

Predictive ADAS: A Predictive Trajectory Guidance Scheme for Advanced Driver Assistance in Public Traffic*

Thomas Weiskircher¹ and Beshah Ayalew¹

Abstract—In recent years, semi-autonomous vehicle control solutions have been aggressively developed in the form of various advanced driver assistance systems. It is expected that these developments will facilitate the eventual customer acceptance of fully autonomous vehicles. In this paper, we present a configuration of a nonlinear model predictive control scheme for predictive driver assistance. The human driver’s steering and acceleration inputs to the vehicle are first interpreted into the driver-desired vehicle responses. These are then configured to be tracked by a coordination of the predictive and the lower level control systems as long as safe motion is predicted for these driver inputs. As soon as a violation of motion constraints is predicted from the driver inputs, the control scheme augments the driver inputs with active steering, brake and/or engine interventions so that the constraints of the vehicle dynamics, road boundaries and dynamic objects are respected. The performance of this proposed scheme is illustrated and discussed considering closed-loop-simulations of some typical driving events in public traffic.

I. INTRODUCTION

The eventual adoption of fully-autonomous (driver-less) vehicles is expected to build on broad market penetration of various semi-autonomous control functions that in turn build on the now mandated active safety systems such as Electronic Stability Control. Such active safety systems have been shown to boost safety in critical maneuvers before passive safety systems need to be deployed for the protection of passengers. These active systems change the vehicle’s dynamics and as such need to have a proper coordination with the human driver to guarantee a stable control behavior and ensure safety. The coordination of the active control action with the driver’s is a challenging Human-In-The-Loop (HITL) problem. It is prevalent in recent driver assist systems such as lane keeping assist or automatic lane change [1], [2].

There is a volume of recent work which address the HITL problem for semi-autonomous vehicles or driver assistance systems (ADAS). In [3]–[5], a predictive control scheme is proposed that uses a fusion of models for the human driver and the vehicle dynamics (often described with the nonlinear bicycle model) assuming that the parameters of the driver model can be estimated online. In [6], a Hidden Markov Model is used to implement a predictive lane keeping assist system. A path tracking controller for the human control behavior is identified in [4]. However, there is uncertainty

in any driver model, and therefore robustness considerations need to be incorporated for a practical semi-autonomous control framework that explicitly use a driver model. To this end, a tube-based Model Predictive Control (MPC) is proposed in [3], [7]. Therein, invariant set theory [8] is utilized to calculate the safe region of vehicle motion assuming a given uncertainty bound on the driver model. The main ideas were also proposed earlier in [4]. An alternative threat assessment method was proposed in [9] which predicts the vehicle trajectory by a MPC-based approach and uses a blending function to combine the control and driver input.

In the present contribution, we seek to build on our previous work for semi- and fully-autonomous vehicles [10], [11]. The framework proposed therein uses the nonlinear model predictive control (MPC) approach to control the vehicle motion under several constraints such as a lane boundaries, dynamic obstacle objects, traffic regulations and vehicle handling/acceleration limits. The present work re-configures this framework as an advanced driver assistance system (ADAS). Instead of adding a human driver model, this new approach uses the MPC implementation alone to find the optimal trade-off between safe vehicle motion and accommodating driver input. Assuming a driver giving steering and acceleration/braking input to the vehicle, the new ADAS mode focuses on the soft transition from no driver support to full autonomous driving. The paper outlines the necessary modifications of the former framework to accommodate the HITL problem.

The rest of the paper is organized as follows. In Section II, we introduce the new mode for the (semi-)autonomous vehicle control framework named predictive ADAS (pADAS). There, the particle model for vehicle motion prediction is recited and the modifications required for the ADAS mode are given. A driver interpreter and the updated lower-level control (LLC) systems are discussed in Section III. For simulation and illustration purposes only, we adopt a model of a human driver for longitudinal and lateral vehicle control. Section IV includes simulation results and discussions considering typical maneuvers in public traffic incorporating other objects/vehicles on the road and illustrate the effectiveness of the framework.

II. CONTROL FRAMEWORK FOR PREDICTIVE ADAS

Figure 1 depicts the proposed control structure. The human driver is the main operator of the vehicle. The driver actuation inputs to the vehicle dynamics are the front axle steering angle and the acceleration request signals generated via the brake and/or acceleration pedals. The driver request

*This work was supported by a fellowship within the Postdoc-Program of the German Academic Exchange Service (DAAD)

¹Thomas Weiskircher and Beshah Ayalew are with the Applied Dynamics & Control Research Group at the Clemson University - International Center for Automotive Research, 4 Research Drive, 29607, Greenville, SC, USA, {tweiski, beshah}@clemson.edu

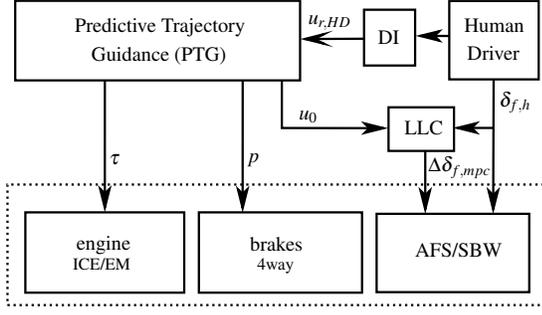


Fig. 1. Proposed control structure in ADAS mode for cooperative driving

is then transformed with a Driver Interpreter (DI) module to generate appropriate signals for the predictive trajectory module. The latter implements a nonlinear Model Predictive Control (MPC) scheme taking into account the interpreted driver input signals as well as a host of constraints related to obstacles, lanes, and vehicle limits. The lower-level control (LLC) follows the references of the PTG but is also connected to the driver signals. The final control actions are to be achieved via the specific actuators indicated (engine, braking, active or by-wire steering).

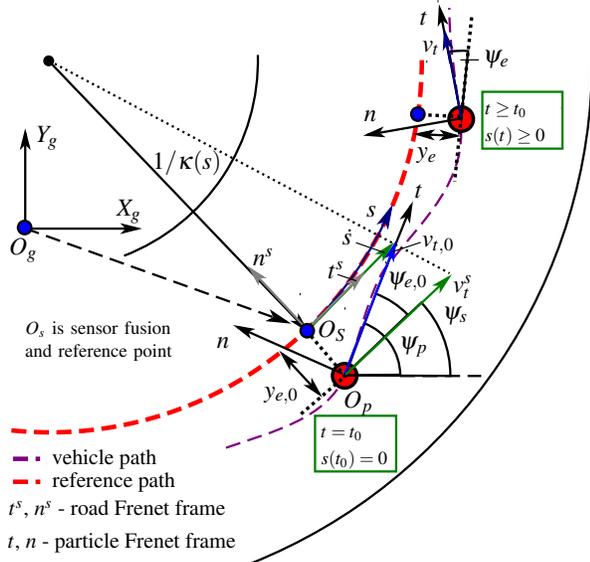


Fig. 2. Definitions of the curvilinear vehicle motion

There is a major difference between this structure and other control structures proposed in the literature that include human driver models (see e.g. [3], [4]). The connection between the human driver and the PTG in this proposed structure allows for a more direct consideration of the driver's control intent in the PTG. Thus, no explicit human driver model is required within the PTG itself. The motivation of doing so is to avoid the generally questionable human driver models within the predictive control loop. As long as the MPC of the PTG results in a safe trajectory, a control action will be generated to control the vehicle by augmenting the driver's input. Specifically, if the driver's input seems to lead

to a future trajectory that violates the constraints, the MPC calculates the control input required to prevent this violation.

A. Particle Error Motion in Frenet Frame

For computational expediency of the MPC on real-time hardware, we adopt the particle motion model for the ideal vehicle motion as shown in [10]. The required parts of this model are recited here. The basis is a 2D curvilinear particle motion described in the Frenet frame with the definitions depicted in Fig. 2. The motion of the particle with respect to the local reference path/lane centerline is given by the angular alignment error ψ_e and lateral error y_e . It can be shown that the motion dynamics can be described by the following equations:

$$\dot{v}_t = a_t, \quad (1a)$$

$$\dot{\psi}_e = \dot{\psi}_p - v_t \cos(\psi_e) \left(\frac{\kappa(s)}{1 - y_e \kappa(s)} \right), \quad (1b)$$

$$\dot{y}_e = v_t \sin(\psi_e), \quad (1c)$$

$$\dot{a}_t = 1/T_{a_t} (a_{t,d} - a_t), \quad (1d)$$

$$\dot{\psi}_p = 1/T_{\psi_p} (\dot{\psi}_{p,d} - \dot{\psi}_p), \quad (1e)$$

$$\dot{s} = v_t \cos(\psi_e) \left(\frac{1}{1 - y_e \kappa(s)} \right). \quad (1f)$$

In this description, the desired acceleration $a_{t,d}$ and the desired yaw rate $\dot{\psi}_{p,d}$ are used as inputs to maneuver the particle along a given reference path. The reference path curvature $\kappa(s)$ is assumed to be known along the reference path coordinate s and v_t, a_t are the particle speed and acceleration along its path. $\dot{\psi}_p$ is the yaw rate, and T_{a_t}, T_{ψ_p} are the time constants of the first-order lag approximation of the controlled longitudinal and lateral vehicle dynamics. \dot{s} is the speed projected on the reference path as shown in Fig. 2. With the knowledge of the curvature of the reference path $\kappa(s)$ and the assumption that the vehicle is required to follow the reference path, the control input can be substituted with

$$\dot{\psi}_{p,d} = \dot{\psi}_{p,r} + \Delta \dot{\psi}_{p,d} = v_t \kappa(s) + \Delta \dot{\psi}_{p,d}. \quad (2)$$

and the MPC algorithm only needs to control the deviation with the new input $\Delta \dot{\psi}_{p,d}$.

B. Description of Objects and Constraints

The MPC objective and constraints for the pADAS implementation of the PTG are described in this section. First, to incorporate a description of other objects (obstacles or other vehicles) on the road in the prediction, the motion of other objects in the road frame $O_s(t^s, n^s)$ is given by (see Fig. 3):

$$s_{o_i} = s_{o_i,0} + v_{t,o_i}^s x_t + 0.5 a_{t,o_i}^s x_t^2, \quad (3a)$$

$$y_{e,o_i} = y_{e,o_i,0} + v_{n,o_i}^s x_t + 0.5 a_{n,o_i}^s x_t^2, \quad (3b)$$

$$\dot{x}_t = 1, \quad (3c)$$

where we included an additional state x_t that represents the internal time in the MPC prediction model. x_t helps to calculate the object positions by using only the parameters $v_{t,o_i}^s, v_{n,o_i}^s, a_{t,o_i}^s, a_{n,o_i}^s$ obtained at each MPC update. Now, the vehicle needs to stay away from each object with a

approaches have been proposed in the literature including neural networks, Hidden Markov Models (HMM), classic path following, and optimal controllers [12]–[14]. For our purposes, we adopt the lateral motion controller model for the driver from [4]. This model has been shown to have good tracking performance and has been validated via some experimental data. The model treats the driver as a steering controller that acts to minimize the error between the future vehicle aligning error ψ_e, l_p and the present lateral offset y_e :

$$\delta_{f,HD} = K_{y_e}(v_t)y_e + K_{\psi_e, l_p}(v_t)\psi_e, l_p \quad (9a)$$

$$\psi_e, l_p = \psi_e + \kappa_0 l_p \quad (9b)$$

Herein, the curvature is assumed to remain constant up to the preview point $l_p = v_t T_{pr}$ in front of the road with the constant preview time T_{pr} .

The longitudinal control mode of the driver model is formulated as a cascade controller for speed and extended with a control action for distance to objects. This allows the driver to follow the reference speed and to keep a safe distance from other vehicles/objects in the same lane. Thus, the control uses the distance to the vehicle/object when (4) with $\zeta_{Do} = 0$ is violated:

$$d_{oi} = s_{oi} - s \quad (10a)$$

$$\underline{d}_e = d_{oi} - (c \cdot v_t + \Delta s_{oi}) \quad (10b)$$

$$\underline{d}_e^{ctrl} = \min(0, \underline{d}_e) \quad (10c)$$

Here, c is a constant value to adjust the driver's desired minimum speed dependent distance to the vehicle/object in front. Then, the speed control loop reads

$$v_{t,dmin} = P_{dmin} \underline{d}_e^{ctrl} \quad (11a)$$

$$v_{t,e} = v_{t,ref} + v_{t,dmin} - v_t \quad (11b)$$

$$a_{t,HD} = P_{at} v_{t,e} \quad (11c)$$

If no vehicle/object is detected in front $v_{t,dmin} = 0$ holds, and the speed control law is a proportional controller with gain P_{dmin} . The speed error is then used for the calculation of the acceleration in the same manner with gain P_{at} . Finally, $a_{t,HD}$ is saturated to an upper and lower bound imitating driving styles (comfort, sport) of the human driver. For both controllers (longitudinal, lateral), a first order lag is used to approximate human neuromuscular dynamics [15].

Extensive simulation analyses show that this simple driver model for speed and path tracking is able to control the car in several maneuvers in the linear vehicle handling regime. However, as we point out in the results section, this and all other such driver models have limitations for the purposes of evaluating semi-autonomous vehicle control functions.

Given the proposed structure depicted in Fig. 1, the driver's inputs pass through the driver interpreter (DI) block to generate the references for the PTG in the pADAS mode. This block is fundamentally independent of the driver model adopted above for the simulation analysis. In the real time implementation, it interprets the inputs from the actual human driver. Nevertheless, here for continuity, we shall refer to the above driver model. For the longitudinal control of the

vehicle, $u_{r,0}(1) = a_{t,HD}$ is already a proper reference signal for the PTG. By contrast, the yaw rate reference is not given directly. Thus, we make use of the customary approach of passing the driver's steering angle through the yaw rate gain $\hat{\psi}(v_t)$ (of the linearized bicycle vehicle model) to generate the driver-desired/reference yaw rate [16]. It is computed as follows:

$$\hat{\psi}_{HD} = \hat{\psi}(v_t) \delta_{f,HD}, \quad (12a)$$

$$u_{r,0}(2) = \hat{\psi}_{HD,r} - v_t \kappa(s). \quad (12b)$$

The modification of Eq. (12b) is required to ensure compatibility with the control input transformation of Eq. (2).

To conclude this topic, the driver generated reference for the first prediction step of the MPC reads:

$$u_{r,0} = u_{r,HD} = \begin{bmatrix} a_{t,HD} \\ \hat{\psi}_{HD} - v_t \kappa(s) \end{bmatrix}. \quad (13)$$

Here, the desired acceleration $a_{t,HD}$ of the *real* driver is interpreted via the pedal positions and the powertrain potential of the car. For simulations, we shall use the output of the above driver model.

B. Lower-Level Control (LLC)

The control inputs computed by the PTG are passed directly as references for the lower-level control systems that are connected to the actuators. Fig. 4 depicts the structure of the lower-level controllers configured to work with the pADAS mode of the PTG. The human driver's control input needs to be considered in the computation of the control signals at the lower-level as well. The left side shows the longitudinal control which has feed-forward (FF) and PID-feedback components. Normally, the driver's longitudinal input serves as the base wheel torque command for the FF part. It is then manipulated with the error between driver input and PTG generated reference. If the PTG generates a reference different from the human driver, the FF part from the human driver is corrected with $a_{t,d} - a_{t,HD}$. This prevents from having two FF signals. As long as the MPC reference $a_{t,d} = a_{t,HD}$, the PID controller calculates an appropriate correction to follow this reference while overcoming disturbances as driving resistances. The rest of the lower-level longitudinal control is taken from the previous version [11] and omitted here for brevity.

The right side of Fig. 4 shows the lower-level lateral vehicle motion control with a speed dependent PID feedback controller. In case of full autonomous control with the MPC control input, a by-wire steering device like a steer-by-wire (SBW) system is assumed to control the vehicle motion with $\Delta \delta_{f,d}^{lim}$. For the pADAS mode of the PTG, the driver steering input acts directly on the front axle and the PTG computed input acts only as a correction for $\delta_{f,HD}$ when the pADAS is not following the drivers ($a_n \neq \hat{\psi}_{v,d} v_t$). The steering angle is limited when a side slip limit of the front tires is reached.

IV. RESULTS AND DISCUSSIONS

The pADAS mode proposed here for semi-autonomous vehicle control is evaluated via the numerical simulation environment with a high-fidelity vehicle dynamics model.

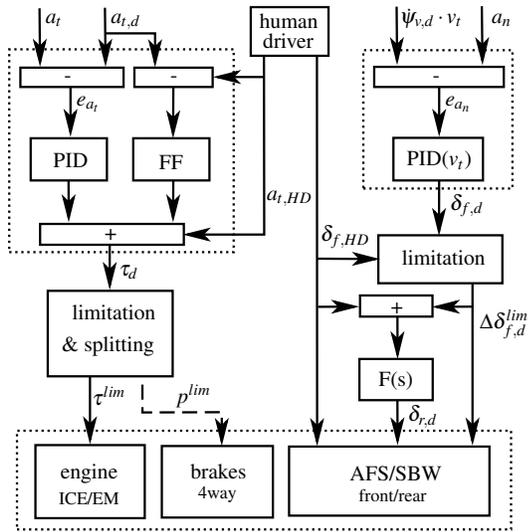


Fig. 4. Lower-level control for trajectory guidance in tracking mode

First, we consider a simple cornering maneuver to illustrate the performance in normal situations. The results are shown in Fig. 5. After the vehicle has entered the corner, it follows the curvature with minor error and the control input is smooth. As no constraint violation is detected, the pADAS follows the driver’s reference very closely.

To illustrate the effect of a pADAS intervention, we first consider an emergency braking maneuver with a distracted driver, and one object vehicle in the same lane traveling with a much slower speed. To emulate a distracted driver, the braking level in the human driver model is set to -0.5 m/s^2 . This reduced deceleration would lead to a collision when only the human driver is acting. The results depicted in Fig. 6 show that the pADAS input follows the human driver (HD) input until a constraint violation is predicted, at which point additional deceleration is commanded to the LLC. Figure 7 shows that the pADAS considers the possibility to pass the object vehicle but after a safe distance has been reached it allows the driver to return the vehicle back to the reference lane. This is also seen in the upper right plot in Fig. 6, which shows the position of the vehicle and the object.

Next, we analyze the tuning utility of the pADAS mentioned above via the component for the yaw rate of the first sample control weight $R_{k=0}$. For the same emergency braking maneuver described above, Fig. 8 shows the lateral deviation and the longitudinal control input. For a higher control weight on tracking the human driver’s lateral control, the lateral deviation is reduced as the human driver tracks the reference itself ($y_e = 0$). Also, the higher weight on lateral control shows no clear influence on a_t . For vehicles equipped with means for detection of driver distraction, the control weight components could serve as tuning variables for the desired level of pADAS intervention. Higher values of the weights force the pADAS to follow the driver input very closely, while smaller values force the pADAS to act more like autonomous driving, which ignores the driver inputs.

Finally, we present results from a collision avoidance

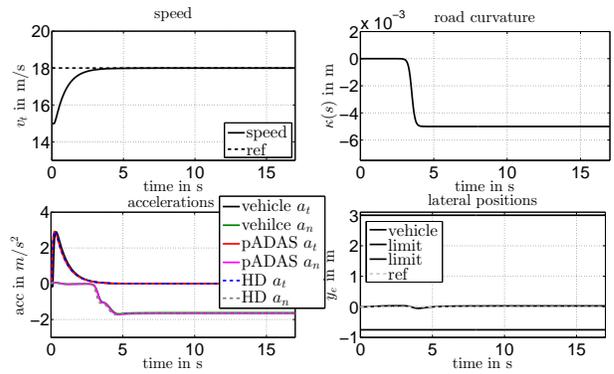


Fig. 5. Cornering: states and control input

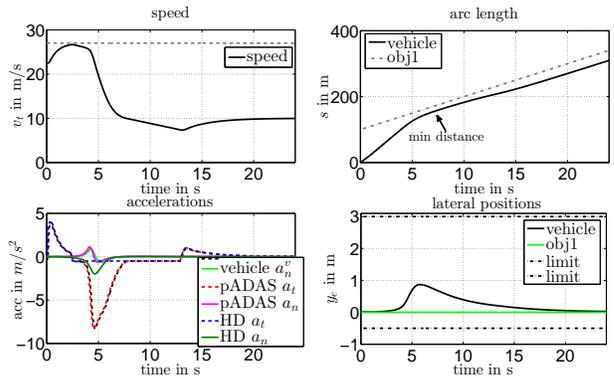


Fig. 6. Emergency Braking: states and control input

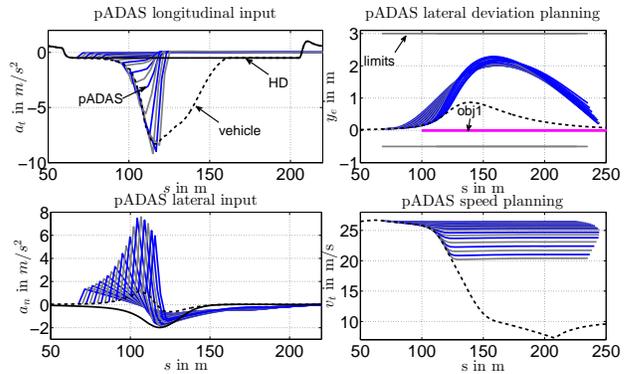


Fig. 7. Emergency Braking: Prediction

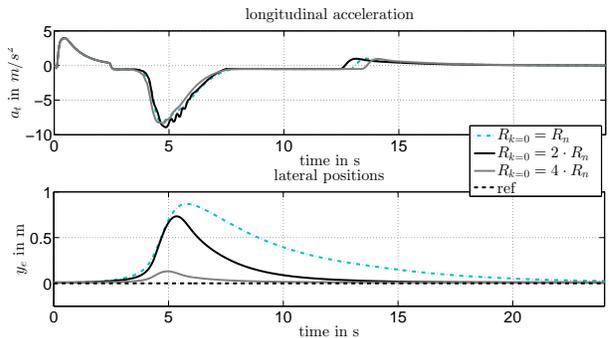


Fig. 8. Emergency Braking: variation of control input weighting

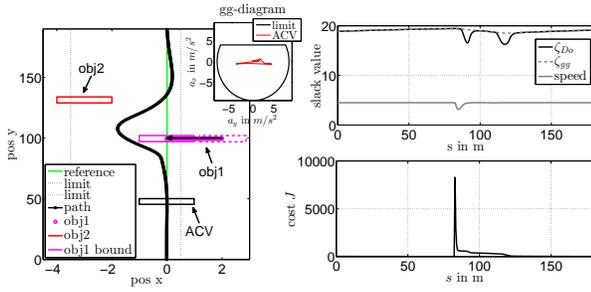


Fig. 9. Collision Avoidance: path, cost, and slack variable

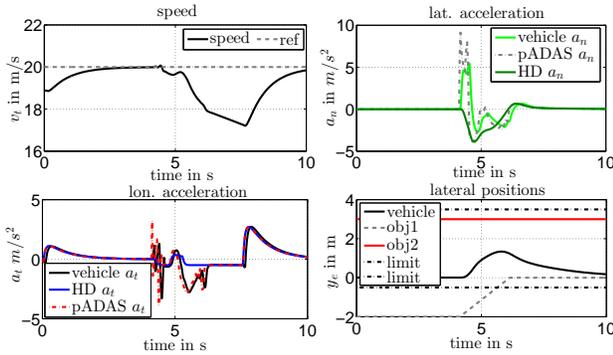


Fig. 10. Collision Avoidance: states and control input

maneuver. Here, one object tries to cross the road and stops suddenly in the middle of the lane of the semi-autonomous vehicle with pADAS. Braking is not sufficient to prevent a collision; only passing will result in a feasible solution. Additionally, on the left lane a static object is in front of the vehicle but with a higher distance than the first object. Figure 9 shows the scene with the road lane, the final vehicle path and two objects and the final path of the controlled vehicle. From Fig. 10 we can see that the pADAS input is clearly different from the driver input when the first object (obj1) starts to move on the road. The pADAS decelerates the vehicle and so does the driver (with a low deceleration of -0.5 m/s^2). When the space between the objects does become less than what is permitted with traffic regulation-based parametrization of constraints, the pADAS formulation proposed here reduces the distance by means of the slack variable ζ_{Do} (see Fig. 9 and discussions following Eq.(4)). This leads to a higher cost but allows the MPC algorithm to find a feasible solution to prevent a crash.

In the above collision avoidance maneuver, the adopted driver model always tries to follow the reference in the middle of the lane regardless of the action of the pADAS. One can expect that a *real* driver would not counteract the pADAS in such a situation. A real driver can be expected to generate a control input which helps to prevent a collision. Such a cooperative control behavior would result from the human decision making process e.g., deciding whether to pass the obstacle or not.

V. CONCLUSIONS

This paper presented a detailed scheme for implementing a predictive advanced driver assistance system (pADAS)

for semi-autonomous vehicle control. Particular emphasis is given in configuring the model predictive controller of a computationally feasible level predictive trajectory guidance (PTG) module to work in cooperation with a human driver. The PTG in this pADAS mode follows the driver control input as long as this input leads to a solution of the underlying optimization problem that satisfies constraints from public driving e.g. obstacles, and speed and handling limits. The proposed pADAS mode of the PTG requires no blending or switching laws for intervention. Furthermore, no human driver model is required in the controller formulation. Thus, the number of model parameters is significantly reduced. The control performance of the system has been illustrated by means of simulations of a high-fidelity vehicle model in collision avoidance and emergency braking events involving a distracted driver.

REFERENCES

- [1] T. Lee, J. Kang, K. Yi, and K. Noh, "An investigation on the integrated human driver model for closed-loop simulation of intelligent safety systems," *Journal of Mechanical Science and Technology*, vol. 24, no. 3, pp. 761–767, 2010.
- [2] P. Bolia, T. Weiskircher, and S. Müller, "Driver steering model for closed-loop steering function analysis," *Vehicle System Dynamics*, vol. 52, no. SP1, pp. 16–30, 2014.
- [3] A. Gray, Y. Gao, J. K. Hedrick, and F. Borelli, "Robust predictive control for semi-autonomous vehicles with an uncertain driver model," in *Proc. of the IEEE Intelligent Vehicles Symposium*, Gold Coast, Australia, 2013, pp. 208–213.
- [4] P. Falcone, M. Ali, and J. Sjberg, "Predictive threat assessment via reachability analysis and set invariance theory," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1352 – 1361, 2011.
- [5] A. Carvalho, Y. Gao, S. Lefevre, and F. Borelli, "Stochastic predictive control of autonomous vehicles in uncertain environments," in *12th International Symposium on Advanced Vehicle Control*, 2014.
- [6] S. Lefevre, Y. Gao, D. Vasquez, E. Tseng, R. Bajcsy, and F. Borelli, "Lane keeping assistance with learning-based driver model and model predictive control," in *12th International Symposium on Advanced Vehicle Control*, 2014.
- [7] Y. Gao, A. Gray, H. E. Tseng, and F. Borelli, "A tube-based robust nonlinear predictive control approach to semiautonomous ground vehicles," *Vehicle System Dynamics*, vol. 52, no. 6, pp. 802–823, 2014.
- [8] F. Blanchini, "Set invariance in control - survey paper," *Automatica*, vol. 35, pp. 1747–1767, 1999.
- [9] S. Anderson and S. Peters, "An optimal-control-based framework for trajectory planning, threat assessment, and semi-autonomous control of passenger vehicles in hazard avoidance scenarios," *Int. J. of Vehicle Autonomous Systems*, vol. 8, no. 2-4, pp. 190–216, 2010.
- [10] T. Weiskircher and B. Ayalew, "Predictive trajectory guidance for (semi-)autonomous vehicles in public traffic," in *American Control Conference*, 2015.
- [11] —, "Frameworks for interfacing trajectory tracking with predictive trajectory guidance for autonomous road vehicles," in *American Control Conference*, 2015.
- [12] R. Sharp, C. de Witt, D. Casanova, and P. Symonds, "Mathematical model for driver steering control, with design, tuning and performance results," *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, vol. 33, no. 5, pp. 289–326, 2000.
- [13] I. Delice and S. Ertugrul, "Intelligent modeling of human driver: A survey," *Intelligent Vehicles Symposium*, pp. 648 – 651, 2007.
- [14] M. Pläächl and J. Edelmann, "Driver models in automobile dynamics applications," *Vehicle System Dynamics*, vol. 45, no. 7-8, pp. 699–741, 2007.
- [15] A. Pick and D. Cole, "Neuromuscular dynamics in the driver-vehicle system," *Vehicle System Dynamics*, vol. 44, no. SP1, pp. 624–631, 2006.
- [16] A. v. Zanten, "Evolution of electronic control systems for improving the vehicle dynamic behavior," in *International Symposium on Advanced Vehicle Control (AVEC)*, Hiroshima, Japan, 2002.