Autonomous Driving in Highway Scenarios through Artificial Potential Fields and Model Predictive Control

Autonomous Driving in Highway Scenarios through Artificial Potential Fields and Model Predictive Control

M. Canale*, A. E. Belvedere, V. Razza

***Abstract*— An approach for automated driving in highway scenarios in the context of a two levels hierarchical architecture is proposed. In particular, we define suitable artificial potential functions (APF) combinations that can effectively handle the most relevant maneuvers of highway driving, such as speed and distance tracking, lane keeping, overtaking and returning. Parameters of the APF functions are dynamically tuned according to the acquired scenario. The defined APF are included in the cost function of a Model Predictive Control (MPC) control problem to generate the path trajectory. A behavioral logic described by a finite state machine (FSM), based on sensor acquired data and suitable dynamic conditions is defined to select the most appropriate maneuver to realize. Extensive simulation tests are introduced to show the effectiveness of the proposed approach.**

I. Introduction

In the last decades, advanced driving assistance systems (ADAS) have gained the attention from both academia and industry. Car manufactures have equipped their vehicles with several automated driving features, such as lane keeping and adaptive cruise control, that are leading to more complex fully autonomous driving cars. In this perspective, several manufactures are already facing autonomous vehicles (AV), e.g., Waymo [1], Tesla [2], and Audi [3]. According to the society of automotive engineers (SAE) classification ([4]), the safety systems are the lower levels of vehicle automation.

The key component of autonomous driving is the path planner, that aims at finding the optimal collision free path to the target destination. In the literature, several approaches have been investigated. In low complexity environments, A* algorithm is deeply analyzed (see, e.g. [5], [6]).

Another effective approach is based on the use of potential fields functions, where the final target is represented by an attractive potential and the obstacles are described by repulsive potentials. The environment is represented by the combination of different fields and the path is given by the minimum energy trajectory. Artificial potential fields (APF) have been firstly introduced in [7] as a collision avoidance method for robotic manipulator and mobile robots. The main advantage of this method is the possibility to adapt the path generation in closed loop with the environment ([8]). However, the APF approach cannot account for overall vehicle limitations, e.g., on steering angle ratio and longitudinal acceleration. These constraints can be taken into account through customized potential fields (see, e.g., [9]), or combining APF scenario description together with control methodologies well suited for vehicle dynamics control. In particular, model predictive control (MPC) is an effective control technique in different automotive applications, which takes into account the vehicle dynamical limits ([10]). The possibility to dynamically build the APF at each sample time can be exploited from the MPC strategy through its intrinsic receding horizon principle. In [11], the authors combine APF and MPC for path planning, while a collision avoidance algorithm is presented in [12]. In both [11] and [12], path generation and vehicle control are performed exploiting a single level hierarchical architecture where the MPC algorithm performs at the same time as path generator and motion controller. In particular, in such approaches, vehicle dynamics are accounted for by a single track dynamic model, while closed loop stability is ensured by means of suitable additional constraints in the MPC underlying optimization problem. Other approaches, instead, exploit two levels architectures: the upper level, made up by an MPC path planner, computes the vehicle trajectory and the speed to be tracked by the vehicle, while the lower level implements the vehicle longitudinal and lateral control functions that realize the needed maneuver.

In this paper, we propose an approach for automated driving in highway scenarios in the context of a two levels hierarchical architecture. In particular, we define suitable APFs combinations that can effectively handle the most relevant maneuvers of highway driving, such as speed and distance tracking, lane keeping, overtaking and returning. Parameters of the APF functions are dynamically tuned according to the acquired scenario. The defined APF are included in the cost function of an MPC control problem to generate the path trajectory. A behavioral logic, described by a finite state machine (FSM) and based on sensor-acquired information and suitable dynamic conditions, is defined to select the most appropriate maneuver to realize.

II. Vehicle Model

Here, we assume that an autonomous vehicle, denoted as the host vehicle (HV), is moving in a highway scenario according to the SAE automation level 3, see [4]. In the considered highway driving scenario, characterized by maneuvers with small steering angles, the single track kinematic model described in Fig. 1 appears a reasonable choice to describe the lateral behavior for path planning purpose, see, e.g., [13]. Furthermore, in Fig. 1, the frame (x,y) is introduced to define the vehicle behaviour and the local frame(X,Y) to describe its interaction with the highway environment. The nonlinear kinematic equations of

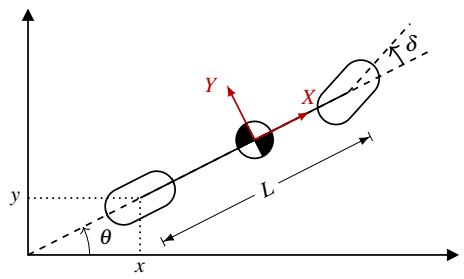

Fig. 1. Single-track kinematic model schematic and frames.

Authors are with the Dipartimento di Automatica e Informatica, Politecnico di Torino, corso Duca degli Abruzzi 24, Torino, 10129, Italy.
* Corresponing author massimo.canale@polito.it

the single-track model are reported below.

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

In (1), x, y and θ describe the pose of the HV rear axle in the frame, v is the longitudinal speed, δ is the steering angle and L is the wheelbase. The relevant variables in (1), can be collected in a state vector ξ defined as

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

Moreover, the longitudinal acceleration $a = \dot{v}$ and the angular speed $\omega_\delta = \dot{\delta}$ acts as exogenous inputs.

In this paper, we assume that the HV is equipped with a quite complete sensor configuration as described in the list below.

1) Vehicle dynamic sensor for acquisition of the relevant longitudinal and lateral variables.
2) Front camera, to detect and reconstruct lane edge and curvature.
3) Surround view cameras to detect lateral scenarios such as vehicles in adjacent lanes.
4) Front radar (long and short range) to detect preceding vehicles relevant variables such as relative distance d_R and speed v_R.
5) IMU unit for the most relevant inertial measurements.
6) GPS to locate the vehicle in world coordinate.

III. Artificial Potential Fields for Autonomous Driving

Artificial Potential Fields (APF) are an efficient tool to handle quite complex scenarios in autonomous driving. In fact, through the definition of suitable virtual attractive or repulsive forces, APF can generate safe regions to keep the vehicle trajectory and to reach target locations while keeping a safety distance to obstacles.

In the considered context, an APF is a real non-negative scalar function $P(X,Y)$ of the local coordinates (X,Y). The negative gradient $-\nabla P(X,Y)$, see, e.g. [14], provides the resulting virtual force magnitude and direction that the APF exerts on the HV. The combination of two or more APF can account for standard maneuvers in highway scenarios such as lane keeping, overtaking and maintaining target distance to a preceding vehicle. In the next subsections, III-A, III-B and III-C, details about how such maneuvers can be handled through the choice of suitable combinations of APF are introduced.

A. Lane Keeping

During the lane-keeping phase, the control system drives the vehicle to track the centre of the lane, keeping the correct orientation. In terms of APFs, this control feature can be accounted for through an attractive field such that the virtual force pulls the vehicle towards the centre of the lane. To this purpose, the combination P_ℓ of two second-order Gaussian functions P_{ℓ_l} and P_{ℓ_r} centered at the left and the right border lane, respectively, is employed as described in (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 is the height of the field, γ_0 is the width and d_i is the Euclidean distance between the vehicle and the left and right borders of the lane fixed at points (X_{0_l}, Y_{0_l}) and (X_{0_r}, Y_{0_r}) respectively. A suitable selection of the parameters P_0 and γ_0 allows one to describe the lane keeping task. In particular, since the task objective is to prevent the vehicle from escaping the lane and not to overtake, it is sufficient to permanently assign P_0 a high value. Parameters γ_{0_i}, can be computed by assigning a specific value to the field function at the target distance d_{tar}. In this case the target distance is the center of the lane whose width is w_L, i.e. when $d_{\text{tar}} = \dfrac{w_L}{2}$. In this way, according to (4) and (5), γ_0 can be computed as

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

Fig. 2 shows an example of the resulting APF for a curved road with width $w_L = 3.6$ m, $d_{\text{tar}} = 1.8$ m and curvature radius $R = 500$ m. The other parameters are $P_0 = 100$, $P_{L_i}(d_{\text{tar}}) = 0.1, i = l, r$. In the proposed situation, the right lane is the road boundary and it is represented by a straight line. The red line is the center lane reconstruction that matches with the local minimum of the function P_{ℓ_i}.

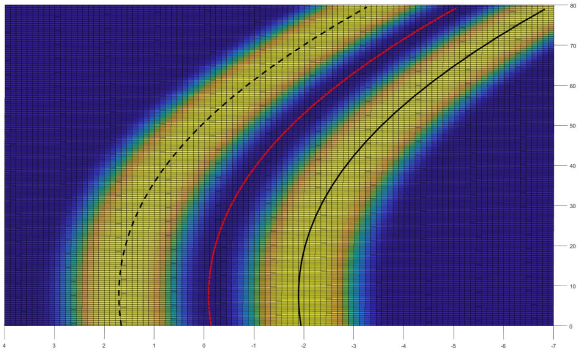

Fig. 2. Lane keeping APF. Blue areas represent low-value points while yellow areas represent high-value points.

B. Lane Changing

The goal of Lane Changing is to move the vehicle from one lane to the adjacent one. Similarly as the lane keeping task, the combination of two repulsive fields is employed to perform the maneuver. For simplicity, we analyze a lane change to the left, e.g., for an overtaking maneuver. Similar considerations hold for the case of right lane change. A first repulsive field is centered on the right line, while a second field is responsible for damping and limiting the vehicle motion as it enters the new lane. As the vehicle enters into the new 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 the same fields proposed for lane keeping with minor modifications. In particular, since the first field must push the vehicle to the new lane, a part of it must overlap the new lane. Thus, γ_0 is computed according to (6) with $d_{\text{tar}} = w_L$. In this way, the objective becomes to track the center of the destination lane. Fig. 3 reports an example of the APF employed to perform a line change.

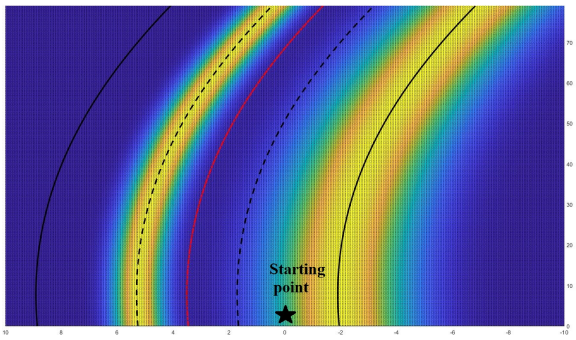

Fig. 3. Example of lane change PF built to move on the left line.

C. Obstacle Potential Field

The obstacles APF describes the influence of the obstacle vehicles (OV) on HV and thus, several fundamental aspects must be considered when building the APF. Here, it is assumed that the HV sensors can provide either directly or indirectly information about the obstacle pose (X_O, Y_O, ψ_O) and speed v_O. Notice that, with respect to the HV frame, the obstacle kinematic behavior can be described by a single-track model similar to the one reported in (1).

The final goal of the obstacle APF is to regulate the HV speed v to keep a desired distance d_{tar} from a preceding vehicle that travels at given speed v_O. Basically, we consider the case when an obstacle is in the same lane of the vehicle and it is not possible to perform an overtaking maneuver because of the presence of vehicles in the left lane. The target distance is defined as

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

where a_{dec} is the desired deceleration magnitude value, d_0 is a safety distance that includes the size of the obstacle, v_O is the obstacle speed and t_H is the time headway, defined as $t_H = \frac{d_O}{v}$, where d_O is the relative distance between the vehicle and the obstacle. To impose a smooth maneuver during the target distance approaching, two phases are considered:

1) the vehicle decelerates thanks to a repulsive field;
2) the vehicle tracks the target position through an attractive field.

Considering the coordinates (X_O, Y_O) and the direction ψ_O of the obstacle in the local HV frame, the following first-order Gaussian function is introduced to describe the repulsive APF for phase 1) described 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

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

The parameters γ_X and γ_Y define the Gaussian function sizes along the principal obstacle axes, while P_0 defines the function height. A similar procedure, as the one described by (6), is employed to compute both γ_X and γ_Y

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

Assuming to travel on a semi-straight road, it is reasonable to choose $d_{\text{tar,o},X} = d_{\text{tar,o}}$, see (7). Notice that $d_{\text{tar,o},X}$ depends on the relative speed between HV and obstacle. As to $d_{\text{tar,o},Y}$, a suitable fixed value that depends on the obstacle width is selected. Moreover, $P_{\text{o},r}(d_{\text{tar},X}) = P_{\text{o},r}(d_{\text{tar},Y}) = \bar{P}_{\text{o}}$ is chosen. Fig. 4 reports an example of the repulsive obstacle APF, where the relevant parameter values are: $X_O = 50$ m, $Y_O = -2.6$ m, $\psi_0 = -4.7°$, $v_O = 20$ m/s, $d_0 = 2$ m, $t_H = 1.5$ s, $P_0 = 100$, $\bar{d}_{\text{tar,o},Y} = 3$ m, and $\bar{P}_{\text{o}} = 0.1$. For the second phase,

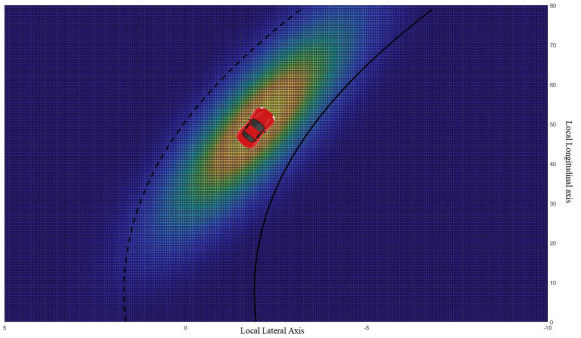

Fig. 4. Obstacle repulsive field: Blue areas represent low-value points, while yellow areas high-value points.

the attractive APF function 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 $(\tilde{X}'_O, \tilde{Y}'_O) = (X_O - d_{\text{tar}}, Y_O)$. The parameters in (12) are tuned through the same procedure described for (8).

IV. MPC Path Planner

In this section, we describe how the path planning task is effectively approached through the use of Model Predictive Control (MPC) methodologies. Since the MPC control is a discrete time methodology, the vehicle model (1) has been discretized using the forward Euler method and sampling time T_s. In this application, the sampling time is chosen as $T_s = 200$ ms. The discretized model is reported in (13).

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

Given the discrete time setting, the last two equations of the state representation (13) are written as

$$\begin{aligned} v(k+1) &= v(k) + \Delta v(k) \\ \delta(k+1) &= \delta(k) + \Delta\delta(k) \end{aligned} \tag{14}$$

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. As a consequence, the control input of (13) becomes

$$u(k) = \begin{bmatrix} \Delta v(k) & \Delta\delta(k) \end{bmatrix}^T. \tag{15}$$

Considering the system state defined in (2) and the control input introduced in (15), the state equation (13)-(14) are rewritten in the compact nonlinear form

$$\xi(k+1) = f(\xi(k), u(k)) \tag{16}$$

The MPC path planner controller is then designed according to the following objectives.

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 possible.
3) Keep the center of the lane when travelling in a bend.
4) Perform overtaking of a slower vehicle and returning in the rightmost lane when overtaking is completed.

At this point, in (17), we introduce a suitable cost function J as a weighted sum of contributions that account for the control objectives listed above.

$$\begin{aligned} J(U(k)) \doteq \sum_{i=k}^{k+H_P} (&\|P_o(i)\|_O + \|P_l(i)\|_L + \|v(i) - v_{des}\|_Q \\ &+ \|\Delta v(i)\|_{R_1} + \|\Delta\delta(i)\|_{R_2}) \end{aligned} \tag{17}$$

where

$$U(k) = \begin{bmatrix} \Delta v(k) & \Delta\delta(k) \\ \Delta v(k+1) & \Delta\delta(k+1) \\ \vdots & \vdots \\ \Delta v(k+H_P-1) & \Delta\delta(k+H_P-1) \end{bmatrix} \tag{18}$$

The cost function $J(U(k))$, introduced in (17), is defined over a prediction horizon H_P and includes several terms that describes how the control objectives can be considered. More specifically:

- $\|P_o(\tau)\|_O$ implements the distance keeping functionality.
- $\|P_\ell(\tau)\|_L$ is responsible for either the lane keeping or the lane changing feature.
- $\|v(\tau) - v_{\text{des}}\|_Q$ accounts for the achievements of the target speed.
- $\|\Delta v(\tau)\|_{R_1}$ and $\|\Delta\delta(\tau)\|_{R_2}$ handle intensity and physical reliability of the control action to guarantee safety, vehicle stability, fuel consumption optimization and driving comfort.

Weight matrices O, L, Q, R_1 and R_2 are chosen to regulate the desired trade-off among all the control objectives through a trial and error procedure. Given the cost function (17), the underlying optimization problem for MPC design is formulated as

$$\begin{aligned} &\min_{U(k)} J(k) \\ &\text{s.t.} \\ &(16) \\ &0 \le v(i) \le v_{\text{des}}, i = 0, \dots, H_P - 1 \\ &-\bar{\Delta}_v T_s \le \Delta v(i) \le \bar{\Delta}_v T_s, i = 1, \dots, H_P - 1 \\ &-\bar{\delta} \le \delta(i) \le \bar{\delta}, i = 0, \dots, H_P - 1 \\ &-\bar{\Delta}_\delta T_s \le \Delta\delta(i) \le \bar{\Delta}_\delta T_s, i = 1, \dots, H_P - 1. \end{aligned} \tag{19}$$

In (19), constraints on the steering angle δ and its rate $\Delta\delta$ account for physical limitations of the actuator speed. Limitations on v and Δv introduce a suitable speed limitation and comfort performance during acceleration maneuvers. The desired velocity v_{des} is chosen as

$$v_{\text{des}} = \min\left(v_{\text{tar}}, v_0 + a_{\max} T_p, \sqrt{a_{lat_{\max}} R}\right). \tag{20}$$

As to the prediction horizon, a value $H_P = 15$ is chosen, that corresponds to a prediction time $T_P = H_P T_s = 3$ s. Furthermore, a *control horizon* $H_C = 8$ is introduced to reduce the number of optimization variables. The remaining manipulable inputs are set as $U(i) = U(H_C), i = H_C + 1, \dots, H_P - 1$. The control input is then computed according to 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

$$\begin{aligned} v(k) &= v(k-1) + T_s \Delta v(k) \\ \delta(k) &= \delta(k-1) + T_s \Delta\delta(k) \end{aligned} \tag{21}$$

Table I resumes the values of the parameters employed in the MPC path planner design.

Parameter Description	Symbol	Value
Sampling time	T_s	200 ms
Prediction horizon	H_P	15 steps
Control horizon	H_C	8 steps
Rate limit for v	$\bar{\Delta}_v$	$2.5T_s$ ms^{-1}
Limit for δ	$\bar{\delta}$	25°
Rate limit for δ	$\bar{\Delta}_\delta$	$0.47T_s$ ° s^{-1}

TABLE I
MPC PARAMETERS.

The just described MPC problem represents the higher level of the autonomous driving architecture, that must generate the path way-points and the speed reference to be tracked by the low level lane keeping and speed motion controller. At this point, we recall that the path planner computation occurs at a lower rate with respect to the lower lever motion controller. More specifically, the sampling time for the path planner is $T_s = 200$ ms, while for the motion controller we have $T_s^m = 10$ ms. This means that, to allow the low level controller to run properly at the correct rate, the way-points provided by the planner must be interpolated at a higher rate. In this regard, while a simple staircase interpolation of the speed reference is adopted, such a procedure for the way-points leads to unsatisfactory performance. For the latter case, a Bézier polynomial fitting, see, e.g. [15], has been employed to obtain a smoother generation of the way points, see for details, [16]. The low level controller, i.e. the one that realizes the longitudinal and lateral motion controller is not described here. Basically, such a controller implements a speed tracking control through a PID regulator and a lane keeping strategy using Linear Quadratic methodologies according to the standard approaches described in [17] and [18].

V. BEHAVIORAL LOGIC

As introduced in Section IV, in highway driving scenarios, the vehicle must accomplish the following tasks.

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 possible.
3) Keep the center of the lane when travelling in a bend.
4) Perform overtaking of a slower vehicle and returning in the rightmost lane when overtaking is completed.

The *switching* between one task and another depends on the driving environment that is read by the on-board sensors. Thus, a switching logic is introduced to decide:

- whether to track the target speed or a distance with respect to a preceding vehicle speed;
- to keep the lane if no lateral maneuver is allowed;
- to overtake if a slower preceding vehicle is present and if left lane is free and to return if the right lanes are free.

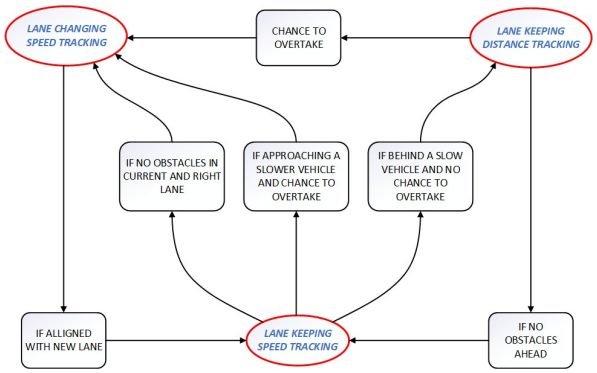

Fig. 5. State diagram of the planner.

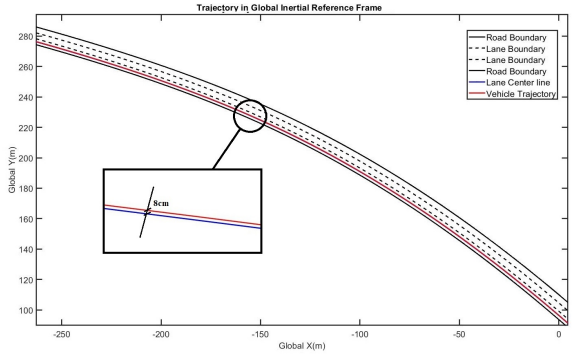

Fig. 7. Lane keeping performance on curved lane.

The finite-state machine (FSM) introduced in Fig. 5 is used to describe the combined behavior of lateral and longitudinal switching logic. Red ellipses represent the finite states of the planner, while black rectangles the conditions to pass from one to another. Quantitative switching rules are introduced to set the state transitions described in Fig. 5. For example, a lane change maneuver can be activated only if suitable safety distance exists between the autonomous vehicle and the surrounding obstacles. The distance d_i of the vehicle with respect to the i^{th} surrounding obstacle can be computed as

$$d_i = s_0 + t_H\, v_i \pm max(0, \mp t_H(v - v_i)) \tag{22}$$

where t_H is a fixed headway time, v is the autonomous vehicle speed, v_i is the corresponding obstacle speed and s_0 a constant safety distance contribution. The sign of the operation depends on whether the obstacle is located in the rear or the front lateral area. Thus, if such distances are greater than a given value, it means that there is free space to engage a safety lane change maneuver. Fig. 6 shows all the relevant distances considered performing an overtaking maneuver. Such spaces must be free before engaging any lateral maneuver. Another interesting case is the switching between *Lane Keeping / Speed Tracking* and *Lane Keeping / Distance Tracking*. Such a case occurs when a vehicle running at lower speed precedes the autonomous and there is no chance for overtaking due to the presence of vehicles in the left lane. This case can be handled as in standard Adaptive Cruise Control driving. In particular, the hysteresis logic (23) based on the relative distance d between the vehicles is employed to switch between the Speed Tracking (ST) and the Distance Tracking (DT) mode.

$$\text{mode} \rightarrow \begin{cases} \text{ST} \rightarrow \text{DT if } d < d_{\text{tar}} - d_{m1} \\ \text{DT} \rightarrow \text{ST if } d > d_{\text{tar}} + d_{m2} \end{cases} \tag{23}$$

where $d_{\text{tar}} = d_i$ and d_{m1}, d_{m2} are suitable thresholds. As the vehicle approaches an obstacle the working function, the FSM switches from ST to DT at a distance $d_{\text{tar}} - d_{m1}$. On the contrary, if the obstacle distance increases over $d_{\text{tar}} + d_{m2}$ threshold, the working function switches from DT to ST. Based on the selected driving task, the MPC controller formulation described in (19) must be set accordingly. In this regard, the contributions in the cost function (17) should be suitably chosen. In general, for each task described in the FSM of Fig. 5 there is the corresponding formulation of the MPC problem.

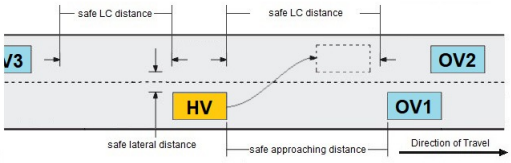

Fig. 6. Safe distances during lane changing maneuver.

VI. Simulation Results

In this Section, we introduce simulation results aimed at showing the effectiveness of the proposed approach. The methodology is tested on a 3 degrees of freedom nonlinear single-track dynamical model. The MPC optimization problem is solved through the MatLab function `fmincon`. With this strategy, the average computation time is about 65 ms with a maximum of 120 ms employing an AMD® FX-8350 CPU, 8 GB RAM running at 4 GHz. MatLab *Automated Driving Toolbox* is used to create the highway scenario and to perform the simulation tests. The driving scenario is made up by a three-lanes road with length 4 km, and it is divided into three sections introduced to test different driving situations. In the next subsections, we describe the settings and the obtained results for each of the considered driving settings.

A. Lane keeping and distance tracking

In the first section, lane keeping and distance tracking are tested in a bend according to the following setup.

1) The HV placed in the rightmost lane is driving at 100 km/h with both lateral and orientation errors, with respect to the center of the lane.
2) In the same lane, a vehicle is driving at a lower speed, about 70 km/h, and precedes HV of about 86 m.
3) A further vehicle is driving in the left lane with speed variable in the range $75-80$ km/h, preventing to overtake the preceding vehicle and, thus, HV is forced to perform a distance tracking maneuver.
4) The whole maneuver is performed in a bend with length of about 550 m with curvature radius [1] of 500 m.

While the lane keeping logic is working during all the simulation, the most challenging situation occurs during the first tract of the driving scenario that includes a bend with the maximum allowed curvature radius. As a consequence of the initial error, the trajectory reaches a satisfactory steady state condition after a settling phase. The trajectory during the curve is shown in Fig. 7 As it can be seen, the second-order Gaussian functions provide good results in terms of lateral precision. The most significant results for the distance keeping performance are reported in Fig. 8, where the HV

[1] The minimum radius of curvature allowed in Italian highways is 500 m

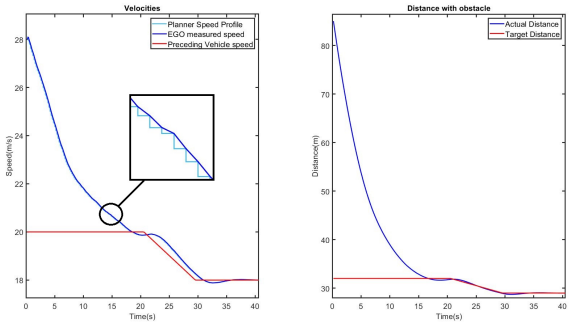

Fig. 8. Speed and Distance Plot of HV with respect to preceding vehicle.

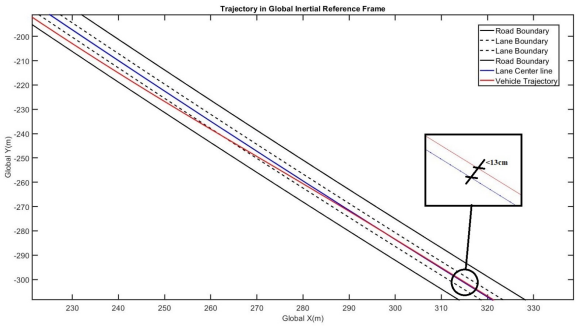

Fig. 9. First overtaking maneuver.

speed together with the distance from the preceding vehicle are reported. Between $t = 0.4$ s and $t = 16$ s, the HV undergoes the effect of the first obstacle field. As a result of the attractive field action, it decelerates until the actual speed reaches the preceding vehicle speed. From $t = 21$ s, the preceding vehicle starts decelerating before the HV speed is settled. However, due to the stiff effect of the second field, the vehicle is forced to keep the imposed distance. As the preceding vehicle speed settles to $v_i = 18$ m/s, the distance converges to a value of 29 m. The presented results show a quite good behavior of distance tracking performance.

B. Double Overtaking and Curve Following

In the second part, overtaking maneuvers are tested in the following setup.

1) The relative distance between HV and the middle lane vehicle is increased to allow the overtaking maneuver of the preceding vehicle.
2) After overtaking of the preceding vehicle, the middle lane vehicle is slower than HV and the leftmost lane is free.
3) HV performs overtaking of the middle lane preceding vehicle and moves to leftmost lane.

Thus, in this phase, HV performs a double overtaking maneuver. The double overtaking maneuver is made up of 4 stages:

1) Lane changing from lane 1 to lane 2
2) Lane keeping to stabilize the vehicle in lane 2
3) When aligned with lane 2, lane changing from lane 2 to lane 3
4) Lane keeping to stabilize the vehicle in lane 3

In Fig. 9, we show only the trajectory of the first overtaking maneuver, since the second one follows a similar path. From Fig. 9, we see that quite good performance are obtained in terms of smoothness and accuracy of the overtaking maneuver where the maximum deviation with respect to the new lane center is $\varepsilon_{err} < 13$ cm.

C. Returning maneuver

After the double overtaking maneuver, the driving scenario allows the HV to perform a returning maneuver to the rightmost lane. The maneuver is realized in two steps, i.e., the first from the leftmost lane to the middle lane and the second from the middle lane to the rightmost one. The path characteristics are similar as in the overtaking maneuvers described in the previous subsection.

VII. Conclusion

An approach for automated driving in highway scenarios in the context of a two layers hierarchical architecture has been introduced. Suitable APFs combinations that can effectively handle the most relevant maneuvers of highway driving, such as speed and distance tracking, lane keeping, overtaking, and returning have been defined. The defined APF are included in the cost function of an MPC control problem to generate the path trajectory. Parameters of the APF functions are dynamically tuned according to the acquired scenario. A behavioral logic, described by a FSM and based on acquired data and suitable dynamic conditions, is defined to select the most appropriate maneuver to realize. A quite complete simulation scenario has been realized to analyze the performances of the proposed method. The obtained results shows that the introduced approach achieves quite good performance in all the considered maneuvers.

References

[1] Waymo. (21) Autopilot ai. [Online]. Available: https://waymo.com/tech/

[2] Tesla. (21) Former google self-driving car project. [Online]. Available: https://www.tesla.com/autopilotAI

[3] Audi. (21) Audi autonomous driving. [Online]. Available: https://www.audi.com/en/experience-audi/mobility-and-trends/autonomous-driving.html

[4] S. international, "Surface vehicle recommended practice j3016," 2018.

[5] 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 *2018 IEEE 4th International Conference on Computer and Communications (ICCC)*, 2018, pp. 510–514.

[6] S. Sedighi, D.-V. Nguyen, and K.-D. Kuhnert, "Guided Hybrid A-star Path Planning Algorithm for Valet Parking Applications," in *2019 5th International Conference on Control, Automation and Robotics (ICCAR)*, 2019, pp. 570–575.

[7] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, 1985, pp. 500–505.

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

[9] N. Noto, H. Okuda, Y. Tazaki, and T. Suzuki, "Steering assisting system for obstacle avoidance based on personalized potential field," in *2012 15th International IEEE Conference on Intelligent Transportation Systems*, 2012, pp. 1702–1707.

[10] L. del Re, F. Allgöwer, L. Glielmo, C. Guardiola, I. Kolmanovsky, and Eds., *Automotive Model Predictive Control: Models, Methods and Applications*. Lecture Notes in Control and Information Sciences book series (LNCIS, volume 402), Springer, 2012.

[11] Y. Rasekhipour, A. Khajepour, S.-K. Chen, and B. Litkouhi, "A Potential Field-Based Model Predictive Path-Planning Controller for Autonomous Road Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1255–1267, 2017.

[12] 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 Transactions on Vehicular Technology*, vol. 66, no. 2, pp. 952–964, 2017.

[13] 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, 2016, nonlinear Dynamics and Complexity.

[14] L. Evans, "An introduction to mathematical optimal control theory version 0.2," 02 2013.

[15] K. Harada and E. Nakamae, "Application of the bézier curve to data interpolation," *Computer-Aided Design*, vol. 1, no. 1, pp. 55–59, 1982.

[16] A. E. Belvedere, *Artificial Potential Fields-based Predictive Control for Autonomous Vehicles in Highway Scenarios*. MSc Dissertation in Mechatronic Engineering, Politecnico di Torino, 2021.

[17] R. Rajamani, *Vehicle Dynamics and Control*. Springer, 2012.

[18] M. Canale, L. Fagiano, F. Ruiz, and M. Signorile, "A study on the use of virtual sensors in vehicle control," in *47th IEEE Conference on Decision and Control*, 2008, pp. 4402 – 4407.