

Control Reconfiguration of Lateral ADAS Steering Control in the Presence of Driver Errors Using Combined Parity Space / LPV Approaches*

Ariel Medero^{1,2}, Olivier Sename¹ and Vicenç Puig²

Abstract—This paper proposes a lateral Advance Driver Assistance System (ADAS) steering controller that uses the detection of driver errors to reconfigure a Linear Parameter Varying (LPV) controller acting on the combined driver-vehicle lateral steering system. The detection of driver errors is proposed to be formulated as a Fault Detection problem using the Parity Space approach, and the computed residual signal then schedules the LPV/\mathcal{H}_∞ controller. The overall goal is to compute an ADAS controller that helps in stabilizing the vehicle when driver errors are detected while otherwise minimizing the level of intrusiveness. With this goal in mind, the proposed method was tested in simulation using a full dynamical model of a Renault Megane car and driver models, for simulation of the human steering action, during a critical scenario.

I. INTRODUCTION

There have been many propositions in the field of Advanced Driver Assisted Systems (ADAS) to design automated systems with the objective of increasing vehicle's safety. Many of these works consider the presence of obstacles in the path or adaptability to road condition and environment changes. However, few works did consider the effect of potential errors in the driver steering actions on the vehicle stability. Indeed some studies have considered for control design the driver's lateral steering action through Driver Models. In [1] the authors consider a supervisory LPV/\mathcal{H}_∞ control problem with combined braking and steering actions, where the driver steering action is modelled by a SISO transfer function. Meanwhile, a MIMO driver model with simplified dynamics is used in [2] to compute an LPV/\mathcal{H}_∞ steering controller robust to parametric uncertainties to account for different levels of driving performances.

One prominent field in the vehicle lateral control problem with human drivers in the loop is the shared control proposition. Let us mention some recent works in shared control, e.g. [3]–[5], where the main control approach is

the optimal control theory, in order to solve some Model Predictive Control or LQ/\mathcal{H}_2 problems. However, it is worth mentioning that the shared control problem is based on the idea that human and automated system work simultaneously, aiming at minimizing the contradiction criteria between human desire and the automatic control command. However, in shared control it is assumed that automation can take over control in a large percentage, which may be felt invasive by the driver and trigger some users to disengage such systems.

In this paper we present a control method for ADAS lateral steering control, as represented in Fig. 1, by evaluating the driver's performance in real-time and only allowing the ADAS system to act on the steering input when needed, thus aiming at not being invasive for the human driver as much as possible. To evaluate the driver performance, a parity space approach is proposed, and combined with an LPV/\mathcal{H}_∞ controller scheduled according to the detected driver errors.

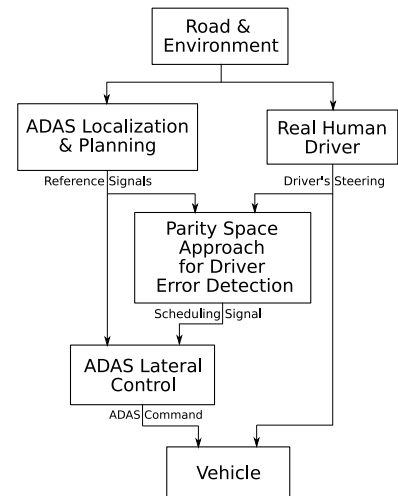


Fig. 1. Proposed ADAS Structure for Lateral Steering Control with Controller Reconfiguration Based on Driver Error Detection.

II. DRIVER MODEL

As highlighted during the introduction, it is of high importance to understand the driver steering action for both lateral control formulation and design, as well as for the driver performance evaluation based on a Fault Detection residual signal. Therefore, a Driver Model (DM) is used to model the human lateral steering task. In this work, we

*This work is supported by the French National Research Agency in the framework of the "Investissements d'avenir" program ANR-15-IDEX-02. This work has also been partially funded by the Spanish State Research Agency (AEI) and the European Regional Development Fund (ERFD) through the project SCAV (ref. MINECO DPI2017-88403-R) and by FPI UPC grant 2020FPI-UPC-008.

¹Ariel Medero and Olivier Sename with GIPSA-Lab, Univ. Grenoble Alpes, CNRS, Grenoble INP, 38000 Grenoble, France ariel.medero@grenoble-inp.fr, olivier.sename@grenoble-inp.fr

²Ariel Medero and Vicenç Puig with Institut de Robòtica i Informàtica Industrial (CSIC-UPC), C/. Llorens i Artigas 4-6, 08028 Barcelona, Spain ariel.medero@upc.edu, vicenc.puig@upc.edu

have selected as a base the DM proposed by Donges [6] as represented in Fig. 2, with $n_u = 2$ multiple inputs acting as feedforward and feedback actions that explain the $n_y = 1$ output of the model representing the driver steering action applied to the vehicle.

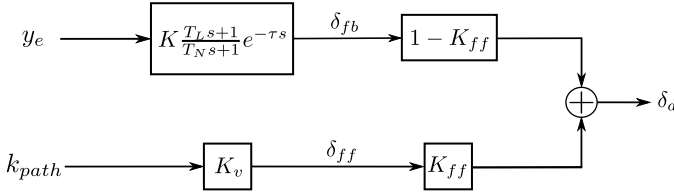


Fig. 2. Driver Model for the Lateral Steering Task.

The feedback path models the neuromuscular human system and explains the physical actions carried by the driver in order to reduce the lateral error y_e between the vehicle's Center of Mass and the road center-line, using the Simplified Driver Precision Model (SDPM) [7] with $n_x = 1$ number of states. Meanwhile, the feedforward steering is modelled as a constant gain [8], and represents the driver's anticipating action to the incoming road path curvature k_{path} .

Finally, the multilevel driver model from Donges was modified to introduce the weighting strategy proposed in [8], with $K_{ff} \in [0, 1]$ as the weighting constant used to explain the reliance of drivers mainly on the feedforward visual information. Additionally, this constant is also useful to differentiate the driver's skill level between novice and experienced drivers, as more experienced drivers rely more on feedforward steering than novice ones.

Finally, given the DM structure in Fig. 2, let us define the set of model parameters as:

$$\mathcal{P} = \{K, T_L, T_N, \tau, K_v, K_{ff}\} \in \mathbb{R}^6 \quad (1)$$

III. INTEGRATED DRIVER-VEHICLE SYSTEM MODELLING

For the design of ADAS system, it is not enough to consider the vehicle as an independent entity, as the human interaction with the vehicle has an important impact in the overall system performance and stability. Therefore, the control-oriented model should be a combined Driver-Vehicle one.

Let us consider a nominal DM with parameters $P_0 \in \mathcal{P}$. Note that the choice of the nominal set of parameters P_0 is so that in open-loop the nominal driver is able to steer the vehicle in a way such that lateral accelerations and overshoots are minimized, even during obstacle avoidance maneuvers. On the other hand, to obtain a combined Driver-Vehicle model for control purposes two modifications are carried out.

First, the road curvature k_{path} input should be expressed in terms of desired yawrate $\dot{\psi}_{ref}$. So that the DM inputs can be easily related to vehicle dynamic variables. Using the expression for $\dot{\psi}_{ref}$ in [9], k_{path} is then substituted by:

$$k_{path} = \frac{\dot{\psi}_{ref}}{v_x} \quad (2)$$

where v_x represents the vehicle longitudinal velocity.

The second modification is to approximate the pure delay present in the SDPM by means of a Padé approximation, considered in this work as of order $n_d = 4$. Then, the modified DM can be expressed in continuous-time state-space form as:

$$\begin{cases} \dot{x}_c = a_c \cdot x_c + [b_c \ 0] \cdot u_c \\ \delta_0 = (1 - K_{ff0})c_c \cdot x_c + [(1 - K_{ff0})c_c \ K_{ff0}K_{v0}/v_x] \cdot u_c \end{cases} \quad (3)$$

where $x_c \in \mathbb{R}^{n_x+n_d}$ is the DM continuous-time state vector, $a_c \in \mathbb{R}^{n_x+n_d \times n_x+n_d}$, $b_c \in \mathbb{R}^{n_x+n_d}$, $c_c \in \mathbb{R}^{n_y \times n_x+n_d}$, $d_c \in \mathbb{R}^{n_u \times n_y}$ are the state-space matrices of the modified SDPM and their values depend on $[T_{L0}, T_{N0}, \tau_0, K_0] \in P_0$ as well as $[K_{ff0}, K_{v0}] \in P_0$ and the input vector $u_c(t) \in \mathbb{R}^{n_u}$ is

$$u_c(t) = \begin{bmatrix} y_e(t) \\ \dot{\psi}_{ref}(t) \end{bmatrix} \quad (4)$$

On the other hand, the vehicle lateral dynamics are modelled using the well known bicycle model given as:

$$\begin{bmatrix} \ddot{y} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} -\frac{C_{\alpha f} + C_{\alpha r}}{m v_x} & -v_x - \frac{C_{\alpha f} l_f - C_{\alpha r} l_r}{m v_x} \\ -\frac{C_{\alpha f} l_f - C_{\alpha r} l_r}{I_z v_x} & -\frac{C_{\alpha f} l_f^2 + C_{\alpha r} l_r^2}{I_z v_x} \end{bmatrix} \begin{bmatrix} \dot{y} \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} \frac{C_{\alpha f}}{m} \\ \frac{C_{\alpha f} l_f}{I_z} \end{bmatrix} \delta \quad (5)$$

For the details on the bicycle model parameters, as well as its values, they can be found on [10], [11]. The parameter values used come from the identification of a full car model for a Renault Megane vehicle, which have been empirically validated using data from a real car.

To conclude, the combined Driver-Vehicle model for control purposes considered in this work, is the series connection of the state-space representation for the nominal driver and vehicle dynamics model. Note that both the nominal driver and the vehicle lateral dynamics models do depend on the longitudinal vehicle speed v_x , which is assumed to be constant in this work. Note that this is coherent with the paper objective which is to use the ADAS system in a critical driving situation (at a given speed). The extension to varying speed models can be the topic of future works.

IV. PARITY SPACE APPROACH FOR DRIVER ERROR DETECTION

In the previous section, for the control oriented model a nominal driver model has been assumed. However, it is to be expected that most of human drivers would perform differently than the nominal one. Specifically, it can be assumed that, in a critical driving situation, a human driver would perform worst than the nominal one (so making errors), since the parameters of the nominal driver model $P_0 \in \mathcal{P}$ are chosen to get an ideal performance. In order to detect driver errors, it is proposed to formulate a Fault Detection problem, for which the fault-free system is the nominal driver model, while the difference in steering between the real human driver δ_f and the nominal driver δ_0 will be considered as the fault signal.

$$\delta_f(k) = \delta_0(k) + f(k) \quad (6)$$

The chosen fault detection method on this paper is the parity space approach, which requires a discrete state-space model of the nominal DM. Assuming the pure time delay present in the SDPM as an input delay, then using a sampling period $T_s = 10ms$ and an exact Zero-Order Hold discretization method, the discrete-time state-space representation of the nominal DM, with parameters $P_0 \in \mathcal{P}$ and the chosen DM structure shown in Fig. 2, is defined by the following system matrices:

$$A_0 = a_0 \in \mathbb{R} \quad (7)$$

$$B_0 = [b_0 \ 0] \in \mathbb{R}^{n_x \times n_u}, \quad (8)$$

$$C_0 = (1 - K_{ff0})c_0 \in \mathbb{R}, \quad (9)$$

$$D_0 = [(1 - K_{ff0})d_0 \ K_{ff0}K_{v0}] \in \mathbb{R}^{n_u \times n_y}, \quad (10)$$

and can be expressed as:

$$\begin{cases} x_d(k+1) = A_0 \cdot x_d(k) + B_0 \cdot u_d(k) \\ \delta_0(k) = C_0 \cdot x_d(k) + D_0 \cdot u_d(k) \end{cases} \quad (11)$$

with $x_d(k) \in \mathbb{R}$ is the DM discrete-time state vector, $a_0, b_0, c_0, d_0 \in \mathbb{R}$ are the elements of the state-space representation for the discretized nominal SDPM, whose values depend on the chosen sampling time T_s and the DM parameters $[K_0, T_{L0}, T_{N0}] \in P_0$ as well as $[K_{ff0}, K_{v0}] \in P_0$ and the input vector $u_d(k) \in \mathbb{R}^{n_u}$ is

$$u_d(k) = \begin{bmatrix} y_e(k - \tau_0/T_s) \\ k_{path}(k) \end{bmatrix} \quad (12)$$

with $\tau_0 \in P_0$.

Let us assume now that the real human driver is doing an erroneous action, considered as an additive fault $f(k)$ with respect the nominal driver as previously defined in (6). The faulty driver model is then given as:

$$\begin{cases} x_d(k+1) = A_0 x_d(k) + B_0 u_d(k) \\ \delta_f(k) = C_0 x_d(k) + D_0 u_d(k) + f(k) \end{cases} \quad (13)$$

Then, for the detection of the driver error $f(k)$ the Parity Space representation of the system is defined as:

$$Y(k) - HU(k) = Wx(k-L) + MF(k) \quad (14)$$

With $Y(k) \in \mathbb{R}^{n_y \cdot (L+1)}$ a buffer of $\delta_f(k)$ and $U(k) \in \mathbb{R}^{n_u \cdot (L+1)}$ a buffer of $u_d(k)$, both buffers of length $L+1$. $F(k) \in \mathbb{R}^{n_y \cdot (L+1)}$ is the fault vector. And matrices $H \in \mathbb{R}^{L+1 \times n_u \cdot (L+1)}$, $W \in \mathbb{R}^{n_y \cdot (L+1) \times n_x}$ and $M \in \mathbb{R}^{n_y \cdot (L+1) \times n_y \cdot (L+1)}$ defined in the standard manner of the parity space approach according to [12].

Note that in the parity representation of system (14) all the elements for the left-side of the equality are known in real time. On the other hand, in the right side of the equality appears the system state, which may be not completely measured, and the fault vector $F(k)$ which is unknown. In order to make the fault detection independent of the system state, (14) can be left-multiplied by the so called parity vector V , which is defined in such a way that the following condition is fulfilled:

$$V \cdot W = 0 \quad (15)$$

If such a parity vector V exists for a number L of past considered data samples, then the residual signal is given by:

$$r(k) = V(Y(k) - HU(k)) \quad (16)$$

The interpretation of $r(k)$ is the following: since the parity vector V fulfills condition (15), left-multiplying (14) by V leads to:

$$r(k) = VMF(k) = VF(k) \quad (17)$$

Where using the definition for the matrix M in [12] and the fault model definition (13) it can be observed that $M = I$, thus $r(k)$ depends only on the fault vector value. Therefore, in the considered driver error detection problem, the residual signal $r(k)$ equals zero (or almost) when the driver's action match or are close to those expected from the nominal driver, on the other hand, when the human steering differs significantly with respect the considered nominal driver it is considered as an erroneous action and the value of $r(k)$ becomes important.

V. H_∞ LATERAL ADAS CONTROLLER

This section presents the main result of the paper, namely an LPV fault-scheduling controller allowing to reconfigure the control action in the presence of driver errors. In what follows, we present first the general structure of the control scheme, then we define the scheduling parameter function of the residual, then the LPV/ H_∞ control problem is formulated, and finally the control synthesis method is explained.

A. Structure of the control system scheduled by the detected driver error

The proposed structure for the integrated fault detection/control system is presented in Fig. 3. In this architecture, high levels elements of the ADAS system will provide with reference signals for the desired path, mainly y_e, k_{path} and ψ_{ref} . The first two signals in conjunction with measurements from the human steering δ_f are used to evaluate the driver performance as explained in Sec. IV, using the Parity Space approach in order to generate a residual signal $r(k)$. This indicator for the driver's performance is then transformed into an adequate scheduling parameter ρ for the real-time reconfiguration of the LPV/ H_∞ controller $K(\rho)$ acting on the combined Driver-Vehicle System.

The considered control problem for the design of $K(\rho)$ is given in Fig. 4, which illustrates how the controller acts on the combined Driver-Vehicle System presented in Sec. III by acting on parallel to the driver and being scheduled by the signal ρ . As proposed, the objective with such a scheduled controller, is that when the levels of performance of the driver are within the accepted levels, the ADAS system will have low control authority. Meanwhile, when an important driver error is detected, the automated lateral steering assistance system will be given more freedom to act on the vehicle to help overcome critical situations.

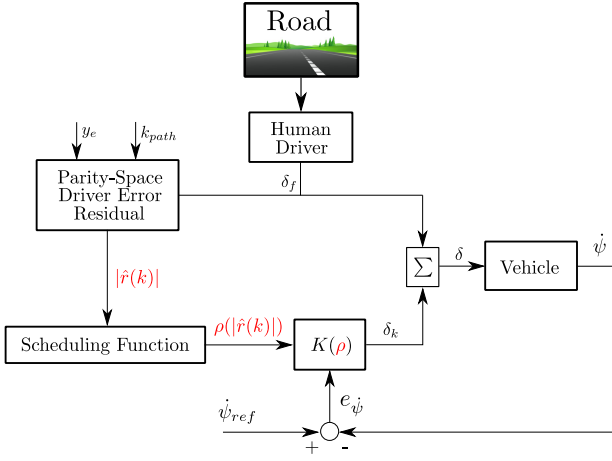


Fig. 3. Global Lateral Driver-in-the-Loop ADAS Control Scheme

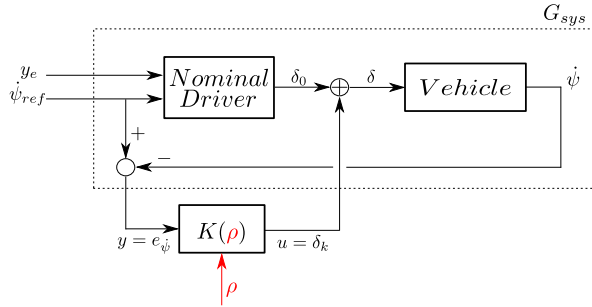


Fig. 4. Driver plus ADAS Closed-Loop System

B. Residual Dependant Scheduling Variable ρ

In order to define the scheduling function, let us first define the relative residual signal as:

$$\hat{r}(k) = \frac{r(k)}{\bar{r}_0} \quad (18)$$

where, $r(k)$ is the residual signal as computed according to Sect. IV and \bar{r}_0 is the threshold residual, which is defined as the maximum value achieved by the residual when the Driver-Vehicle system, described in Sect. III, is evaluated during the nominal scenario. Its determination follows a three-step procedure:

- 1) Simulation of a steering maneuver of the Driver-Vehicle system with a given nominal driver defined by a set of nominal parameters $P_0 \in \mathcal{P}$
- 2) Computation of the residual vector r_0 (of length N) using a constant sampling period T_s .
- 3) Definition of the threshold residual \bar{r}_0 as:

$$\bar{r}_0 = \sup |r_0(k)|, \quad \forall k \in [0, N] \quad (19)$$

Then, let us define the scheduling signal ρ , as the following hyperbolic function:

$$\rho(\nu) = \frac{\bar{\rho}}{2} \cdot \frac{e^{a(\nu-\nu_0)} - e^{-b(\nu-\nu_0)}}{e^{a(\nu-\nu_0)} + e^{-b(\nu-\nu_0)}} + \left(\frac{\bar{\rho}}{2} + \varepsilon\right) \quad (20)$$

with $\nu := |\hat{r}(k)|$, $\bar{\rho}$ is the maximum value desired for the scheduling function $\rho(|\hat{r}(k)|)$, ε is the minimum value

desired for $\rho(|\hat{r}(k)|)$, ν_0 can be used to shift the x axis of the function if desired, a and b determine the slope of the hyperbolic function.

Finally, note that the scheduling variable ρ due to its dependency on the residual signal is a discrete signal. For implementation, it is converted from digital to analog to be used to schedule the LPV/\mathcal{H}_∞ controller.

C. LPV/\mathcal{H}_∞ Control Problem

The control problem is here formulated as an LPV/\mathcal{H}_∞ problem where the objective is to minimize the induced \mathcal{L}_2 norm of the LPV closed-loop system from exogenous inputs w to exogenous outputs z .

$$\|z\|_2 \leq \gamma_\infty \|w\|_2 \quad (21)$$

where the exogenous inputs w represent external references and disturbances inputs and the exogenous outputs z represent the control performance channels. The closed-loop interconnection for \mathcal{H}_∞ design is shown in Fig. 5, and is given by the Lower Fractional Transformation between the generalized plant $P(\rho)$ and the controller $K(\rho)$: $\Sigma_{CL}(\rho) = LFT(P(\rho), K(\rho))$. For control design, W_e and W_u are used to set performances on the closed-loop system by means of frequency shaping [13].

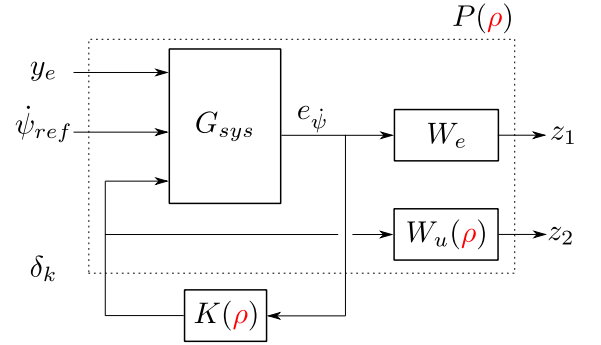


Fig. 5. Control Loop Interconnection for LPV/\mathcal{H}_∞ Design

The weight W_e shapes the tracking performances (through the sensitivity function S) from the reference $\dot{\psi}_{ref}$ to the tracking error e_ψ , and is given by:

$$W_e(s) = \frac{s/M + 2\pi f_1}{s + 2\pi f_1 \varepsilon_e} \quad (22)$$

with $M = 2$ to ensure robustness at all frequencies, $f_1 = 3Hz$ to set the closed-loop system bandwidth and $\varepsilon_e = 0.1$ to ensure a closed-loop steady-state error less than 10%.

The weight $W_u(\rho)$ is dedicated to the control input performance (through the control sensitivity function KS from $\dot{\psi}_{ref}$ to the control output u). It is chosen as an LPV system of the form:

$$W_u(\rho) = \rho G_\delta^0 \frac{(s/2\pi f_2 + 1)(s/2\pi f_3 + 1)}{(\alpha s/2\pi f_2 + 1)(s/\alpha 2\pi f_3 s + 1)} \quad (23)$$

$$G_\delta^0 = \frac{(\alpha \Delta_f/2\pi f_2 + 1)(\Delta_f/\alpha 2\pi f_3 + 1)}{(\Delta_f/2\pi f_2 + 1)(\Delta_f/2\pi f_3 + 1)}$$

$$\Delta_f = 2\pi(f_2 + f_3)/2$$

which is a band-pass filter extended from [10], [14]. The objective of such a filter is to constraint the controller commands in the frequency range between $f_2 = 1Hz$ and $f_3 = 10Hz$, where it can affect the vehicle dynamics while not being felt intrusive to the driver, who is mainly sensitive to low steady-state frequencies ($\leq 1Hz$) and very high frequencies vibrations ($\geq 10Hz$) acting on the steering wheel. The constant α is used to shape the filter.

The originality is here to use a parameter dependent weight $W_u(\rho)$, where the scheduling parameter ρ is obtained from the residual signal as explained previously in (20). Note that for the maximum value of $\rho = 100$, the control action is heavily penalized so the controller does not have much control authority, meanwhile for the minimum value $\rho = 0.1$ it is given extra authority in order to help the driver to overcome a critical driving situation.

D. LPV/ \mathcal{H}_∞ Controller Synthesis

The synthesis of the \mathcal{H}_∞ dynamic output feedback controller, as represented in Fig. 5 consists in applying the Bounded Real Lemma to the closed loop system $\Sigma_{CL}(\rho) = LFT(P(\rho), K(\rho))$. Such a problem being defined for an infinite set of parameter values, several methods do exist to reduce it to a finite dimensional problem. We here consider the referred-to-as grid based approach considering a parameter dependent Lyapunov function $X(\rho)$, and using a gridding of the parameter space to solve the optimisation problem [15]. The grid-based LPV model, consists of a series of LTI models "frozen" along the trajectories of the varying parameters. For both modelling and synthesis of the grid-based LPV/ \mathcal{H}_∞ controller the toolbox LPVTools [16] has been used. Note that since the combined driver-vehicle model is LTI, as v_x is assumed constant in this paper, the varying parameter is then ρ , appearing in the performance weight W_u of the generalized plant P that defines the formulated \mathcal{H}_∞ control problem. The grid for the parameter is here defined by: $\rho_i = [0.1, 1, 100]$ with the parameter variation bounded by $\dot{\rho} \in [-400, 400]$. Finally, the basis for the parameter dependent Lyapunov function has been chosen as:

$$X(\rho) = X_0 + \rho X_1 \quad (24)$$

where X_0 and X_1 are constant matrices to be computed during the LMI optimization.

VI. RESULTS

To assess the performances of the proposed control strategy, simulations have been carried out using a full vehicle model based on a real Renault Megane car [11] thus performing a truthful vehicle dynamics simulation. Moreover, to increase the accuracy of the driver action for simulation, the full driver precision model [7] is used instead of the SDPM shown in Fig. 2.

$$\frac{\delta_{fb}(s)}{y_e(s)} = K \frac{T_L s + 1}{(T_N s + 1) \cdot \left(\frac{s^2}{\omega_n} + \frac{2\zeta}{\omega_n} s + 1 \right)} e^{-\tau s} \quad (25)$$

The other elements of the DM remain as presented in Sect. II.

The test scenario presented to show the performance of the proposed system is the following:

- A double lane-change (DLC) at high speeds, where the controller and the nominal driver parameters are chosen for a longitudinal speed of $v_x = 35m/s$, while the test has been done at a initial vehicle speed of $v_x = 40m/s$.

Note that the differences between the vehicle velocity for design and the simulation scenario allows to show the controller robustness faced to parameter uncertainties. The parameters that define the considered nominal and the faulty driver are given in the following table:

Parameters	Nominal	Faulty
K	1/20	1/25
T_L	0.3	0.3
T_N	0.1	0.15
τ	0.1	0.18
K_v	1	1.7
K_{ff}	0.85	0.81
ω_n	-	19
ζ	-	0.17

For the shown results, the Driver-Vehicle system has been simulated first in Open-Loop, where only the faulty driver acts on the steering system without ADAS intervention (case referred to as Only Driver in Fig. 7). Then, a second simulation concerns the driver error reconfiguration controller scheme in Fig. 3, acting in parallel to the faulty driver (case referred to as With ADAS in Fig. 7).

The collected data from the simulations are¹:

- in Fig. 6 information related to the controller reconfiguration is given: the relative residual signal (up), the scheduling function (center) and finally the controller command (down)
- in Fig. 7 it is shown useful information to compare the gains in performance and safety thanks to the proposed scheme, the global car trajectory (up-left), the longitudinal velocity along the simulation (up-right), the steering actions by the faulty driver δ_f together with the total combined steering applied to the car $\delta = \delta_f + \delta_k$ and the expected steering according to the nominal driver δ_0 (down-left), and the lateral acceleration (down-right)

From Fig. 6, it can be seen that the high speed DLC maneuver is demanding for the faulty driver. Which can be quantified thanks to the value of the the relative residual, which reaches an order of 100, an important difference with respect the nominal driver expected steering behaviour. With the fault driver error quantification, the scheduling function ρ becomes small and the LPV controller is scheduled to be able to act. Bringing the system to the nominal scenario ($\hat{r}(k) \simeq 0$) with a soft and safe trajectory.

Important differences for vehicle performance and safety between the open-loop case and the case with ADAS can be

¹Note that it is assumed that the steering action δ can be explained as the combination of driver δ_f and controller steering's δ_k . In reality, the steering action applied to the car would be a combination of driver and controller torques acting on the steering wheel/system.

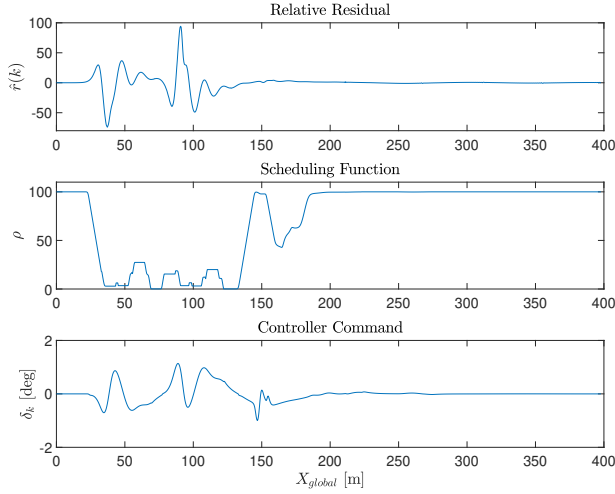


Fig. 6. Controller Scheduling and Command (DLC).

seen in Fig. (7). In the open-loop case with only driver, the vehicle trajectory has an important oscillation which last for a significant time after the DLC maneuver is completed, which in a real scenario would be far from safe. Meanwhile, when the ADAS is active, the applied steering by the controller allows to produce a smoother trajectory. Which also transmits in a reduction by half of the experienced vehicle's lateral acceleration during the maneuver. Which leads to an increase in passenger comfort and most importantly an increase in vehicle safety, as high lateral accelerations at high speeds could cause the vehicle to roll-over.

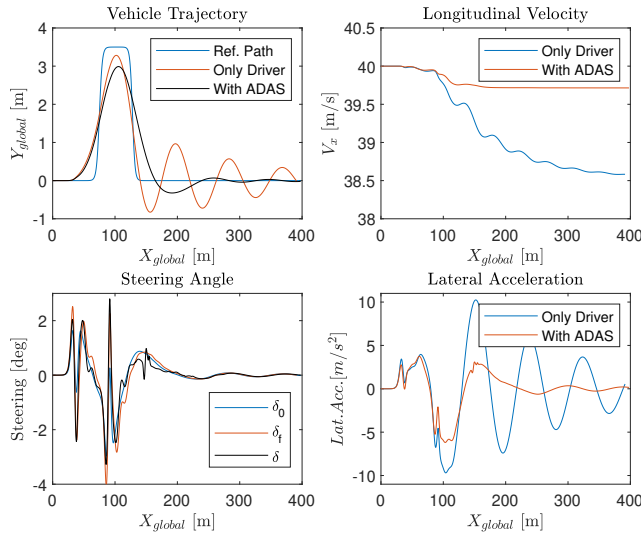


Fig. 7. System Evaluation with/without Proposed ADAS (DLC).

VII. CONCLUSION

In this paper a novel approach to ADAS systems design for lateral vehicle steering have been introduced. The method combines the Parity Space approach to quantify the

driver performance as a residual signal and a reconfigurable controller is designed using LPV/\mathcal{H}_∞ control techniques. The strategy was proven in simulation using a full vehicle dynamical model of a Renault Megane Car. It was seen how the control action is low while the driver's performance is good and when important driver errors are detected the controller helps in improving the vehicle yaw stability by greatly decreasing trajectory oscillations, overshoots and lowering lateral accelerations. Thus showing that the proposed architecture allows indeed to obtain a robust and safe ADAS controller while not being invasive to the human driver in non-critical situations.

REFERENCES

- [1] P. Gáspár, B. Németh, and J. Bokor, "Design of integrated vehicle control using driver models," *IFAC Proceedings Volumes*, vol. 45, no. 13, pp. 517–522, 2012.
- [2] Y. Chen, X. Zhang, and J. Wang, "Robust vehicle driver assistance control for handover scenarios considering driving performances," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019.
- [3] L. Saleh, P. Chevrel, F. Claveau, J.-F. Lafay, and F. Mars, "Shared steering control between a driver and an automation: Stability in the presence of driver behavior uncertainty," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 974–983, 2013.
- [4] S. M. Erlien, S. Fujita, and J. C. Gerdes, "Shared steering control using safe envelopes for obstacle avoidance and vehicle stability," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 441–451, 2015.
- [5] X. Ji, K. Yang, X. Na, C. Lv, and Y. Liu, "Shared steering torque control for lane change assistance: a stochastic game-theoretic approach," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 4, pp. 3093–3105, 2018.
- [6] E. Donges, "A two-level model of driver steering behavior," *Human factors*, vol. 20, no. 6, pp. 691–707, 1978.
- [7] D. T. McRuer and E. S. Krendel, "Mathematical models of human pilot behavior," Advisory Group for Aerospace Research and Development, Neuilly-sur-Seine, France, Tech. Rep., 1974.
- [8] C. Moon and S. B. Choi, "A driver model for vehicle lateral dynamics," *International journal of vehicle design*, vol. 56, no. 1-4, pp. 49–80, 2011.
- [9] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.
- [10] C. Poussot-Vassal, O. Sename, L. Dugard, and S. M. Savaresi, "Vehicle dynamic stability improvements through gain-scheduled steering and braking control," *Vehicle System Dynamics*, vol. 49, no. 10, pp. 1597–1621, 2011.
- [11] S. Fergani, "Robust multivariable control for vehicle dynamics," Theses, Université de Grenoble, Oct. 2014. [Online]. Available: <https://tel.archives-ouvertes.fr/tel-01303752>
- [12] J. Chen and R. J. Patton, *Robust model-based fault diagnosis for dynamic systems*. Springer Science & Business Media, 2012, vol. 3.
- [13] S. Skogestad and I. Postlethwaite, *Multivariable feedback control: analysis and design*. Citeseer, 2007, vol. 2.
- [14] B. A. Guvenc, L. Guvenc, and S. Karaman, "Robust yaw stability controller design and hardware-in-the-loop testing for a road vehicle," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 2, pp. 555–571, 2008.
- [15] F. Wu, *Control of linear parameter varying systems*. University of California, Berkeley, 1995.
- [16] A. Hjartarson, P. Seiler, and A. Packard, "Lpvttools: A toolbox for modeling, analysis, and synthesis of parameter varying control systems," *IFAC-PapersOnLine*, vol. 48, no. 26, pp. 139–145, 2015.