Automated Driving Control in Highway Scenarios Through a Two-Level Hierarchical Architecture

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 18 April 2024, accepted 14 June 2024, date of publication 19 June 2024, date of current version 26 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3416670

RESEARCH ARTICLE

Automated Driving Control in Highway Scenarios Through a Two-Level Hierarchical Architecture

MASSIMO CANALE[1], **(Member, IEEE), AND VALENTINO RAZZA**[2]
[1]Department of Control and Computer Engineering, Politecnico di Torino, 10129 Turin, Italy
[2]Department of Management and Production Engineering, Politecnico di Torino, 10129 Turin, Italy
Corresponding author: Massimo Canale (massimo.canale@polito.it)

ABSTRACT We introduce an approach for automated driving in highway scenarios based on a two-level hierarchical architecture. The high-level consists of a path planner implemented through a Model Predictive Control algorithm that, using a simple kinematic model of the vehicle, effectively manages the relevant maneuvers of highway driving, such as lane keeping, lane change, velocity, and distance tracking, by means of suitable combinations of artificial potential field functions. Parameters of such functions are dynamically tuned according to the acquired scenario. A switching logic described by a finite state machine, based on acquired sensor data, selects the most appropriate maneuver to realize in the present driving scenario. At the low-level, the motion controller regulates the longitudinal and lateral dynamics through an original decentralized architecture to track the generated reference trajectory. Robustness issues in the presence of plant uncertainty are handled by H_∞ synthesis. Extensive simulation tests show the effectiveness of the proposed approach.

INDEX TERMS Autonomous vehicles, model predictive control, robust control, path planning.

I. INTRODUCTION

In the past years, the increase of traffic density led to the need to develop appropriate driving support systems to improve safety and prevent accidents and fatalities mainly due to distracted drivers. In such a context, enhancements were initially introduced by Advanced Driver Assistance Systems (ADAS) in commercial cars through Automated Lane Keeping systems and Adaptive Cruise Control functions (see, e.g., [1]). More recently, thanks to significant research efforts by both academia and industry, vehicles with fully automated driving functionality have been developed to further improve safety and driveability issues, see, e.g., [2]. As a result, car manufacturers, like, e.g., Waymo [3], Tesla [4], Audi [5], Mercedes-Benz [6] and Stellantis [7] deployed advanced prototypes of autonomous vehicles (AV) and ongoing field tests are giving promising results, especially in low complexity scenarios such as highway routes. In highway scenarios, the vehicle is expected to perform autonomously standard maneuvers such as lane keeping, distance tracking with respect to preceding vehicles, and lane change to overtake and return when allowed by the surrounding traffic conditions.

The principal ingredient for automated driving is the path planner that is responsible for finding an optimal collision-free path to perform the requested maneuver. In this context, an effective approach is made up by the use of artificial potential field (APF) that describes obstacles as repulsive virtual forces, while attractive forces represent target locations. Suitable combinations of APF can account for environment virtual obstacles such as lane borders to generate minimum energy-constrained paths even for lane change maneuvers. APFs have been originally introduced in [8] in industrial and mobile robotics as a collision avoidance method. The main advantage of this method is the possibility of adapting the path generation in a closed-loop fashion with the environment, see, e.g., [9]. Furthermore, APF can overcome computational drawbacks related to grid cells methods such as A* algorithm, see, e.g., [10], [11], [12] and Rapidly exploring Random Trees (RTTs) methods, see, e.g., [13] and [14]. However, as observed in [15], APF cannot directly include physical limitations

The associate editor coordinating the review of this manuscript and approving it for publication was Zhong Wu.

in vehicle behavior, such as on the steering angle and lateral acceleration. A method to include such constraints in the APF description is introduced in [16] by considering customized APF. In this regard, more systematic approaches combine the driving scenario and maneuver setting described by APF with control methods appropriated for vehicle dynamics control. As relevant examples, in [17], [18], and [19], the combined use of APF and model predictive control (MPC) is introduced for collaborative automated driving, path planning, and collision avoidance, respectively. In [20], an MPC controller is employed to track the path generated by a planner based on the combination of APF and Reinforcement Learning methodologies. Finally, [21] proposes an optimal path planning based on enhanced APF and PID control. To realize the considered automated driving functions, a suitable tracking of the trajectory provided by the path planner is performed by means of a motion controller that computes the steering and acceleration actions needed to accomplish the required maneuvers and accounting for the vehicle lateral and longitudinal dynamics. Motion controller must guarantee, besides smooth driving maneuver for comfort purposes, robustness issues in the presence of plant uncertainty. The integrated action of path planning and vehicle dynamic control is implemented through either single-level or two-level architectures, according to whether the planner and the controller are embedded together or are placed in two different hierarchical layers. In [18], [19], and [22], a single-level architecture is adopted where an MPC algorithm performs at the same time as both path generator and motion controller. Possible critical points with such approaches are represented by the complexity of the optimization problem that needs additional constraints to manage stability and robustness issues (see, e.g., the methods presented in [23] and [24]) and by consequent higher computational times.

Two-level architectures are mainly characterized by an upper layer made up of a path planner that computes the trajectory way-points and the speed to be tracked by the vehicle and a lower layer that implements the motion control functions that realize the requested maneuver. In general, two-level architectures provide more flexible solutions to handle the integration between path planning and motion control since different control methodologies can be employed for each of the two levels to exploit their peculiar characteristics in the view of achieving significant overall performance improvements. The advantages of using two-level approaches are also shown in the field of cooperative driving, as discussed in [25]. Several interesting and effective approaches to two-layer architectures have been recently introduced in the relevant literature. Basically, such approaches can be classified according to the methodologies employed in the path planner, the motion controllers, and the maneuver scenario that is accounted for. As an example, in [15], the path planner and motion controller are both made up of an MPC strategy based on, respectively, kinematic and dynamic linear time-varying (LTV) single-track models. Such an approach, is successfully tested in overtaking maneuvers of quasi-static obstacles performed at medium speed in straight roads. In [21], instead, the upper level employs suitable APF to describe moving obstacles, road boundaries, and target positions combined with a fireworks optimization algorithm to generate safe and smooth paths effectively. The motion control is implemented through a standard PID controller that provides the steering action needed to realize the considered maneuver. The effectiveness of this approach is shown by considering lane change and overtaking maneuvers in straight roads. In the two-layer solution presented in [26], the upper level exploits a given reference path to compute the steering command through an MPC controller that accounts for dynamic changes in driving conditions. The lower level implements the steer actuator control using a PID controller whose parameters are updated online through a recursive algorithm based on neural networks. In this approach, the driving scenario is predetermined, and trajectory modifications cannot be introduced to perform, e.g., overtaking slower vehicles traveling in the same lane. In all the methods presented above, the attention is mainly focused on lateral maneuvers without a thorough analysis of robustness issues in the presence of plant uncertainty.

In this paper, we introduce a two-layer architecture for SAE automated driving level 3 in highway scenarios that aims at

1) managing the lateral and longitudinal maneuvers as well as their combination that may occur in the considered driving scenario;
2) handling robustness issue in the presence of parametric and dynamic uncertainty of the vehicle.

In the proposed solution, the path planner of the higher level is based on the approach described in [27], where a dynamic tuning of the APF parameters manages the relevant maneuvers based on the current driving scenario. An MPC control algorithm that includes the maneuver objectives described by suitable APF in the cost function computes the path way-points and the corresponding target vehicle speed. The low-level controller computes the steering and acceleration actions needed to track the path computed at the higher level. A finite state machine (FSM) manages the switching among the considered maneuvers. An original decoupled robust architecture based on H_∞ methodology is introduced to control the lateral and the longitudinal vehicle dynamics. Since the higher-level control is working at a slower rate, a suitable path interpolator based on Bezier curves is included to generate the trajectory to be tracked by the motion controller implemented at the lower level. The main features of the proposed approach are summarized and highlighted below.

- The proposed path planner architecture can handle at the same time both lateral and longitudinal maneuvers by choosing suitable combinations of APF functions whose

parameters are adapted online based on the scenario acquired by the vehicle onboard sensors.
- The actual traffic conditions are considered by an FSM, which selects the most suitable maneuver to perform.
- Obstacle dynamic is explicitly considered in the path-planner MPC model to generate collision-free trajectories.
- The original decentralized motion control architecture can guarantee robust stability in the presence of parametric and/or dynamic uncertainty of both lateral and longitudinal vehicle dynamics.
- Both the control levels can be designed independently, allowing a more flexible management of the interaction between the path planning task and the automated driving functions when either different planners or motion control functions are already implemented in the existing control hardware and exploit different sampling times.

To show the advantages of the proposed architecture, we compare its performance with respect to a single-level approach based on an MPC controller designed including APFs in the cost function and a two-layer structure that exploits linear quadratic (LQ) methodologies for the motion controller. In both cases, extensive simulation tests that include sensitivity analysis are included to show the effectiveness of the method. The main contributions of this paper can be summarized as follows.

1) To propose a two-layer architecture that exploits the potentials of MPC and APF in path generation and H_∞ methodologies for motion control to effectively manage robustness issues in both longitudinal and lateral driving maneuvers.
2) To show the advantages of two-layer with respect to one-layer solutions that employ a similar approach.
3) To introduce extensive simulation tests that include a wide set of maneuvering situations performed in the considered scenario and to show the potential of the method in handling robustness issues through a suitable sensitivity analysis.

The paper is organized as follows. Section II defines the main settings of the considered automated driving problem and the two-level architecture. Section III introduces the high-level controller by describing the APF functions employed in the MPC path planner and how they are customized according to the specific maneuvers to be realized. Details on FSM are included as well. In Section IV, the path interpolation procedure is introduced. Section V is devoted to describing the structure and the design settings of the low-level motion controller. Extensive simulation tests that show the effectiveness of the proposed approach during the execution of the relevant driving maneuvers and in the presence of model uncertainty are presented in Section VI. Comparisons with other approaches are introduced as well to highlight the advantages of the proposed method. Some concluding remarks end the paper.

II. PROBLEM FORMULATION

We consider an autonomous vehicle, denoted as the host vehicle (HV), traveling in a highway scenario according to the SAE automation level 3, see [28]. In particular, the HV is requested to perform standard highway maneuvers such as velocity regulation, either in the presence or the absence of a preceding vehicle, overtaking and returning while keeping the center of the lane in both straight and curved roads. Thus, emergency maneuvers are not considered in this context. We assume the HV is equipped with a complete sensor configuration as described in the list below.

- Vehicle dynamic sensor for acquisition of the relevant longitudinal and lateral variables.
- Front camera to detect and reconstruct lane edges.
- Surround-view cameras to detect lateral scenarios such as vehicles in adjacent lanes.
- Front radar (long and short-range) to detect front vehicles' relevant variables such as the relative distance and speed.
- Inertial Measurements Unit (IMU) for the most relevant inertial measurements.
- Differential Global Positioning System (DGPS) to locate the vehicle in the world coordinate.

We introduce the two-layer hierarchical control architecture reported in Fig. 1 to handle the automated driving functions for the considered highway application. It consists of the two functional blocks described below.

1) High-level controller, which includes:
   a) a path planner, implemented by an MPC algorithm based on APF and a kinematic single track model of the vehicle, that generates every 200 ms the way-points of the cartesian path needed to perform the requested maneuver;
   b) a trajectory planner, starting from the way-points provided by the path planner, that employs the Bezier curve interpolation method to obtain at a rate of 10 ms the trajectory to be tracked by the vehicle in terms of the desired speed and acceleration, cartesian position, and desired yaw rate.
2) Low-level controller that implements the motion control unit and aims at providing the vehicle with suitable acceleration and steering actions needed to accomplish the requested maneuver. Such acceleration and steering action are obtained through a decentralized control architecture, including a longitudinal and lateral controller. Robustness issues in the presence of model uncertainty are managed as well.

Using a two-layer architecture can bring some advantages in the context of practical implementation with respect to a single-layer one. Single-layer approaches are based on the same principles, but path planning and motion control are performed simultaneously, employing more complex models to describe the vehicle dynamics. Here, as described in Section III, a simpler model is used in the path generation

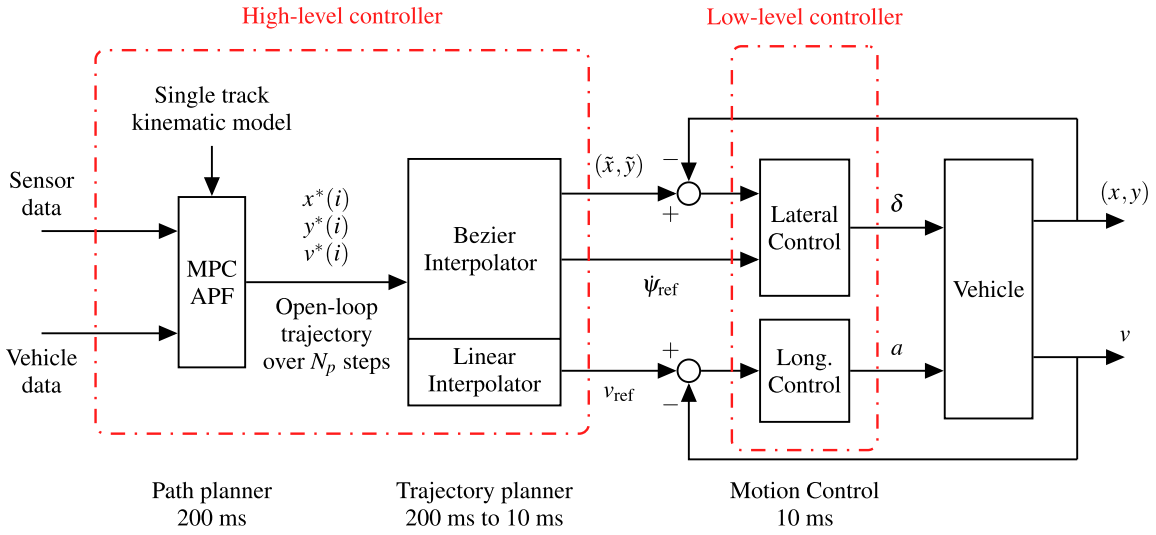


FIGURE 1. The proposed two-layer hierarchical control architecture.

phase leading to a more efficient solution of the underlying MPC optimization problem in terms of complexity and computation time. In this context, the trajectory tracking performed by the low-level controller can be implemented more effectively to account for possible plant uncertainty in the control design, as it will be explained in Section V. Moreover, since each architecture layer can be seen as a standalone block, the path and trajectory generation level can be easily interfaced with commercial motion control modules (see, e.g., [29]), already implemented in the vehicle Electronic Control Unit board.

III. HIGH-LEVEL CONTROLLER

In this Section, we provide details about the high-level control architecture introduced in Section II. First, we describe the vehicle model employed in this context. Then, details on how Artificial Potential Fields (APF) are employed to characterize the most relevant maneuvers during highway automated driving and how to include them in a suitable path planner formulation based on Model Predictive Control (MPC) methodologies. Finally, we show how Bezier curve interpolation methods can be customized to obtain an effective trajectory planner.

A. VEHICLE MODEL

Small steering angles characterize the maneuvers of the considered driving scenario. In this context, the single-track kinematic model reported in Fig. 2 adequately describes the lateral behavior for path generation purposes (see, e.g., [17]). In fact, the use of more complicated models may lead to more complex MPC controllers. The schematic of such vehicle model is reported in Fig. 2, together with the frame (x, y) that defines the vehicle behavior and the local frame (X, Y) employed to formalize the interaction with the highway scenario.

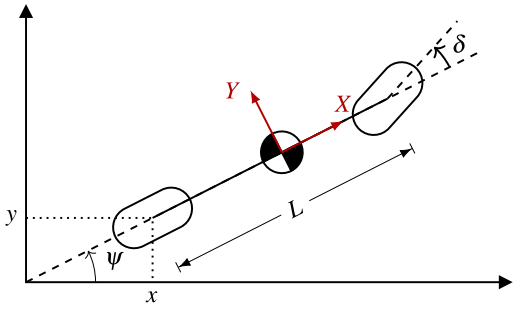


FIGURE 2. Single-track kinematic model schematic and frames.

Equation (1) introduces the single-track kinematic state equations.

$$\begin{cases} \dot{x}(t) = v(t)\cos(\psi(t)) \\ \dot{y}(t) = v(t)\sin(\psi(t)) \\ \dot{\psi}(t) = \dfrac{v(t)}{L}\tan(\delta(t)) \\ \dot{v}(t) = a(t) \\ \dot{\delta}(t) = \omega_\delta(t) \end{cases} \tag{1}$$

In (1), x, y and ψ represent the pose of the HV rear axle, v is the longitudinal speed, δ is the steering angle while L is the wheelbase. The state vector ξ defined in (2) includes the relevant variables of model (1).

$$\xi = [x, y, \psi, v, \delta]^T \tag{2}$$

The longitudinal acceleration $a = \dot{v}$ and the angular speed $\omega_\delta = \dot{\delta}$ are the inputs of the system.

B. ARTIFICIAL POTENTIAL FIELDS FOR AUTOMATED DRIVING

Artificial Potential Fields (APF) allow effective automated driving thanks to their capability to generate virtual forces,

either attractive or repulsive, that affect the vehicle motion, see, e.g., [30]. In the following subsections, we describe how suitable combinations of APF can be employed to handle typical maneuvers in highway scenarios such as lane keeping, overtaking and returning, and distance-tracking with respect to a preceding vehicle. In particular, we show how the standard parameters of APF functions like height and width can be tuned on the basis of the maneuver to be performed and according to the acquired sensors data.

1) APF FOR LANE KEEPING AND LANE CHANGE

The lane-keeping function is activated when the HV should track the lane center. Such a task can be implemented through the combined action P_ℓ of two APF defined by a 2$^{\text{nd}}$ order Gaussian function P_{ℓ_l} and P_{ℓ_r} centered at the left and the right borders of the lane, respectively, and providing a pulling action towards the lane center, as described in (3) and shown in Fig. 3.

$$P_\ell = P_{\ell_l} + P_{\ell_r} \tag{3}$$

where

$$P_{\ell_i} = P_0 \exp\left(-\frac{d_i}{\gamma_0}\right)^4, \; i = l, r \tag{4}$$

with

$$d_i = \sqrt{(X - X_{0_i})^2 + (Y - Y_{0_i})^2}, \quad i = l, r. \tag{5}$$

In (4), P_0 and γ_0 represent the height and width of the field, respectively, while d_i is the Euclidean distance between the vehicle and the left and right edges of the lane located at (X_{0_l}, Y_{0_l}) and (X_{0_r}, Y_{0_r}) respectively. Parameters P_0 and γ_0 are chosen to make the vehicle track the lane center to prevent lane escaping and avoid overtaking when not allowed. The field height P_0 is set to a sufficiently high value while the field widths γ_{0_i}, $i = l, r$ are obtained by assigning a suitable value $P_{\ell_i}(d_{\text{tar}})\, i = l, r$ to the field functions at the target distance $d_{\text{tar}} = \frac{w_L}{2}$ being w_L the lane width. According to (4) and (5), γ_0 can be computed as

$$\gamma_{0_i} = \sqrt[4]{\frac{-d_{\text{tar}}}{\log\left(\frac{P_{\ell_i}(d_{\text{tar}})}{P_0}\right)}}. \tag{6}$$

For example, Fig. 3 shows the resulting APF combination for a road with curvature radius $R = 500$ m and lane width $w_L = 3.65$ m. The other parameters are $P_0 = 100$, $P_{\ell_i}(d_{\text{tar}}) = 0.1$, $i = l, r$. In Fig. 3, the road boundary is the right lane represented by a straight line. The red line is the center of the lane where the functions P_{ℓ_i}, $i = l, r$ are minimal.

Lane change maneuvers such as overtaking and returning are handled through the combination of two repulsive fields, as in the case of lane keeping. Considering an overtaking maneuver, i.e., a lane change to the left, the first field is centered on the right line, while the second field is suitably shaped to damp and limit the vehicle motion as it enters the new lane. As soon as the vehicle enters into the destination lane, the high-level logic switches the working mode to lane-keeping, and the vehicle is driven to the lane center. The actual implementation employs quadratic Gaussian fields as in lane keeping with minor changes. In particular, a part overlaps the destination lane since the first field pushes the vehicle to the new lane. Then, γ_0 is obtained as in (6) with $d_{\text{tar}} = w_L$. Fig. 4 shows an example of the APF employed to perform an overtaking maneuver. Similar considerations hold for the returning maneuver.

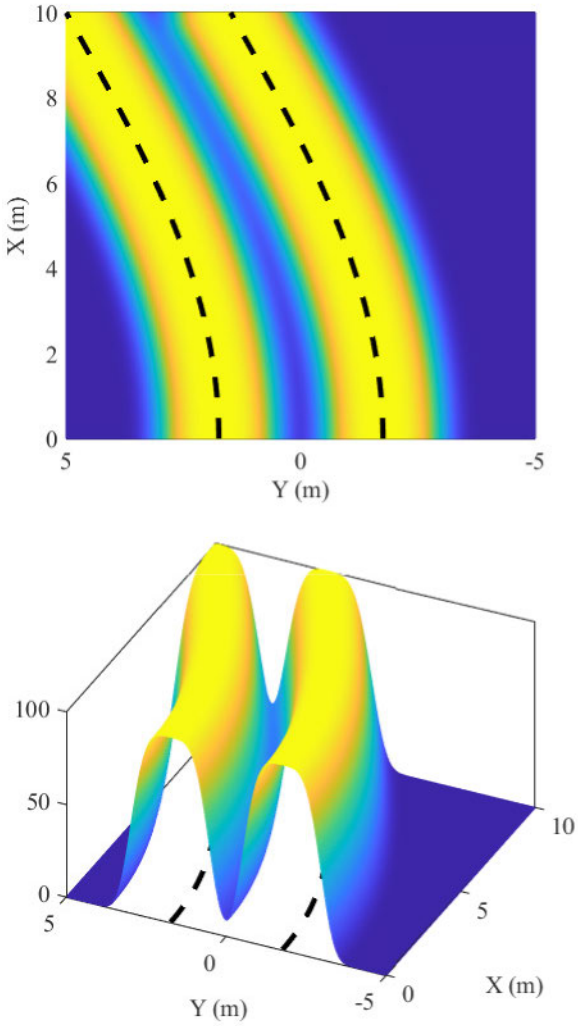


FIGURE 3. Lane keeping APF. The black dashed lines represent the lane boundaries, and the background color is relative to the APF value (i.e., blue is the lowest, and yellow is the highest APF value).

2) APF FOR OBSTACLE VEHICLES

Obstacle vehicles (OV), i.e., vehicles moving ahead of the HV that can not be overtaken because the left lane is not empty, are managed by an appropriate combination of an attractive and a repulsive APF. A single-track model, similar to the one introduced in (1), describes the OV kinematic behavior. In this context, we assume that information on the obstacle pose (X_O, Y_O, ψ_O) and velocity v_O is achieved either directly

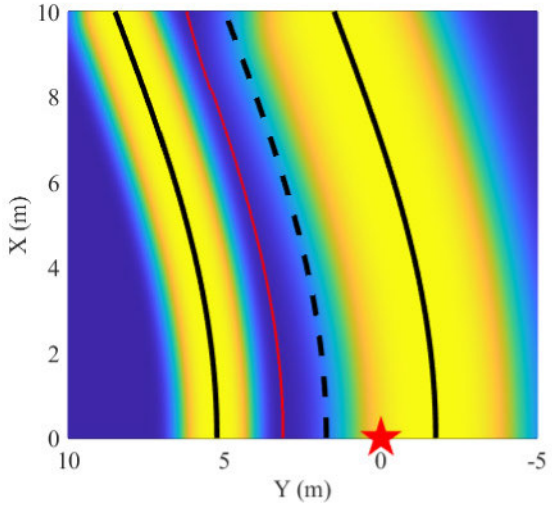


FIGURE 4. Example of lane change PF built to move on the left line. The red star represents the vehicle position, the black lines are the lane boundaries and the red line is the minimum APF value within the road boundaries.

or indirectly by the considered onboard sensors. Obstacle APFs are designed to regulate the velocity v of the HV in order to keep a target distance $d_{\text{tar},o}$ from the OV. Such a target distance is defined as

$$d_{\text{tar,o}} = \begin{cases} d_0 + t_H v_i, & v \le v_O \\ d_0 + t_H v_P + \dfrac{(v - v_O)^2}{2a_{dec}}, & v > v_O \end{cases} \tag{7}$$

where a_{dec} is the desired magnitude of the HV deceleration, d_0 represents a safety distance including the vehicle size, v_O is the obstacle speed and t_H is the time headway, defined as $t_H = \frac{d_O}{v}$, being d_O the relative distance between HV and OV.

The distance-approaching maneuver is obtained thanks to a suitable interaction between

1) a repulsive field that performs a smooth deceleration;
2) an attractive field that implements the target position tracking.

Given the pose (X_O, Y_O, ψ_O) of the OV in the local HV frame, the first-order Gaussian function reported in (8) describes the repulsive APF considered above.

$$P_{\text{o},r} = P_0 \cdot \exp\left(-\left(\frac{(\tilde{X} - \tilde{X}_O)^2}{\gamma_X^2}\right) - \left(\frac{(\tilde{Y} - \tilde{Y}_O)^2}{\gamma_Y^2}\right)\right) \tag{8}$$

where

$$\begin{bmatrix} \tilde{X} \\ \tilde{Y} \end{bmatrix} = T \cdot \begin{bmatrix} X \\ Y \end{bmatrix}, \qquad \begin{bmatrix} \tilde{X}_O \\ \tilde{Y}_O \end{bmatrix} = T \cdot \begin{bmatrix} X_O \\ Y_O \end{bmatrix} \tag{9}$$

and

$$T = \begin{bmatrix} \cos\psi_O & \sin\psi_O \\ -\sin\psi_O & \cos\psi_O \end{bmatrix}. \tag{10}$$

In (8), γ_X and γ_Y define the Gaussian function sizes along the coordinate axes X and Y, respectively, while P_0 stands for the function height. Similarly as in (6), γ_X and γ_Y are computed as

$$\gamma_X = \sqrt{\frac{-d_{\text{tar,o},X}}{\log\left(\dfrac{P_0}{P_{\text{o},r}(d_{\text{tar},X})}\right)}}, \gamma_Y = \sqrt{\frac{-d_{\text{tar,o},Y}}{\log\left(\dfrac{P_0}{P_{\text{o},r}(d_{\text{tar},Y})}\right)}} \tag{11}$$

A reasonable choice for the field parameter in case of a semi-straight road is $d_{\text{tar,o},X} = d_{\text{tar,o}}$, see (7). Notice that $d_{\text{tar,o},X}$ depends on the relative speed between HV and OV. $d_{\text{tar,o},Y}$, is chosen according to the OV width. Moreover, $P_{\text{o},r}(d_{\text{tar},X}) = P_{\text{o},r}(d_{\text{tar},Y}) = \bar{P}_{\text{o}}$ is selected.

The attractive APF function considered above is chosen as

$$P_{\text{o},a} = P_0 \cdot \left(1 - \exp\left(-\left(\frac{(\tilde{X} - \tilde{X}'_O)^2}{\gamma_X^2}\right) - \left(\frac{(\tilde{Y} - \tilde{Y}'_O)^2}{\gamma_Y^2}\right)\right)\right) \tag{12}$$

where $(\tilde{X}'_O, \tilde{Y}'_O) = (X_O - d_{\text{tar}},\ Y_O)$. Parameters in (12) are tuned through the same procedure described for (8). An example of the repulsive (8) and the attractive fields (12) can be found in [27].

C. MODEL PREDICTIVE CONTROL PATH PLANNER

Here, we introduce how MPC methodologies effectively implement the path planning task. In such a context, the vehicle model (1) is discretized through the forward Euler method and using sampling time $T_s = 200$ ms. The obtained discretized model is reported in (13).

$$\begin{cases} x(k+1) = x(k) + T_s\, v(k) \cos(\psi(k)) \\ y(k+1) = y(k) + T_s\, v(k) \sin(\psi(k)) \\ \psi(k+1) = \psi(k) + T_s \dfrac{v(k)}{L} \tan(\delta(k)) \\ v(k+1) = v(k) + \Delta v(k) \\ \delta(k+1) = \delta(k) + \Delta\delta(k) \end{cases} \tag{13}$$

where $\Delta v(k) = T_s\, a(k)$ and $\Delta\delta(k) = T_s\, \omega_\delta(k)$ represent the speed and steering angle finite increments respectively. The control input of (13) is

$$u(k) = \left[\ \Delta v(k)\ \ \Delta\delta(k)\ \right]^T. \tag{14}$$

Considering (2) and (14), the state equation (13) is represented in the compact nonlinear form

$$\xi(k+1) = f(\xi(k), u(k)) \tag{15}$$

The MPC path planner is designed by considering the objectives described below.

1) Track a constant speed v_{des} in the absence of a preceding vehicle.
2) Track a (speed-depending) distance from a preceding vehicle when overtaking is not allowed.
3) Keep the center of the lane when traveling on a curved road.
4) Perform overtaking of a slower vehicle and returning to the rightmost lane when overtaking is completed.

Remark 1: It is worth noting that the goal of the designed control structure is to reach the SAE level 3 automation level, where no emergency maneuvers are considered. The assumed sampling time $T_s = 200$ ms is sufficient for the MPC motion planner to schedule a suitable trajectory in highway scenarios. The effective vehicle input, i.e., steering angle and acceleration, is computed by the lower layer controller, which works at a higher frequency to track the desired trajectory precisely.

The cost function J introduced in (16) accounts for all the control objectives listed above as a weighted sum of the relevant contributions.

$$J(U(k)) \doteq \sum_{i=k}^{k+N_p} (\|P_o(i)\|_O^2 + \|P_\ell(i)\|_L^2 + \|v(i) - v_{\text{des}}\|_Q^2 + \|\Delta v(i)\|_{R_1}^2 + \|\Delta\delta(i)\|_{R_2}^2) \tag{16}$$

where

$$U(k) = \begin{bmatrix} \Delta v(k) & \Delta\delta(k) \\ \Delta v(k+1) & \Delta\delta(k+1) \\ \vdots & \vdots \\ \Delta v(k+N_p-1) & \Delta\delta(k+N_p-1) \end{bmatrix} \tag{17}$$

The cost function $J(U(k))$, introduced in (16), is defined over a prediction horizon N_p and depends on a sum of terms, each describing how the control objectives are treated. More specifically:

- $\|P_o(\tau)\|_O$ is the weighted 2-norm related to the obstacle field that implements the distance-keeping functionality.
- $\|P_\ell(\tau)\|_L$ is the weighted norm of the lane field responsible for either the lane keeping or the lane changing feature.
- $\|v(\tau) - v_{\text{des}}\|_Q$ is the weighted 2-norm of the speed tracking error that accounts for the achievements of the target speed.
- $\|\Delta v(\tau)\|_{R_1}$ and $\|\Delta\delta(\tau)\|_{R_2}$ are the weighted 2-norm of the two inputs increments, needed to handle the control action effort.

Weight matrices O, L, Q, R_1, and R_2 are chosen to regulate the desired trade-off among all the control objectives. Typically, a trial and error procedure is employed to tune the entries of such matrices. Given the cost function (16), the underlying optimization problem for MPC design is formulated as

$$\begin{aligned} &\min_{U(k)} J(k) \\ &\text{s.t.} \quad (15) \\ &\quad 0 \le v(i) \le v_{\text{des}}, i = 0, \dots, N_p - 1 \\ &\quad -\bar{\Delta}_v T_s \le \Delta v(i) \le \bar{\Delta}_v T_s, i = 1, \dots, N_p - 1 \\ &\quad -\bar{\delta} \le \delta(i) \le \bar{\delta}, i = 0, \dots, N_p - 1 \\ &\quad -\bar{\Delta}_\delta T_s \le \Delta\delta(i) \le \bar{\Delta}_\delta T_s, i = 1, \dots, N_p - 1 \\ &\quad Y_{0_l} \le y(k) \le Y_{0_r}, i = 1, \dots, N_p - 1 \\ &\quad |\Delta\psi(k)| \le \bar{\Delta}_\psi(k), i = 1, \dots, N_p - 1, \end{aligned} \tag{18}$$

where $\Delta\psi(k) = \psi(k) - \psi(k-1)$ and $\bar{\Delta}_\psi(k) = T_s \mu g / v(k)$ is the limit of the yaw angle increment, being μ the road friction coefficient and g the gravity acceleration. Notice that, since the optimization is performed with respect to the rates Δv and $\Delta\delta$, an implicit integral action is included in the control input computation. In (18), constraints on the δ and $\Delta\delta$ account for physical limitations of the steering actuator; in this case we impose $\bar{\delta} = 25°$ and $\bar{\Delta}_\delta = 0.47°$ s^{-1}. Limitations on v and Δv introduce a suitable speed limitation and comfort performance during acceleration maneuvers. In particular, v_{des} is either the user set-speed or the velocity limit for the road regulations, while $\bar{\Delta}_v = 2.5$ ms^{-2}. The two latter constraints in (18) account for the stability of the generated trajectory. As to the prediction horizon, a value $N_p = 15$ is chosen, corresponding to a prediction time $T_p = N_p T_s = 3$ s. Furthermore, a control horizon $N_c = 8$ is introduced to reduce the number of optimization variables in the optimization problem. The remaining manipulable inputs are set as $U(i) = U(N_c), i = N_c + 1, \dots, N_p - 1$.

Remark 2: The control input is computed by predicting the relevant variables over the time horizon N_p. In particular:

- The kinematic variables of HV are obtained through (13) represented in the local frame of the HV.
- The values of the center of the lane involved in the APF (4) are provided by the vision system, which updates the information at every sampling instant.
- The prediction of the OV kinematic variables is obtained through a suitable discretized single-track model expressed in the OV coordinate.

The control input is then computed according to the receding horizon principle as the first time component of the optimizer $U^*(k) = \arg\min_{U(k)} J(k)$. In particular, at a given sampling time k, the optimal vehicle speed and the steering angle are provided as

$$\begin{aligned} v(k) &= v(k-1) + T_s \Delta v(k) \\ \delta(k) &= \delta(k-1) + T_s \Delta\delta(k) \end{aligned} \tag{19}$$

The previously described MPC procedure allows us to generate the waypoints of an optimal geometric path and the required longitudinal speed needed to perform the requested driving maneuver. In particular, based on the solution to the optimization problem (18), at every sampling time T_s, the MPC path planner provides

1) the optimal target way-point coordinate $x^*(k+1), y^*(k+1)$ obtained by applying the first time component of the optimizer

$$U^*(k) = \begin{bmatrix} \Delta v^*(k) & \Delta\delta^*(k) \end{bmatrix} \tag{20}$$

to the model (13) starting from the previous step solution $x^*(k), y^*(k)$;

2) the longitudinal speed target v_{tar} obtained as

$$v_{\text{tar}} = v^*(k+1) = v^*(k) + \Delta v^*(k). \tag{21}$$

Remark 3: The MPC-based trajectory planner is based on the discretized system equations (1), i.e., the single-track

kinematic model. The only physical vehicle parameter is the wheelbase L, whose uncertainty is related to the production tolerance that is typically negligible and does not affect significantly the path planning procedure. Robust stability issues due to the vehicle uncertain parameters are managed by the lower-layer controller described later in Section V.

The path way-points and the reference speed described above are obtained at a rate of $T_s = 200$ ms. The low-level motion controller runs at a sampling time of $T_s^m = 10$ ms. Thus, the way-point and speed interpolation procedures must be applied to obtain a suitable trajectory reference to be tracked by the motion controller at the prescribed rate. Section IV describes the details of how trajectory generation is performed through way-point and speed interpolation.

D. BEHAVIORAL LOGIC

The transition between one driving task and another, e.g. between lane keeping and overtaking, depends on the driving environment that is detected by the on-board sensors. A switching logic must be is introduced to decide when the HV has

- to track a target speed or regulate the relative distance with respect to a preceding OV;
- to keep the center of the lane when no lane change maneuver is permitted;
- to overtake in the presence of a slower preceding OV and the left lane is free;
- to return after an overtaking maneuver when the right lane is free.

To manage the transition among the different driving functions, we introduce the finite-state machine (FSM) reported in Fig. 5. In the FSM of Fig. 5, red ellipses are the

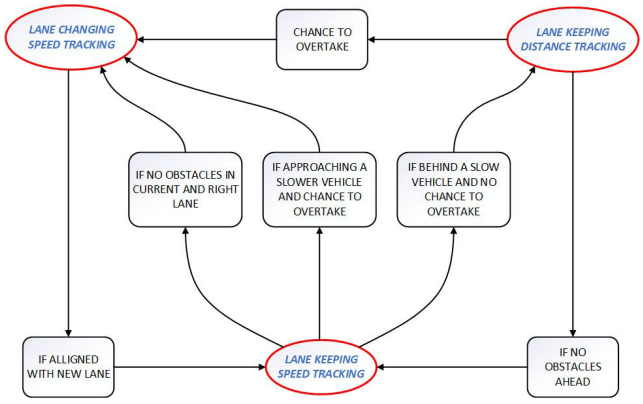


FIGURE 5. Path planner finite state machine diagram.

states of the path planner while black rectangles define the conditions for state switching. State transitions are described by quantitative switching rules. As an example, we consider the transition between the task *Lane Keeping / Speed Tracking* and *Lane Keeping / Distance Tracking*. This case occurs when a preceding OV, e.g., a truck, is running in the same lane at a lower speed with respect to the HV, and overtaking is prevented because of the presence of another OV in the left lane. Supposing that the Lane Keeping function is always active, the transition *Speed Tracking* (ST) $\rightarrow$ *Distance Tracking* (DT) is governed by the hysteresis mechanism based on the relative distance d_O between HV and OV and defined in (22).

$$\text{mode} \rightarrow \begin{cases} \text{ST} \rightarrow \text{DT if } d_O < d_{\text{tar}} - d_{m_1} \\ \text{DT} \rightarrow \text{ST if } d_O > d_{\text{tar}} + d_{m_2} \end{cases} \tag{22}$$

where d_{tar} is a suitable target distance that depends on the relative speed between HV and OV and d_{m_1}, d_{m_2} are appropriate hysteresis thresholds introduced to avoid chattering during the switching ST $\leftrightarrow$ DT.

On the basis of the selected driving task, the MPC controller formulation described in (18) must be set accordingly. In this regard, the contributions in the cost function (16) should be suitably chosen. As an example, during *Lane Keeping / Speed Tracking* task, the cost function (16) must include the terms described below.

1) $\|P_l\|_L$, that accounts for the lane keeping task and its relevant field function is obtained as in (3) and (5);
2) $\|v - v_{\text{des}}\|_Q$, that describe the speed tracking performance;
3) $\|\Delta v\|_{R_1}$ and $\|\Delta\delta\|_{R_2}$, that penalize the input rates to obtain smooth control actions.

The weight matrices L, Q, R_1, and R_2, are chosen to set the desired performance trade-off. In general, for each task described in the FSM of Fig. 5 there is the corresponding formulation of the MPC problem.

IV. PATH INTERPOLATION AND TRAJECTORY GENERATION

As described in Section III, the output of the MPC path-planner is the one-step-ahead prediction of the path way-point and a constant reference speed value v_{ref}, both generated at the planner sampling time $T_s = 200$ ms while the low-level motion controller is running at $T_s^m = 10$ ms. This means that between every run of the path-planner there are twenty runs of the motion control. Thus, to obtain effective and smooth path tracking, a suitable interpolation procedure is implemented to generate twenty trajectory target points between the present vehicle position and speed and those provided by the path-planner. The following Subsections describe how desired speed and path way-points are interpolated.

A. SPEED REFERENCE INTERPOLATION

The speed reference sequence v_{ref} for the motion controller is obtained through a linear interpolation between the speed measured at the present planner sample time v_0 and the target value v_{tar} as described in (23).

$$v_{\text{ref}}(t) = \int_0^t \frac{1}{T_s}(v_{\text{des}} - v_0)\, d\tau, v_{\text{ref}}(0) = v_0 \tag{23}$$

B. WAY-POINTS INTERPOLATION

The use of a linear approach for target position interpolation leads to non-smooth trajectories that significantly affect the comfort performance of the resulting automated driving system. In such a context, polynomial curves are a more appropriate solution to obtain regular paths to be tracked by the motion controller. In particular, Bezier curves are demonstrated to be an effective tool for trajectory computation (see, e.g., [31] and [32]). In fact, such an approach based on Bernstein polynomial basis can generate the needed trajectory reference guaranteeing smooth sharp borders and perfect fitting of starting and ending points. As a matter of fact, the Bezier method applied for interpolating the way-points $(x^*(k), y^*(k))$ and $(x^*(k+1), y^*(k+1))$, see (20), may lead to non-smooth solutions. Thus, to improve the smoothness properties of the interpolated trajectory, we compute, at first, an interpolated trajectory that includes $n_T \leq N_p$ way-points obtained as the vehicle positions $(x^*(k+i), y^*(k+i)), i=0,\ldots,n_T \leq N_p$ within the prediction horizon. Then, as reference trajectory, we consider the first tract that includes the first 20 samples that belong to the obtained curve. A Bezier curve of order n_T that interpolates n_T+1 way-points is described by the parametric expression introduced in (24).

$$\begin{bmatrix}\tilde{x}(\upsilon)\\ \tilde{y}(\upsilon)\end{bmatrix} = \sum_{i=0}^{n_T}\binom{n_T}{i}\upsilon^i(1-\upsilon)^{n_T-i}\begin{bmatrix}x^*(k+i)\\ y^*(k+i)\end{bmatrix} \tag{24}$$

In (24), $\upsilon = \frac{t}{T_s} \in [0,1]$ is the time-dependent parameter and $(\tilde{x}(\upsilon), \tilde{y}(\upsilon))$ are the coordinates of the generic point of the curve. In particular, $(\tilde{x}(0), \tilde{y}(0)) = (x^*(k), y^*(k))$ and $(\tilde{x}(1), \tilde{y}(1)) = (x^*(k+n_T), y^*(k+n_T))$. In general, the coordinates of the j^{th} point of the curve is given by $(\tilde{x}(\frac{j}{n_T-1}), \tilde{y}(\frac{j}{n_T-1}))$.

As already observed, the trajectory to be tracked is obtained by computing $\frac{T_s}{T_s^m} = 20$ points of the curve (24) between $(\tilde{x}(0), \tilde{y}(0))$ and $(\tilde{x}(\frac{1}{n_T-1}), \tilde{y}(\frac{1}{n_T-1}))$ as $(\tilde{x}(\frac{\kappa}{20(n_T-1)}), \tilde{y}(\frac{\kappa}{20(n_T-1)})), \kappa = 1,\ldots,20$.

Remark 4: It is well known that the Bezier curves do not cross the way-point employed for its definition (24), except for the first and the last., see e.g. [31]. However, since, in general, small steering angles and almost constant speed values characterize the considered application, the difference between the actual target and its approximation in the interpolating Bezier curve is usually negligible. In Fig. 6, we compare the trajectory target computed by the planner and the approximated ones, showing the small difference between the real and interpolated targets.

The high-level controller must provide the yaw rate reference $\dot{\psi}_{\text{ref}}$ to the motion controller as defined in (25)

$$\dot{\psi}_{\text{ref}} = \frac{v_x}{\rho} \tag{25}$$

where

$$\rho(\upsilon) = \frac{\dot{\tilde{x}}(\upsilon)\cdot\ddot{\tilde{y}}(\upsilon) - \dot{\tilde{y}}(\upsilon)\cdot\ddot{\tilde{x}}(\upsilon)}{(\dot{\tilde{x}}(\upsilon)^2 + \dot{\tilde{y}}(\upsilon)^2)^{\frac{3}{2}}} \tag{26}$$

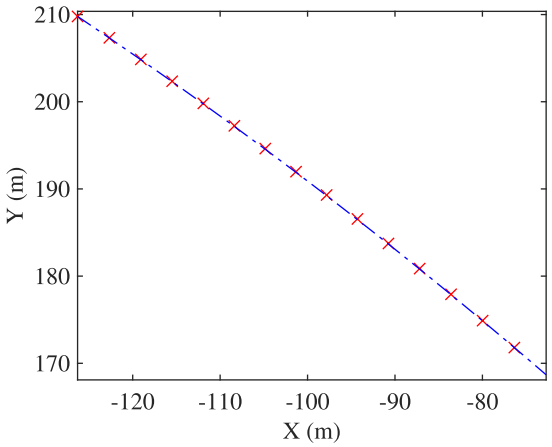


FIGURE 6. Bezier interpolating curve. In red are the trajectory points from the high-level MPC controller and, in blue, the Bezier interpolating curve. The data is collected from the simulation example described in Section VI.

is the road curvature (see, e.g., [33]), computed by evaluating the first and the second derivative of (24) with respect to parameter υ. From $\dot{\psi}_{\text{ref}}$, we get

$$\psi_{\text{ref}}(t) = \int_0^t \dot{\psi}_{\text{ref}}(\tau)d\tau,\ \psi_{\text{ref}}(0) = \psi_0, \tag{27}$$

where ψ_0 is the yaw angle measured at the present planner sample time.

V. LOW-LEVEL CONTROLLER

The low-level controller aims to regulate vehicle dynamics to track the reference trajectory computed at the higher level.

The longitudinal behavior is described by the kinematic relationship between the desired acceleration a_{des} to be tracked and the actual longitudinal speed v_x is (see, e.g., [1])

$$G_{\text{long}}(s) = \frac{v_x(s)}{a_{\text{des}}(s)} = \frac{1}{s(1+\tau s)}. \tag{28}$$

In (28), the time constant τ depends on the characteristic of the vehicle driveline and on the dynamics of the actuator devices, such as the throttle body, the engine and brake circuit, see e.g., [1] for details. The lateral dynamics are accounted for by considering the single-track state equation (30) where the state variables are chosen as the lateral speed v_y, the yaw rate $\dot{\psi}$, the lateral deviation with respect to the trajectory center computed by the path planner

$$e_y = (\tilde{y}-y)\cos(\psi_{\text{ref}}) - (\tilde{x}-x)\sin(\psi_{\text{ref}}) \tag{29}$$

and the yaw rate deviation defined as $e_{\dot{\psi}} = \dot{\psi}_{\text{ref}} - \dot{\psi}$. The manipulable input is the steering angle δ while the yaw rate reference $\dot{\psi}_{\text{ref}}$ acts as a measurable disturbance.

$$\begin{bmatrix}\dot{v}_y\\ \ddot{\psi}\\ \dot{e}_y\\ \dot{e}_{\dot{\psi}}\end{bmatrix} = A\begin{bmatrix}v_y\\ \dot{\psi}\\ e_y\\ e_{\dot{\psi}}\end{bmatrix} + B_1\delta + B_2\dot{\psi}_{des} \tag{30}$$

where

$$A = \begin{bmatrix} -\frac{c_f+c_r}{m v_x} & -v_x - \frac{c_f l_f - c_r l_r}{m v_x} & 0 & 0 \\ \frac{c_r l_r - c_f l_f}{J_\psi v_x} & -\frac{c_r l_r^2 + c_f l_f^2}{J_\psi v_x} & 0 & 0 \\ 1 & 0 & 0 & v_x \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{31}$$

and

$$B_1 = \begin{bmatrix} \frac{c_f}{m} \\ 0 \\ \frac{c_f l_f}{J_\psi} \\ 0 \end{bmatrix}, \; B_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}. \tag{32}$$

In (30), the symbols c_f and c_r represent the front and the rear cornering stiffness coefficients, m and J_ψ are the vehicle mass and the yaw moment of inertia respectively, while l_f and l_r are the front and the rear semi-wheelbase. Since in the considered autonomous driving application, the longitudinal velocity v_x is varying during the considered maneuvers, it results that the longitudinal and the lateral dynamics described by (28) and (30) are coupled. In this regard, a combined longitudinal/lateral approach to the vehicle dynamics control can be used. Several solutions have been employed in such a context, leading to a nonlinear controller. For example, MPC methodologies are used in [18] and [22], while sliding mode approaches are considered, e.g., in [34]. Linear Parameter Varying solutions are discussed in [35]. In this paper, a robust decentralized approach is introduced to obtain a linear control architecture. In particular, we consider a decoupled longitudinal controller based on (28) and a robust lateral controller based on (30) considering fixed the value of v_x and, at the same time, imposing robust stability in the presence of variations of v_x and other relevant parameters of model (30).

A. LONGITUDINAL CONTROLLER

The longitudinal controller is represented by the one degree of freedom architecture reported in Fig. 7 and aims at regulating the longitudinal velocity v_x to track the reference speed v_{ref} provided by the path planner. The control input is a_{des}, which is realized by a suitable regulation of the throttle and brake actuators that is not described here. The design of

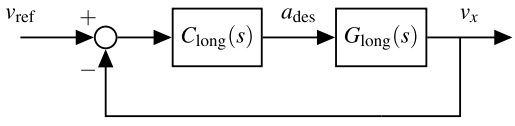


FIGURE 7. Longitudinal control architecture.

the cascade controller $C_{\text{long}}(s)$ is performed considering the plant transfer function $G_{\text{long}}(s)$ with the nominal value of the delay $\tau = 500$ ms. The design procedure is developed in the frequency domain through loop-shaping techniques by imposing robust stability in the presence of a variation of the time constant τ within $\pm 10\%$ of its nominal value, null steady state tracking error for a linear ramp velocity reference signal and nominal closed-loop bandwidth of about 15 rad/s. The resulting Nichols diagram is reported in Fig. 8.

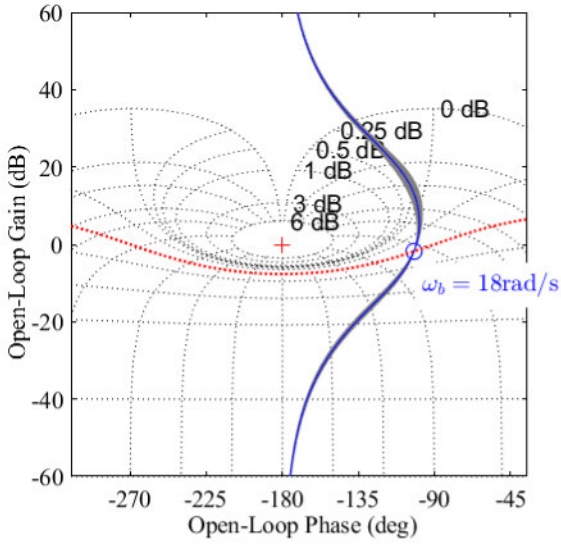


FIGURE 8. Nichols diagram of the longitudinal control loop transfer function $L_{\text{long}} = C_{\text{long}} G_{\text{long}}$. We compare the nominal loop transfer function (blue) with the uncertain functions (gray), where τ has variations within $\pm 10\%$ of its nominal value. The red dashed line represents the -3 dB constant magnitude locus for the complementary sensitivity function, and ω_b is the nominal closed-loop transfer function bandwidth.

B. LATERAL CONTROLLER

The lateral controller aims at minimizing the lateral deviation e_y during lane-keeping and overtaking maneuvers. Robust stability must also be guaranteed in the presence of longitudinal speed and relevant parameter variations. The design is performed by considering the transfer functions $G_{\delta,e_y}(s)$ and $G_{\delta,\dot{\psi}}(s)$ that describe the relationship between the input δ and the outputs of interest e_y and $\dot{\psi}$ obtained from the model (30) with the nominal value of the longitudinal velocity $v_x = 110$ km/h and the other physical parameters set to the nominal values reported in Table 1.

TABLE 1. Lateral model parameters nominal values.

Symbol	Value	Unit
m	1715.0	kg
J_ψ	2697.0	Nms^2
l_f	1.07	m
l_r	1.47	m
c_f	87330	N/rad
c_r	114100	N/rad

To account for perturbation of the physical parameters of the lateral dynamics model (30), we consider that the longitudinal speed v_x varies between 80 and 130 km/h, i.e., the range of velocities allowed in typical highway scenarios. Moreover, we suppose that the relevant parameters, i.e., the cornering stiffness coefficients c_f and c_r and the vehicle mass m, are affected by an uncertainty of 10%. The perturbation on the mass induces a variation on the polar moment of inertia

J_ψ that can be computed, as explained in [36], supposing that the 30% is concentrated on the front axle and the 70% on the rear axle.

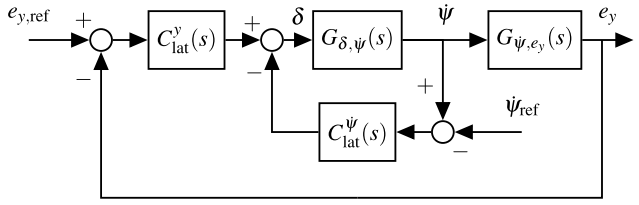


FIGURE 9. Lateral control architecture.

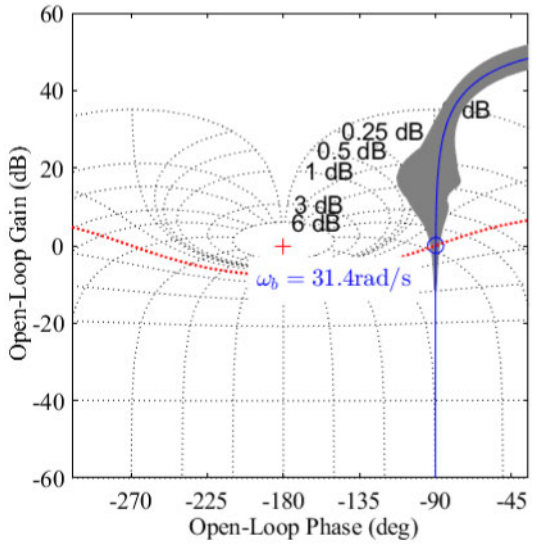


FIGURE 10. Nichols diagram of the lateral control inner loop transfer function $L_{\dot{\psi}}$. We compare the nominal loop transfer function (blue) with the uncertain parameter functions (gray). The red dashed line represents the −3 dB constant magnitude locus for the complementary sensitivity function, and ω_b is the nominal closed-loop transfer function bandwidth.

The nested architecture reported in Fig. 9, where $G_{\dot{\psi},e_y}(s) = G_{\delta,e_y}(s) \cdot G^{-1}_{\delta,\dot{\psi}}(s)$ and $e_{y,\text{ref}} = 0$, implements the lateral controller. The lateral control design consists of a two-step procedure. In the first step, the inner loop controller $C^{\dot{\psi}}_{\text{lat}}(s)$ is obtained to pre-stabilize the yaw rate dynamics described by $G_{\delta,\dot{\psi}}(s)$. The second step is in charge of guaranteeing robust stability of the overall architecture through the controller $C^{y}_{\text{lat}}(s)$. A loop-shaping procedure is employed to design the inner loop controller $C^{\dot{\psi}}_{\text{lat}}(s)$ to obtain stability in the presence of the described parameter perturbations and imposing a closed loop bandwidth greater than the one of the outer loop. Fig. 10 displays the obtained uncertainty bounds of the open loop function $L_{\dot{\psi}}(s) = C^{\dot{\psi}}_{\text{lat}}(s)G_{\delta,\dot{\psi}}(s)$ in the Nichols plane showing the achievement of the design objectives.

The outer loop controller $C^{y}_{\text{lat}}(s)$ is designed to obtain robust stability of the lateral control system introduced in Fig. 9. An H_∞ approach is adopted considering a plant transfer function

$$G^{p}_{\text{lat}}(s) = Q_{\dot{\psi}}(s)G_{\dot{\psi},e_y}(s) \tag{33}$$

where

$$Q_{\dot{\psi}}(s) = \frac{G_{\delta,\dot{\psi}}(s)}{1 + G_{\delta,\dot{\psi}}(s)C^{\dot{\psi}}_{\text{lat}}(s)}$$

To account for the parameter variation in the control design, we introduce the unstructured multiplicative uncertainty described by $\Delta_{m,\text{lat}}$

$$\Delta_{m,\text{lat}}(s) = \frac{\tilde{G}^{p}_{\text{lat}}(s) - G^{p}_{\text{lat}}(s)}{G^{p}_{\text{lat}}(s)} \tag{34}$$

where $\tilde{G}^{p}_{\text{lat}}(s)$ represents the perturbed plant transfer function. The multiplicative uncertainty (34) induces a multiplicative model set of the form

$$\begin{aligned}&\mathcal{G}(G^{p}_{\text{lat}}, (s), \Gamma_{\delta,e_y})\\ &= \left\{G^{p}_{\text{lat}}(s)(1 + \Delta_{m,\text{lat}}(s)) : |\Delta_{m,\text{lat}}(j\omega)| \leq |\Gamma_{m,\text{lat}}(j\omega)|\right\}.\end{aligned} \tag{35}$$

Such a model set is computed through uniform gridding of the considered vehicle parameter values and is reported in Fig. 11. As overbound of the multiplicative uncertainty $\Delta_{m,\text{lat}}(s)$ the low order real rational function $\Gamma_{m,\text{lat}}(s)$ reported in (36), is employed.

$$\Gamma_{m,\text{lat}}(s) = \frac{0.22(s^2 + 42.42s + 900)}{s^2 + 28.59s + 408.9}. \tag{36}$$

The closed-loop performance is described by the nominal

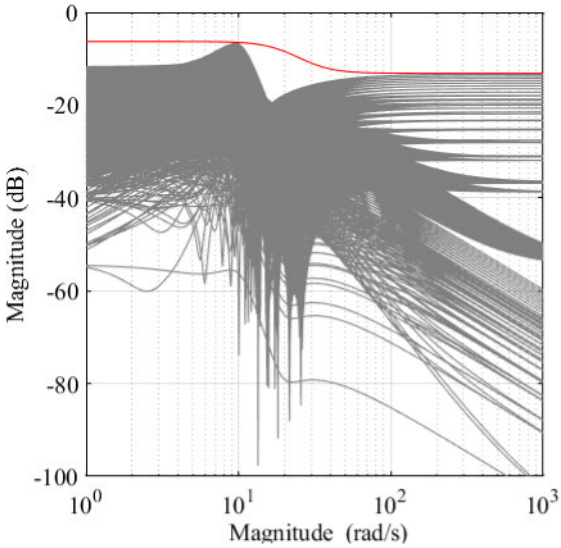


FIGURE 11. Comparison of the multiplicative uncertain functions magnitude $\Delta_{m,\text{lat}}(s)$ (gray) with the magnitude of the over-bounding function $\Gamma_{m,\text{lat}}(s)$ (red).

sensitivity function $S_{\text{lat},\dot{\psi}}(s)$ of the lateral deviation e_y with respect to the yaw rate reference disturbance $S_{\text{lat},\dot{\psi}}(s)$ and

the closed loop complementary sensitivity function $T_{\text{lat}}(s)$, defined as

$$S_{\text{lat},\dot{\psi}}(s) = \frac{G_{\delta,\dot{\psi}}(s)\,G_{\dot{\psi},e_y}(s)\,C^{\dot{\psi}}_{\text{lat}}(s)}{1+G_{\delta,\dot{\psi}}(s)\left(G_{\dot{\psi},e_y}(s)\,C^{y}_{\text{lat}}(s)+C^{\dot{\psi}}_{\text{lat}}(s)\right)}$$
$$T_{\text{lat}}(s) = \frac{G^{p}_{\text{lat}}(s)\,C^{y}_{\text{lat}}(s)}{1+G^{p}_{\text{lat}}(s)\,C^{y}_{\text{lat}}(s)}. \tag{37}$$

An H_∞ control synthesis method is adopted to handle robust stability in the presence of the model uncertainty described in (35) and, simultaneously, disturbance rejection and bandwidth requirements. More specifically, the lateral controller transfer function $C^{y}_{\text{lat}}(s)$ is designed through the following H_∞ optimization problem

$$C^{y}_{\text{lat}}(s) = \arg \min_{\|T_{\text{lat}}(s)\Gamma_{m,\text{lat}}(s)\|_\infty<1} \|W_S^{-1}(s)S_{\text{lat},\dot{\psi}}(s)\|_\infty, \tag{38}$$

where

$$W_S(s) = \frac{22s}{s^2+0.0707\,s+0.0025} \tag{39}$$

represents a proper real rational transfer function that describes the desired requirements on $S^{\text{nom}}_{\text{lat},\dot{\psi}}(s)$ in terms of closed-loop bandwidth and steady-state disturbance rejection for e_y when $\dot{\psi}_{des}$ is a step function The controller $C^{y}_{\text{lat}}(s)$ is computed through the MatLab function `hinfopt`, and a model order reduction procedure is performed before its implementation. We have implemented all the controllers transfer functions, i.e., C_{long}, $C^{\dot{\psi}}_{\text{lat}}$, and C^{y}_{lat}, by discretization using the Tustin method and sampling time T^{m}_{s}.

VI. SIMULATION RESULTS

In this section, we demonstrate the efficiency of the proposed approach through a simulation example.

A. EXECUTION ENVIRONMENT

The simulation is performed using MatLab and Simulink environments. Specifically, we utilized the *Automated Driving Toolbox* to simulate the highway road and the *Vehicle Dynamics Blockset* to describe the vehicle dynamics. The underlying MPC optimization problem was solved using the MatLab function `fmincon`, with a maximum peak of 0.17 s, using a computer equipped with 16 GB of RAM and an Intel i7-12700H processor.

B. COMPARISON AMONG CONTROL TECHNIQUE

To show the effectiveness of the proposed two-layer control architecture, denoted here as MPC-H_∞, we compare its performance with respect to

- a single-layer architecture based on a nonlinear single-layer MPC controller indicated as MPC-SL;
- a two-layer architecture denoted as MPC-LQ where a standard LQ control law as described in [1] replaces the lateral controller introduced in sub-section V-B and the path planner is the same MPC APF based controller described in subsection III-C.

In particular, the architecture MPC-SL embeds both the path planner and the motion controller and provides the required inputs directly to the HV. The design is performed by considering the dynamical single tack model (30) augmented by the longitudinal dynamics described by (28). As a consequence, the vehicle model becomes nonlinear. The underlying optimization problem employed to compute the control move is similar to (16), where the same APFs account for lateral and longitudinal maneuver performance. The optimization is performed with respect to the steering angle δ and the vehicle acceleration a. Constraints on the lateral acceleration a_y are also included to guarantee closed-loop stability. Because of the nonlinear characteristics of the optimization problem, a sampling time of 100 ms is chosen for such MPC-SL architecture.

As to the two-layer MPC-LQ architecture, the LQ controller is designed by considering the cost function introduced in (40) that aims at minimizing the lateral and the yaw rate deviations e_y and $e_{\dot{\psi}}$

$$J_{\text{LQ}} = \int_0^\infty (q_y\,e_y^2(\tau)+q_{\dot{\psi}}\,e_{\dot{\psi}}^2(\tau)+r_\delta\,\delta^2(\tau)\,d\tau) \tag{40}$$

where q_y, $q_{\dot{\psi}}$ and r_δ are suitable positive weights. To improve the nominal steady-state performance of e_y, a feed-forward contribution based on the knowledge of $\dot{\psi}_{des}$ is added as an additional control action of the LQ controller.

C. MANEUVERS DESCRIPTION

We have designed a 3.8 km long three-lane highway road scenario to simulate all possible situations that may occur in standard conditions. The width of each lane is 3.65 m. Fig. 12 shows the shape of the considered scenario. The simulation is composed of three consecutive sections to test different driving situations.

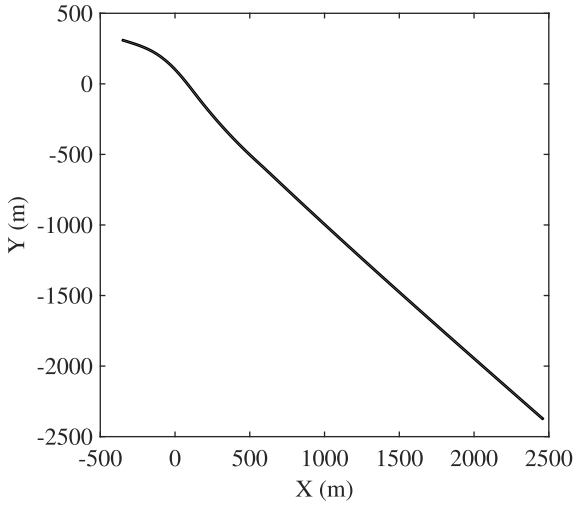


FIGURE 12. Road profile of the testing scenario. The simulation starts from the top-left vertex of the path.

In the first section, the road path bends to the right with a minimum radius of 500 m, the minimum curvature radius

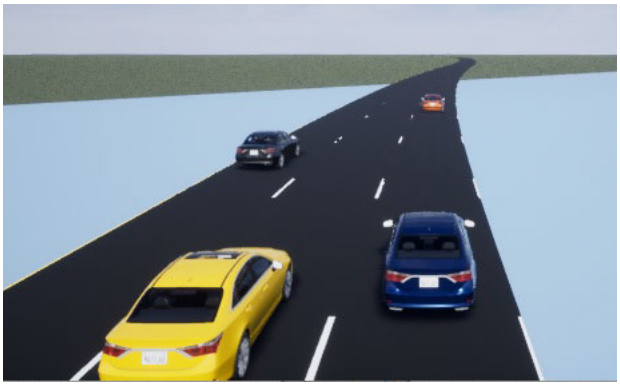


FIGURE 13. Vehicle positions during the first section of the simulation. The orange vehicle moves 70 km/h ahead of the HV (blue). The yellow vehicle is preventing the overtaking maneuver.

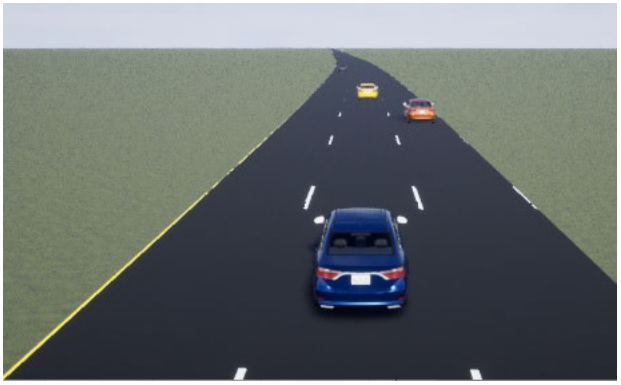


FIGURE 14. Vehicle positions during the second section of the simulation. The HV (blue) overtakes the orange car in the first lane. On the third lane, the distance from the purple vehicle is enough to begin the second overtaking maneuver of the yellow car.

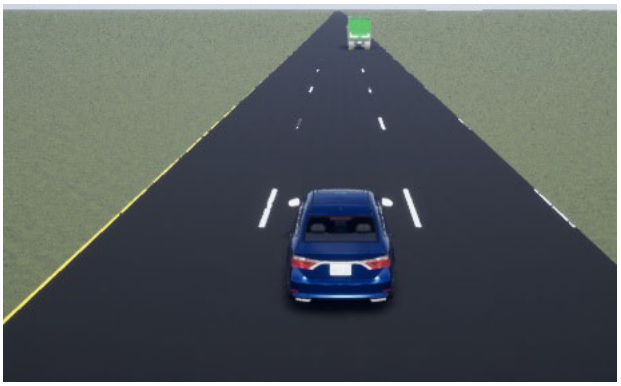


FIGURE 15. Vehicle positions during the final section of the simulation. The HV (blue) moves on the second lane, while a slower truck (green) travels on the rightmost lane.

allowed on Italian Highways. The HV starts on the rightmost lane at 100 km/h. Another vehicle is placed 90 m ahead on the same lane, moving at 70 km/h. On the second lane, a vehicle moves between 75 and 80 km/h, preventing the HV from performing the overtaking maneuver. Fig. 13 shows the vehicles' positions in this simulation stage. Until the central lane is empty, the HV has to follow the preceding vehicle, regulating the speed to maintain the right safety distance while traveling on the bending path. The objective of this first section of the path is to demonstrate the ability of the HV to perform the lane-keeping maneuver together with distance tracking from a slower vehicle ahead.

The HV can start the overtaking maneuver as soon as the vehicle in the second lane leaves the space. It is worth noting that the vehicle on the second lane is moving at 80 km/h, while the HV objective speed is set at 120 km/h. Thus, this second section of the path leads to a *double overtaking*, with the HV moving to the third and leftmost lane. The path section is completed when the HV vehicle moves correctly to the third lane, where a vehicle moving at 100 km/h is moving ahead. We show a snapshot of the vehicle positions during this stage in Fig. 14. In this second path section, we test the ability of the controlled system to perform the lane-change maneuver while maintaining the distance from preceding vehicles.

Finally, in the last section of the path, the preceding vehicle on the third lane moves to the second one. This allows the HV vehicle to reach the objective speed, complete the overtaking maneuver, and return to the second lane. The second lane is free till the end of the simulation, while the rightmost is still occupied by a truck moving at 70 km/h (see Fig. 15). As soon as the truck is overtaken, the HV moves to the first lane. Here, we demonstrate the speed-tracking ability while performing lane-change and lane-keeping maneuvers.

In such a driving scenario, the HV lateral performance are evaluated considering the position error ε^y_{ss} during steady state LK operation and the maximum lateral deviation $\varepsilon^y_{\max}$ in transient phases such as overtaking maneuvers. Similar indices are considered for distance tracking maneuvers, i.e., ε^x_{ss} and $\varepsilon^x_{\max}$ to account for longitudinal position errors at steady state and during the transient of the approaching maneuver respectively. Velocity tracking errors $\varepsilon^{v_x}_{ss}$ and $\varepsilon^{v_x}_{\max}$ are evaluated as well.

D. SIMULATION RESULTS IN NOMINAL CONDITIONS

This simulation is performed considering the nominal values of the vehicle parameters.

Fig. 16 shows the vehicle lateral position with respect to the rightmost lane border along the considered 3.8 km path. The simulation starts with the vehicle not exactly centered in the lane, leading to small initial oscillations. The first section of the path, when the HV performs the distance-keeping maneuver, is about 900 m long. After this, the double overtaking maneuver is performed. After 1050 m from the starting point of the simulation, the HV moves to the second lane to perform the first overtaking and immediately starts the successive overtaking maneuver. The second section ends when the HV is 1600 m from the initial point. At last, the HV performs the final returning maneuver to the rightmost free lane. In Fig. 16, we can observe quite a good performance of the lateral behavior of the HV. Let us consider the MPC-H_∞ HV. We see that

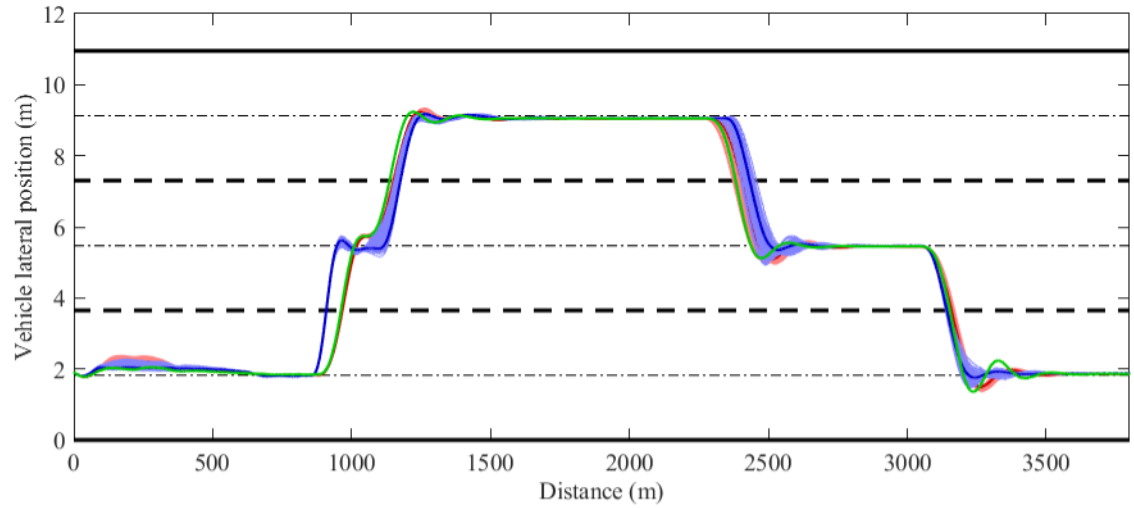


FIGURE 16. Vehicle lateral position along the designed scenario, together with the lanes limits (thick black) and center (dash-dotted). The dark red, blue, and green lines show the MPC-H_∞, MPC-SL, and MPC-LQ nominal trajectories, respectively. Light color lines are the trajectories for uncertain vehicle parameters.

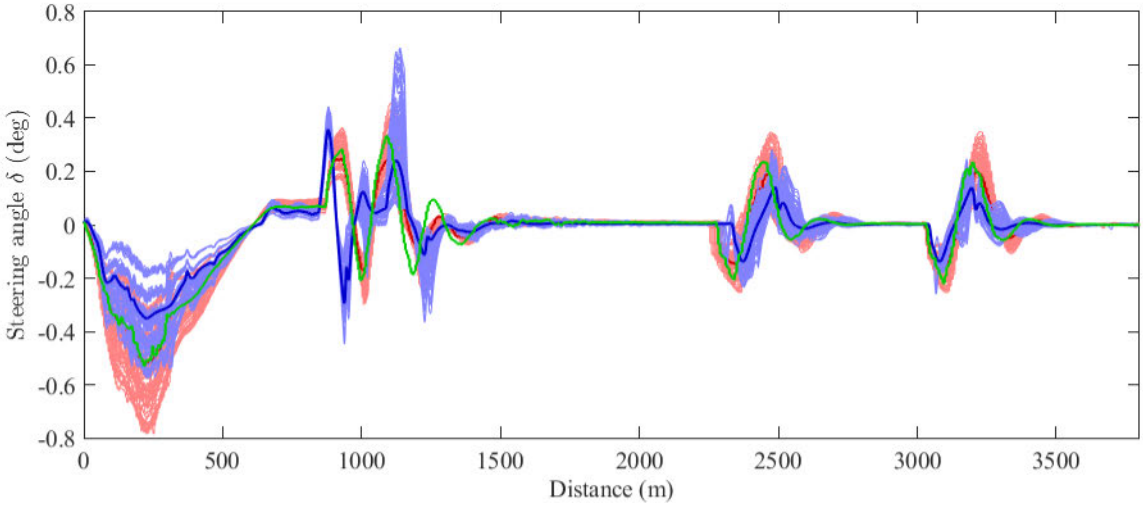


FIGURE 17. Steering angle δ along the trajectory. The dark red, blue, and green lines show the MPC-H_∞, MPC-SL, and MPC-LQ nominal inputs, respectively. Light color lines are the inputs for uncertain vehicle parameters.

1) in the first phase, when a slower car ahead induces a speed reduction of about 30 km/h, the HV can effectively keep the center of the lane in the right bend;
2) the lateral position error is about 4 cm from the lane center in all the steady state situations;
3) the maximum spatial overshoot is less than 3% with respect to the lane width during the lane change transient as happens at about 3000 m from the starting point when the maneuver is performed at 120 km/h within 200 m.

The steering angle δ employed to carry out the considered driving maneuvers is reported in Fig. 17. The plot of δ shows an even course characterized by small values, i.e., less than 0.5° that lead to smooth trajectories that ensure passengers' comfort, especially during the bend of the first 700 m of the road.

The MPC-LQ and MPC-H_∞ HV trajectories almost overlap. It is worth noting that these approaches differ only for the low-level lateral controller. The LQ controller is tuned to get the best performances along the scenario, where the HV speed changes from 80 km/h in the first part to 120 km/h in the last. Thus, the LQ controller has been designed by linearizing the vehicle dynamic around the average speed equal to 100 km/h. As the speed increases, the MPC-LQ HV lateral behavior has worse performances than the MPC-H_∞ HV, leading to less damped oscillations in the final first lane return (see Fig. 16).

The MPC-SL controller shows comparable nominal performances to the MPC-H_∞ HV. Due to the higher computational effort required by the MPC-SL approach, we have set the H_c and H_p horizons equal in both the MPC approaches. Through the same computer and numerical

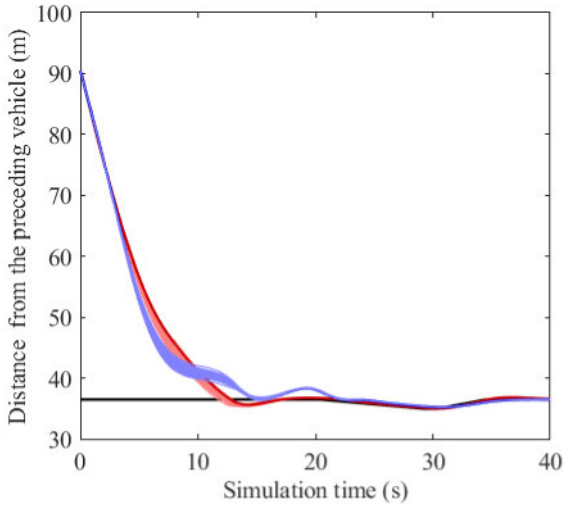


FIGURE 18. HV distance from preceding car during the distance-keeping maneuver. Comparison between the two vehicles' distance and the safety distance (black) computed as a function of the preceding vehicle speed. The dark red and blue lines show the MPC-H_∞ and, MPC-SL nominal trajectories, respectively. Light color lines are the trajectories for uncertain vehicle parameters.

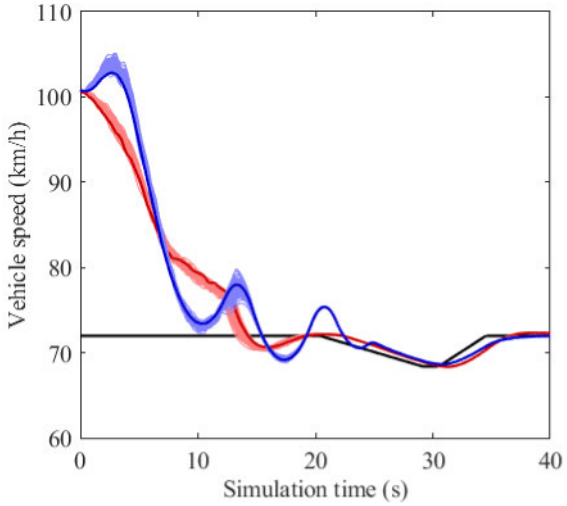


FIGURE 19. Comparison between the HV and slower vehicle ahead (black) velocities during the distance-keeping maneuver. The dark red and blue lines show the MPC-H_∞ and, MPC-SL nominal trajectories, respectively. Light color lines are the trajectories for uncertain vehicle parameters.

solver, the MPC-SL problem is solved in 0.5 s, almost two times more than the proposed MPC path planner. Moreover, it is worth noting that the solver time is six times larger than the MPC-SL sampling time, making this approach not applicable as it is to a real vehicle. The MPC-SL controlled AV shows slightly different longitudinal dynamics. In fact, from Fig. 16, we see that the MPC-SL HV vehicle performs the lane change maneuvers at different times with respect to the MPC-H_∞ HV. However, the more considerable longitudinal difference is only 130 m in performing the lane change. The MPC-SL shows good performances in the lateral behavior since the maximum overshoot during the lane change is 1.5% of the lane width. Fig. 18 compares the relative distance between HV and the preceding vehicle with the target value introduced in (7) and proves the achieved good tracking performance. Similar accuracy is obtained while tracking the preceding vehicle speed, as shown in Fig. 19. The plots do not show the MPC-LQ trajectory since it is equal to the one from the MPC-H_∞ HV. The MPC-H_∞ approaches the preceding vehicle with a smoother trajectory. The desired safety distance is reached in 15 s by the MPC-H_∞ HV and in 22 s by the MPC-SL HV. The MPC-SL HV shows large speed oscillations approaching the preceding slower vehicle, leading to reduced comfort for the HV passengers.

To evaluate the comfort performance, Fig. 20 shows the G-G diagram of the lateral and longitudinal accelerations obtained along the simulation. The maximum longitudinal acceleration achieved by the MPC-H_∞ HV is 1.5 ms^{-2}, while the lateral acceleration is always less than 0.25 ms^{-2}. The lateral acceleration evaluated for the MPC-SL HV reaches higher values, up to 0.6 ms^{-2}, often in combination of large longitudinal acceleration phases. On the overall, user comfort is achieved by both MPC-H_∞ and MPC-SL HVs throughout the simulation since the equivalent acceleration is less than the preset boundary of 2.5 ms^{-2}. An animation video of the simulation is available online [37].

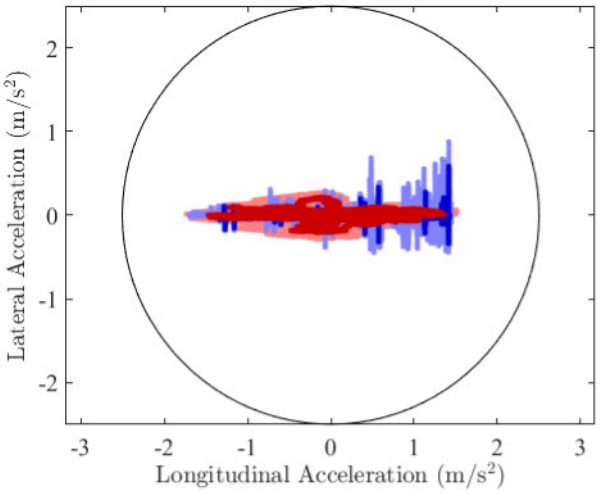


FIGURE 20. G-G diagram showing the set of longitudinal and lateral accelerations estimated during the simulation. The black circle represents the comfort region. The dark red and blue dots show the MPC-H_∞ and MPC-SL nominal G-G values, respectively. Light color lines are the G-G values for uncertain vehicle parameters.

E. CONTROL SENSITIVITY ANALYSIS

In this subsection, we perform a sensitivity analysis to test the robustness properties of the proposed control strategy in the presence of vehicle parameter variations. Through uniform gridding of the vehicle parameter ranges described in Subsection V-B, we have performed 100 simulations to evaluate the control performances in the proposed highway scenario. The robustness analysis for the MPC-LQ-controlled autonomous vehicle led to unstable behavior, even for small vehicle parameter variations.

TABLE 2. Comparison of the tested HV control techniques performances along the considered scenario.

	Nominal vehicle				Robustness tests				
Control strategy	ε_{ss}^{y} (cm)	$\varepsilon_{\max}^{y}$ (cm)	$\varepsilon_{ss}^{v_x}$ (km/h)	$\varepsilon_{\max}^{v_x}$ (km/h)	$\Delta\varepsilon_{ss}^{y}$ (cm)	$\Delta\varepsilon_{\max}^{y}$ (cm)	$\Delta\varepsilon_{ss}^{v_x}$ (km/h)	$\Delta\varepsilon_{\max}^{v_x}$ (km/h)	Computation time
MPC-H_∞	4	10	0.2	1.4	0.5	18	0.1	1.5	0.17 s
MPC-SL	3	10	0.1	3.3	0.9	40	0.1	3	0.5 s
MPC-LQ	4	13	0.2	1.4	Vehicle unstable				0.17 s

In Fig. 16, we show the vehicle's lateral position with respect to the rightmost lane. The trajectories in all the tests almost match the nominal performance. The MPC-H_∞ HV shows lateral displacement trajectories within a range of 18 cm during the lane change maneuvers, while the MPC-SL HV has larger lateral displacement, up to 40 cm. In steady-state conditions, the vehicle parameter uncertainty leads to a lateral error up to 5 mm and 7 mm with respect to the nominal trajectory while considering, respectively, the MPC-H_∞ and MPC-SL controller. The larger difference in the trajectory is due to the longitudinal behavior since variations in the vehicle mass are responsible for different accelerations when the same longitudinal input is applied. This difference is evident in Fig. 16 after 2300 m from the starting point of the simulation when the HV performs the return to the second lane.

The controller also shows robust performances during the distance-keeping maneuver. From Fig. 18, we see that the controller keeps the target distance from the preceding car despite uncertain parameters. On average, the perturbed vehicle is farther with respect to the nominal HV: the maximum error in the relative distance is about 5 cm for both approaches. User comfort is preserved as well. In fact, in the G-G plot reported in Fig. 20, the uncertainty still ensures the comfort performance. The differences are due to the low-level controller input when applied to cars with different masses and moments of inertia. In the worst case, the longitudinal acceleration reaches 1.9 ms^{-2}, while the lateral acceleration is less than 0.3 ms^{-2} for the MPC-H_∞ HV. The effects of the uncertain parameters affects the G-G plot for the MPC-SL HV as well. In this case, the lateral acceleration reaches 1 ms^{-2}, while the longitudinal acceleration is comparable to the MPC-SL controlled vehicle. The uncertain parameter values lead to larger activity for the steering angle δ (see Fig. 17), especially in the bending section of the path, where the input values are within a 0.8 degrees range. However, it is worth noting that the applied input allows the HV to follow the trajectory (Fig. 16) and ensure passengers' comfort performance (Fig. 20).

The overall comparison between the three approaches is reported in Table 2.

VII. CONCLUSION

We presented an original approach for automated driving in highway scenarios using a two-layer control architecture. The higher level includes a planner that computes the road path through an MPC algorithm that exploits a simplified kinematic model of the vehicle and suitable APFs combinations to handle the most relevant maneuvers of highway driving effectively. The motion controller of the lower level is implemented through a decentralized architecture to regulate the vehicle longitudinal and lateral dynamics for the required trajectory tracking. Robust synthesis methods of both longitudinal and lateral controllers allow the overall automated driving architecture to account for plant uncertainty. A complete simulation setting that includes all the relevant highway driving maneuvers such as lane keeping, distance end speed-tracking, overtaking, and returning has been built to evaluate the achieved performance. In particular, a sensitivity analysis has been carried out in the presence of perturbation of the vehicle model relevant parameters. The results show that the introduced approach achieves quite good robust performance in all the maneuvers considered regarding path tracking, accuracy, and comfort. The effectiveness of the proposed approach is shown by comparing the performance obtained with a single-level architecture based on an MPC-APF strategy and a two-layer structure that includes an LQ strategy as a motion controller.

REFERENCES

[1] R. Rajamani, *Vehicle Dynamics and Control*. New York, NY, USA: Springer, 2012.

[2] B. Paden, M. Cáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Trans. Intell. Vehicles*, vol. 1, no. 1, pp. 33–55, Mar. 2016.

[3] *Waymo. (21) Autopilot AI*. Accessed: Jun. 19, 2024. [Online]. Available: https://waymo.com/tech/

[4] *Tesla. (21) Former Google Self-Driving Car Project*. Accessed: Jun. 19, 2024. [Online]. Available: https://www.tesla.com/autopilotAI

[5] *Audi. (21) Audi Autonomous Driving*. Accessed: Jun. 19, 2024. [Online]. Available: https://www.audi.com/en/innovation/future-technology/autonomous-driving.html

[6] *Mercedes-Benz. (21) Mercedes-Benz Autonomous Driving*. Accessed: Jun. 19, 2024. [Online]. Available: https://group.mercedes-benz.com/innovation/product-innovation/autonomous-driving/

[7] *Stellantis. (23) Stellantis Autonomous Driving*. Accessed: Jun. 19, 2024. [Online]. Available: https://www.stellantis.com/en/technology/autonomous-driving

[8] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, vol. 2, Mar. 1985, pp. 500–505.

[9] P. B. Kumar, H. Rawat, and D. R. Parhi, "Path planning of humanoids based on artificial potential field method in unknown environments," *Expert Syst.*, vol. 36, no. 2, Apr. 2019, Art. no. e12360.

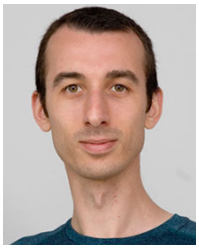
[10] X. Tang, T. Zhou, J. Yu, J. Wang, and Y. Su, "An improved fusion algorithm of path planning for automated guided vehicle in storage system," in *Proc. IEEE 4th Int. Conf. Comput. Commun. (ICCC)*, Dec. 2018, pp. 510–514.

[11] L. Cheng, C. Liu, and B. Yan, "Improved hierarchical A-star algorithm for optimal parking path planning of the large parking lot," in *Proc. IEEE Int. Conf. Inf. Autom. (ICIA)*, Jul. 2014, pp. 695–698.

[12] S. Sedighi, D.-V. Nguyen, and K.-D. Kuhnert, "Guided hybrid A-star path planning algorithm for valet parking applications," in *Proc. 5th Int. Conf. Control, Autom. Robot. (ICCAR)*, Apr. 2019, pp. 570–575.

[13] Y. Kuwata, G. A. Fiore, J. Teo, E. Frazzoli, and J. P. How, "Motion planning for urban driving using RRT," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2008, pp. 1681–1686.
[14] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 846–894, Jun. 2011.
[15] Q. Shi, J. Zhao, A. E. Kamel, and I. Lopez-Juarez, "MPC based vehicular trajectory planning in structured environment," *IEEE Access*, vol. 9, pp. 21998–22013, 2021.
[16] N. Noto, H. Okuda, Y. Tazaki, and T. Suzuki, "Steering assisting system for obstacle avoidance based on personalized potential field," in *Proc. 15th Int. IEEE Conf. Intell. Transp. Syst.*, Sep. 2012, pp. 1702–1707.
[17] Z. Huang, Q. Wu, J. Ma, and S. Fan, "An APF and MPC combined collaborative driving controller using vehicular communication technologies," *Chaos, Solitons Fractals*, vol. 89, pp. 232–242, Aug. 2016.
[18] Y. Rasekhipour, A. Khajepour, S.-K. Chen, and B. Litkouhi, "A potential field-based model predictive path-planning controller for autonomous road vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1255–1267, May 2017.
[19] J. Ji, A. Khajepour, W. W. Melek, and Y. Huang, "Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 952–964, Feb. 2017.
[20] Z. Qi, T. Wang, J. Chen, D. Narang, Y. Wang, and H. Yang, "Learning-based path planning and predictive control for autonomous vehicles with low-cost positioning," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 2, pp. 1093–1104, Feb. 2023.
[21] H. Li, W. Liu, C. Yang, W. Wang, T. Qie, and C. Xiang, "An optimization-based path planning approach for autonomous vehicles using the DynEFWA-artificial potential field," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 2, pp. 263–272, Jun. 2022.
[22] J. Nilsson, P. Falcone, M. Ali, and J. Sjöberg, "Receding horizon maneuver generation for automated highway driving," *Control Eng. Pract.*, vol. 41, pp. 124–133, Aug. 2015.
[23] M. Nezami, H. S. Abbas, N. T. Nguyen, and G. Schildbach, "Robust tube-based LPV-MPC for autonomous lane keeping," in *Proc. 5th IFAC Workshop Linear Parameter Varying Syst. (LPVS)*, 2022, vol. 55, no. 35, pp. 103–108.
[24] T. Brüdigam, M. Olbrich, D. Wollherr, and M. Leibold, "Stochastic model predictive control with a safety guarantee for automated driving," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 1, pp. 22–36, Jan. 2023.
[25] M. Li, H. Cao, G. Li, S. Zhao, X. Song, Y. Chen, and D. Cao, "A two-layer potential-field-driven model predictive shared control towards driver-automation cooperation," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 5, pp. 4415–4431, May 2022.
[26] Z. He, L. Nie, Z. Yin, and S. Huang, "A two-layer controller for lateral path tracking control of autonomous vehicles," *Sensors*, vol. 20, no. 13, p. 3689, Jul. 2020.
[27] M. Canale, A. E. Belvedere, and V. Razza, "Autonomous driving in highway scenarios through artificial potential fields and model predictive control," in *Proc. Eur. Control Conf. (ECC)*, Jul. 2022, pp. 8–13.
[28] *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*, SAE Int., Warrendale, PA, USA, 2021.
[29] *Bosch. (21) Bosch Autonomous Driving*. Accessed: Jun. 19, 2024. [Online]. Available: https://www.bosch-mobility.com/en/mobility-topics/automated-driving-sense-think-act/
[30] C. W. Warren, "Global path planning using artificial potential fields," in *Proc., Int. Conf. Robot. Autom.*, Sep. 1989, pp. 316–321.
[31] K. G. Jolly, R. S. Kumar, and R. Vijayakumar, "A Bezier curve based path planning in a multi-agent robot soccer system without violating the acceleration limits," *Robot. Auto. Syst.*, vol. 57, no. 1, pp. 23–33, Jan. 2009.
[32] K. Kawabata, L. Ma, J. Xue, C. Zhu, and N. Zheng, "A path generation for automated vehicle based on Bezier curve and via-points," *Robot. Auto. Syst.*, vol. 74, pp. 243–252, Dec. 2015.
[33] J. W. Rutter, *Geometry Curves*. London, U.K.: Chapman & Hall, 2000.
[34] R. Attia, R. Orjuela, and M. Basset, "Combined longitudinal and lateral control for automated vehicle guidance," *Vehicle Syst. Dyn.*, vol. 52, no. 2, pp. 261–279, Feb. 2014.
[35] H. Atoui, O. Sename, V. Milanés, and J. J. Martinez, "LPV-based autonomous vehicle lateral controllers: A comparative analysis," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 13570–13581, Aug. 2022.
[36] M. Canale and L. Fagiano, "Comparing rear wheel steering and rear active differential approaches to vehicle yaw control," *Vehicle Syst. Dyn.*, vol. 48, no. 5, pp. 529–546, May 2010.
[37] M. Canale and V. Razza. (2023). *Autonomous Driving in Highway Scenarios Simulation Example*. [Online]. Available: https://youtu.be/AxeZtKjNT7s

MASSIMO CANALE (Member, IEEE) received the M.Sc. degree in electronic engineering and the Ph.D. degree in information and system engineering from Politecnico di Torino, Italy, in 1992 and 1997, respectively. From 1997 to 1998, he was a Software Engineer with the Research and Development Department, Comau Robotics, Italy. In 1998, he joined the Department of Control and Computer Engineering, Politecnico di Torino, where he is currently an Associate Professor of automatic control. At present, he is on a Research Staff with the Center for Automotive Research and Sustainable Mobility (CARS), Politecnico di Torino. His research interests include robust control, model predictive control, set membership approximation, and application to automotive and sustainable energy problems. He is a member of the IEEE Control System Society (CSS) Technical Committee on Automotive Control and IEEE Intelligent Transportation Systems Society. In 2011, he has been a co-recipient of IEEE Transactions on Control Systems Technology Outstanding Paper Award.



VALENTINO RAZZA received the M.S. degree in computer engineering and the Ph.D. degree in information and system engineering from Politecnico di Torino, Italy, in 2008 and 2012, respectively. From 2009 to 2013, he was a Post-doctoral Researcher with the System Identification and Control (SIC) Group, Control and Computer Department, Politecnico di Torino. From January 2014 to June 2015, he was with the Multi-Layer Wireless Solutions Group, Istituto Superiore Mario Boella. From 2015 to 2019, he was a Research Assistant with the Control and Computer Department, SIC Group, Politecnico di Torino. After that, he was a Researcher with the Artificial Mechanical Intelligence Group, Italian Institute of Technology, in 2022. He currently holds the position of an Assistant Professor with the Management and Production Engineering Department, Politecnico di Torino. His main research interests include identification, robust control, and model predictive control of linear and nonlinear systems, with an application to real-world problems in industrial process technologies and the automotive and green-energy fields.

Open Access funding provided by 'Politecnico di Torino' within the CRUI CARE Agreement