

A Nonlinear Model-Predictive Contouring Controller for Shared Control Driving Assistance in High-Performance Scenarios

Enrico Picotti^{id}, *Graduate Student Member, IEEE*, Mattia Bruschetta^{id}, Enrico Mion, and Alessandro Beghi, *Senior Member, IEEE*

Abstract—An increasing number of vehicles today are equipped with advanced driver-assistance systems that provide humans involved in the driving tasks with continuous and active support. State-of-the-art implementations of these systems frequently rely on an underlying vehicle controller based on the model-predictive control strategy. In this article, we propose a nonlinear model-predictive contouring controller for a driving assistance system in high-performance scenarios. The design follows specific features to ensure the effectiveness of the interaction, namely, adaptability with respect to the current vehicle state, high-performance driving capabilities, and tunability of the assistance system. First, the control algorithm performance is evaluated offline and compared with a commercial lap-time minimizer, then experimental implementation of the assistance system with the human driver (HD) in the loop has been accomplished on a professional dynamic driving simulator, where an evaluation of the specific features has been performed: 1) a *gg-bound* is exploited to adapt the controller's behavior to different driver abilities; 2) the controller's adaptability to unexpected HD behavior is tested; and 3) the controller's ability to handle the vehicle at the limit of maneuverability is established. The obtained strategy, then, demonstrates to be suitable as an underlying vehicle controller for a driver-assistance system on a racing track.

Index Terms—Cooperative systems, cybernetics, predictive control, road vehicle control.

I. INTRODUCTION

IN THE last decade, the use of advanced driver-assistance systems (ADASs) has been spreading in all types of vehicles [1]. A large research effort has been dedicated to the design of systems with autonomy-like characteristics, aimed at assisting the driver in increasing both safety in emergency situations and comfort in highway and urban scenarios. When safety is the main goal, control systems are

usually designed to intervene promptly only in dangerous situations [2]–[4], while when driving comfort is of interest, ADASs provide continuous assistance to the human driver (HD) in a shared control mode, effectively considering the driver-in-the-loop and operating typically under mild dynamic conditions [5]–[10]. Specifically, these implementations rely on a vehicle controller, often based on model-predictive control (MPC) techniques, which generates input signals for the actual vehicle commands (e.g., accelerator, brake, gear, or wheel angle) and/or for the HD interaction system (e.g., haptic interaction through active pedal or steering, visual or auditory cues, etc.). In such a shared control strategy, the human–machine interaction guarantees that drivers can always overwhelm the controller (under simple implementation assumptions, e.g., saturation of the controller maximum force application), allowing an effective handling of the human–machine contradictory. Moreover, the controller is designed to provide continuous feedback to the driver, aiming at reducing the workload of the driver and increasing the overall performance [8], [11], [12].

Notably, research on shared driver-assistance systems for high-performance driving, e.g., in track driving scenarios, is still scarce, although availability of such systems may impact on both industrial applications [13] and training processes [14]. In this context, in fact, an ADAS could be used to ensure safety even at the limit of handling, to allow safe practicing and to ease the testing of high-performance vehicles by *common drivers*. In such a driving scenario, the main challenge is to provide continuous assistance and feedback to the driver while driving at the edge of vehicle maneuverability. Indeed, due to the presence of the human-in-the-loop (HIL), the vehicle controller has to continuously adapt to the actual vehicle state, which might deviate significantly from the expected one. Ensuring such adaptability at the limit of vehicle performance requires the custom design of a vehicle controller. In this framework, we propose an innovative high-performance vehicle control system, based on a nonlinear MPC (NMPC) strategy, which is specifically tailored for driver-assistance systems. The NMPC technique, in fact, is an optimization-based control scheme that is spreading in industrial applications thanks to the availability of *fast* or distributed algorithms that usually rely on the specific structure of the problem and smart initialization, often seeking for suboptimal solutions [15]–[17]. The proposed strategy allows computing both the vehicle commands and the resultant

Manuscript received February 2, 2022; accepted May 8, 2022. This article was recommended by Associate Editor M. Dotoli. (Enrico Picotti, Mattia Bruschetta, and Enrico Mion contributed equally to this work.) (Corresponding author: Enrico Picotti.)

This work involved human subjects or animals in its research. The authors confirm that all human/animal subject research procedures and protocols are exempt from review board approval.

Enrico Picotti, Mattia Bruschetta, and Alessandro Beghi are with the Department of Information Engineering, University of Padua, 35131 Padua, Italy (e-mail: picottie@dei.unipd.it).

Enrico Mion is with the Institute for Dynamic Systems and Control, ETH Zürich, 8092 Zürich, Switzerland (e-mail: enmion@ethz.ch).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TSMC.2022.3177271>.

Digital Object Identifier 10.1109/TSMC.2022.3177271

trajectory for use in different levels of human–automation cooperation, ranging from overriding the human controls to suggesting the optimal actions.

To fit the HIL high-performance scenario, a controller that operates at high sampling rate (50 Hz) has been implemented. Then, inspired by the vehicle control literature [18]–[20], a contouring NMPC formulation [21] (NMPCC) has been adopted, addressing a minimum travel time objective in the NMPC cost function. This strategy allows the optimal trajectory and the relative velocity profile to be computed online by the controller, ensuring a high-performance control objective consistent with the current state (position and attitude) of the vehicle.

To ensure high performance in the contouring formulation framework, it is crucial to balance the need to provide both adequate knowledge of the track in the near future (i.e., a long enough prediction horizon) and an accurate model description with limited computational load. In this regard, a detailed control model that grants real-time capabilities of the controller has been implemented. MPC implementations for shared control ADASs are usually based on simple vehicle models, whereas state-of-the-art high-performance vehicle controllers typically implement more accurate and complex models [18], [19], [22]–[25]. The four-wheel vehicle model described in [24] is adopted and modified to include load transfer, gear shift predictions, longitudinal forces saturation, and an ellipsoid tire friction constraint [20]. The latter characteristic has been also used as a calibration parameter to tailor the performance of the assistance system to different driver abilities.

Since, in the application at hand, the steering action of the controller ultimately defines the haptic feedback for the driver, a specific feature has been introduced in the model. To this end, additional local information about the coupling of longitudinal and lateral tire forces is supplied through a time-varying parameter, thus allowing precise control actions under high slip conditions.

Recent ADASs operate at different levels and provide different types of feedback to the driver [1], e.g., by lifting control of the vehicle, applying torque/force on vehicle controls, giving visual or auditory feedback, etc. In this regard, recent studies have shown that user acceptance is highly dependent on the degree of autonomy of the system [26]. Accordingly, to validate the control algorithm at different levels of cooperation between the human and the vehicle controller, three representative schemes have been defined for the design and validation: 1) *full autonomy*, where the controller overrides the human commands; 2) *longitudinal autonomy*, where the NMPCC controls the longitudinal controls and interacts on the steering wheel with the human; and 3) *full interaction*, where the HD controls the car, while the controller corrects the actions on the steering wheel and suggests the correct braking action to the HD through a LED on the dashboard. Note that the last scheme exploits information about future optimal controls, requiring reliable predictions, which are obtained by the combination of accurate modeling, the ellipsoidal constraint, and high control frequency.

The algorithm has been tested on a professional driving simulator (see Fig. 1), to obtain efficient validation with the HD in the loop while ensuring safety and repeatability. Furthermore, co-simulation with high-fidelity vehicle simulation software



Fig. 1. *Compact dynamic driving simulator* is composed of a cockpit, an active belt and active seat system, four legs for tilting and vertical motion, a high-resolution cylindrical screen, and a high-fidelity vehicle simulation software (VI-CRT).

within the simulator makes the difference between the control model and the simulation model comparable to experimental conditions.

Summarizing, the main contributions of this article are: 1) the definition of a contouring controller, tailored for ADAS at the limit of handling and 2) its validation on a high-fidelity, real-time, simulation platform with the HIL. The following features are introduced to implement the proposed innovative strategy.

- 1) *Real-time* capabilities with 50-Hz control frequency and *sufficiently long* prediction horizon, obtained by a specific implementation with a fast NMPC solver [15];
- 2) The capability of high-performance driving, obtained by the definition of a tailored modeling that revealed to be effective in high sideslip conditions;
- 3) The adaptability to unexpected behavior of the HD, obtained by the combination of the contouring formulation and the soft constraint on the track bounds;
- 4) The adaptability to the HD's ability, obtained from the *gg-diagram* soft constraint, which can be used to specify the driver's skills in terms of maximum accelerations.

In Section II, the internal model used by the controller is described and in Section III, the NMPC control problem is defined. In Section IV, a comparison with a commercial lap-time minimizer (VI-MaxPerformance [27]) in a high-fidelity vehicle simulation environment, and an analysis of the impact of the local information about the combined tire slip is reported. In Section V, the experimental environment is presented. Experimental results are reported in Section VI, using the three different interaction schemes, showing the capability of the system to achieve the proposed characteristics. Conclusions are drawn in Section VII.

II. VEHICLE MODEL FOR CONTROL

In the vehicle control literature, different models have been proposed for four-wheel vehicles, ranging from complex four wheels to simple bicycle models. A complete four-wheel model, including load transfer, wheel dynamics, and the full Pacejka tire model is proposed in [22]. Due to the presence of wheel dynamics, however, the ordinary differential equation (ODE) system is stiff, thus computationally unsuitable for long prediction horizons and high-frequency control. Less onerous four-wheel models with simplified longitudinal tire dynamics have been proposed in [23] and [24]. Alternatively,

TABLE I
MAIN MODEL QUANTITIES AND PARAMETER DESCRIPTION

Symbol	Description
x	longitudinal position
y	lateral position
ψ	yaw angle
β	sideslip angle of the vehicle
δ_f	steering angle of the front wheels
m	vehicle mass
I_z	inertia around the vertical axis of the vehicle
a	front wheels to CM longitudinal distance
b	rear wheels to CM longitudinal distance
c	wheels to CM lateral distance
h_{CM}	height of the vehicle's CM
$F_{\{l,c\}\{i,j\}}$	wheels lateral/longitudinal forces in the tire frames ¹
$F_{\{x,y\}\{i,j\}}$	wheels lateral/longitudinal forces in the vehicle's frame
$F_z^{\{i,j\}}$	normal forces on the wheels
F_x^d	downforce in the vehicle's frame
F_z^d	longitudinal drag force in the vehicle frame
s	curvilinear abscissa
e_y	lateral tracking error
e_ψ	angular tracking error

in the presence of simple bicycle models, additional control strategies are adopted, as in [20], where an ellipsoidal friction constraint is included to account for the combined tire behavior, and specific strategies, designed for a vehicle with independent electric motors on the wheels, are implemented to account for the simplifying assumptions.

The proposed internal NMPCC model is a four-wheel vehicle based on the description in [24], supplemented by load transfers, gear shift predictions, longitudinal force saturation, and an ellipsoidal tire friction constraint, improving the overall prediction capabilities of the controller. Moreover, a time-varying parameter has been used to model the lateral force dependency on the longitudinal slip, in order to handle locally the tire forces coupling. The model has been reformulated with respect to the curvilinear abscissa s .

A. Model Dynamics

The vehicle dynamics model is described as

$$\dot{\xi} = \phi(\xi(t), u(t); p(t)) \quad (1)$$

where $\xi(t) \in \mathbb{R}^{n_x}$ is the state of the vehicle, $u(t) \in \mathbb{R}^{n_u}$ is the input, and $p(t) \in \mathbb{R}^{n_p}$ is the time-varying parameter vector. The main quantities and parameters of the model are described in Table I.

The system dynamics ϕ is characterized by the equation of motion of the vehicle center of mass (CM) [23], i.e.,

$$\begin{aligned} \ddot{x} &= \dot{y}\dot{\psi} + \frac{1}{m} \left(\sum_{i,j} F_{x_{i,j}} - F_x^d \right), \quad \ddot{y} = -\dot{x}\dot{\psi} + \frac{1}{m} \left(\sum_{i,j} F_{y_{i,j}} \right) \\ \ddot{\psi} &= \frac{1}{I_z} \left[a \left(\sum_j F_{y_{f,j}} \right) - b \left(\sum_j F_{y_{r,j}} \right) \right. \\ &\quad \left. + c \left(\sum_i F_{x_{i,r}} - \sum_i F_{x_{i,l}} \right) \right]. \end{aligned} \quad (2)$$

¹Subscripts $i \in \{f, r\}$ refer to front or rear wheels, $j \in \{l, r\}$ left or right wheels.

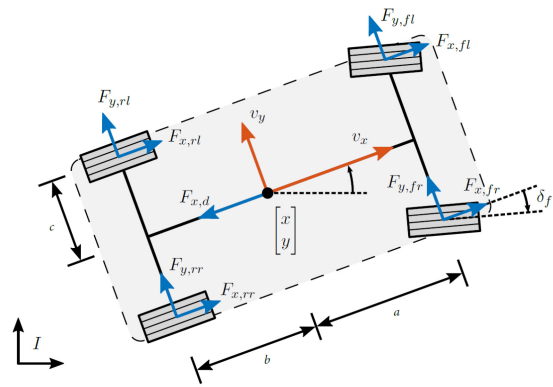


Fig. 2. Quantities defined in the vehicle model for control.

The projection of cornering and longitudinal forces in the vehicle frame yields

$$\begin{aligned} F_{x_{f,j}} &= F_{l_{f,j}} \cos(\delta_f) - F_{c_{f,j}} \sin(\delta_f), \quad F_{x_{r,j}} = F_{l_{r,j}} \\ F_{y_{f,j}} &= F_{l_{f,j}} \sin(\delta_f) + F_{c_{f,j}} \cos(\delta_f), \quad F_{y_{r,j}} = F_{c_{r,j}}. \end{aligned} \quad (3)$$

The steering angle is here assumed to be the same for both front wheels.

In Fig. 2, the physical quantities involved and directions and application point of the forces are described.

The dynamics in the inertial frame are then computed as

$$\begin{aligned} \dot{X} &= \dot{x} \cos(\psi) - \dot{y} \sin(\psi) \\ \dot{Y} &= \dot{x} \sin(\psi) + \dot{y} \cos(\psi) \end{aligned} \quad (4)$$

where (X, Y) is the position of the vehicle's CM in the inertial frame.

Finally, the *sideslip angle* of the vehicle is defined as

$$\beta = \text{atan} \left(\frac{\dot{y}}{\dot{x}} \right). \quad (5)$$

B. Forces

The longitudinal drag force and the downforce are modeled as [28]

$$F_{x,z_i}^d = \frac{1}{2} \rho C_{x,z_i} A_{x,z_i} \dot{x}^2 \quad (6)$$

where ρ is the air density, C_{x,z_i} are the drag coefficients, and A_{x,z_i} are the interested section area. The longitudinal tire forces in each wheel reference frame are computed as

$$F_{l_{i,j}} = f_{\text{eng}_{i,j}} - f_{\text{brk}_{i,j}} \quad (7)$$

where the engine and braking forces are

$$f_{\text{eng}_{i,j}} = \text{sat} \left(\frac{\tau_{\text{eng}_i}}{r_w}, \mu F_{z_{i,j}} \right), \quad f_{\text{brk}_{i,j}} = \text{sat} \left(\frac{\tau_{\text{brk}_i}}{r_w}, \mu F_{z_{i,j}} \right) \quad (8)$$

where μ is the tire friction coefficient, r_w is the wheel radius, and the saturation function is defined as

$$\text{sat}(f_a, f_b) = \frac{f_b}{1 + \exp \left(-5 \left(\frac{f_a}{f_b} - \frac{1}{2} \right) \right)}.$$

Then, the engine and braking torques at the wheels are

$$\tau_{\text{eng}_i} = \gamma_t \left(\tau_{\text{eng}_i}^{\text{MAX}} - \tau_{\text{eng}_i}^{\text{min}} \right) + \tau_{\text{eng}_i}^{\text{min}}, \quad \tau_{\text{brk}_i} = \gamma_b \tau_{\text{brk}_i}^{\text{MAX}} \quad (9)$$

where $\gamma_{t,b}$ are the normalized throttle and braking efforts, $\tau_{\text{brk},i}^{\text{MAX}}$ is the maximum torque given by the braking system to front/rear wheels, and $\tau_{\text{eng},i}^{\text{MAX},\text{min}}$ are the maximum and minimum torque values expressed by the engine at front/rear wheels at a given gear and are changed as a time-varying parameter to model gearshift. To compute the torques in the prediction horizon, an iterative strategy predicting the engine rpm and, hence, gearshift, based on the predicted velocity is used. Specifically, the engine rpm is computed as

$$\text{rpm}^{\text{pred}} = \frac{v_x^{\text{pred}}}{r_w} \frac{\text{diff}_{\text{ratio}}}{\text{gear}_{\text{ratio}}} \frac{60}{2\pi} \quad (10)$$

where $\text{diff}_{\text{ratio}}$ and $\text{gear}_{\text{ratio}}$ are the input/output torque ratios at the differential and at the gearbox (in a specific gear), respectively. The dependence of $\tau_{\text{eng},i}^{\text{MAX},\text{min}}$ with respect to the engine rotational velocity has been neglected.

The normal forces are modeled as

$$\begin{aligned} F_{z_{f,l}} &= \frac{F_{zf}^0}{2} + \frac{F_{zf}^d}{2} - \Delta_{F_z}^{\text{lon}} - \Delta_{F_z}^{\text{lat}} \\ F_{z_{f,r}} &= \frac{F_{zf}^0}{2} + \frac{F_{zf}^d}{2} - \Delta_{F_z}^{\text{lon}} + \Delta_{F_z}^{\text{lat}} \\ F_{z_{r,l}} &= \frac{F_{zr}^0}{2} + \frac{F_{zr}^d}{2} + \Delta_{F_z}^{\text{lon}} - \Delta_{F_z}^{\text{lat}} \\ F_{z_{r,r}} &= \frac{F_{zr}^0}{2} + \frac{F_{zr}^d}{2} + \Delta_{F_z}^{\text{lon}} + \Delta_{F_z}^{\text{lat}} \end{aligned} \quad (11)$$

where the load transfer in steady-state condition Δ_{F_z} is computed by

$$\Delta_{F_z}^{\text{lon}} = F_x^{\text{sat0}} \frac{h_{\text{CM}}}{a+b}, \quad \Delta_{F_z}^{\text{lat}} = F_y^{\text{static}} \frac{h_{\text{CM}}}{2c}. \quad (12)$$

To avoid an algebraic loop in the model, the forces used for the load transfer dynamics computation are F_x^{sat0} (total longitudinal force expressed in the vehicle frame saturated at nominal F_z) and F_y^{static} (the sum of the lateral forces computed at nominal F_z on each wheel).

The lateral forces model is based on *Pacejka's Magic Formula* [29]. The lateral forces expression is

$$F_{c_{i,j}}^0 = D \sin(C \arctan(B \beta_{i,j} - E(B \beta_{i,j} - \arctan(B \beta_{i,j})))) \quad (13)$$

where B is the *stiffness factor*, which regulates the slope, C is the *shape factor*, which determines the *stretching* with respect to the side slip, D is the *peak factor*, which defines the peak value, E is the *curvature factor*, which affects the transition to the constant saturation value, and $\beta_{i,j}$ are the wheels side slip angles. Moreover, local information about the tire coupling has been included through the introduction of the term $G_{y_{k_{i,j}}}$ as a time-varying parameter, computed as [29]

$$G_{y_{k_{i,j}}} = \frac{1}{G_{y_{k0}}} \left\{ \cos(C_{y_k} a \tan(B_{y_k}(\beta_{i,j}) k_{i,j} - E_{y_k}(B_{y_k}(\beta_{i,j})) \cdot k_{i,j} - a \tan(B_{y_k}(\beta_{i,j}) k_{i,j}))) \right\} \quad (14)$$

where $k_{i,j}$ is the current longitudinal slip, B_{y_k} is a function of the lateral slip, E_{y_k} , C_{y_k} , and $G_{y_{k0}}$ are tire dependent parameters. The structure of the lateral force factors leads to a cornering force model of the form

$$F_{c_{i,j}} = F_{c_{i,j}}^0 \cdot G_{y_{k_{i,j}}} = \Psi(\beta_{i,j}, F_{z_{i,j}}; G_{y_{k_{i,j}}}). \quad (15)$$

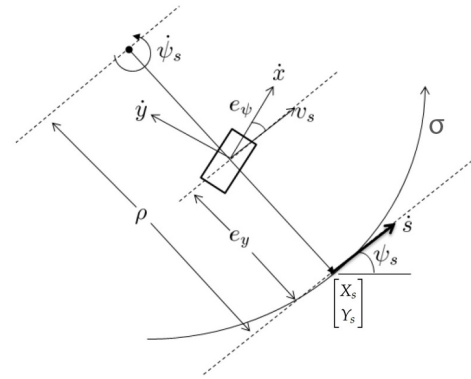


Fig. 3. Spatial coordinates system.

The sideslip angle of each wheel is computed as

$$\beta_{i,j} = \arctan\left(\frac{v_{c_{i,j}}}{v_{l_{i,j}}}\right) \quad (16)$$

where the longitudinal and lateral velocities in each wheel frame are

$$\begin{aligned} v_{c_{f,j}} &= v_{y_{f,j}} \cos(\delta_f) - v_{x_{f,j}} \sin(\delta_f), \quad v_{c_{r,j}} = v_{y_{r,j}} \\ v_{l_{f,j}} &= v_{y_{f,j}} \sin(\delta_f) + v_{x_{f,j}} \cos(\delta_f), \quad v_{l_{r,j}} = v_{x_{r,j}} \end{aligned} \quad (17)$$

and the velocities along x and y directions are

$$\begin{aligned} v_{y_{f,j}} &= \dot{y} + a\dot{\psi}, \quad v_{y_{r,j}} = \dot{y} - b\dot{\psi} \\ v_{x_{i,l}} &= \dot{x} - c\dot{\psi}, \quad v_{x_{i,r}} = \dot{x} + c\dot{\psi}. \end{aligned} \quad (18)$$

C. Spatial Reformulation

The dynamics have been reformulated in spatial coordinates with respect to the arc length s along the track in order to allow time to be a minimization variable, to set the spatial constraints, and to eliminate the dependency on the velocity in the trajectory reference for the controller. This has been previously studied, e.g., in [30] and [31], and the procedure is summarized in the following for completeness. The reference trajectory σ is parameterized in s and the *curvature* of the trajectory $\zeta = (1/\rho)$, where ρ is the instantaneous curvature radius, is locally defined as

$$\zeta = \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{\frac{3}{2}}} \quad (19)$$

where $f' = ([df(s)]/ds)$ $e f'' = ([d^2f(s)]/ds^2)$.

The lateral and angular *tracking errors* are computed as

$$\begin{aligned} e_{\psi} &= \psi - \psi_s \\ e_y &= (Y_s - Y) \cos(\psi_s) + (X_s - X) \sin(\psi_s) \end{aligned} \quad (20)$$

where ψ_s is the reference yaw angle and (X_s, Y_s) is the absolute reference position at the current spatial coordinate while (X, Y) and ψ are the absolute current position and yaw angle (see Fig. 3). The tracking errors dynamics are [32]

$$\begin{aligned} \dot{e}_{\psi} &= \dot{\psi} - \zeta \dot{s} \\ \dot{e}_y &= v_x \sin(e_{\psi}) + v_y \cos(e_{\psi}). \end{aligned} \quad (21)$$

The state vector ξ is differentiated w.r.t s using the *chain rule* as

$$\xi' = \frac{d\xi}{ds} = \frac{d\xi}{dt} \frac{dt}{ds} = \frac{d\xi}{dt} \frac{1}{\dot{s}} = \frac{\dot{\xi}}{\dot{s}} \quad \forall \dot{s} \neq 0 \quad (22)$$

where $\dot{s} = [1/(1 - \zeta e_y)](\dot{x} \cos(e_\psi) - \dot{y} \sin(e_\psi))$.

D. Full Model

The minimization of the travel time over the prediction horizon must be considered in the definition of the cost function to enable the contouring formulation; hence, the time has been included as a state variable with the following ODE $\dot{t} = (1/\dot{s})$. The full-state vector is then given by

$$\xi = [\dot{x}, \dot{y}, \dot{\psi}, e_\psi, e_y, \delta_f, \gamma_t, \gamma_b, t]^T \quad (23)$$

and the input computed by the algorithm is

$$u = [\dot{\delta}_f, \dot{\gamma}_t, \dot{\gamma}_b, \epsilon_{\text{slip}}, \epsilon_{\text{err}}, \epsilon_{\text{gg}}]^T \quad (24)$$

where $\dot{\delta}_f$, $\dot{\gamma}_t$, and $\dot{\gamma}_b$ are the derivatives of the actual input to the vehicle and ϵ are slack variables, whose role will be discussed in Section III. This formulation allows a smooth action of the controller and avoids too aggressive, unrealistic behaviors.

The dynamics equation of the model used in the NMPC algorithm can be compactly written as

$$\xi' = \phi(\xi(s), u(s); p(s)) \quad (25)$$

where $p(s) = [\zeta(s), \tau_{\text{eng},i}^{\text{MAX},\min}(s), G_{y_{k_i,j}}(s)]^T$ is the time-varying parameter vector expressed as function of the curvilinear abscissa s .

III. NMPCC ALGORITHM

In this work, an NMPCC algorithm for assisted driving in a high-performance scenario is designed. The goal of the controller is to compute steering, throttle, brake commands, and their derivatives in a future window, being reliable at the limit of performance. The NMPCC algorithm is based on MATMPC [15], [33], [34], an open-source software built in MATLAB for a real-time NMPC solution. MATMPC has a number of algorithmic modules and can achieve real-time simulation in MATLAB by coding each module in MATLAB API for C. Ease of prototyping and debugging is maintained thanks to the MATLAB environment.

In MATMPC, a nonlinear programming problem (NLP) is formulated at sampling instant i by applying direct multiple shooting [35] to an optimal control problem over the prediction horizon $S = [s_0, s_f]$, which is divided into N shooting intervals $[s_0, s_1, \dots, s_N]$, as follows:

$$\min_{\xi_{|i}, u_{|i}} \sum_{k=0}^{N-1} \frac{1}{2} \|h_k(\xi_{k|i}, u_{k|i})\|_W^2 + \frac{1}{2} \|h_N(\xi_{N|i})\|_{W_N}^2 \quad (26a)$$

$$\text{s.t. } 0 = \xi_{0|i} - \hat{\xi}_0 \quad (26b)$$

$$0 = \xi_{k+1|i} - \phi_k(\xi_{k|i}, u_{k|i}; p_{k|i}), \quad k \in [0, N-1] \quad (26c)$$

$$r_{k|i} \leq r_k(\xi_{k|i}, u_{k|i}) \leq \bar{r}_{k|i}, \quad k \in [0, N-1] \quad (26d)$$

$$r_{N|i} \leq r_N(\xi_{N|i}) \leq \bar{r}_{N|i} \quad (26e)$$

where $\xi_{|i} = (\xi_{0|i}^\top, \xi_{1|i}^\top, \dots, \xi_{N|i}^\top)^\top$, and $u_{|i} = (u_{0|i}^\top, u_{1|i}^\top, \dots, u_{N-1|i}^\top)^\top$; and $\hat{\xi}_0$ represents the measurement of the current state. System states $\xi_{k|i} \in \mathbb{R}^{n_\xi}$ are defined at the discrete arc-length point s_k for $k = 0, \dots, N$ and the control inputs $u_{k|i} \in \mathbb{R}^{n_u}$ for $k = 0, \dots, N-1$ are piecewise constant. Their definitions are given in (23) and (24). Here, (26d) is defined by $r(\xi_{k|i}, u_{k|i}) : \mathbb{R}^{n_\xi} \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_r}$ and $r(\xi_{N|i}) : \mathbb{R}^{n_\xi} \rightarrow \mathbb{R}^{n_r}$ with lower and upper bounds $r_{k|i}, \bar{r}_{k|i}$. Equation (26c) refers to the *continuity constraint* where $\phi_k(\xi_{k|i}, u_{k|i}; p_{k|i})$ is a numerical integration operator that solves (25) with initial condition $\xi(0) = \xi_{0|i}$ and returns the solution at s_{k+1} .

The real-time iteration (RTI) scheme [16] is employed to reduce the time required to solve (26). The RTI scheme performs a single sequential quadratic programming (QP) iteration to solve (26). By this means, (26) is linearized only once and a QP problem is solved at each sampling instant. Details on the QP formulation can be found in [15]. Moreover, a nonuniform grid (NUG) strategy [36] has been used for lowering the computational burden and let the controller predict a sufficiently long horizon (as specified in Section IV-A). The local convergence of the algorithm is ensured by the RTI guarantees on contractivity and boundness of the loss of optimality compared to optimal feedback control [37], [38]. Specifically, the RTI scheme achieves convergence of the NMPC problem *on-the-fly*, i.e., as the system dynamics evolve [38].

To effectively generate the vehicle commands, the cost function for the NMPC is defined as

$$\begin{aligned} h_k(\xi_k, u_k) &= [\beta, \gamma_t \cdot \gamma_b, \zeta \cdot \gamma_t, t, \dot{\delta}_f, \dot{\gamma}_t, \dot{\gamma}_b, \epsilon_{\text{slip}}, \epsilon_{\text{err}}, \epsilon_{\text{gg}}]^T \\ h_N(\xi_N) &= [\beta, \gamma_t \cdot \gamma_b, \zeta \cdot \gamma_t, t, e_y - e_y^{\text{ref}}, \dot{e}_y, e_\psi - e_\psi^{\text{ref}} \\ &\quad + \beta, \dot{e}_\psi]^T. \end{aligned} \quad (27)$$

The penalty on the sideslip β is used to choose how restrained the behavior of the vehicle should be. The cost $\gamma_t \cdot \gamma_b$ penalizes simultaneous throttling and braking, whereas the $\zeta \cdot \gamma_t$ cost is used to make the controller accelerate smoothly while exiting the curves. The objective variable time t supports the computation of a time-minimizing path and its weight can be used to tune the importance of the performance index. The terms on the inputs ensure a smooth control action. The slack variables are used to define the *soft constraints* [39] discussed in the next paragraph. Finally, the terms related to errors e_y and e_ψ , used only as terminal objective variables, are introduced to integrate information about the trajectory over the prediction horizon.

The constraints are defined as

$$\begin{aligned} r_k &= \left[\delta_f, \gamma_t, \gamma_b, \dot{\delta}_f, \dot{\gamma}_t, \dot{\gamma}_b, \epsilon_{\text{slip}}, \epsilon_{\text{err}}, \epsilon_{\text{gg}}, \beta + \epsilon_{\text{slip}} \right. \\ &\quad \left. e_y + \epsilon_{\text{err}}, \left(\mu_x \frac{\ddot{x}_{\text{ext}}}{g} \right)^2 + \left(\mu_y \frac{\ddot{y}_{\text{ext}}}{g} \right)^2 + \epsilon_{\text{gg}} \right]^T \\ r_N &= [\delta_f, \gamma_t, \gamma_b]^T \end{aligned} \quad (28)$$

where the constraints on δ_f , γ_t , and γ_b are intrinsic bounds of the actual vehicle controls, while those on $\dot{\delta}_f$, $\dot{\gamma}_t$, and $\dot{\gamma}_b$

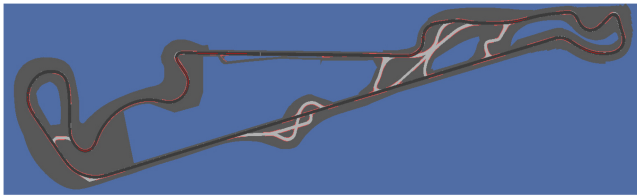


Fig. 4. Paul Richard race track used for testing. The chosen course is highlighted in black.

are added in order to improve the smoothness of the computed inputs and can be used to easily tune the aggressivity of the NMPC driving commands. Also, the slack variables have been constrained in order to help the optimization procedure restricting the *search space* of the inputs. Finally, three *soft constraints* have been added to make the controller predict a trajectory with the desired features while supporting the robustness of the overall procedure. In fact, due to the interaction with the human actually driving on the simulator, the behavior of the vehicle could be significantly different from the predicted one, and the presence of hard constraints would easily lead to failures in the NLP solution. Specifically, the first soft constraint is introduced on the sideslip of the vehicle in order to force the controller to regain control of the vehicle in case of skidding; the second one is used to firmly correct the human trajectory in case of out-of-the-track future situations. Finally, the last soft constraint is designed to make the controller respect the required *gg diagram*, which represent the maximum combined longitudinal-lateral acceleration that can be induced by the combined longitudinal-lateral behavior of the tire forces [40]. μ_x and μ_y are the longitudinal and lateral friction coefficients of the tires, respectively, whereas the considered accelerations on the vehicle are

$$\begin{aligned}\ddot{x}_{\text{ext}} &= \frac{\sum_{i,j} F_{x_{i,j}} - F_d^x}{m} \\ \ddot{y}_{\text{ext}} &= \frac{\sum_{i,j} F_{y_{i,j}}}{m}.\end{aligned}\quad (29)$$

At the i th sampling instant, the control input is updated by

$$u^{i*} = u^{i-1*} + \Delta u^{i*} \quad (30)$$

where Δu^{i*} is the solution of the QP. The first sample of u^{i*} is applied to the vehicle, the prediction horizon is shifted forward and the optimization procedure is repeated with the updated state measurement.

IV. OFFLINE PERFORMANCE VALIDATION

Since the controller must be able to handle challenging high-performance situations, a validation test has been made, by comparing with a commercial dynamic minimum lap-time tool, VI-MaxPerformance. VI-MaxPerformance is based on an iterative procedure that acts on the velocity profile to minimize lap time [27]. The comparison has been made on virtualization of the Paul Ricard race track (Le Castellet, France), shown in Fig. 4, which is 5665-m long.

The vehicle dynamics are computed using the simulation tool VI-CarRealTime (VI-CRT), a multibody software

specifically designed to reproduce vehicle behavior for high-performance driving in real time [27]. The underlying simulation model is composed of 14 degrees of freedom, 6 for the chassis and 2 for each wheel, and it includes comprehensive dynamics of tire, chassis, suspensions, brakes, engine, and transmission.

The co-simulation is performed in Simulink, connecting a VI-CRT simulation block with the NMPC controller. In particular, the controls δ_f , $\dot{\gamma}_t$, and $\dot{\gamma}_b$ computed by the NMPC controller are integrated and given as inputs to the simulation block. VI-CRT is used to simulate the dynamics of the vehicle at 1000 Hz while the control action is updated by the NMPC controller at $f_c = 50$ Hz.

A. Controller Setup

The NMPC tool MATMPC has been set to use Explicit Runge-Kutta 4 integrator, HPIPM as the QP sparse solver [41] with the RTI scheme [16]. To achieve satisfactory performance in trajectory planning, the prediction horizon should be long enough to predict the entire dynamics of both the longest corner on the track, to start positioning and braking in advance, and of consecutive corners in the chicanes, to calculate a trajectory consistently. However, to maintain real-time capabilities of the controller, the number of shooting points was limited to $N = 140$. Moreover, to maintain good tracking performances, a dense integration at the beginning of the control horizon is needed. Therefore, NUG steps [36] have been set as follows:

$$G = [2, 3, 4, 6, 8, 10, 13 : 3 : 390, 392 : 2 : 396, 397 : 1 : 400] \quad (31)$$

where $a : b : c$ means from a to c with step b in meters, enabling a prediction length of 400 m on track. Note that the first step, that is 2-m long, allows computing coherently the first input, i.e., valid for more than one control period ($T_c = f_c^{-1} = 20$ ms), until the velocity is less than 100 (m/s). The implemented configuration, using the RTI scheme and NUG, is specifically realized as to effectively control the vehicle at the limit of handling. In this conditions, in fact, the control frequency must be high enough, e.g., to react in a high sideslip condition with a countersteering action, while the prediction horizon must be long enough, as stated before. To ensure both features, a very fast solution method is implemented through the RTI scheme, while a reduction of the needed points in the prediction horizon is achieved by NUG.

The tuning of the NMPC controller parameters is essential to obtain an effective and proficient controller. The weight choice is accomplished in two phases: a first tuning operation is carried out in offline simulation, in order to obtain the best achievable performance, while a further refinement is obtained during HIL experiments, in order to enhance the interaction between the controller and the HD. The choice of weights initially obtained on the offline simulation, indeed, led to rapid and abrupt changes in control actions and, when interacting with the driver on the steering wheel, generated annoying high-frequency vibrations. In this sense, increasing the weights on the control derivatives promoted smoother controller behavior.

The described empirical analysis has been conducted and the weights for the cost function have been manually set as

$$W = \text{diag}\left(\left[5 \cdot 10^3, 10^5, 6 \cdot 10^{-1}, 10^1, 4 \cdot 10^1, 10^{-2}, 10^{-2}, 3 \cdot 10^2, 5 \cdot 10^{-1}, 10^2\right]\right). \quad (32)$$

In particular, the sideslip weight and the time weight have been set as to maintain the correct tradeoff between a restrained behavior and high performance, a high weight on $\gamma_t \cdot \gamma_b$ is used for preventing contemporary throttling and braking, the weight on $\zeta \cdot \gamma_t$ has been set to obtain a throttle action at the curve exit that does not lead to tire spinning, the input derivatives have been penalized to obtain the desired smoothness of the control actions, the weights on the slack variables for sideslip and lateral errors have been increased until the controller firmly corrects the vehicle behavior in unstable or out-of-the-track future situations, and finally, a high penalty on the *gg constraint* slack variable has been set to comply with the desired maximum performance.

The terminal weighting matrix has then been set as

$$W_N = \text{diag}\left(\left[5 \cdot 10^3, 10^5, 6 \cdot 10^{-1}, 10^1, 10^2, 10^{-2}, 10^3, 10^0\right]\right). \quad (33)$$

The terminal weights have been set unchanged for the present variables already analyzed, whereas the weights on the reference errors have been set as to address that the predicted trajectory concludes, at the end of the prediction horizon, within a reference corner-cutting trajectory, whose generation is discussed in the next paragraph.

The reference trajectory for the final point of the prediction horizon employed in this test has been computed using the *Corner Cutting* procedure embedded in the commercial tool VI-Road. The process, given the centerline, the track width, and the vehicle width, computes a path that minimizes

$$J_{\text{path}} = \int \left(w_l \cdot \iota(s)^2 + w_\zeta \cdot \zeta(s)^2 \right) ds \quad (34)$$

where $\iota(s)$ is a measure of the path length, $\zeta(s)$ is the local path curvature, and $w_l = 6 \cdot 10^{-3}$ and $w_\zeta = 4$ are the respective weights. The minimization is constrained in order to maintain the vehicle within the track.

Finally, the bounds in (28) have been set as

$$\begin{aligned} \underline{r}_k &= \left[-\frac{\pi}{20}, 0, 0, -2, -10, -100, -2\pi, -10^5, -10^5, -10^5, \right. \\ &\quad \left. -\frac{\pi}{15}, -5, 0 \right]^\top \\ \bar{r}_k &= \left[+\frac{\pi}{20}, 1, 1, +2, 1.1, +100, +2\pi, +10^5, +10^5, +10^5, \right. \\ &\quad \left. +\frac{\pi}{15}, +5, 1.9 \right]^\top \\ \underline{r}_N &= \left[-\frac{\pi}{20}, 0, 0 \right]^\top, \quad \bar{r}_N = \left[+\frac{\pi}{20}, 1, 1 \right]^\top \end{aligned} \quad (35)$$

where the bound $\delta_f \in [-(\pi/20), +(\pi/20)]$ is the limit of the vehicle steering system; γ_t and $\gamma_b \in [0, 1]$ are normalized; the bounds on the control derivatives are set as to ensure a smooth action of the controller; the bounds on the slack variables $\epsilon \in [-10^5, +10^5]$ are set to increase the robustness of the optimization procedure; the limit on the soft

TABLE II
COMPARISON BETWEEN NMPCC AND VI-MXP

	NMPCC	VI-MXP
lap time	112.16s	112.46s
mean longitudinal velocity	50.3 m/s	50.0m/s
maximum lateral acceleration	1.9 m/s ²	1.9 m/s ²
maximum sideslip angle	3.4deg	5.1deg
maximum throttle and braking derivative	37.6 s ⁻¹	200 s ⁻¹

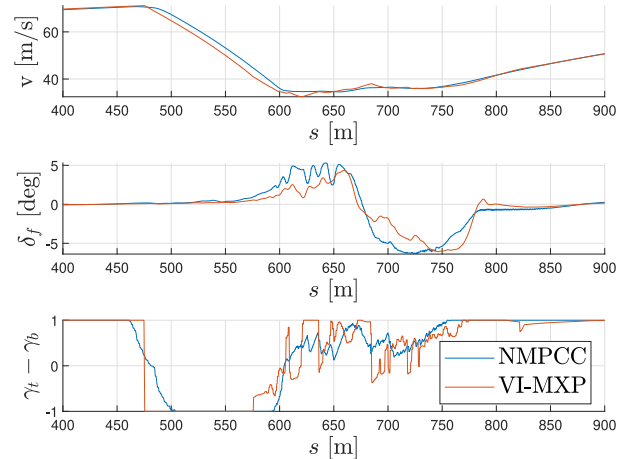


Fig. 5. Velocity maintained and the controls applied by the NMPCC and the commercial tool VI-MXP on the first chicane of the track (between 400 and 900 m).

constraint $\beta + \epsilon_{\text{slip}} \in [-(\pi/15), +(\pi/15)]$ is set to heavily penalize excessive oversteering and the one on the lateral error $e_y + \epsilon_{\text{err}} \in [-5, +5]$ as to force the vehicle within the track boundaries; and finally, the bound on the soft constraint for the *gg* is set as the vehicle acceleration limit.²

B. Comparison With VI-MaxPerformance

The results on the *Paul Ricard* track show the capability of the controller to achieve performance comparable with those of the commercial tool VI-MaxPerformance (VI-MXP). In particular, VI-MXP has been tested both on the *corner-cutting* path and on the NMPCC controller trajectory, leading to a lap time of 113.27 and 112.46 s, respectively, while for the NMPCC controller the lap-time is 112.16 s. Interestingly, the NMPCC trajectory results to be faster than the corner-cutting one also for the commercial lap time minimizer (see Table II).

The commands given by VI-MXP following the NMPCC path and the NMPCC are shown in Fig. 5, where it can be observed that the NMPCC exhibits a smoother behavior than VI-MXP. This affects in particular the velocity profile, which is smoother and more representative of a human behavior in the NMPCC case. Note that the smoothness of the control actions is fundamental for the assistance system in order to propose consistent and human-like behavior.

The mean computational time for the controller is 12 ms while the maximum is 16 ms, which is compatible with the 50-Hz control frequency in this real-time application.

²Note that, as the vehicle used for testing is a race car with aerodynamic downforce, the limit is greater than 1.

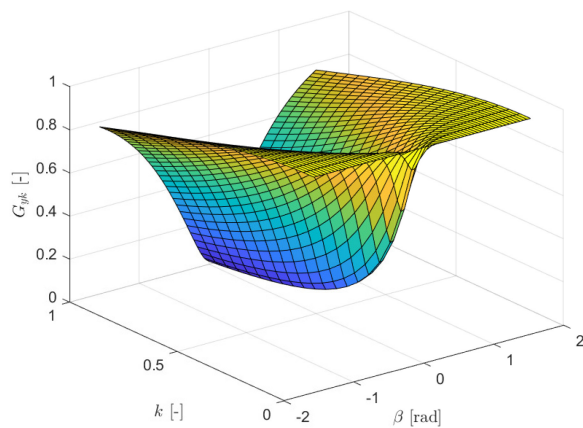


Fig. 6. Representation of the combined lateral coefficient G_{yk} w.r.t. lateral and longitudinal slip of the tire.

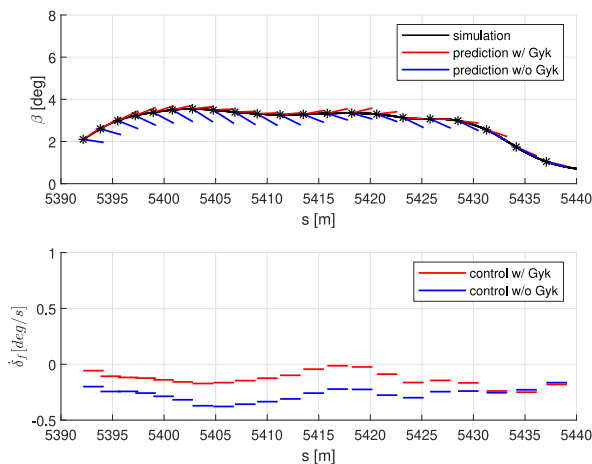


Fig. 7. One-step-ahead sideslip predictions and the steering velocity control in Sim w/G_{yk} . An offline evaluation of the same quantities computed without G_{yk} is overimposed for comparison.

C. Impact of the Local Information on the Combined Tire Condition

Controlling a vehicle at the limit of handling requires a sufficiently accurate knowledge of tire forces. In fact, under conditions of high longitudinal forces, e.g., originated by accelerating after a curve or braking before it, a significant reduction in tire lateral force occurs, with a relevant impact on vehicle behavior. The information described through the time-varying parameter G_{yk} , see (14) and Fig. 6, is then crucial to correctly characterize such conditions. Since G_{yk} varies at very high frequency, the parameter is provided to the controller and processed as linearly fading in the first 5 steps of the prediction horizon. The impact of such a solution is evaluated on the last turn of the track through two closed-loop simulations obtained with and without scaling of the lateral forces with G_{yk} (Figs. 7 and 8), namely, Sim w/G_{yk} and Sim $w/o G_{yk}$. In both cases, one-step-ahead sideslip predictions and steering derivative control are shown. An offline evaluation of the same quantities computed with the alternative approach, i.e., with G_{yk} in the Sim $w/o G_{yk}$ and vice versa, is overimposed for comparison.

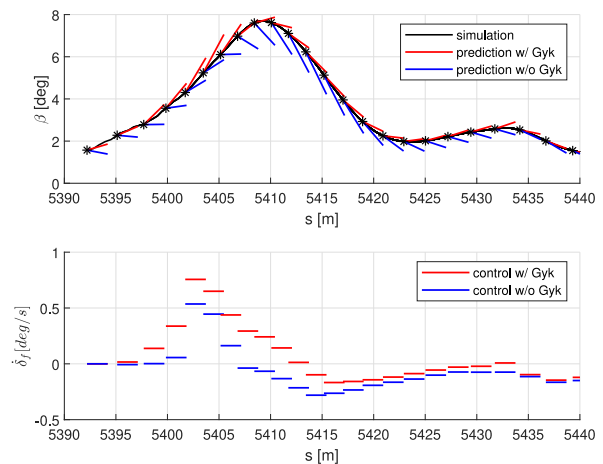


Fig. 8. One-step-ahead sideslip predictions and the steering velocity control in Sim $w/o G_{yk}$. An offline evaluation of the same quantities computed with G_{yk} is overimposed for comparison.

It can be noticed that the one-step-ahead sideslip predictions obtained without G_{yk} is highly underestimated (blue line), leading to a high side slip condition in the case G_{yk} is not used in closed loop.

In both cases, without this information, the sideslip evolution is highly underestimated. This fact leads to a curve exit traveled with the expected restrained behavior (sideslip around 4°) in Test $W/$, and to a loss of control of the rear of the car in Test W/O .

The effect of this predictions on the computed steering derivative control is clearly visible, which is particularly important in the application at hand since it is used to generate steering feedback to the driver (see Section VI-C).

V. EXPERIMENTAL SETUP

The NMPCC algorithm has been developed for interacting with the HD on a *compact dynamic driving simulator* (shown in Fig. 1). The driving simulator is composed by a mockup of a cockpit, an active belt and active seat system, four legs for tilting and vertical motion, a high-resolution cylindrical screen, and a high-fidelity vehicle simulation software (VI-CRT), which computes the vehicle's dynamics at 1000 Hz. The steering wheel is equipped with an electric motor to feed torque feedback to the driver. The NMPCC controller, running in Simulink on a dedicated PC, receives the current vehicle states from the simulator, computes the controls δ_f^* , γ_i^* , and $\dot{\gamma}_b^*$ at 50 Hz, which are sent to the simulator and, if necessary, integrated at 1000 Hz. The obtained control actions are then used differently depending on the chosen interaction scheme. The communication between the simulator and the controller is based on UDP protocol. MATMPC configuration is as described in Section IV. The controller runs on a PC in WINDOWS 10, with Intel Core i7-7700K @ 4.20-GHz CPU.

VI. EXPERIMENTAL RESULTS

Different experimental activities, incrementally involving the HD, have been carried on in order to validate the algorithm in representative interaction schemes and to test the specific

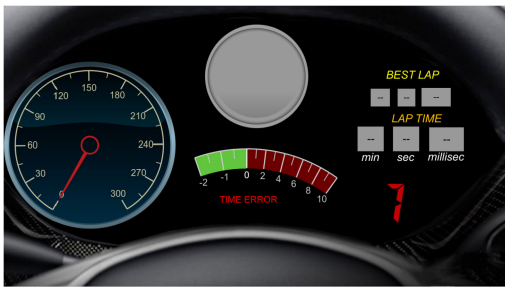


Fig. 9. Visualization of the dashboard proposed to drivers in full interaction scheme. The dashboard shows, starting from the left side: the current speed reached by the vehicle; the light used for visual feedback about braking; the current deviation with respect to the best lap time obtained offline with a fully autonomous configuration; the current and the best lap time in the whole driving session; and the current gear state.

features described in the introduction. In particular, the three different interaction schemes used are as follows.

- 1) *Full Autonomy Scheme*: In this setup, where the controller is fully in charge of the vehicle, the *tunability* characteristic, i.e., the possibility to adapt to different driving skills, has been tested.
- 2) *Longitudinal Autonomy Scheme*: The test, where the NMPCC controls the longitudinal commands and interact on the steering wheel with the human, has been used to evaluate the controller *adaptability* to the HD unexpected behavior and the reliability of the NMPCC predictions.
- 3) *Full Interaction Scheme*: In this scheme, the HD controls the car, while the controller corrects/suggests the actions on the steering wheel and on the dashboard. The test case is used to demonstrate the NMPCC *support capability* in highly unstable and fast dynamics vehicle behavior.

The system has been tested by different drivers, from novice to expert, with similar results. Specifically, the reported experiments have been conducted with a nonprofessional driver. The interaction between the system and the human is given, for *longitudinal autonomy scheme* and *full interaction scheme*, by a steering axis torque τ_s , generated by a proportional controller (with $P = 3.5$) on the control input δ_f computed by the NMPCC. In addition to the steering torque, in the *full interaction scheme*, the algorithm leverages the predictive capabilities of the NMPCC to provide the driver with visual feedback about the exact moment to brake. Specifically, a light within a dashboard (see Fig. 9) is used as follows: when the blue light is on, no brake actions are required from drivers in the very next future. A yellow light turns on 1 s before the predicted brake action, in order to prepare the driver for braking and reduce reaction time. A red light turns on when the driver is supposed to brake, given the current vehicle state.

A. Full Autonomy Scheme

In this setup, we highlight the tunability of the controller, by manipulating the constraint on the *gg-bound*. The *gg* is the representation of longitudinal acceleration with respect to lateral acceleration, used to characterize the performance of a vehicle, and the skill of the driver. This bound is manipulated here

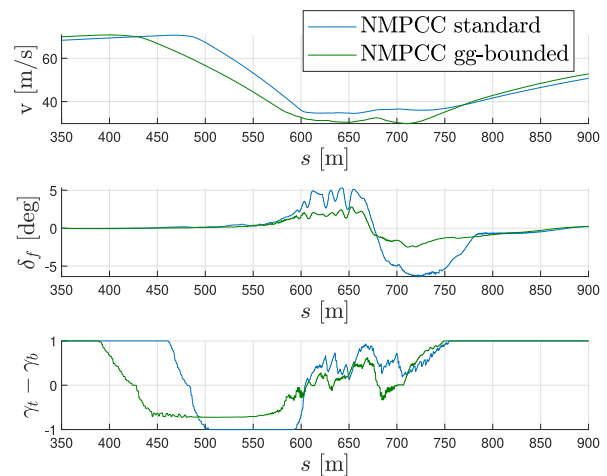


Fig. 10. Velocity and command profiles for the *standard* and *gg-bounded* configurations. Due to the request for less acceleration in the latter configuration, the braking action starts earlier and the curve is traveled at a lower velocity.

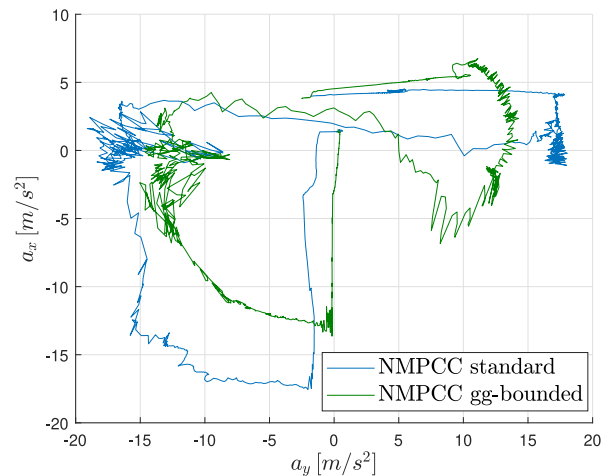


Fig. 11. *gg-diagram* for the *standard* and *gg-bounded* configurations, showing longitudinal acceleration with respect to lateral acceleration. The effect of the constraint is to force the accelerations within a smaller ellipsoid.

to modify the behavior of the controller to emulate different abilities of the HD, making the interaction more effective. In this test, the *standard* configuration described in Section IV-A is compared with a configuration where the soft constraint on the *gg-bound* [see (28)] has been reduced to $1.3g$. This configuration shows the possibility to reduce the aggressiveness of the controller in a direct way. In Fig. 10, the velocity and the controls for the first chicane are shown, while in Fig. 11, the effect of this reduction is shown in terms of *gg-diagram*. In particular, the constrained configuration brakes less but starting earlier, to achieve a lower longitudinal acceleration, and travels through the curve at lower speed, to maintain lower lateral acceleration.

B. Longitudinal Autonomy Scheme

In this configuration, the NMPCC controls the throttle and brake, while the HD forces the vehicle onto an unexpected trajectory. In this case, the controller must adapt the predicted trajectory and velocity to the actual state of the vehicle. In

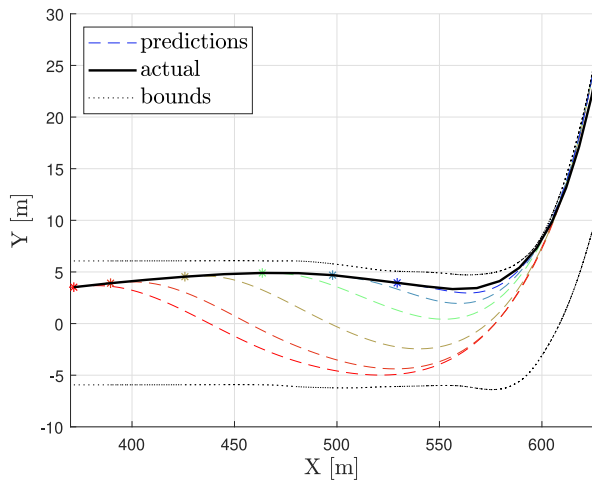


Fig. 12. Predictions of the future trajectory in the prediction horizon for the first chicane of the track. The color changes as the time passes, to show the predictions in different time instants. In the prediction horizon, the controller calculates faster trajectories, but constantly adapts to the actual position.

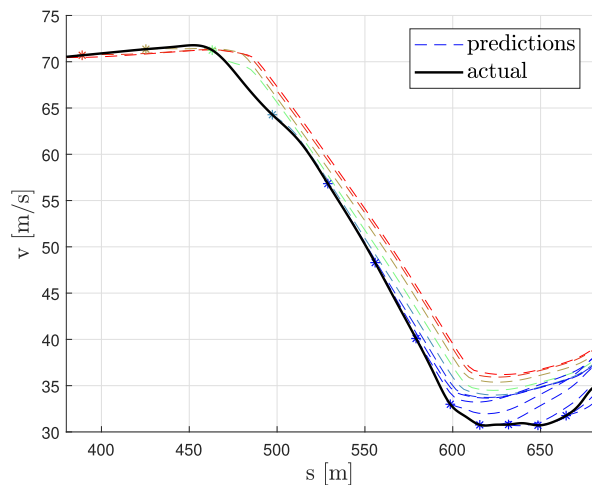


Fig. 13. Predictions of the velocity in the prediction horizon for the first chicane of the track. The color changes as the time passes, to show the predictions in different time instants. In the prediction horizon, the controller calculates faster velocity profiles, but adapts constantly to lower velocities as the human approaches the curve in unexpected positions.

Fig. 12, the trajectory predictions of the NMPCC in the first chicane of the track are shown. In particular, the NMPCC predicts a control sequence that should move the vehicle toward the outside of the track before the curve, with the aim of driving through the curve at a higher speed. However, since the HD does not behave as predicted, the NMPCC is forced to adapt the trajectory and gradually reduce the speed as the car approaches the curve (shown in Fig. 13). In all predictions, however, the controller computes trajectories at the limit of handling, always around the *soft* acceleration limit of 1.9 g (see Fig. 14).³ NMPCC predictions and controls, therefore, constantly adapt to the unexpected behavior of the HD, while maintaining a constant performance.

³In the peak of the curve the longitudinal acceleration is 0, hence the *gg* bound applies directly to the lateral acceleration.

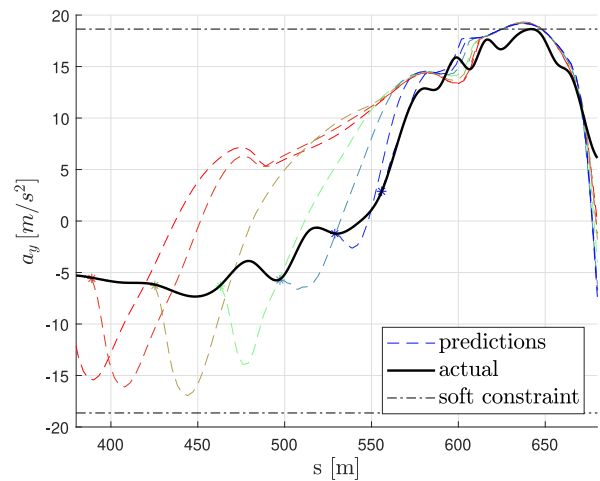


Fig. 14. Predictions of the lateral acceleration in the prediction horizon for the first chicane of the track. The color changes as the time passes, to show the predictions in different time instants. In the prediction horizon, the controller computes trajectories at the limit of handling, always around the *soft* acceleration limit of 1.9 g.

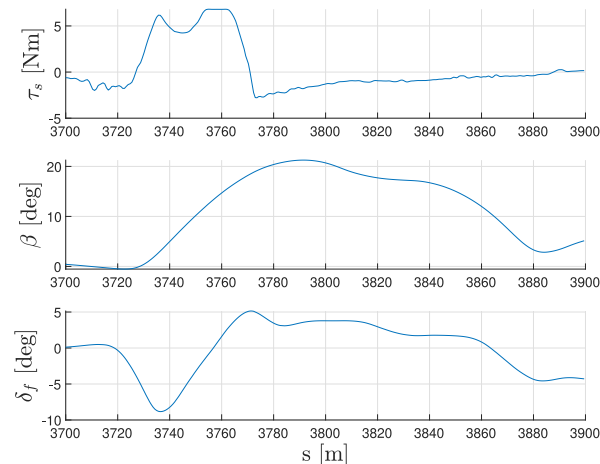


Fig. 15. Steering axis torque, the sideslip angle and the steering angle on the ground signals during the oversteer maneuver at the exit of the longest straightway of the track. The controller steering command leads to a torque on the steering wheel that anticipates the sideslip growth, hence promoting the counter steering action in advance.

C. Full Interaction Scheme

In this experiment, in which the HD has full control of the vehicle, an oversteer test has been performed in order to evaluate the controller's support capabilities at the limit of handling of the vehicle. During a high-velocity turn at the end of a straightway, oversteer occurs due to the HD acting too fast on the steering wheel and an analysis of the controller activity in this situation was conducted. In Fig. 15, the steering axis torque, the sideslip angle β , and the ground steering angle δ_f are shown, whereas in Fig. 16, a visualization of the maneuver is represented. As the vehicle's sideslip is starting to increase (at 3730 m), the controller leads to a torque on the steering wheel in opposition to the human action. The torque, in fact, is anticipating the growth of the sideslip, thus promoting the counter steering action in advance. Interestingly, once the steering wheel is in a counter-steering position (at

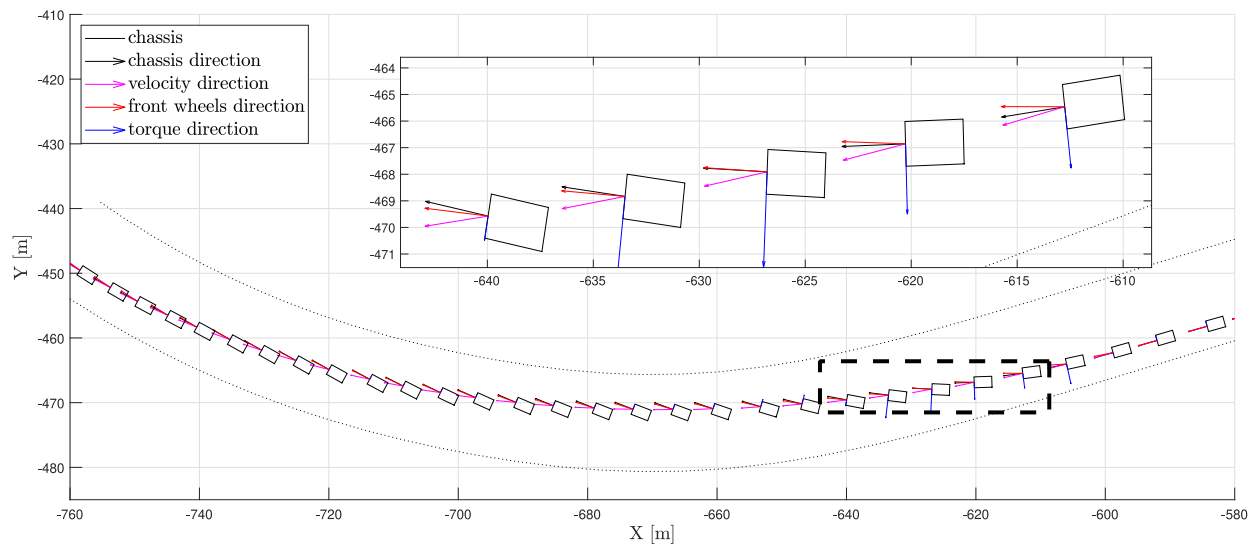


Fig. 16. Visualization of the oversteer maneuver at the exit of the longest straightway of the track. The torque is promoting the counter steering action in advance, then, once a stable skidding condition is reached, the controller torque goes to zero.

3770 m), the torque is clearly lowered, hence maintaining the sideslip condition *stable*. Notably, the traveling velocity during this maneuver is around 45 m/s. In this scenario, the NMPC control strategy demonstrates its ability to support the HD in highly unstable vehicle behavior with fast dynamics, promoting its use even in extreme situations.

VII. CONCLUSION

In this article, the implementation of a nonlinear model predictive contouring controller for driving assistance in high-performance scenarios has been proposed. Explicit characteristics for the controller have been stated, and their fulfillment has been proven through different experimental activities on a professional driving simulator, with increasing HD involvement. The validation has been made on a challenging racetrack, testing the controller at the handling limit of the vehicle.

The main results of this article are the following: 1) a virtual driver faster than a commercial solution has been obtained; 2) importance of the local information on the combined tire condition has been demonstrated; 3) the possibility to use a *gg-bound* to adapt the controller behavior to different driver skills has been exhibited; 4) the controller adaptability to unexpected HD behavior has been verified; and 5) the controller capability to stabilize the vehicle at the limit of handling properly interacting on the steering wheel with the HD has been established.

Future research will comprehend the development of a structured torque generation procedure and the application of the implemented controller for the design of an adaptive coaching procedure.

REFERENCES

- [1] A. Lindgren and F. Chen, "State of the art analysis: An overview of advanced driver assistance systems (ADAS) and possible human factors issues," in *Proc. Swedish Human Factors Netw. (HFN) Conf.*, Linköping, Sweden, Apr. 2006, pp. 38–50.
- [2] C. E. Beal and J. C. Gerdes, "Model predictive control for vehicle stabilization at the limits of handling," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 4, pp. 1258–1269, Jul. 2013.
- [3] A. Balachandran, M. Brown, S. M. Erlien, and J. C. Gerdes, "Predictive haptic feedback for obstacle avoidance based on model predictive control," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 1, pp. 26–31, Jan. 2016.
- [4] N. Chowdhri, L. Ferranti, F. S. Iribarren, and B. Shyrokau, "Integrated nonlinear model predictive control for automated driving," *Control Eng. Pract.*, vol. 106, Jan. 2021, Art. no. 104654.
- [5] C. Guo, C. Sentouh, J. Popieul, and J. Haué, "MPC-based shared steering control for automated driving systems," in *Proc. IEEE Int. Conf. Syst. Man Cybern. (SMC)*, 2017, pp. 129–134.
- [6] Z. Ercan, A. Carvalho, H. E. Tseng, M. Gökaşan, and F. Borrelli, "A predictive control framework for torque-based steering assistance to improve safety in highway driving," *Veh. Syst. Dyn.*, vol. 56, no. 5, pp. 810–831, 2018.
- [7] F. Flemisch, M. Heesen, T. Hesse, J. Kelsch, A. Schieben, and J. Beller, "Towards a dynamic balance between humans and automation: Authority, ability, responsibility and control in shared and cooperative control situations," *Cogn. Technol. Work*, vol. 14, no. 1, pp. 3–18, 2012.
- [8] D. A. Abbink, M. Mulder, and E. R. Boer, "Haptic shared control: Smoothly shifting control authority?" *Cogn. Technol. Work*, vol. 14, no. 1, pp. 19–28, Mar. 2012. [Online]. Available: <https://doi.org/10.1007/s10111-011-0192-5>
- [9] M. Marcano, S. Díaz, J. Pérez, and E. Irigoyen, "A review of shared control for automated vehicles: Theory and applications," *IEEE Trans. Human-Mach. Syst.*, vol. 50, no. 6, pp. 475–491, Dec. 2020.
- [10] W. Schwarting, J. Alonso-Mora, L. Pauli, S. Karaman, and D. Rus, "Parallel autonomy in automated vehicles: Safe motion generation with minimal intervention," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2017, pp. 1928–1935.
- [11] D. A. Abbink and M. Mulder, "Exploring the dimensions of haptic feedback support in manual control," *J. Comput. Inf. Sci. Eng.*, vol. 9, pp. 1–9, Mar. 2009.
- [12] B. Soualmi, C. Sentouh, J.-C. Popieul, and S. Debernard, "A shared control driving assistance system : Interest of using a driver model in both lane keeping and obstacle avoidance situations," *IFAC Proc. Vol.*, vol. 46, no. 15, pp. 173–178, 2013.
- [13] R. C. Macarthur, "Vehicle race track driving assistance," U.S. Patent 9 475 494, Oct. 25, 2016.
- [14] E. Remolina, S. Ramachandran, D. Fu, R. Stottler, and W. R. Howse, *Intelligent Simulation-Based Tutor for Flight Training*, Defense Techn. Inf. Center, Fort Belvoir, VA, USA, 2004.
- [15] Y. Chen, M. Bruschetta, E. Picotti, and A. Beghi, "MATMPC—A MATLAB based toolbox for real-time nonlinear model predictive control," in *Proc. 18th Eur. Control Conf. (ECC)*, 2019, pp. 3365–3370.
- [16] M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer, "Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations," *J. Process Control*, vol. 12, no. 4, pp. 577–585, 2002.

- [17] X. Huo, H. R. Karimi, X. Zhao, B. Wang, and G. Zong, "Adaptive-critic design for decentralized event-triggered control of constrained nonlinear interconnected systems within an identifier-critic framework," *IEEE Trans. Cybern.*, early access, Jan. 5, 2021, doi: 10.1109/TCYB.2020.3037321.
- [18] A. Liniger, A. Domahidi, and M. Morari, "Optimization-based autonomous racing of 1:43 scale RC cars," *Optim. Control Appl. Methods*, vol. 36, pp. 628–647, Sep. 2015.
- [19] R. Verschuere, M. Zanon, R. Quirynen, and M. Diehl, "Time-optimal race car driving using an online exact hessian based nonlinear MPC algorithm," in *Proc. Eur. Control Conf. (ECC)*, 2016, pp. 141–147.
- [20] J. Kabzan *et al.*, "AMZ driverless: The full autonomous racing system," *J. Field Robot.*, vol. 37, no. 7, pp. 1267–1294, 2020.
- [21] D. Lam, C. Manzie, and M. Good, "Model predictive contouring control," in *Proc. 49th IEEE Conf. Decis. Control (CDC)*, 2010, pp. 6137–6142.
- [22] J. V. Frasch *et al.*, "An auto-generated nonlinear MPC algorithm for real-time obstacle avoidance of ground vehicles," in *Proc. Eur. Control Conf. (ECC)*, 2013, pp. 4136–4141.
- [23] A. Carvalho, Y. Gao, A. Gray, H. E. Tseng, and F. Borrelli, "Predictive control of an autonomous ground vehicle using an iterative linearization approach," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, 2013, pp. 2335–2340.
- [24] M. Bruschetta, E. Picotti, E. Mion, Y. Chen, A. Beghi, and D. Minen, "A nonlinear model predictive control based virtual driver for high performance driving," in *Proc. IEEE Conf. Control Technol. Appl. (CCTA)*, 2019, pp. 9–14.
- [25] X.-H. Chang, Y. Liu, and M. Shen, "Resilient control design for lateral motion regulation of intelligent vehicle," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 6, pp. 2488–2497, Dec. 2019.
- [26] C. Rödel, S. Stadler, A. Meschtscherjakov, and M. Tscheligi, "Towards autonomous cars: The effect of autonomy levels on acceptance and user experience," in *Proc. 6th Int. Conf. Autom. User Interfaces Interact. Veh. Appl.*, 2014, pp. 1–8.
- [27] *VI-CarRealTime 18.0 Documentation*, VI-grade Eng. Softw. Serv., Darmstadt, Germany, 2017.
- [28] T. D. Gillespie, *Fundamentals of Vehicle Dynamics*. Warrendale, PA, USA: Soc. Autom. Eng., Inc., 1992.
- [29] H. B. Pacejka, *Tyre and Vehicle Dynamics*. Oxford, U.K.: Butterworth-Heinemann, 2006, pp. 187–189.
- [30] V. Cossalter, M. Da Lio, R. Lot, and L. Fabbri, "A general method for the evaluation of vehicle manoeuvrability with special emphasis on motorcycles," *Veh. Syst. Dyn.*, vol. 31, no. 2, pp. 113–135, 1999.
- [31] M. Bruschetta *et al.*, "Real-time nonlinear model predictive control of a virtual motorcycle," *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 5, pp. 2214–2222, Sep. 2021.
- [32] Y. Gao *et al.*, "Spatial predictive control for agile semi-autonomous ground vehicles," in *Proc. 11th Int. Symp. Adv. Veh. Control*, 2012, pp. 1–6.
- [33] Y. Chen, M. Bruschetta, D. Cuccato, and A. Beghi, "An adaptive partial sensitivity updating scheme for fast nonlinear model predictive control," *IEEE Trans. Autom. Control*, vol. 64, no. 7, pp. 2712–2726, Jul. 2019.
- [34] Y. Chen, D. Cuccato, M. Bruschetta, and A. Beghi, "An inexact sensitivity updating scheme for fast nonlinear model predictive control based on a curvature-like measure of nonlinearity," in *Proc. IEEE 56th Annu. Conf. Decis. Control (CDC)*, 2017, pp. 4382–4387.
- [35] H. G. Bock and K.-J. Plitt, "A multiple shooting algorithm for direct solution of optimal control problems," in *Proc. IFAC World Congr.*, 1984, pp. 1603–1608.
- [36] Y. Chen, N. Scarabottolo, M. Bruschetta, and A. Beghi, "Efficient move blocking strategy for multiple shooting-based non-linear model predictive control," *IET Control Theory Appl.*, vol. 14, no. 2, pp. 343–351, 2019.
- [37] M. Diehl, H. G. Bock, and J. P. Schlöder, "A real-time iteration scheme for nonlinear optimization in optimal feedback control," *SIAM J. Control Optim.*, vol. 43, no. 5, pp. 1714–1736, 2005.
- [38] S. Gros, M. Zanon, R. Quirynen, A. Bemporad, and M. Diehl, "From linear to nonlinear MPC: Bridging the gap via the real-time iteration," *Int. J. Control*, vol. 93, no. 1, pp. 62–80, 2020. [Online]. Available: <https://doi.org/10.1080/00207179.2016.1222553>
- [39] M. N. Zeilinger, C. N. Jones, and M. Morari, "Robust stability properties of soft constrained MPC," in *Proc. 49th IEEE Conf. Decis. Control (CDC)*, 2010, pp. 5276–5282.
- [40] R. Brach and M. Brach, "The tire-force ellipse (friction ellipse) and tire characteristics," SAE Techn. Paper, Warrendale, PA, USA, Rep. 2011-01-0094, 2011.
- [41] G. Frison and M. Diehl, "HPIPM: A high-performance quadratic programming framework for model predictive control," 2020, *arXiv:2003.02547*.



Enrico Picotti (Graduate Student Member, IEEE) received the M.Sc. degree in control system engineering from the University of Padua, Padua, Italy, in 2018, where he is currently pursuing the Ph.D. degree in information engineering.

He was with the Department of Information Engineering, University of Padua from 2018 to 2020, where he has been a Research Fellow with the Human Inspired Technology Research Center. His research interests include model-predictive control, with specific attention to learning-based and nonlinear techniques, and automotive applications, such as four wheels and two wheels vehicle control.



Mattia Bruschetta received the M.S. degree (*cum laude*) in control system engineering, in 2017, and the Ph.D. degree in information engineering from the University of Padua, Padua, Italy, in 2011.

He is currently an Assistant Professor with the Department of Information Engineering, University of Padua. His research interests include model-predictive control, automotive, virtual prototyping, motion cueing, virtual driver/riders, system identification, and numerical integration.



Enrico Mion received the M.Sc. degree in control system engineering and the Ph.D. degree in information engineering from the University of Padua, Padua, Italy, in 2017 and 2021, respectively.

He is currently a Postdoctoral Fellow with the Institute for Dynamic System and Control, ETH Zürich, Zürich, Switzerland. His research interests include model-predictive control, human-machine interaction systems, motion cueing algorithms, and automotive and smart grids.



Alessandro Beghi (Senior Member, IEEE) received the Laurea degree *cum laude* in electrical engineering and the Ph.D. degree in control systems engineering from the University of Padua, Padua, Italy, in 1989 and 1993, respectively.

He joined the Department of Information Engineering, University of Padua, in 1994, where he is currently a Professor of Control Systems Engineering. He held visiting positions with universities and international research centers both in Europe and USA. He has been responsible of research projects funded by the European Union, the Italian Ministry for Instruction and Research, the University of Padua, and of research activities with national and international industrial partners. He is a co-inventor of six Patents on the use of advanced control techniques in different applications fields. He has authored more than 200 publications in journals, books, and conference proceedings. He has worked on a wide range of control applications, from control of fusion devices, guidance algorithms for virtual vehicles, to control of HVAC&R systems, adaptive optics systems, and semiconductor manufacturing processes. His research interests include modeling and control, filtering and identification, model reduction, fault detection and isolation, and their applications.