Design and Experimental Validation of an LPV Pure Pursuit Automatic Steering Controller

Dimitrios Kapsalis, Olivier Sename, Vicente Milanes, John Jairo Martinez Molina

Design and Experimental Validation of an LPV Pure Pursuit Automatic Steering Controller

Dimitrios Kapsalis [*,**] **Olivier Sename** [**] **Vicente Milanes** [*]
John Jairo Martinez Molina [**]

[*] *Research Department, Renault SAS, 1 Avenue de Golf, 78280 Guyancourt, France. (e-mail: dimitrios.kapsalis@renault.com; vicente.milanes@renault.com).*
[**] *Univ. Grenoble Alpes, CNRS, Grenoble INP*, GIPSA-Lab, 38000 Grenoble, France. *Institute of Engineering Univ. Grenoble Alpes (e-mail:dimitrios.kapsalis@grenoble-inp.fr; olivier.sename@grenoble-inp.fr; john-jairo.martinez-molina@grenoble-inp.fr)*

Abstract: This paper addresses the problem of designing control systems for steering an autonomous vehicle, for varying speed, ensuring stability and sustaining performance, taking into account comfort and safety. A novel automatic lateral control architecture is proposed combining Linear Parameter Varying (LPV) control theory and the Pure-Pursuit algorithm. The response of the robust controller is investigated for tracking a reference trajectory. The implementation is based on the reduction of the polytopic LPV model. In addition, to illustrate the results, the synthesized controller has been tested first in simulations, and then to an automated electric Renault Zoe in a test track providing encouraging results.

Keywords: Automotive Control, Connected and Automated Vehicles.

1. INTRODUCTION

Autonomous vehicles could facilitate modern life reducing accidents rate, average transportation time and fuel consumption; see Bagloee et al. (2016). To achieve these goals, vehicles have to be able to accomplish several scenarios to reach full autonomy.

From the control perspective, the main categories can be divided into: Lateral control, Longitudinal control, Integrated Lateral/Longitudinal control and higher-level control issues; see González et al. (2015). The addressed issue for this paper is Lateral Control.

Numerous efforts have been devoted to automate the steering control of a vehicle. Since 90s, automatic lateral control has been studied by the California Path Program at the University of Berkeley. They introduced a new dynamic vehicle model and applied an optimal control law-based controller, including a feed forward term as a function of the road curvature of the trajectory in Peng and Tomizuka (1990).

In the 90s, Ackermann et al. (1999) made a breakthrough decoupling the yaw and lateral dynamics of the vehicle by active steering control. In 1995, Jochem and Pomerleau (1996) from Carnegie Mellon University demonstrated Navlab 5, a test-bed car that completed about 2800 miles journey across America. The vehicle wasn't driving fully autonomously because the pedals were manually controlled. The steering wheel control was based on the pure-pursuit algorithm. The lane-tracking was achieved by minimizing lateral deviation and heading error in a look-ahead distance ahead of the vehicle.

In 1996, Broggi et al. (1999) demonstrated the Argo project. The main target of the project was the autonomous driving of the vehicle through Italy. The steering system was using only two low-cost cameras and was controlled by a gain-scheduled proportional controller to correct the vehicle's offset with respect to the reference trajectory.

In 2005, Stanford presented the Stanley vehicle which won the second Defense Advanced Research Projects Agency (DARPA) Grand Challenge that took place in the desert; see Thrun et al. (2006). Its steering controller was a nonlinear feedback function. The steering system's main purpose was to minimize the cross-track error and secondly correct the heading of the vehicle.

In 2013, Tan and Huang (2014) conducted real experiments with drivers of different levels and tried to formulate the steering mechanism of the humans. They resulted in a PID controller, the T&C controller (Target and Control Driver Steering Model). The controller's command is the steering rate which is the angular error (error between the vehicle's heading and the target point's at a look-ahead distance) multiplied by a gain. The gain could be selected through pole placement and the system was validated in a bus revenue service.

Several control methods have been implemented and tested experimentally for the design of automatic steering wheel systems. Some of them are cascade control architecture combined with fuzzy logic in Pérez et al. (2011), as well as adaptive and predictive control e.g. Ercan et al. (2017). Furthermore, Corno et al. (2020) have implemented successfully LPV control theory and recently reinforcement learning and deep-based lateral control strategy has demonstrated promising results in Li et al. (2019).

⋆ This work was supported by Renault.

1.1 Motivations & Contributions

The above brief review allows us to emphasize three main pillars of Lateral control. First, the control theory applied to the vehicle model i.e Gain scheduling, MPC, Adaptive control theory, fuzzy logic etc. Second, the strategy for the formalization of the model such as T&C, Pure Pursuit or Stanley method. Third, the control objectives to be satisfied for better results i.e the minimization of the key variables, such as the yaw rate error between the vehicle and the reference trajectory, the angular error of the vehicle etc.

The main motivation behind this study is the use of LPV theory for the design over a range of speed, allowing a smooth transition for speed variations. Indeed, even if stability is guaranteed, transient effects of switching between controllers can become important for the system performance.

LPV techniques offer capabilities to handle the whole operating domain of a varying parameter with respect to time. In automotive problems where parameters variations as speed exist and play a significant role, LPV can provide stability and performance against them; see Sename et al. (2013). Finally, LPV problems are convex and amenable to Linear Matrix Inequality (LMI) computations, the latter being supported by efficient and reliable software tools.

This paper proposes a novel architecture combining LPV control theory, the Pure-Pursuit strategy and the control objectives by minimizing: 1) the lateral offset at a given look-ahead distance, 2) the angular error between the vehicle and the reference trajectory at that distance and 3) the yaw rate difference between the yaw rate of the vehicle and the reference trajectory.

1.2 Structure of the paper & Vehicle Parameters

This paper is organized as follows. Sections 2 introduces the vehicle model and the errors to be minimized according to the Pure-Pursuit strategy. Section 3 presents the LPV synthesis of the controller, according to the polytopic approach, including the polytope reduction. In section 4 is presented the simulation results and in section 5 the experimental validation of the proposed algorithm to an electric Renault Zoe. Some concluding remarks and future work directions are given at the end of the paper.

The following notations will be referred in the rest of the paper expressing useful information about the vehicle, as well as for the model used. Throughout the paper, the following notation will be adopted: $\{f,r\}$ are used to identify vehicle front and rear positions respectively. $\{x,y\}$ holds for dynamics in the longitudinal and lateral axes respectively. Then, let $v = \sqrt{v_x^2+v_y^2}$ denote the vehicle speed and K is the road curvature. y_L is the offset from the centerline at the look-ahead, ε_L is the angle between the tangent to the road and the vehicle orientation and $\dot{\psi}$ is the yaw rate of the vehicle (see Fig. 1). In addition, L and T denote the look-ahead distance and time (Fig. 2). m is the vehicle mass, I_z is the yaw inertia, L_f COG-front distance, L_r the COG-rear distance, C_f cornering stiffness of each front tire and C_r cornering stiffness of each rear tire.

2. VEHICLE AND ERRORS MODELING

Combining the lateral dynamics derived from the bicycle model of Rajamani (2011), as shown in Fig. 1, and the geometry from the pure pursuit algorithm; see Taylor et al. (1999) in Fig. 2 for the design of an automatic steering controller, the following extended state space model is obtained.

2.1 Lateral Dynamics

The bicycle model, depicted in Fig. 1, is used with states the lateral speed v_y, the yaw-rate $\dot{\psi}$ of the vehicle and as input the steering wheel angle δ. The corresponding equations are presented below:

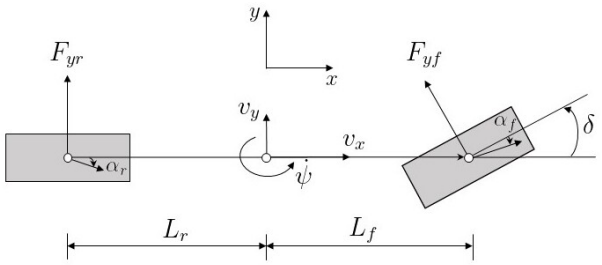

Fig. 1. A two wheeled bicycle model describing the lateral dynamics

$$\begin{aligned}\dot{v}_y &= \left(-\frac{C_f+C_r}{mv_x}\right)v_y + \left(-v_x+\frac{C_rL_r-C_fL_f}{mv_x}\right)\dot{\psi}+\frac{C_f}{m}\delta \\ \ddot{\psi} &= \left(\frac{-L_fC_f+L_rC_r}{I_zv_x}\right)v_y + \left(-\frac{L_f^2C_f+L_r^2C_r}{I_zv_x}\right)\dot{\psi}+\frac{L_fC_f}{Iz}\delta\end{aligned} \tag{1}$$

2.2 Error Dynamics

Navigation provides at each instant a generated trajectory taking into account the speed, heading and current position of the vehicle. That reference trajectory is discretized to a set of points that each one of them contains as information the coordinates and the heading. The lateral y_L and the angular ε_L errors are calculated as the projection of the vehicle pose at the look-ahead distance onto the generated trajectory, Fig.2. The equations which captures the evolution of these variables at the look-ahead distance are the following:

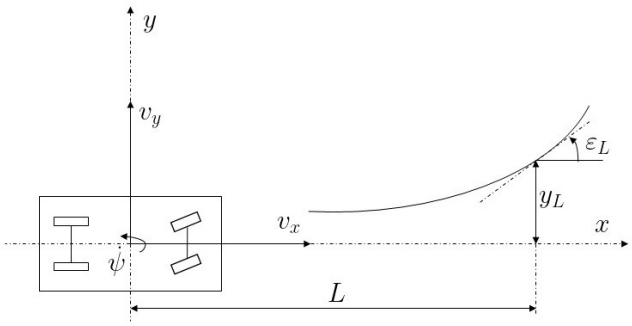

Fig. 2. Pure pursuit geometry describing the errors created at the target point.

$$\begin{aligned} \dot{y}_L &= -v_y - L\dot{\psi} + v_x \varepsilon_L \\ \dot{\varepsilon}_L &= -\dot{\psi} + v_x K \end{aligned} \tag{2}$$

where K is the road curvature of the trajectory at the target point. The extended state space model created is the following:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 w(t) + B_2 u(t) \\ y(t) &= Cx(t) \end{aligned} \tag{3}$$

$$x(t) = \begin{bmatrix} v_y \\ \dot{\psi} \\ y_L \\ \varepsilon_L \end{bmatrix}, B_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}, B_2 = \begin{bmatrix} \frac{C_f}{m} \\ \frac{L_f C_f}{Iz} \\ 0 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A = \begin{bmatrix} -\frac{C_f + C_r}{m v_x} & -v_x + \frac{C_r L_r - C_f L_f}{m v_x} & 0 & 0 \\ \frac{-L_f C_f + L_r C_r}{I_z v_x} & -\frac{L_f^2 C_f + L_r^2 C_r}{I_z v_x} & 0 & 0 \\ -1 & -L & 0 & v_x \\ 0 & -1 & 0 & 0 \end{bmatrix},$$

and $w(t) = \dot{\psi}_{traj} = v_x K$ is considered as an exogenous input.

3. LPV PURE PURSUIT CONTROLLER SYNTHESIS

3.1 LPV Model

The above state space representation (3) can be written in an LPV form by expressing the parameter-dependent matrix $A(v_x, L)$ in an LPV form, with v_x and L as parameters, known at each instant and bounded, and vary with respect to time. Therefore, defining the vector of varying parameters $\rho^T = [v_x \; L]^T$ we obtained that the state matrix is of the form

$$A(\rho) = \rho_1 A_{\rho_1} + \frac{1}{\rho_1} A_{\frac{1}{\rho_1}} + \rho_2 A_{\rho_2} \tag{4}$$

In order to be able to express the model system in a proper LPV form for the chosen design method (here the polytopic one); see Mohammadpour and Scherer (2012), the matrices that depend on time-varying parameters must be affine with respect to the the vector of varying parameters. From (4), $A(\rho)$ is not affine with respect to the parameter vector ρ. To get such an affine form, it is necessary to define $\frac{1}{v_x}$ as a new additional parameter, which leads to a vector of 3 varying parameters.

It is known that such a choice may increase the conservatism due to the larger convex set to be considered during the optimisation process. However, as seen later, some reduction of the polytopic set might be considered.

On the other hand, the look-ahead time T is chosen constant to further reduce the size of polytope, by considering $L = v_x T$ which is a coherent choice for the path tracking control. Therefore the vector of varying parameters is considered as:

$$\rho^T = [v_x \; \frac{1}{v_x}]^T \tag{5}$$

Considering as bounds of the parameters $\underline{\rho_i} \leq \rho_i \leq \overline{\rho_i}$, the LPV matrix $A(\rho)$ can be written as:

$$A(\rho) = \sum_{i=1}^{4} a_i(\rho) A_i \tag{6}$$

where 4 are the number of vertices ω_i of the polytope i.e all the possible combinations of the lower and upper bounds $\underline{\rho_i}$ and $\overline{\rho_i}$. Additionally, $a_i(\rho)$ are parameter-dependent variables computed on-line, as explained later for the on-line implementation of the LPV controller.

Replacing the LPV matrix (6) in (3), the polytopic system, by Apkarian et al. (1995), is:

$$\begin{aligned} \dot{x}(t) &= A(\rho)x(t) + B_1 w(t) + B_2 u(t) \\ y(t) &= Cx(t) \end{aligned} \tag{7}$$

3.2 Reduction of the Polytope

It is worth to mention that the parameters defined previously depend actually on the same variable v_x. This may lead to a conservatism using the polytopic approach (this is indeed a drawback of such an approach; see Robert et al. (2009)). As a result, the performances of the closed-loop system may be degraded or even the controller might not be implementable.

The modification presented below is to define the LPV controller based on a reduced polytope that expresses at least the possible combinations of the bounds of the parameters (see Fig 3). The operating domain of speed is considered $v_x \in [1, 20] m/s$, so the bounds of the parameters ρ_i are the following:

$$\begin{aligned} 1 \leq \rho_1 \leq 20 \\ 0.05 \leq \rho_2 \leq 1 \end{aligned} \tag{8}$$

The two realistic vertices of the polytope (see left subfigure in Fig.3) are ω_2 and ω_4 but in order to keep the convexity property and solve the equivalent LMIs, the reduced polytope should contain all the intermediate combinations of the parameters; see Boyd et al. (1994). For that reason, the third selected vertex for the reduced polytope is ω_1 (right subfigure in Fig.3).

The whole procedure for the reduction of a polytope is detailed in Robert et al. (2009), where the authors implemented the synthesized controller based on a reduced polytope to a T inverted pendulum.

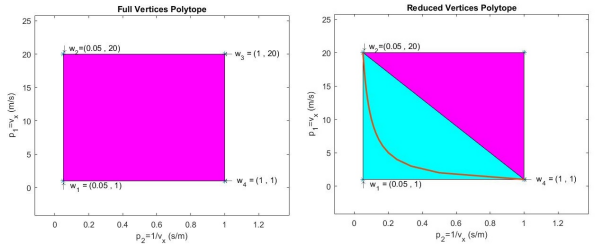

Fig. 3. Full and Reduced Conservatism Polytopes

3.3 Control Structure and Synthesis

Control problem formulation in the LPV framework is depicted in Fig. 4 where the LPV controller's performance is achieved by tuning the weighting functions; see Poussot-Vassal (2008). The MIMO generalized plant considers the LPV extended model (7) defining the vector of variables z that have to be minimized in order to achieve the desired performance.

The z variables are formed by introducing some appropriate weight functions $W_u(s,\rho), W_e(s,\rho)$ to the measurements $e_{\dot{\psi}} = \dot{\psi}_{ref} - \dot{\psi}$, y_L, ε_L as to the output of the controller u as shown in Fig. 4.

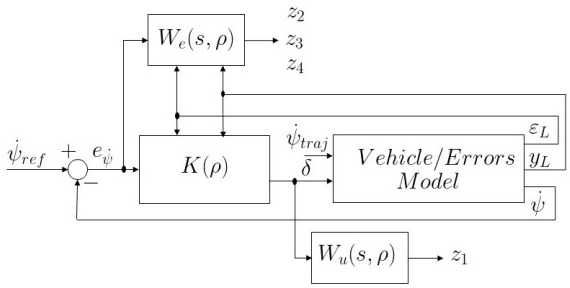

Fig. 4. Control Scheme

The augmented state vector x_g consists of the states of the extended model (3) plus the states x_z, which express the dynamics of the weighting functions i.e $x_g(t) = \begin{bmatrix} x(t) \\ x_z(t) \end{bmatrix}$.
The generalized LPV plant is:

$$\begin{bmatrix} \dot{x}_g(t) \\ y(t) \\ z(t) \end{bmatrix} = \begin{bmatrix} \mathscr{A}(\rho) & \mathscr{B}_1(\rho) & \mathscr{B}_2(\rho) \\ \mathscr{C} & 0 & 0 \\ \mathscr{C}_z(\rho) & \mathscr{D}_{z1}(\rho) & \mathscr{D}_{z2}(\rho) \end{bmatrix} \begin{bmatrix} x_g(t) \\ w(t) \\ u(t) \end{bmatrix} \tag{9}$$

The generalized plant in (9) is also in polytopic form consisted of the three vertices ω_i, $i = 1,2,3$ of the reduced polytope.

Problem Definition: The LPV control synthesis problem consists in finding a LPV controller $K(\rho)$, for the polytopic approach the vertex controllers K_i, by solving off-line the appropriate set of LMIs, so that the closed-loop system represented in Fig. 4 is stable and there exists a $\gamma > 0$ s.t.

$$\sup_{||w||\neq 0} \frac{\|z\|_2}{\|w\|_2} < \gamma, \ \forall \ \omega_i \tag{10}$$

The solution of that problem are the vertices of the LPV controller's $K(\rho)$ corresponding polytope. This allows to get the vertex controllers in state space form:

$$\begin{aligned} \dot{x}_c(t) &= A_i x_c(t) + B_i y(t) \\ u(t) &= C_i x_c(t) + D_i y(t) \end{aligned} \tag{11}$$

where x_c is the state space vector of the controller.

In the sequel, the LPV controller $K(\rho)$ is computed on-line as a convex combination of the vertex controllers, for the reduced-polytope in Fig. 3, $K_i = \left[\begin{array}{c|c} A_i & B_i \\ \hline C_i & D_i \end{array}\right]$,

$$K(\rho) = \sum_{i=1}^{3} a_i(\rho) K_i \tag{12}$$

where the constants a_i are the solution of the linear system shown below and θ_i are the vectors that include the coordinates of the vertices ω_i; from Poussot-Vassal (2008).

$$\begin{bmatrix} \theta_1 & \theta_2 & \theta_3 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ 1 \end{bmatrix} \tag{13}$$

4. SIMULATION RESULTS

To validate the proposed designed controller, a given scenario was implemented using the aforementioned extended model. A straight lane is used as a reference trajectory. All simulations started with an initial offset of $3m$ away from the reference trajectory and the vehicle's longitudinal speed is set to $v_x = 1 : 19m/s$ and $T = 1.5s$ to show indicatively the convergence and the performance of the controllers for different cases.

In Fig. 5, the trajectory of the vehicle is presented in the frame of the global coordinates where convergence is achieved for lower speeds at $50m$ and for higher after $80m$, having overshoots below $0.5m$ at all speeds.

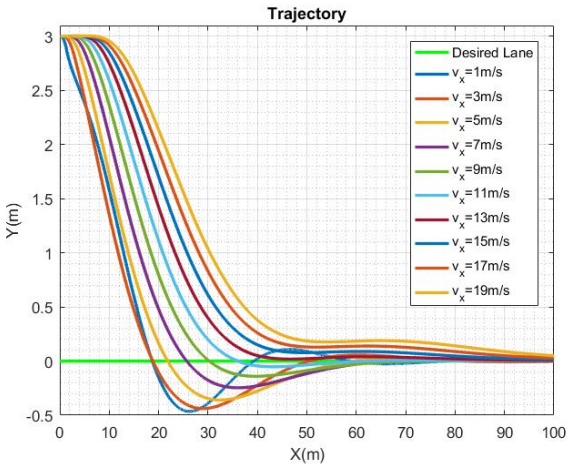

Fig. 5. Trajectory

5. EXPERIMENTAL VALIDATION

This section demonstrates the results of the proposed control, which is implemented and tested on an electric Renault Zoe that has been automated for allowing steering computed control (see Fig. 6).

The vehicle is equipped with a Real-Time Kinematic Differential Global Positioning System (RTK-DGPS) providing information for the localisation. In addition, a MicroAutoBox is installed at the back of the car to integrate the Matlab code to the vehicle's architecture. The vehicle acceptes steering command at 100 H_z. Consequently, the vertex controllers K_i, $i = 1,2,3$ are discretized for sampling time $T_s = 0.01s$ and their on-line implementation is achieved following (12), (13).

In a test track in Satory, France a straight lane followed by a turn is used (see Fig. 7). The vehicle started in manual mode and the autonomous mode is activated in presence of an initial lateral offset. For that reason, it can be seen that there is an initial steering wheel angle in figure 8 but also initial errors to be minimized in the figures 9, 10, 11.

Furthermore, vehicle's velocity followed a speed profile provided by the navigation system (see Fig. 12). At the beginning, the vehicle accelerated gradually from 6m/s to 14m/s, continued with a constant speed of 14m/s, decelerated from 14m/s to 8m/s and accelerated to enter the turn, as shown in Fig. 12.

In Fig. 7 shows the trajectory followed by the vehicle and its position for different instants throughout the test. It has to be

remarked that in the straight lane the mean error value of the lateral offset of the vehicle is $(0.15m)$ and at the turn $(0.63m)$. That could be remarked as a flaw of the method to increase more the bandwidth of the closed-loop system for the whole operating domain.

In Fig. 9, 10 and 11 are shown the evolution in time of the yaw rate error, the lateral and angular errors at the target point respectively. As it can be seen, these errors converged close to zero without having significant oscillations, apart from the overshoot that occurred at the beginning cause of the brute acceleration of the vehicle in combination with the presence of the initial lateral offset.

It's worth to mention that the vehicle has a good performance and the steering was comfortable enough as can be seen in Fig. 8. The steering angle increased abruptly only at the beginning cause of the initial errors and the acceleration of the vehicle. In the rest of the track, the corrections were smooth, as it can be seen there are no sudden changes of the steering wheel angle.

Fig. 6. Automated electric Renault Zoe used throughout the experiments.

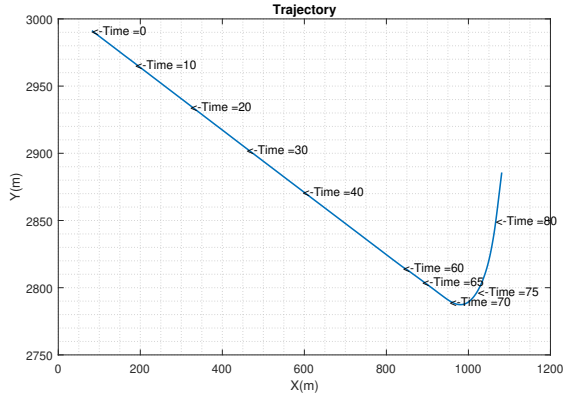

Fig. 7. Trajectory followed in the test track.

6. CONCLUSIONS

This paper has presented that using LPV theory and especially the polytopic approach, an automatic lateral controller could be designed for lane-tracking.

Even though the Polytopic approach is a conservative method, experimental results have shown that it can work for different variations of speed reducing errors smoothly only by computing the convex combination of three linear dynamic output controllers guaranteeing stability and performance at the same time.

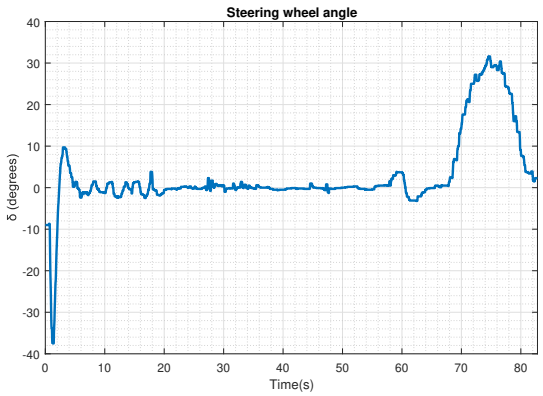

Fig. 8. Steering wheel angle.

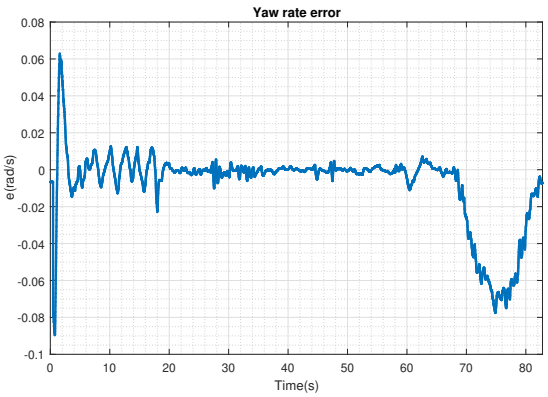

Fig. 9. Yaw rate error of the vehicle.

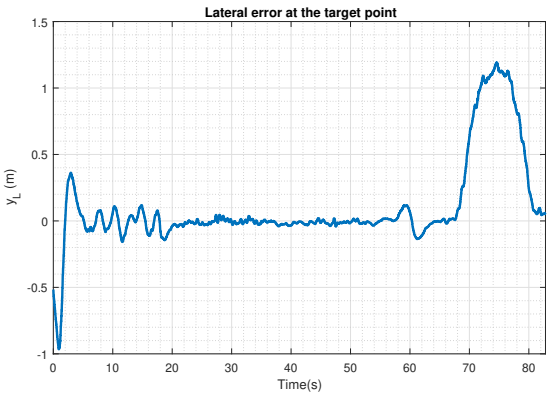

Fig. 10. Lateral error at the target point.

Future works will include the use of different LPV techniques, such as Gridding or Linear Fractional Transformation (LFT) for reduced conservatism, where different look-ahead time T could be selected as a function of speed and ameliorate even better the performance.

Additionally, LPV identification method expressing the actuator model dependent on speed will be explored in order to design a controller using full information of the real vehicle and achieve the optimal performance for the whole operating domain of speed.

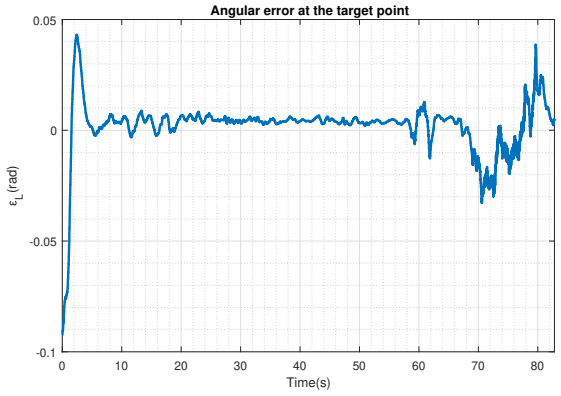

Fig. 11. Angular error at the target point.

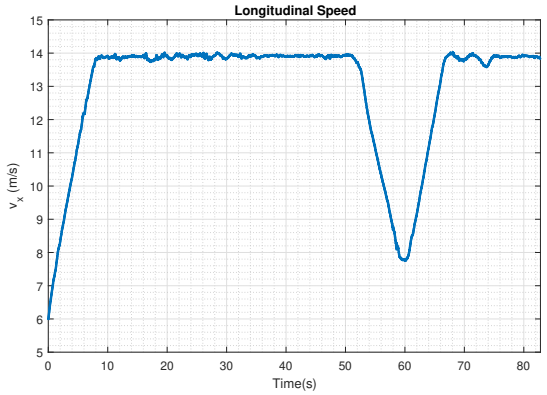

Fig. 12. Vehicle's varying longitudinal speed.

ACKNOWLEDGMENT

This paper reflects solely the views of the authors and not necessarily the view of the company they belong to.

Authors want to thank Renault research department and David Gonzalez for the fruitful discussion and support on the controller integration.

REFERENCES

Ackermann, J., Bünte, T., and Odenthal, D. (1999). Advantages of active steering for vehicle dynamics control.

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

Bagloee, S.A., Tavana, M., Asadi, M., and Oliver, T. (2016). Autonomous vehicles: challenges, opportunities, and future implications for transportation policies. *J. Mod. Transp.*, 24(4), 284–303.

Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). *Linear matrix inequalities in system and control theory*. SIAM, Philadelphia, USA.

Broggi, A., Bertozzi, M., Fascioli, A., Bianco, C.G.L., and Piazzi, A. (1999). The argo autonomous vehicle's vision and control systems. *Int. J. Intell. Syst.*, 3(4), 409–441.

Corno, M., Panzani, G., Roselli, F., Giorelli, M., Azzolini, D., and Savaresi, S.M. (2020). An lpv approach to autonomous vehicle path tracking in the presence of steering actuation nonlinearities. *IEEE Trans. Control Syst. Technol.*, 1–9.

Ercan, Z., Gokasan, M., and Borrelli, F. (2017). An adaptive and predictive controller design for lateral control of an autonomous vehicle. In *IEEE Inter. Conf. Veh. Electron. Safe. (ICVES)*, 13–18.

González, D., Pérez, J., Milanés, V., and Nashashibi, F. (2015). A review of motion planning techniques for automated vehicles. *IEEE Trans. Intell. Transp. Syst.*, 17(4), 1135–1145.

Jochem, T. and Pomerleau, D. (1996). Life in the fast lane: The evolution of an adaptive vehicle control system. *AI Mag.*, 17(2), 11–50.

Li, D., Zhao, D., Zhang, Q., and Chen, Y. (2019). Reinforcement learning and deep learning based lateral control for autonomous driving. *IEEE Comput. Intell. Mag.*, 14(2), 83–98.

Mohammadpour, J. and Scherer, C.W. (2012). *Control of linear parameter varying systems with applications*. Springer Science & Business Media, New York, USA.

Peng, H. and Tomizuka, M. (1990). Vehicle lateral control for highway automation. In *IEEE Proc. Am. Control Conf.*, 788–794.

Pérez, J., Milanés, V., and Onieva, E. (2011). Cascade architecture for lateral control in autonomous vehicles. *IEEE Trans. Intell. Transp. Syst.*, 12(1), 73–82.

Poussot-Vassal, C. (2008). *Robust LPV multivariable automotive global chassis control*. Ph.D. thesis, Institut National Polytechnique de Grenoble, France.

Rajamani, R. (2011). *Vehicle dynamics and control*. Springer Science & Business Media, New York, USA.

Robert, D., Sename, O., and Simon, D. (2009). An h_∞ lpv design for sampling varying controllers: Experimentation with a t-inverted pendulum. *IEEE Trans. Control Syst. Technol.*, 18(3), 741–749.

Sename, O., Gaspar, P., and Bokor, J. (2013). *Robust control and linear parameter varying approaches: application to vehicle dynamics*, volume 437. Springer, New York, USA.

Tan, H.S. and Huang, J. (2014). Design of a high-performance automatic steering controller for bus revenue service based on how drivers steer. *IEEE Trans. Robot.*, 30(5), 1137–1147.

Taylor, C.J., Košecká, J., Blasi, R., and Malik, J. (1999). A comparative study of vision-based lateral control strategies for autonomous highway driving. *Int. J. Rob. Res.*, 18(5), 442–453.

Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., Diebel, J., Fong, P., Gale, J., Halpenny, M., Hoffmann, G., et al. (2006). Stanley: The robot that won the darpa grand challenge. *J. Field Robot.*, 23(9), 661–692.