's small size and the controlled indoor environment of the platform.

Applying these assumptions to the longitudinal vehicle model given in Rajamani (2012), the longitudinal dynamics of the microcar can be described by:

$$m\dot{v}_x = F_{xrl} + F_{xrr} - R_x \tag{19}$$

where m is the mass of the microcar and v_x is the longitudinal microcar velocity.

Under the assumption of small slip conditions, the longitudinal tire forces can be modeled using linear relationships:

$$F_{xrl} = C_{\sigma_l} \cdot s_l, \quad F_{xrr} = C_{\sigma_r} \cdot s_r, \tag{20}$$

where C_{σ_l} and C_{σ_r} are the longitudinal stiffness coefficients, and s_l and s_r, described by (21), denote the slip ratios of the left and right rear tires, respectively:

$$s_l = \frac{R\omega_l - v_x}{v_x}, \quad s_r = \frac{R\omega_r - v_x}{v_x} \tag{21}$$

with ω_l and ω_r, angular velocity for the left (l) and right (r) wheels respectively; R is the tire radius.

The rolling resistance is defined as :

$$R_x = C_r \cdot m \cdot g, \tag{22}$$

where C_r is the rolling resistance coefficient and g is gravitational acceleration.

Additionally, the rotational dynamics of the driven rear wheels are given by:

$$J_w\dot{\omega}_l = \Gamma_l - R \cdot F_{xrl}, \quad J_w\dot{\omega}_r = \Gamma_r - R \cdot F_{xrr} \tag{23}$$

where Γ_l and Γ_r denote the applied torque, for the left (l) and right (r) wheels respectively; J_w is the wheel moment of inertia.

Finally, the motor dynamics are modeled using the classical electromechanical equations of a DC motor. Both motors are torque-controlled in a closed-loop configuration, with the reference torques denoted as Γ_l^{ref} and Γ_r^{ref}. The parameter estimation process, not detailed here since it is not the core of this study, was performed using data obtained from the experimental platform. The overall simulator configuration is presented in Fig.4.

A microcar platoon can then be simulated by replicating the individual microcar simulator multiple times. Finally, each microcar in the simulator is equipped with a simple longitudinal cruise controller, as implemented on the real microcars of the platform. A proportional-integral (PI) controller is used to ensure that each vehicle follows its longitudinal dynamic equation, as defined in (3), with a distinct time constant τ_i, enabling heterogeneous vehicle responses within the platoon.

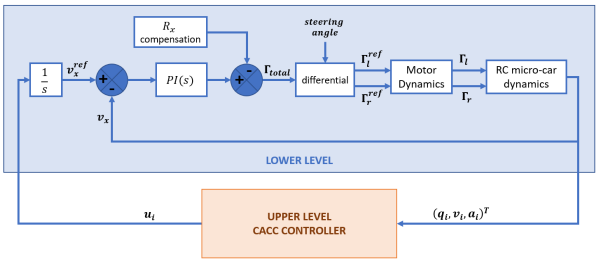

Fig. 4. Microcar simulator overview

5. APPLICATION TO A GIVEN SPACING POLICY FOR MICROSCALE PLATOON.

In this section, a specific case in which the time headway is defined as an affine function of the vehicle velocity, given by (24), is considered.

$$h(v_i(t)) = \lambda_1 + \lambda_2 \cdot v_i(t), \quad (\lambda_1, \lambda_2) \in \mathbb{R}^2, \quad i \in [1, N], \tag{24}$$

where the parameters λ_1 and λ_2 are selected such that the time headway satisfies

$$h(v_{\min}) = 0.5\,\text{s} \quad \text{and} \quad h(v_{\max}) = 1\,\text{s},$$

with $v_i \in [2, 5]$ m/s representing the operating velocity range of the microscale platoon. This spacing policy is not intended to be optimal; rather, it serves to validate the performance of the proposed LPV state feedback controller under a realistic and representative headway strategy.

5.1 Polytopic Approach: Reduction of the Polytope

Under the spacing policy described by (24), (15) becomes:

$$\rho_i = \begin{pmatrix} \rho_{i_1}(v_i(t)) \\ \rho_{i_2}(v_i(t)) \end{pmatrix} = \begin{pmatrix} 2 \cdot \lambda_2 \cdot v_i(t) + \lambda_1 \\ (\lambda_1 + \lambda_2 \cdot v_i(t))^{-1} \end{pmatrix}, \quad i \in [1, N]. \tag{25}$$

Since ρ_{i_1} and ρ_{i_2} are not independent, it may be advantageous to reduce the number of vertices in the parameter polytope from four to three. This reduction helps eliminate unreachable vertices, thus reducing the conservatism of the LPV model and the computational complexity of the controller synthesis, while improving closed-loop performance. The reduction is performed by first computing the tangents to the curve (ρ_{i_1}, ρ_{i_2}) at the two extreme (reachable) boundary points. The third vertex is then defined as the intersection point of these tangents, as illustrated in Fig.5, resulting in the smallest convex triangle that fully encloses the parameter trajectory.

Offline: the controller is synthesized solving an LMI problem considering only the three vertices of the reduced polytope ω_A, ω_B, and ω_C. The solution of the LMI problem using YALMIP (Lofberg (2004)) and SeDuMi (Sturm (2024)) yields a value of $\gamma_\infty = 8.48$ and $\gamma_\infty = 8.73$ for the polytopic approach with reduced and full polytope respectively.

Online: the convex coefficients α_{i_A}, α_{i_B}, and α_{i_C} are determined by solving an RLS algorithm as described in Kapsalis et al. (2022).

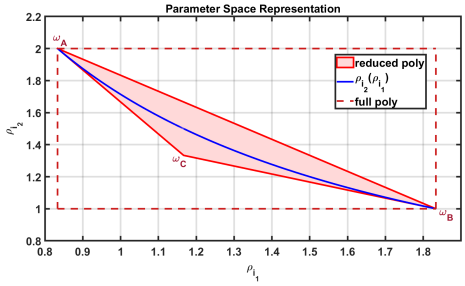

Fig. 5. Representation of the scheduling trajectory in the (ρ_{i_1}, ρ_{i_2}) plane for $v_i \in [2,5] m/s$.

5.2 *Grid-Based Approach :* LPV TOOLS *(Pfifer and Burgin, 2025)*

As explained in Section 3.4, $P(v_i)$ and $Y(v_i)$ are parameter-dependent matrices constructed, for $i \in [1,N]$, as:

$$P(v_i) = P_0 + P_1 v_i + P_2 \frac{1}{v_i}, \quad Y(v_i) = Y_0 + Y_1 v_i + Y_2 \frac{1}{v_i},$$

where P_0, P_1, P_2, Y_0, Y_1, and Y_2 are constant matrices determined by solving the LMI problem. This form is chosen to effectively capture the system nonlinearities represented in the qLPV formulation. Moreover, to solve the LMI problem, the maximum acceleration capability of the microcars is taken into account and is limited to the range $[-0.8, 0.8]$ m/s^2.

The grid-based approach has been implemented using LPV TOOS 2.0 (Pfifer and Burgin, 2025). Several tests were conducted to determine an appropriate number of grid points, with Z varying from 2 to 10. Approximately the same value of $\gamma_\infty \approx 6$ was obtained as the solution to the corresponding LMI problem for all $Z \in [2,10]$. Note that in LPV TOOLS 2.0, γ is first minimized and then adjusted using a back-off factor, whose default value is 1.2.

However, while the frequency-domain analysis did not provide a clear criterion, the time-domain responses for $Z \geq 4$ revealed some undesired irregularities in the controller output, likely due to minor numerical issues. Consequently, three grid points are selected at $v_i = 2$m/s, $v_i = 3.5$m/s, and $v_i = 5$m/s. Note that, in this configuration, the polytopic approach with a reduced polytope and the grid-based approach exhibit comparable computational complexity, since the controller synthesis is performed at only three operating points in both cases.

5.3 *Frequency Domain Analysis*

The bode magnitude of the sensitivity functions corresponding to the above three objectives and related to the external input a_{i-1}, namely $\frac{e_i}{a_{i-1}}$, $\frac{a_i}{a_{i-1}}$, and $\frac{\xi_i}{a_{i-1}}$ are shown in Fig. 6 for both the polytopic and the grid-based approaches. For the polytopic approach, the sensitivity functions are evaluated at each vertex of the reduced polytope while for the grid-based approach, they are computed at each grid point. The magnitude of the sensitivity functions are compared with the desired performance templates, given by the inverses of the weighting functions (11), (12), and (13).

Figs. 6a (polytopic approach with reduced polytope) and 6b (grid-based approach) show that the magnitude of the input-output relationship between e_i and a_{i-1} remains well below the desired template for each vertex of the reduced polytope

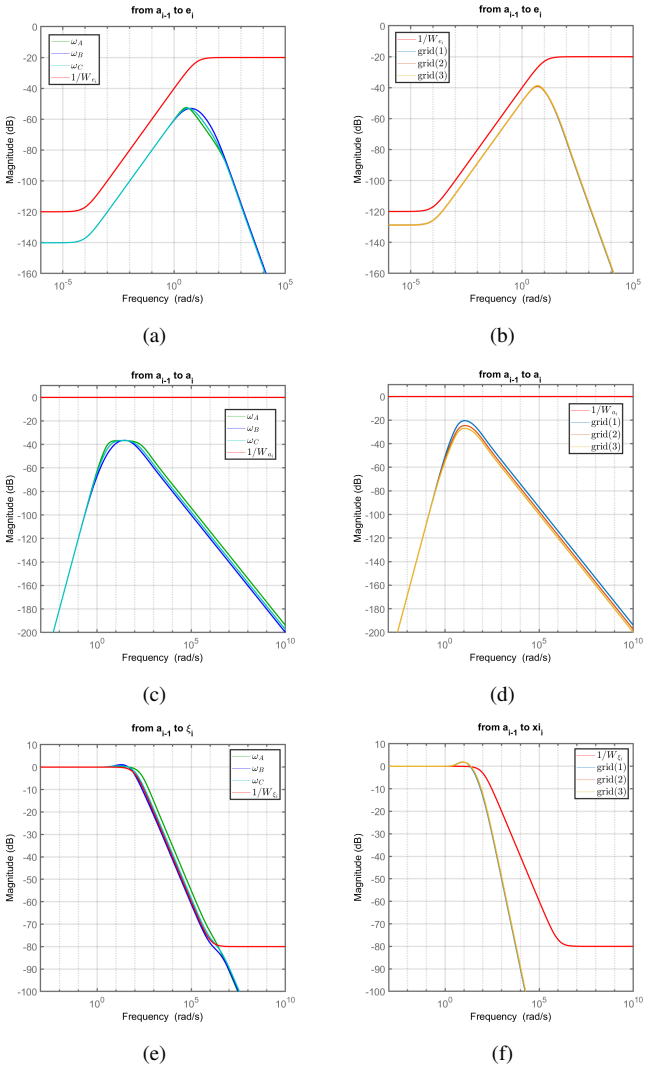

Fig. 6. Bode magnitude of the sensitivity functions related to a_{i-1} in the frequency domain for both LPV controller design approaches: (left column) reduced polytopic approach and (right column) grid-based approach. Each row corresponds to a performance objective: (a,b) objective 1, (c,d) objective 2, and (e,f) objective 3.

(respectively for each grid point), ensuring that the constraints in terms of bandwidth, maximal overshoot, and steady state are not violated.

Moreover, as shown in Figs. 6c and 6d, condition (10) is satisfied, thereby ensuring that string stability is achieved over the entire frequency range for both LPV control design approaches.

Finally, as depicted in Fig. 6e, for the polytopic approach, the magnitude of the sensitivity function $\frac{\xi_i}{a_{i-1}}$, remains below its associated template at high frequencies. In the mid-frequency range, a slight violation of the template occurs, although the response closely follows its shape; such deviation is minor and acceptable from a practical standpoint. For the grid-based approach, as shown in Fig. 6f, the magnitude generally satisfies the template over the entire frequency range, with only a small deviation observed around 10 rad/s, which also remains within acceptable limits.

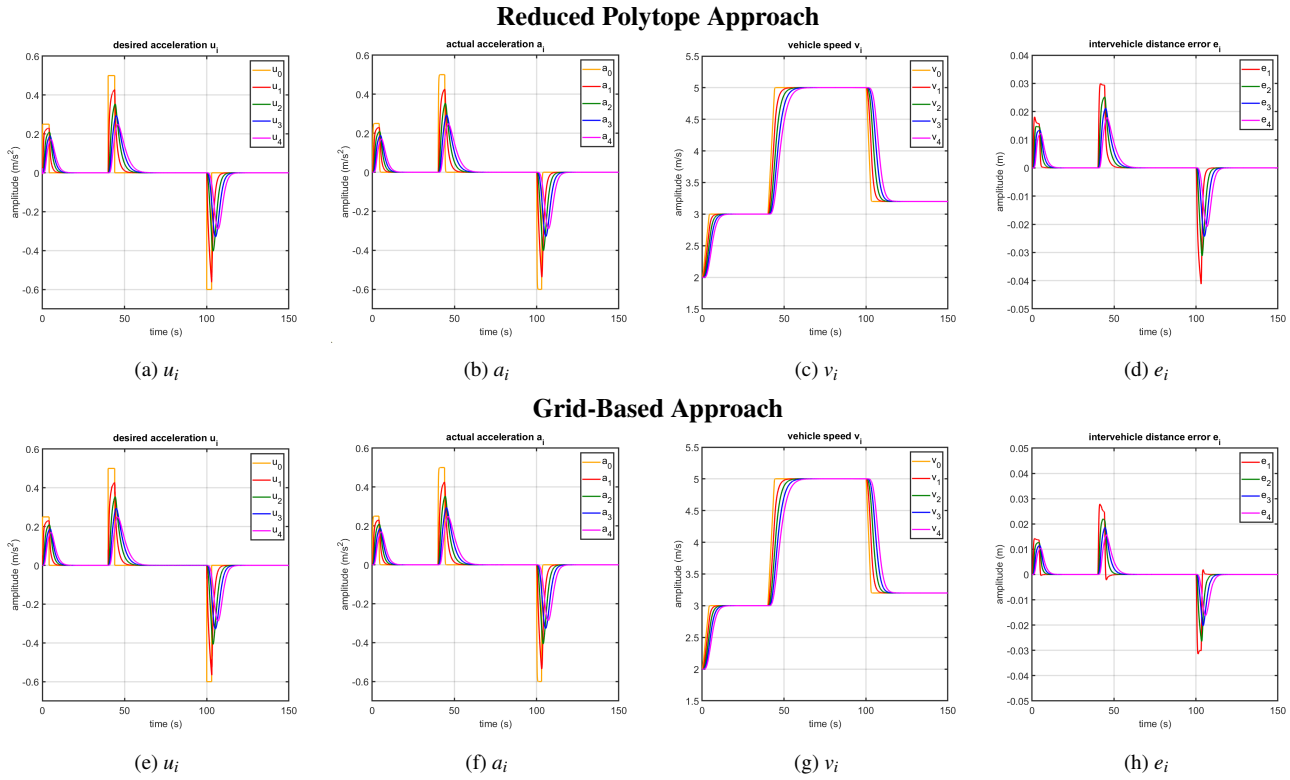

Fig. 7. Time-domain simulation results for both LPV controller design approaches: reduced polytope (top) and grid-based (bottom). Subplots show the evolution of the desired accelerations u_i, actual accelerations a_i, velocities v_i, and tracking errors e_i.

In conclusion, the frequency domain analysis confirms that the three control objectives are, *a priori*, achieved, and yields qualitatively similar results regardless of the method employed—either the reduced polytope or the grid-based approach—although the grid-based method achieves a better γ_∞ value.

6. TIME-DOMAIN SIMULATIONS ON MICROSCALE PLATOON SIMULATOR

A heterogeneous microscale platoon of five microcars is considered, the simulation scenario is as follows. The leader is acceleration controlled while the others are CACC-equipped. $\tau_0 = 0.5s$, $\tau_1 = 0.9s$, $\tau_2 = 0.4s$, $\tau_3 = 0.6s$, and $\tau_4 = 0.8s$ reflect, respectively, the engine dynamics of microcar $i \in [0,4]$. The desired acceleration profile of the leader is described by

$$u_0(t) = \begin{cases} 0.25 & if \quad 0,1 < t \leq 4,1 \\ 0.5 & if \quad 40 < t \leq 44 \\ -0.6 & if \quad 100 < t \leq 103 \\ 0 & otherwise. \end{cases} \tag{26}$$

The communication delay $\theta_i = 0.02s$ is assumed to be the same for all i. The initial velocity and acceleration are set to $v_i(0) = v_{\min}$ and $a_i(0) = 0$ m/s^2, respectively, for all vehicles $i \in [0,4]$. The remaining states of $x_{\mathscr{P}_i}$ must be carefully initialized so that $x_{\mathscr{P}_i}$ satisfies the closed-loop system equations when evaluated at $v_i = v_i(0)$ in order to avoid unexpected transient behavior or numerical issues at $t = 0$s.

The simulation results in the time domain for both LPV control approaches — the reduced polytope method and the grid-based method — are presented in Fig.7. All relevant system variables, namely the desired acceleration u_i, the actual acceleration a_i, the vehicle velocity v_i, and the distance tracking error e_i, are depicted for each vehicle to comprehensively assess the dynamic behavior of each control strategy. For comparative analysis, the root mean square error (RMSE), computed from the distance tracking error e_i over the entire simulation time for each vehicle and each LPV control design approach—namely, the full polytope, reduced polytope, and grid-based methods—is presented in Table 1. To further support the analysis, the corresponding absolute minimum and maximum peak values of the distance tracking error e_i are reported in Tables 2 and 3, respectively. The minimum values are extracted during the braking phase ($100 < t \leq 103$), while the maximum values are obtained during the acceleration phase ($40 < t \leq 44$).

Figs. 7a and 7e confirm that the controller outputs result in achievable desired acceleration profiles. Figs. 7b and 7f illustrate the string-stable behavior of the platoon; the acceleration amplitudes decrease along the vehicle string and converge to zero at steady state. As shown in Figs. 7c and 7g, the velocity profiles remain smooth, even under sudden acceleration or braking of the leader. Figs.7d and 7h illustrate that the intervehicle distance tracking error converges to zero despite the presence of disturbances.

While the overall time-domain responses are similar for both LPV control strategies—the reduced polytope and the grid-based methods—a notable difference can be observed in the transient behavior of the tracking error. Specifically, the peak values (absolute minimum and maximum) are higher for the reduced polytope approach than for the grid-based method. Moreover, the RMSE values computed on e_i are also larger for the polytopic approach than for the grid-based one. It is also worth noting that the full polytope yields even higher peak and RMSE values than the reduced polytope, confirming the benefit

of polytope reduction. Finally, both the RMSE and the peak values show a gradual decrease along the platoon.

Table 1. RMSE computed on e_i over the entire simulation time.

Method	Vehicle 1	Vehicle 2	Vehicle 3	Vehicle 4
Full poly	0.0086	0.0072	0.0063	0.0057
Reduced poly	0.0084	0.0069	0.0059	0.0054
Gridding	0.0082	0.0068	0.0058	0.0050

Table 2. Absolute minimum of e_i during the braking phase ($100 < t \leq 103$).

Method	Vehicle 1	Vehicle 2	Vehicle 3	Vehicle 4
Full poly	0.0429	0.0325	0.0253	0.0217
Reduced poly	0.0412	0.0312	0.0243	0.0208
Gridding	0.0314	0.0265	0.0202	0.0162

Table 3. Maximum of e_i during the acceleration phase ($40 < t \leq 44$).

Method	Vehicle 1	Vehicle 2	Vehicle 3	Vehicle 4
Full poly	0.0311	0.0262	0.0221	0.0186
Reduced poly	0.0299	0.0252	0.0212	0.0179
Gridding	0.0279	0.0220	0.0186	0.0158

7. CONCLUSION

In this work, a state-feedback CACC controller for a heterogeneous vehicle platoon—accounting for variations in driveline dynamics—was designed using a velocity-dependent spacing policy and a PF communication topology. The control problem was formulated within the $\mathscr{H}_\infty$/LPV framework and addressed using both polytopic and grid-based synthesis approaches.

Frequency-domain analysis and time-domain simulations on a microscale platoon simulator confirmed satisfactory controller performance in terms of inter-vehicle distance tracking and string stability. The grid-based approach appears to provide better overall performance. However, a compromise must be considered regarding the stability guarantees, which differ between the two methods.

Future work will focus on the discrete-time implementation of the proposed control strategies on the microcar platooning platform, in order to further validate the obtained simulation results under real-time conditions.

ACKNOWLEDGEMENTS

We warmly thank Peter Seiler and Harald Pfifer for the help in implementing the new LPV TOOS v2.0.

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

During the preparation of this work the author(s) used ChatGPT to assist with linguistic improvements. After using this tool, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

REFERENCES

Apkarian, P., Gahinet, P., and Becker, G. (1995). Self-scheduled H-inf control of linear parameter-varying systems: A design example. *Automatica*, 31(9), 1251–1261.

Feng, S., Zhang, Y., Li, S.E., Cao, Z., Liu, H.X., and Li, L. (2019). String stability for vehicular platoon control: Definitions and analysis methods. *Annual Reviews in Control*, 47, 81–97.

Flores, C., Milanes, V., and Nashashibi, F. (2017). A time gap-based spacing policy for full-range car-following. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, 1–6. IEEE, Yokohama.

Gao, F., Li, S.E., Zheng, Y., and Kum, D. (2016). Robust control of heterogeneous vehicular platoon with uncertain dynamics and communication delay. *IET Intelligent Transport Systems*, 10(7), 503–513.

Gáspár, P., Szabó, Z., Bokor, J., and Nemeth, B. (2017). *Robust Control Design for Active Driver Assistance Systems: A Linear-Parameter-Varying Approach*. Advances in Industrial Control. Springer International Publishing, Cham.

Kapsalis, D., Sename, O., Milanes, V., and Molina, J.J. (2022). A reduced lpv polytopic look-ahead steering controller for autonomous vehicles. *Control Engineering Practice*, 129, 105360.

Lefeber, E., Ploeg, J., and Nijmeijer, H. (2020). Cooperative Adaptive Cruise Control of Heterogeneous Vehicle Platoons. *IFAC-PapersOnLine*, 53(2), 15217–15222.

Lofberg, J. (2004). YALMIP: A toolbox for modeling and optimization in MATLAB. In *Computer Aided Control Systems Design, 2004 IEEE International Symposium On*, 284–289. IEEE.

Pfifer, H. and Burgin, E. (2025). Lpvtools 2.0 and its application to spacecraft attitude control. *IFAC-PapersOnLine*, 59(15), 1–6.

Ploeg, J., Shukla, D.P., Van De Wouw, N., and Nijmeijer, H. (2014a). Controller Synthesis for String Stability of Vehicle Platoons. *IEEE Transactions on Intelligent Transportation Systems*, 15(2), 854–865.

Ploeg, J., Van De Wouw, N., and Nijmeijer, H. (2014b). Lp String Stability of Cascaded Systems: Application to Vehicle Platooning. *IEEE Transactions on Control Systems Technology*, 22(2), 786–793.

Rajamani, R. (2012). *Vehicle Dynamics and Control*. Mechanical Engineering Series. Springer US, Boston, MA.

Sename, O. (2025). *Linear Parameter-Varying Control : Theory and Apllication to Automotive Systemes*. Wiley.

Sturm, J.F. (2024). SeDuMi: A software package for solving optimization problems over symmetric cones.

Wang, J., Zhao, W., Min, H., Huang, Y., and Liu, C. (2025). String stability of the heterogeneous vehicle platoon considering connectivity uncertainty and autonomous driving levels under variable time headway strategy. *Transportation Letters*, 1–22.

Wang, Z., Wu, G., and Barth, M.J. (2018). A Review on Cooperative Adaptive Cruise Control (CACC) Systems: Architectures, Controls, and Applications. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2884–2891. IEEE, Maui, HI.

Zhou, J. and Peng, H. (2005). Range Policy of Adaptive Cruise Control Vehicles for Improved Flow Stability and String Stability. *IEEE Transactions on Intelligent Transportation Systems*, 6(2), 229–237.

Zhuang, X., Zhang, J., Tian, J., Cui, F., and Wang, T. (2024). Variable time headway spacing strategy for connected vehicles platoon based on sliding mode control. *Physica A: Statistical Mechanics and its Applications*, 637, 129588.