HAL
open science

High-Speed Trajectory Planning for Autonomous Vehicles Using a Simple Dynamic Model

Florent Altché, Philip Polack, Arnaud de La Fortelle

To cite this version:

Florent Altché, Philip Polack, Arnaud de La Fortelle. High-Speed Trajectory Planning for Autonomous Vehicles Using a Simple Dynamic Model. IEEE 20th International Conference on Intelligent Transportation , Oct 2017, Yokohama, Japan. hal-01691788

HAL Id: hal-01691788

https://minesparis-psl.hal.science/hal-01691788v1

Submitted on 24 Jan 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

High-Speed Trajectory Planning for Autonomous Vehicles Using a Simple Dynamic Model

Florent Altché[2,1], Philip Polack[1] and Arnaud de La Fortelle[1]

***Abstract*— In this article, we propose a new approach to drive a vehicle at high speed along a predetermined path using Model Predictive Control. Instead of modeling precise vehicle dynamics, which strongly restricts the planning horizon that can be considered in real-time, we use a simplified second-order integrator model which is constrained to match the vehicle's feasible dynamic envelope. Moreover, and contrary to many MPC approaches taking a target speed as input, our formulation also includes velocity planning, making the vehicle able to automatically adjust its speed to the layout of the road. Simulation results on a highly precise vehicle model show that our approach can be used in real-time to provide feasible trajectories that can be tracked using a simple control architecture. Moreover, the use of our simplified model makes the planner more robust and yields better trajectories compared to kinematic models commonly used in trajectory planning.**

I. Introduction

In order to improve safety and energy efficiency and not to surprise other road users, autonomous vehicles are widely expected to drive smoothly; moreover, traffic efficiency concerns will likely lead these vehicles to follow relatively closely the road speed limit. However, such speed limits may not always exist (*e.g.*, on some dirt roads) or they may not be followed safely due to the road topology (*e.g.*, mountain road with sharp curves) or weather conditions (*e.g.*, icy road).

Many trajectory planning algorithms rely on an *a-priori* knowledge of a target velocity that can either be an explicit parameter of the problem [1], [2] or be implicitly given by requiring a set of target positions at fixed times [3], [4]. Some authors have proposed using the road curvature [5] to provide an upper bound on the velocity, which corresponds to a maximum lateral acceleration; this method can be extended to obstacle avoidance by first planning a collision-free path, then adjusting the velocity consequently [6]. However, as path selection and velocity planning are intrinsically linked, this approach can lead to severe inefficiencies.

For this reason, model predictive control (MPC) techniques are often used in the literature, since they allow to simultaneously compute a feasible trajectory and a sequence of control inputs to track it. However, high-speed trajectory planning requires a complex modeling of the vehicle to account for its dynamic limitations, which mainly come from the complex and highly non-linear [7] tire-road interactions. Adding to this difficulty, wheel dynamics are generally much faster (around $1\,\mathrm{ms}$ [8]) than changes of the vehicle's macroscopic state (typically $100\,\mathrm{ms}$). Therefore, the existing literature is generally divided between medium-term (a few seconds) trajectory planning including obstacle avoidance for low-speed applications, mainly relying on simple kinematic models (see, *e.g.* [9], [10]), and short-term (sub-second) trajectory tracking for high-speed or low-adherence applications using wheel dynamics modeling (see, *e.g.*, [11]–[14]). In the second case, obstacle avoidance is generally not considered (with the notable exception of [15]), and the existence of a feasible collision-free trajectory is not guaranteed in the case of an unexpected obstacle. Note that sampling-based approaches [16] have also been proposed in the literature; however, they do not provide optimality guarantees when a finite number of samples is selected, and may have trouble finding a feasible solution in complex scenarios.

In this article, we propose a middle-ground approach to allow trajectory planning over a few seconds in high dynamic situations. This approach relies on the use a simple vehicle model (initially proposed in [17]) derived from a realistic dynamic modeling of the vehicle body and wheels, which accounts for tire slip effects using carefully estimated bounds on the dynamics. Using this model, we design a non-linear trajectory planning framework, which is able to follow a predefined path (*e.g.*, the road centerline) at high speed while avoiding obstacles. Comparison with a more classical kinematic bicycle model shows that our proposed planner provides better trajectories and is more robust. Moreover, our simulations showed that computation time remains below $80\,\mathrm{ms}$ on a standard computer and could be further reduced using auto-generation [4], thus allowing real-time use. To the best of the authors' knowledge, no comparable framework has been published for medium- or long-term planning in high dynamic situations.

The rest of this article is structured as follows: in Section II, we present a 9 degrees of freedom dynamic model of the vehicle's body, that we use throughout the rest of this article. In Section IV, we briefly describe our simplified dynamic model and to formulate a trajectory planner based on Model Predictive Control, which computes an aggressive yet feasible trajectory. This planner is validated and its performance is compared to a classical kinematic bicycle model in Section V using a realistic physics simulation suite; finally, Section VI concludes the study.

II. Vehicle model

In this article, we consider a 9 degrees of freedom modeling of the vehicle body, which provides a satisfying balance

[1] MINES ParisTech, PSL Research University, Centre for robotics, 60 Bd St Michel 75006 Paris, France [florent.altche, philip.polack, arnaud.de_la_fortelle] @mines-paristech.fr

[2] École des Ponts ParisTech, Cité Descartes, 6-8 Av Blaise Pascal, 77455 Champs-sur-Marne, France

between accuracy and computational cost. Alongside with the usual 2D state $[X, Y, \psi]$ (with ψ the yaw rotation) of the vehicle, the model takes into account roll and pitch movements, wheel dynamics and coupling of longitudinal and lateral tire slips. Being a chassis model, it does not take into account the dynamics of the car engine or brakes. The control inputs of the vehicle are the torque T_i applied to each wheel i and the steering angle of the front wheels, δ. We use uppercase letters (*e.g.*, X, Y) to denote coordinates in the ground (global) frame, and lowercase letters for coordinates in the vehicle (local) frame; the x coordinate in the local frame corresponds to the longitudinal component. The notations are given in Table I and illustrated in Figure 1.

TABLE I
NOTATIONS

X, Y, Z	Position of the vehicle's CoM (ground frame)
θ, ϕ, ψ	Roll, pitch and yaw angles of the car body
V_x, V_y	Longitudinal and lat. vehicle speed (vehicle frame)
V_{xw_i}	Longitudinal speed of wheel i (wheel frame)
ω_i	Angular velocity of wheel i
ζ_i	Displacement of suspension i
δ	Steering angle of the front wheels
T_{ω_i}	Total torque applied to wheel i
F_{xw_i}, F_{yw_i}	Longitudinal and lateral forces on wheel i (wheel frame)
F_{x_i}, F_{y_i}	Longitudinal and lat. forces on wheel i (vehicle frame)
F_{z_i}	Normal ground force on wheel i
F_{aero}	Air drag force on the vehicle
M_T	Total mass of the vehicle
I_x, I_y, I_z	Roll, pitch and yaw inertia of the vehicle
I_{r_i}	Inertia of wheel i around its axis
l_f, l_r	Distance between the front/rear axle and the CoM
l_w	Half-track of the vehicle
r_w	Effective radius of the wheels
k_s, d_s	Suspensions stiffness and damping

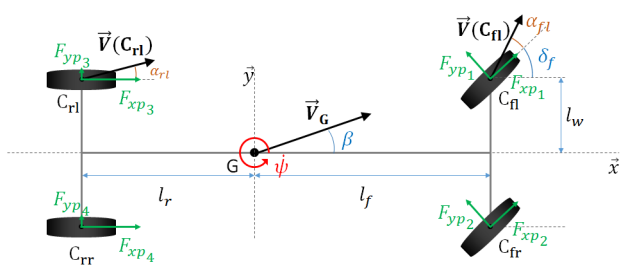


Fig. 1. Simulation model of the vehicle in the (x, y) plane

We make the assumptions that the body of the vehicle rotates around its center of mass, and that the aerodynamic forces do not create a moment on the vehicle. Moreover, we assume that the road remains horizontal, and any slope or banking angle is neglected; this assumption could be relaxed using a slightly more complex vehicle model. Under these hypotheses, the dynamics of the vehicle's center of mass are written as:

$$\dot{X} = V_x \cos\psi - V_y \sin\psi \tag{1a}$$

$$\dot{Y} = V_x \sin\psi + V_y \cos\psi \tag{1b}$$

$$\dot{V}_x = \dot{\psi} V_y + \frac{1}{M_T} \sum_{i=1}^{4} F_{x_i} - F_{aero} \tag{1c}$$

$$\dot{V}_y = -\dot{\psi} V_x + \frac{1}{M_T} \sum_{i=1}^{4} F_{y_i}, \tag{1d}$$

where F_{x_i} and F_{y_i} are respectively the longitudinal and lateral tire forces generated on wheel i, expressed in the local vehicle frame (x, y). The yaw, roll and pitch motions of the car body are computed as:

$$I_z \ddot{\psi} = l_f(F_{y_1} + F_{y_2}) - l_r(F_{y_3} + F_{y_4}) + l_w(F_{x_2} + F_{x_4} - F_{x_1} - F_{x_3}) \tag{2a}$$

$$I_x \ddot{\theta} = l_w(F_{z_1} + F_{z_3} - F_{z_2} - F_{z_4}) + Z \sum_{i=1}^{4} F_{y_i} \tag{2b}$$

$$I_y \ddot{\phi} = l_r(F_{z_3} + F_{z_4}) - l_f(F_{z_1} + F_{z_2}) - Z \sum_{i=1}^{4} F_{x_i} \tag{2c}$$

where $F_{z_i} = -k_s \zeta_i(\theta, \phi) - d_s(\dot{\zeta}_i)(\theta, \phi)$, with $\zeta_i(\theta, \phi)$ the displacement of suspension i for the given roll and pitch angles of the car body. The variation of F_z models the impact of load transfer between tires. Finally, the dynamics of each wheel i can be written as

$$I_r \dot{\omega}_i = T_{\omega_i} - r_w F_{xw_i}. \tag{3}$$

In general, the longitudinal and lateral forces F_{xw_i} and F_{yw_i} depend on the longitudinal slip ratio τ_i, the side-slip angle α_i, the reactive normal force F_{z_i} and the road friction coefficient μ. The slip ratio of wheel i can be computed as

$$\tau_i = \begin{cases} \frac{r_w \omega_i - V_{xw_i}}{r_w \omega_i} & \text{if } r_w \omega_i \geq V_{xw_i} \\ \frac{r_w \omega_i - V_{xw_i}}{V_{xw_i}} & \text{otherwise.} \end{cases} \tag{4}$$

The lateral slip-angle α_i of tire i is the angle between the wheel's orientation and its velocity, and can be expressed as

$$\alpha_f = \delta - \frac{V_y + l_f \dot{\psi}}{V_x \pm l_w \dot{\psi}} \tag{5}$$

$$\alpha_r = -\frac{V_y - l_r \dot{\psi}}{V_x \pm l_w \dot{\psi}} \tag{6}$$

where f and r denote the front and rear wheels respectively.

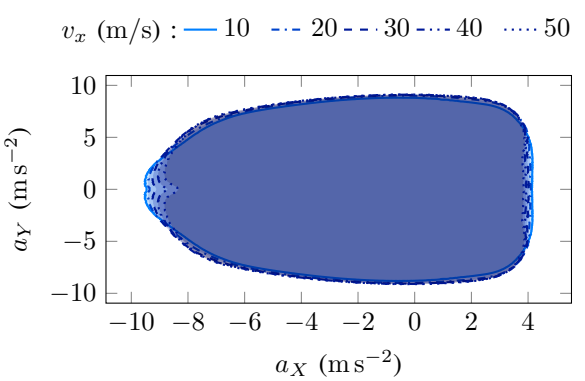
In this article, we use Pacejka's combined slip tire model (equations (4.E1) to (4.E67) in [7]), which takes into account the interaction between longitudinal and lateral slips, thus encompassing the notion of friction circle [18]. For clarity purposes, we do not reproduce the complete set of equations.

III. Constrained second-order integrator model

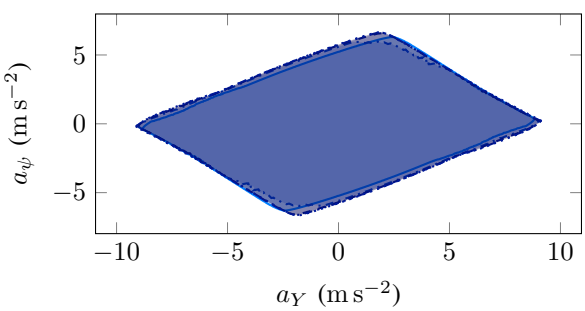
Theoretically, it is possible to use the 9 degrees of freedom model inside a model predictive control framework to directly compute an optimal trajectory and the corresponding controls. However, classical optimization tools often struggle

to solve problems involving highly nonlinear constraints or cost functions, as it is the case in the model presented in Section II – notably due to disjunction (4) which makes τ non-differentiable. Additionally, wheel dynamics generally occur over very small characteristic times – typically a few milliseconds – which requires choosing a correspondingly small discretization time step, making planning over long horizons impractical at best. For this reason, simplified models are very often preferred. Kinematic bicycle (or single-track) models [9], [10], [19], or even simpler second-order integrator models [2] are therefore common in the trajectory planning literature.

One of the main issues of these simplified models is that they are generally considered to be imprecise when nearing the handling limits of the vehicle. To counter this problem, previous work [17] presented a constrained second-order dynamic model derived from simulation data using the 9 degrees of freedom model of Section II. For lack of space, we only briefly described these previous results here.



(a) Lateral vs. longitudinal



(b) Yaw vs. lateral

Fig. 2. Envelope of the sets of feasible accelerations for different longitudinal velocities $v_{x,0}$ and lateral velocity $v_{y,0} \in [-0.2v_{x,0}, 0.2v_{x,0}]$; notice the slight deformation along the a_X axis with increasing $v_{x,0}$..

Using a random sampling method, we first compute (offline) an envelope for the set of feasible longitudinal (a_X), lateral (a_Y) and angular (a_ψ) accelerations, as shown in Figure 2. More specifically, these sets are computed for a given longitudinal velocity $v_{x,0}$ and a lateral velocity $v_{y,0} \in [-0.2v_{x,0}, 0.2v_{x,0}]$. Therefore, only part of this region is actually reachable from an initial vehicle state (corresponding to a determined lateral velocity). In this article, we assume that the lateral dynamics of the vehicle is sufficiently fast to neglect this effect. Simulation results (see Section V) show this assumption seems reasonable.

Using these results, we propose a constrained double integrator model for the vehicle dynamics. This model considers a state vector $\xi = [X, Y, \psi, v_x, v_y, v_\psi]^T$ and a control $\mathbf{u} = [u_x, u_y, u_\psi]^T$, with the same notations and reference frames as presented in Section II. The dynamic equation of the system is $\dot{\xi} = f_{2di}(\xi, \mathbf{u})$ with

$$f_{2di}(\xi, \mathbf{u}) = \begin{bmatrix} v_x \cos\psi - v_y \sin\psi \\ v_x \sin\psi + v_y \cos\psi \\ [v_\psi, u_x, u_y, u_\psi]^T \end{bmatrix}. \tag{7}$$

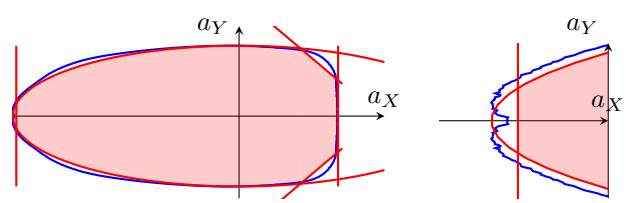
To allow the use of this model in planning, we approximate the sets shown in Figure 2 as a set of convex linear and nonlinear constraints in the (a_X, a_Y) plane as shown in Figure 3. These constraints are expressed as:

$$\left(\frac{a_X}{\alpha}\right)^2 + \left(\frac{a_Y}{\beta}\right)^2 \leq 1 \tag{8}$$

$$a_X^{min}(v_{x,0}) \leq a_X \leq a_X^{max}(v_{x,0}) \tag{9}$$

$$A[a_X, a_Y, a_\psi]^T \leq b \tag{10}$$

where A is a constant matrix, b a constant vector and a_X^{min}, a_Y^{min} depend on $v_{x,0}$. For our model vehicle, the experimental data of Figure 2 yield $\alpha = 9.4\,\mathrm{m\,s^{-2}}$, $\beta = 9.0\,\mathrm{m\,s^{-2}}$, $A = \left(\begin{smallmatrix} 2.6 & 1 \\ 2.6 & -1 \end{smallmatrix}\right)$ and $b = \left(\begin{smallmatrix} 15.3 \\ 15.3 \end{smallmatrix}\right)\,\mathrm{m\,s^{-2}}$. Figure 4 shows the variation of a_X^{min} and a_X^{max} with the initial longitudinal velocity; a polynomial fit yields $a_X^{min}(v_{x,0}) = -9.3 - 0.013v_{x,0} + 0.00072{v_{x,0}}^2$ and $a_X^{max}(v_{x,0}) = 4.3 - 0.009v_{x,0}$ (with $v_{x,0}$ expressed in $\mathrm{m\,s^{-1}}$). In the (a_Y, a_ψ) plane, we suppose a linear relation between a_ψ and a_Y in the form: $a_\psi = \gamma a_Y$, with $\gamma = 0.56\,\mathrm{rad\,m^{-1}}$. Although seemingly restrictive, this approximation provides better results than using a parallelogram acceptable region, and in practice corresponds to minimizing the vehicle slip angle.



(a) Feasible set (blue) and approximation (red) (b) Detail

Fig. 3. Convex approximation of the feasible accelerations in the (a_X, a_Y) plane. The actual region is shown in blue.

IV. MPC FORMULATION FOR TRAJECTORY PLANNING

We now use the second-order integrator model developed in the previous section to design a trajectory planner based on model predictive control. In this section, we assume that the vehicle tries to follow a known reference path, for instance the centerline of a given lane. The reference path γ_{ref} is supposed to be given as a set of positions (X_{ref}, Y_{ref}); as obstacle avoidance is part of the MPC formulation, these positions can simply be given through, *e.g.*, high-definition cartography.

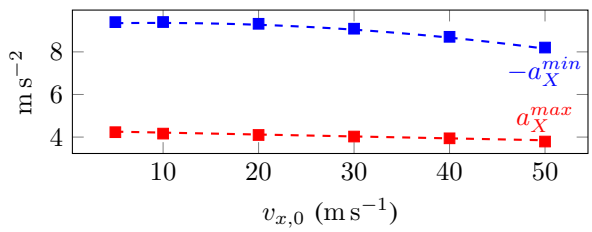


Fig. 4. Variations of the a_X^{min} and a_X^{max} coefficients with the initial longitudinal velocity $v_{x,0}$; the dashed lines show the polynomial fit.

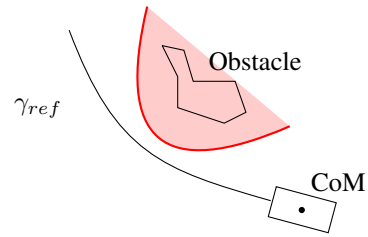


Fig. 5. Modeling of an obstacle a parabola. The vertex and roots of the parabola are chosen with enough margin to ensure that no collision can occur as long as the center of mass is outside of the shaded region containing the obstacle.

In this article, we assume that the vehicle has no information on a safe choice of longitudinal speed. Such situations can arise in various scenarios, such as poor adherence conditions where the speed limit cannot be safely followed, or in the absence of speed limits, for instance on dirt roads or for racing. Moreover, we assume that the vehicle also needs to avoid obstacles on the road; for now, we only consider fixed obstacles with known positions and shapes. Additionally, we do not consider varying road adherence, and we suppose that the tire-road friction coefficient μ is constant and equal for all four wheels. Future work will study real-time estimation of this friction coefficient to adapted the planner accordingly.

Algorithm 1: Planning and control

Data: current state $\xi(t_0)$, γ_{ref}, horizon T
find $x_{closest}$:= point of γ_{ref} closest to $(X(t_0), Y(t_0))$
set s_0 := curvilinear position of $x_{closest}$
set hz := $[s_0, s_0 + v_x(t_0)T]$
set p_X := fitpolynom($X_{ref}|_{hz}$, $s - s_0$, 5)
set p_Y := fitpolynom($Y_{ref}|_{hz}$, $s - s_0$, 5)
find obs := list of relevant obstacles
find κ := max(abs(curvature(p_X, p_Y, hz)))
set v_{max} := min $\left(v_x(t_0) + a_X(B)T, \sqrt{\mu g/\kappa}\right)$
compute ξ_{mpc} := MPC($\xi(t_0)$, p_X, p_Y, v_{max}, T, obs)
apply low level control to track ξ_{mpc}

The planning and control scheme works in several steps, as presented in Algorithm 1. We first approximate X_{ref} and Y_{ref} over the next planning horizon T as fifth order polynomials of the curvilinear position, starting at the point of γ_{ref} closest to the vehicle's current position. Using these polynomials, we compute the maximum (in absolute value) of the path curvature over the planning horizon, noted κ, to determine an upper bound $v_{max} = \sqrt{\mu g/\kappa}$ for the speed of the vehicle in order to limit the lateral acceleration to μg (as proposed, *e.g.*, in [5]). Only the relevant obstacles, *i.e.* those for which a risk of collision exists during the next planning horizon T, are effectively considered for collision avoidance; we note $\mathcal{O}$ the set of these obstacles. Generalizing the ideas of [20], we determine a bounding parabola for each obstacle $o \in \mathcal{O}$ as shown in Figure 5, such that the collision avoidance constraints can be written as $p_o(X, Y) \leq 0$, with p_o a second-order multinomial function. Note that the use of unbounded parabolas is preferred to bounded shapes such as ellipses, since they do not create local minima.

At a time t_0 corresponding to a vehicle state $\xi(t_0)$ (with an initial longitudinal speed v_0), we formulate the motion planning problem using model predictive control with time step duration h and horizon $T = Kh$ as follows:

$$\min_{(\mathbf{u}_k)_{k=0\ldots K-1}} \quad J\big((\xi_k)_{k=0\ldots K}, (\mathbf{u}_k)_{k=0\ldots K-1}\big) \tag{11a}$$
$$\text{subj. to} \quad \xi_{k+1} - \xi_k = h\, f_{2di}(\xi_k, \mathbf{u}_k) \tag{11b}$$
$$\xi_0 = \xi(t_0) \tag{11c}$$
$$\left(\frac{u_X}{\alpha}\right)^2 + \left(\frac{u_Y}{\beta}\right)^2 \leq 1 \tag{11d}$$
$$a_X^{min}(v_{x,0}) \leq a_X \leq a_X^{max}(v_{x,0}) \tag{11e}$$
$$A[u_X, u_Y]^T \leq b \tag{11f}$$
$$u_\psi = \gamma u_Y \tag{11g}$$
$$\forall o \in \mathcal{O}, \quad p_o(X_k, Y_k) \leq 0 \tag{11h}$$
$$\text{for} \quad k = 0 \ldots K-1.$$

Note that, as it is often the case in the planning literature (see, *e.g.*, [21]), collision avoidance (eq. (11h)) is actually implemented as soft constraints to avoid infeasibility caused by numerical errors. Additionally, note that our formulation can be slightly modified in order to take moving obstacles into account, by using a different parabola p_o^k for each obstacle and at each time step.

In this article, we only focus on minimizing the deviation from the reference trajectory. In most of the existing MPC literature where a reference speed is supposed to be known in advance, the cost function J is expressed as:

$$J\big((\xi_k), (\mathbf{u}_k)\big) = \sum_{k=0}^{K} (X_k - X_k^{ref})^2 + (Y_k - Y_k^{ref})^2. \tag{12}$$

In these formulations, X_k^{ref} and Y_k^{ref} implicitly encode the speed at which the reference path should be followed. Since we do not assume that a reference speed is known in advance, this method cannot be applied directly. A possible way to handle this difficulty is to express Y_k^{ref} as a function of X_k (see, *e.g*, [22]). However, this method cannot be applied to all shapes of reference paths, and is notably not suited to sharp turns even when using local instead of global coordinates. For this reason, we introduce an auxiliary state s to denote the curvilinear position of the vehicle along γ_{ref}, so that $X_k^{ref} = p_X(s_k)$ and $Y_k^{ref} = p_Y(s_k)$. In this article, we

use a simple first-order integrator dynamic for s with $\dot{s} = \sqrt{v_x^2 + v_y^2}$. Noting ξ' the extended state of the vehicle, we instead use the objective function:

$$J = \sum_{k=0}^{K-1} w_v {v_{tol}}_k^2 + w_X {X_{tol}}_k^2 + w_Y {Y_{tol}}_k^2 + w_o {O_{tol}}_k^2 \quad (13)$$

where w_v, w_X, w_Y and w_o are positive weighting terms, and we add the following constraints to problem (11):

$$\begin{aligned} v_{tol\,k} &\geq |v_{max} - v_{X\,k}| && (14a)\\ X_{tol\,k} &\geq |X_k - p_X(s_k)| && (14b)\\ Y_{tol\,k} &\geq |Y_k - p_Y(s_k)| && (14c)\\ \forall o \in \mathcal{O},\quad O_{tol\,k} &\geq p_o(X_k, Y_k) && (14d)\\ &\text{for } k = 0 \ldots K-1. \end{aligned}$$

V. Simulation Results

We used the realistic physics simulator PreScan [23] to validate the proposed MPC trajectory planner. The simulator uses the 9 degrees of freedom model presented in Section II, with the same parameters that were used to obtain the sets of feasible accelerations in Section III. Robustness of the planner to variations of the vehicle parameters is a subject for further study.

The MPC problem (11)-(14) is solved using the ACADO Toolkit [24]; due to the inner workings of the simulator, the simulation is paused during resolution. The output of the solver is the set of future target longitudinal and lateral accelerations, as well as target future positions and longitudinal velocities for the vehicle in the horizon T. These outputs are fed into two low-level controllers, one being tasked with velocity tracking and the other with steering.

Low-level tracking of the planned trajectory is achieved using PID controllers; the lateral control also uses a τ seconds look-ahead [25]. At time t, the predicted position of the vehicle at $t+\tau$ is computed as $\hat{X} = X + \tau v_x \cos(\hat{\psi})$ and $\hat{Y} = Y + \tau v_x \cos(\hat{\psi})$, with $\hat{\psi} = \psi + \frac{1}{2}\tau\dot{\psi}$. Instead of tracking the target position at time t, the lateral control uses the error between predicted and desired positions at time $t+\tau$. Using a look-ahead $\tau = 0.2\,\mathrm{s}$, this method was found to provide better performance and stability than a simple PID. The lateral control takes into account a limited angular velocity for the steering wheel of $12\,\mathrm{rad\,s^{-1}}$, which is in the average of recorded steering velocities for human drivers in obstacle avoidance scenarios [26]; we do not consider the dynamics of the engine or brakes, which are supposed to respond instantaneously. Note that, although the PID approach gives overall satisfying performance, we did observe certain rare occurrences of over-correction sending the vehicle into a spin; therefore, more robust control schemes accounting for the tire's friction circle should be explored.

The reference path used in our simulations is presented in Figure 6; the path consists of a $60\,\mathrm{m}$ straight line, a half circle with radius $20\,\mathrm{m}$, a $200\,\mathrm{m}$ straight line, a Bezier arc corresponding to a 135 degrees turn, a $100\,\mathrm{m}$ straight line, a half circle with radius $10\,\mathrm{m}$ and a final -135 degrees turn.

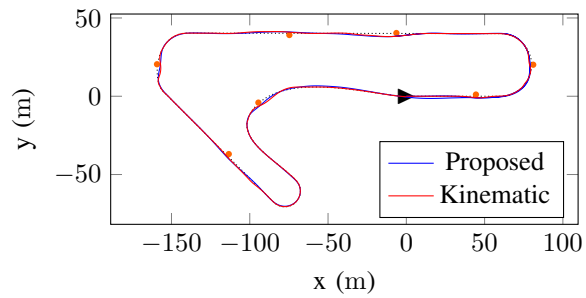


Fig. 6. Detail of the reference path (black, dotted) and actual path followed by the vehicle while avoiding obstacles (represented as orange circles), using both planners.

In all simulations, the weights are chosen as $w_v = 1$, $w_X = w_Y = 10$ and $w_o = 100$, which were found experimentally to provide a good trade-off between speed and precision. The planning horizon is chosen as $T = 3\,\mathrm{s}$ and the time step duration of the MPC is $h = 200\,\mathrm{ms}$; replanning is performed every $100\,\mathrm{ms}$. To achieve real-time computation speeds, the solver is limited to five SQP iterations, which was experimentally shown to be sufficient for a good convergence on our particular problem. Auto-generation techniques can also be used to further reduce computation time [4].

For comparison purposes, we also implemented the same MPC planner with a classical kinematic bicycle model such as presented in [19]; the model is written as follows:

$$\begin{aligned} \dot{X} &= v\cos(\psi + \arctan\beta) && (15a)\\ \dot{Y} &= v\sin(\psi + \arctan\beta) && (15b)\\ \dot{\psi} &= \frac{v}{l_r}\sin(\arctan\beta) && (15c)\\ \dot{v} &= a && (15d) \end{aligned}$$

with v the longitudinal velocity and $\beta = \frac{l_r}{l_r + l_f}\tan\delta$ the side slip angle. The control inputs are the longitudinal acceleration a, and the steering angle of the front wheel δ.

All other parts of the planning and control algorithm are otherwise equal, including the low-level controller. Moreover, this model is only use inside the MPC planner, while the simulation relies on that of Section II. The maximum and minimum acceleration in the bicycle model are chosen equal to a_X^{min} and a_X^{max} (see Figure 3 for the notations) respectively. Note that the solver is slightly faster using this formulation; therefore, the maximum number of SQP iterations is set to 6 for the kinematic model to yield comparable computation times, which increases solution quality.

A. Planning without obstacles

We first consider trajectory planning without obstacles; Figure 7 presents the actual speed of the vehicle during the simulation as a function of its position along the path γ_{ref} for the two MPC planners, as well as the speed bound $v^2 \leq \kappa g$ (with κ the path curvature), corresponding to a centripetal acceleration of $1\,g$ used, *e.g.*, in [5]. First, we notice that both planners have similar performance in the straight portions of the road; however, the planner based on the proposed

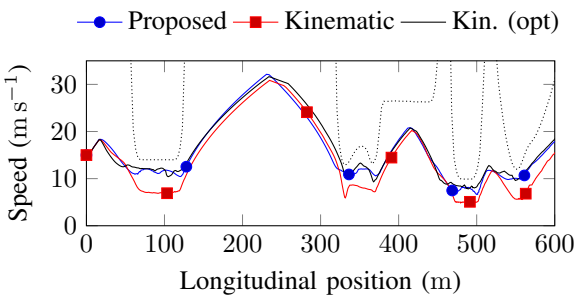


Fig. 7. Comparison of achieved vehicle speed with our proposed model (in blue) and for a kinematic bicycle model (in red); the speed for the kinematic model run until convergence is shown in solid black. The dotted curve represents the bound $v^2 \leq \mu g/\kappa$ where κ is the path curvature.

TABLE II

ABSOLUTE LATERAL POSITIONING ERROR FOR BOTH PLANNERS.

Model	RMS error (m)	Maximum error (m)
Proposed	0.25	0.70
Kinematic	0.16	0.95

second-order model systematically achieves higher speeds in curves. Second, no solver reaches the upper bound $\sqrt{g/\kappa}$, thus confirming that including a speed selection phase during planning (as opposed to tracking a predefined velocity solely computed from path curvature) is useful for proper tracking.

The superior performance of the planner based on the second-order integrator model is likely due to the simpler relation between the optimization variables (the input controls) and the objective value (the deviation from the target state) than in the bicycle model, which allows a much faster convergence towards the solution. Indeed, when allowing the solver to run until convergence with the kinematic bicycle model, the resulting velocity becomes comparable to that obtained with the real-time second-order model (but computation time is above $500\,\mathrm{ms}$).

Figure 8 shows the lateral error when tracking the reference path, for both MPC planners. Table II presents synthetic data about the lateral error of the complete planning and control architecture in both cases, showing satisfying overall performance for high-speed applications. Note that better precision can be achieved (at the cost of speed) by selecting different values for the weighting coefficients. Moreover, a more precise low-level controller can probably achieve better performance.

B. Planning with obstacle avoidance

In this section, we compare the behavior of both planners in the presence of obstacles, modeled as parabolas as explained in Figure 5. Figure 6 shows a detail of the actual path followed by the vehicle using the proposed planner while avoiding obstacles. The attached video file[1] shows the corresponding simulation.

Figure 9 presents the lateral deviation from the reference trajectory while avoiding obstacles using both planners.

[1]Also available at `https://youtu.be/BRpmdIxTz-0`

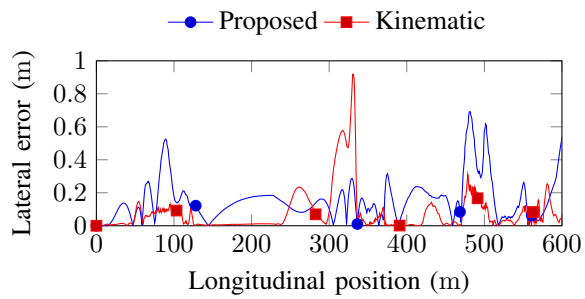


Fig. 8. Lateral positioning error for the proposed model (in blue) and for the kinematic bicycle model (in red), without obstacles.

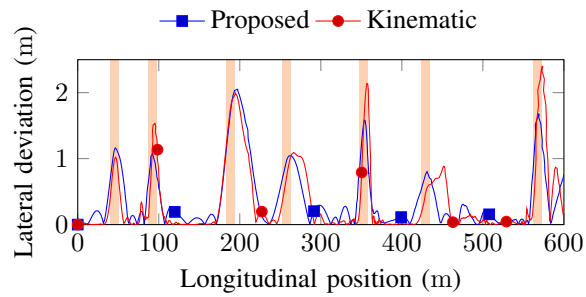


Fig. 9. Lateral deviation for the proposed model (in blue) and for the kinematic bicycle model (in red) while avoiding obstacles (shaded regions).

Generally speaking, the proposed planner allows a smaller deviation from the reference and a higher average speed of $12.7\,\mathrm{m\,s^{-1}}$ compared to $10.2\,\mathrm{m\,s^{-1}}$ using the kinematic bicycle planner (for lack of space, the velocity curves are not shown). More importantly, the kinematic planner is sometimes unable to output a trajectory in less than $100\,\mathrm{ms}$, as shown in Figure 10. This situation happens when approaching obstacles in high-curvature portions of the road; as before, the better behavior of the proposed planner is likely due to the simpler search space since the dynamic model presents much less non-linearity. A more in-depth analysis could provide useful insights on desirable model properties to allow fast and robust convergence.

VI. Conclusion

In this article, we considered the trajectory planning and control of a vehicle at high velocity near the limits of

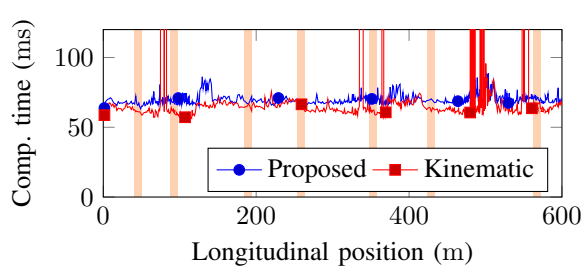


Fig. 10. MPC computation time (with obstacles) for the proposed model, and for a kinematic bicycle model starting from the same state.

handling. Instead of using a highly complex model during online resolution, as it is often the case in the literature, we used a simpler vehicle model obtained from precise dynamic simulations. This model was implemented into a novel MPC-based trajectory planner; contrary to most existing algorithms, our formulation does not need a predefined target speed. Instead, the MPC adjusts the vehicle's speed in real time to track a predefined path (such as the centerline of a road) as fast as possible while maintaining a low tracking error. Using the high fidelity physics simulator PreScan, we demonstrated that the combination of this planner with a PID-based low-level controller is capable of driving along a demanding path at high speed with a low lateral error, although controller robustness could be improved. Moreover, comparison with a simple and widely used bicycle model shows that the use of our simpler model allows the solver to converge faster towards a better quality solution under real-time constraints.

Although this work remains mainly theoretical, it opens several perspectives for future research. First, the good performance of our simple dynamic model even at high speeds allows envisioning longer planning horizons without sacrificing computation speed. Future work should also study the consistency of using fully linear models, that can be coupled with efficient mixed-integer optimization techniques to allow optimal decision-making, for instance for overtaking or lane-change decisions [2]. Second, we believe that the ability of the proposed MPC formulation to automatically adapt the vehicle's velocity can find practical applications, for instance when driving in low-adherence situations. Therefore, the behavior of the planner with a variable friction coefficient should be studied further. A more precise low-level controller, taking into account the very particular response curve of the tires to precisely track a target force, which would be more consistent with the second order integrator model, can also prove interesting. Finally, implementation of the planner and controller on an actual (scale model) vehicle is a necessary step to validate the suitability of the proposed planner for real-world use.

References

[1] R. Benenson, S. Petti, T. Fraichard, and M. Parent, "Towards urban driverless vehicles," *International Journal of Vehicle Autonomous Systems*, vol. 6, no. 1/2, p. 4, 2008.

[2] X. Qian, F. Altché, P. Bender, C. Stiller, and A. de La Fortelle, "Optimal trajectory planning for autonomous driving integrating logical constraints: An MIQP perspective," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, nov 2016, pp. 205–210.

[3] P. Falcone, F. Borrelli, H. E. Tseng, J. Asgari, and D. Hrovat, "A hierarchical Model Predictive Control framework for autonomous ground vehicles," *2008 American Control Conference*, pp. 3719–3724, 2008.

[4] J. Frasch, A. Gray, M. Zanon, H. Ferreau, S. Sager, F. Borrelli, and M. Diehl, "An auto-generated nonlinear MPC algorithm for real-time obstacle avoidance of ground vehicles," *European Control Conference (ECC), 2013*, pp. 4136–4141, 2013.

[5] Y. Koh, K. Yi, H. Her, and K. Kim, "A speed control race driver model with on-line driving trajectory planning," in *The Dynamics of Vehicles on Roads and Tracks: Proceedings of the 24th Symposium of the International Association for Vehicle System Dynamics (IAVSD 2015), Graz, Austria, 17-21 August 2015*. CRC Press, 2016, p. 67.

[6] D. Kim, J. Kang, and K. Yi, "Control strategy for high-speed autonomous driving in structured road," in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, oct 2011, pp. 186–191.

[7] H. Pacejka, *Tire and vehicle dynamics*. Elsevier, 2005.

[8] C. V. Altrock, "Fuzzy logic technologies in automotive engineering," in *WESCON/94. Idea/Microelectronics. Conference Record*, Sep 1994, pp. 110–117.

[9] M. A. Abbas, R. Milman, and J. M. Eklund, "Obstacle avoidance in real time with Nonlinear Model Predictive Control of autonomous vehicles," in *2014 IEEE 27th Canadian Conference on Electrical and Computer Engineering (CCECE)*. IEEE, may 2014, pp. 1–6.

[10] V. Cardoso, J. Oliveira, T. Teixeira, C. Badue, F. Mutz, T. Oliveira-Santos, L. Veronese, and A. F. De Souza, "A Model-Predictive Motion Planner for the IARA Autonomous Car," *arXiv preprint arXiv:1611.04552*, nov 2016.

[11] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive Active Steering Control for Autonomous Vehicle Systems," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 3, pp. 566–580, may 2007.

[12] J.-M. Park, D.-W. Kim, Y.-S. Yoon, H. J. Kim, and K.-S. Yi, "Obstacle avoidance of autonomous vehicles based on model predictive control," *Proc. of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 223, no. 12, pp. 1499–1516, 2009.

[13] Y. Gao, T. Lin, F. Borrelli, E. Tseng, and D. Hrovat, "Predictive Control of Autonomous Ground Vehicles With Obstacle Avoidance on Slippery Roads," in *ASME 2010 Dynamic Systems and Control Conference, Volume 1*. ASME, 2010, pp. 265–272.

[14] A. Liniger, A. Domahidi, and M. Morari, "Optimization-based autonomous racing of 1: 43 scale rc cars," *Optimal Control Applications and Methods*, vol. 36, no. 5, pp. 628–647, 2015.

[15] J. Liu, P. Jayakumar, J. L. Stein, and T. Ersal, "A Multi-Stage Optimization Formulation for MPC-Based Obstacle Avoidance in Autonomous Vehicles Using a LIDAR Sensor," in *ASME 2014 Dynamic Systems and Control Conference*. ASME, oct 2014.

[16] J. Jeon, R. V. Cowlagi, S. C. Peters, S. Karaman, E. Frazzoli, P. Tsiotras, and K. Iagnemma, "Optimal motion planning with the half-car dynamical model for autonomous high-speed driving," in *2013 American Control Conference*. IEEE, jun 2013, pp. 188–193.

[17] F. Altché, P. Polack, and A. de La Fortelle, "A Simple Dynamic Model for Aggressive, Near-Limits Trajectory Planning," *arXiv preprint arXiv:1703.01225*, mar 2017.

[18] R. Guntur and S. Sankar, "A friction circle concept for dugoff's tyre friction model," *International Journal of Vehicle Design*, vol. 1, no. 4, pp. 373–377, 1980.

[19] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Kinematic and dynamic vehicle models for autonomous driving control design," in *2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, jun 2015, pp. 1094–1099.

[20] X. Qian, A. De La Fortelle, and F. Moutarde, "A hierarchical model predictive control framework for on-road formation control of autonomous vehicles," in *Intelligent Vehicles Symposium (IV), 2016 IEEE*. IEEE, 2016, pp. 376–381.

[21] A. Gray, Y. Gao, J. K. Hedrick, and F. Borrelli, "Robust Predictive Control for semi-autonomous vehicles with an uncertain driver model," in *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, jun 2013, pp. 208–213.

[22] F. Borrelli, P. Falcone, T. Keviczky, J. Asgari, and D. Hrovat, "MPC-based approach to active steering for autonomous vehicle systems," *International Journal of Vehicle Autonomous Systems*, vol. 3, no. 2/3/4, p. 265, 2005.

[23] TASS International. http://www.tassinternational.com/prescan.

[24] B. Houska, H. Ferreau, and M. Diehl, "ACADO Toolkit – An Open Source Framework for Automatic Control and Dynamic Optimization," *Optimal Control Applications and Methods*, vol. 32, no. 3, pp. 298–312, 2011.

[25] J. Kosecka, R. Blasi, C. Taylor, and J. Malik, "Vision-based lateral control of vehicles," in *Proceedings of Conference on Intelligent Transportation Systems*, vol. 46. IEEE, 1997, pp. 900–905.

[26] J. Breuer, "Analysis of driver-vehicle-interactions in an evasive manoeuvre - results of "moose test" studies," *Proceedings: International Technical Conference on the Enhanced Safety of Vehicles*, vol. 1998, pp. 620–627, 1998.