Curvature-Based Geometric Approach for the Lateral Control of Autonomous Cars

Alexandre Lombard[a], Jocelyn Buisson[a], Abdeljalil Abbas-Turki[a], Stéphane Galland[a], Abderrafiaa Koukam[a]

[a]*CIAD UMR 7533, Univ. Bourgogne Franche-ComtÃl, UTBM, F-90010 Belfort, France*

Abstract

Several approaches exist for the lateral control of autonomous vehicles. Among them are the geometric approaches. They are shown to be robust to disturbances and able to manage complex tracks. Their main advantage lies on the fact that they are explainable, in the sense that their behavior can be analyzed to provide guarantees about their limitations. However, they do not give the quality of results that can be obtained using other control principles, mostly because of design issues. This paper aims to tackle these issues by proposing a novel geometric approach based on FrenetâĂŞSerret formulas to reach the level of quality proposed by the other approaches, while still benefiting from the advantages of geometric approaches. A numerical analysis of the proposed control approach show its advantage: Simulation results and tests on a real autonomous car are provided.

Keywords: Autonomous vehicle, Lateral control, Path tracking, Simulation

1. Introduction

Researches about autonomous cars have now been a trendy topic for the last three decades, notably because of the numerous expected outcomes (increased safety, improved road capacity, shared vehicles, increased fuel efficiency, etc. [1, 2]), but also because designing a fully autonomous car (reaching SAE level 4 and 5 [3]) is still a challenging task [4]. Recently, a particular attention is given to the interactions of the robot-driver either with the other road users in the case of fully autonomous vehicles, or with the human-driver for safety and acceptability reasons. However, the perquisite of testing the emergent technologies is an automation driving system able to make the vehicle respecting its planned path. Hence, driving automation solutions that

can be easily deployed on a wide variety of vehicles and situations help for saving a valuable amount of time needed for rapidly extending the scope of the autonomous driving systems.

A key point of car driving automation is the ability to follow a predefined path, which is known as "path tracking control" or "lateral control". For Ackermann-like vehicles (like cars), the problem of path tracking control is to compute either the angle to apply on the steering wheel (i.e. the angle of the front wheels) or the effort to apply on it, to keep the vehicle on a predefined track. It is a famous and widely studied control problem [5]. The challenges for the definition of path tracking control are to be able to manage all possible tracks and to ensure passengers comfort and safety while being robust to the latency (the delay between the perception and the action by the control software), and the perturbations (the error in the data provided by the sensors and the error in the application of the action).

To perform the path following task, several approaches have been proposed that can be divided into the following categories: geometric controllers, dynamic controllers, optimal controllers (LQR), adaptive and intelligent controllers (neural networks and fuzzy logic [6, 7]), model-based controllers (MPC [8, 9]), and classical controllers (PID [10, 11]).

Each different category has its specific advantages and drawbacks. For instance, nowadays, neural networks are giving really interesting results [12], and can directly use as input the signal of a camera for lane-keeping [13] (reducing the required amount of sensors), but the drawback is the limited explainability of the command, making it hard to define the limits of the command and to predict its behavior in every situation, and in the end making it hard to get the legal approval required to use it on open roads. Dynamic and optimal controllers are also efficient [14], but needs tuning, are computationally demanding and requires a lot of data about the model and the dynamic of the vehicles, which can be difficult to obtain accurately (the vehicle's split mass, the center of gravity location, the moment of inertia, the cornering stiffness of tires). PID and MPC controllers are difficult to design and to tune and are dependent on the track and the vehicles' characteristics. On the other hand, most of the geometric approaches are easy to understand, study, and explain, and they are also known to be computationally efficient and robust to perturbations, however, the given results are not as good as the results provided by the other classes of solutions, and are also requiring tuning, which may lead to over-fitting a specific track.

The current limits of the geometric approaches are leaving room for im-

provement. A new geometric approach could provide better results while requiring a limited amount of parameters and tuning, and not being track-dependent. The contribution of this paper is to propose a new geometric approach for lateral control, with the benefits of geometric approaches, while overcoming some of their drawbacks.

Thus, after a quick overview of the common solutions for lateral control (section 2), this paper proposes a contribution to the path tracking problem with a new geometric approach to define the angle of the wheel as a function of the state of the car (section 3.1). A numerical analysis of the command is provided to highlight its characteristics (steady-state error, reaction-time, overshoot, etc.) compared to the most common geometric approaches (section 3.3). Then, to ensure the efficiency and the robustness of the command, simulations are performed including simulated perturbations (with simulated latency and inaccuracy of the sensors), and the results are compared to existing commands (section 4.1). Finally, to confirm the feasibility of the approach, implementation is made on a real autonomous car executing the presented command, and the results are then discussed (section 4.2).

2. Related Works

As stated before, the choice is made to put the focus on geometric approaches for lateral control. This field has produced a lot of work and provides several solutions, which are considered standards nowadays thanks to the quality of results they provide while being explainable and thus exposing their benefits and their limits.

These geometric approaches are based on the geometric bicycle model (Fig. 1), which is a simplification of an Ackermann steered vehicle [15]. In this model, the considered parameters are the radius of curvature of the trajectory R (meters), the wheel base of the vehicle E (meters), and the angle of the front wheel δ (radians).

Relying on this model to describe the movement of the car, the principle of these geometric lateral control solutions is to define the angle $\delta(t)$ to be applied to the front wheels of the vehicle at the instant t. This depends on the geometrical characteristics of the vehicle (notably its wheelbase E), the trajectory to follow (commonly modeled as function of $\mathbb{R} \longrightarrow \mathbb{R}^2$), and the current state of the car, which generally includes the position, the heading, and the speed, and can be extended to include the vehicle velocity, the acceleration, and the slipping forces (if the control solution expects to manage

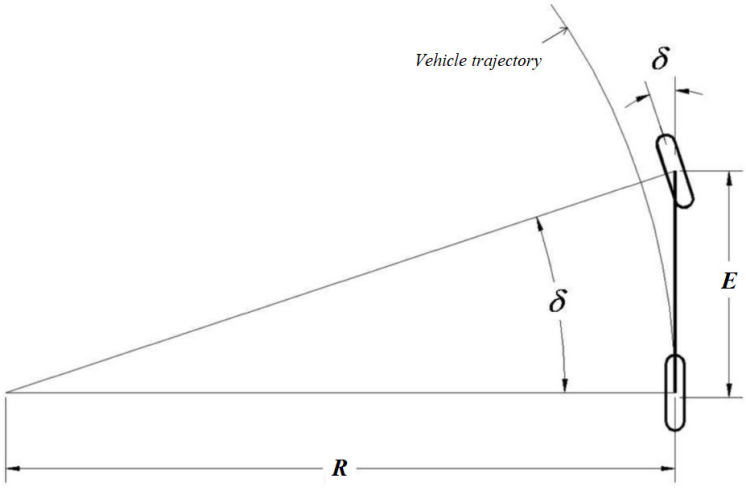

Figure 1: Geometric bicycle model: a model for computing the radius of curvature R and the vehicle trajectory knowing the angle of the front wheels δ and the wheelbase E [16]

vehicles clearly out of the pure rolling conditions). For the sake of simplicity, the movement of the rear wheels is considered, given that the position of the rest of the vehicle can be inferred from the position of the rear wheels.

The model proposed in Fig. 1 is the basis of geometric approaches. It works perfectly for low speeds. As the vehicle speed increases, several forces make wheels slip. This has prompted many studies to move towards the linear differential equation model of forces (kinematic model). However, even if the kinematic model seems satisfactory for controlling slight rotation at high speed, the hypotheses remain just as difficult to check. Several forces are present in practice due to road inclination [17], wind [18], tire conditions [1] and so on. In addition, today's vehicles are equipped to prevent slipping [19] and even adapt the rear wheels rotations accordingly.

The objective is to present an explainable approach on a widely used model in such a way that the solution can be easily tuned according to the vehicle characteristics. Moreover, the control can later be adjusted according to the detected errors due to forces by using artificial intelligence techniques. Hence, in the following, for the purpose of designing a geometric lateral

[1]The entries of the transfer matrix of models are based on the wheel stiffness

control of the vehicle in urban area, we assume that the model presented by Fig. 1 holds in normal urban traffic conditions.

From the classification proposed by Snider et al. [16], the most common solutions and the ones being the most widely adopted, with a focus on geometric approaches, are discussed more in detail further.

2.1. Pure-pursuit

The pure-pursuit was at first applied to the pursuit of a missile to a target [20]. Later, it was applied in the robotics field [21] before being applied to cars [22]. Since then, it is a widely used solution for lateral control of robots or vehicles [23]. Its principle and implementation are simple: the position of the vehicle is projected on a reference trajectory, then a point is defined upstream of this projection on the trajectory, the vehicle determines the angle of the steering wheel so that that the trajectory of the vehicle reaches this point. Given the radius of curvature R of the circle passing through the rear axle of the vehicle and the target point, and admitting the line directed by the vehicle direction as a tangent, the angle δ is given by Eq. (1).

$$\delta(t) = tan^{-1}\left(\frac{E}{R}\right) \tag{1}$$

The position of the target point greatly influences the quality of the monitoring. It is common to place this point at a distance d upstream of the vehicle, with d_0 being a minimum value for the distance of the point, $v(t)$ the speed of the vehicle, and k a dimensionless multiplier of $\mathbb{R}^+$ (often belonging to $[0.5, 2.0]$):

$$d = \max(d_0, k \times v(t)) \tag{2}$$

2.2. Stanley's method

The method of Stanley (Thrun et al. [24]) was used during the *DARPA challenge*. The angle of the wheels is defined by Eq. (3), with $\theta_e(t)$ the difference between the direction of the vehicle and the direction of the trajectory, $e_{fa}(t)$ the lateral error, $v_x(t)$ the speed of the vehicle and k a dimensionless gain parameter.

$$\delta(t) = \theta_e(t) + tan^{-1}\left(\frac{ke_{fa}(t)}{v_x(t)}\right) \tag{3}$$

2.3. Linderoth's method

The Linderoth et al. [25] method is a geometric approach expressing the angle $\delta(t)$ with the relation in Eq. (4).

$$\delta(t) = tan^{-1}\left(\frac{-cos(e_\theta(t))e^{\perp}(t) - (l_1 + l_2)sin(e_{theta}(t))}{l_1 - (l_1 + l_2)cos(e_\theta(t)) + sin(e_\theta(t))e^{\perp}(t)}\right) \tag{4}$$

2.4. Vector pursuit

Another solution to the path-following problem is the vector pursuit [26], based on the theory of screws. Similarly to the pure pursuit, a look-ahead point is computed, but it uses both its position and its orientation (i.e. the tangent to the track at the look-ahead point). Even though this algorithm performs better than the pure pursuit algorithm, it suffers from similar drawbacks as it ignores the influence of the vehicle speed on the path-tracking performance: a look-ahead point placed too close from the vehicle will still induce oscillations when speed increases, while a look-ahead point placed too far will induce too much anticipation in path tracking (i.e. the car will start turning before a curve).

2.5. Overview

Table 1 sums up this breve overview of the main geometric approaches that can be found in the literature. This Table 1 also includes the Linear Quadratic Regulator with Feed-Forward from Cho and Kim [29], which serves here as a reference for non-geometric approaches.

For autonomous cars, the robustness to perturbations is a requirement as the positioning system (mostly GPS) suffers from a lack of precision and accuracy even when relying on an augmentation system (such as Differential GPS or Real-Time Kinematic). Moreover, it is also required to work on a track with sharp curves and rapidly changing curvature, making the solutions well-suited for highway driving un-adapted to the more general situation. Considering these requirements, the pure pursuit has been chosen as a basis for the proposed approach. However, it was important to reduce the drawbacks of the pure pursuit, i.e. the "cutting corners" effect and the implied limited speed. Therefore, researches have been made to improve this solution, taking into account the curvature of the trajectory.

Tracking method	Robustness to disturbances	Path requirements	Cutting corners	Overshooting	Steady-state error	Good applications
Pure pursuit [27]	Good	None with reason	Significant as speed increases	Moderate as speed increases	Significant as speed increases	Slow driving and/or on discontinuous path
Stanley [24]	Fair	Continuous curvature	No	Moderate as speed increases	Significant as speed increases	Smooth highway driving and/or parking maneuvers
Linderoth [25]	Fair	Continuous curvature	Moderate as speed increases	Significant as speed increases	Significant as speed increases	Smooth highway driving
Kinematic [28]	Poor	Continuous curvature through 2nd derivative of curvature	No	Moderate as speed increases, significant during rapidly changing curvature	Significant as speed increases	Smooth parking maneuvers
LQR with FF [29]	Poor	Continuous curvature	No	Significant during rapidly changing curvature	Minimal until much higher speeds	Smooth high speed driving, urban driving at speed
Preview [30]	Fair	Continuous curvature	Moderate in rapidly changing curvature and/or speed	Moderate in rapidly changing curvature and/or speed	Minimal until much higher speeds	Highway driving at relatively constant speed

Table 1: Overview of the existing solutions, adapted from Snider et al. [16]

3. Geometric Lateral Control with Curvature Following

3.1. Parameters and relation with the angle of the front wheels

To apply the existing solutions to real autonomous cars, the limits of the sensors and the actuators of the vehicles must be considered. All of the sensors will suffer from latency (e.g. an RTK GPS can only provide a position at 10Hz) and inaccuracy (e.g. around 10cm for RTK GPS), and the same is true for the actuators.

Thus, the lateral control strategy must be robust and bear a certain latency and inaccuracy, regarding the measure of the state of the vehicle (position, speed), and the application of the command. Simulation tests showed that under these constraints most of the existing solutions did not give satisfactory results.

During these tests, the strategy that gave the best results was pure pursuit. This is due to a higher tolerance to disturbances and lower prerequisites for the accuracy and response time of the system. Nevertheless, pure pursuit tends to cut the curves by the inside, and this, all the more as the speed increases. To overcome this, the tracking strategy can integrate the curvature of the trajectory into the equation. A first approach has been proposed in [31], which is refined in this paper by proposing a new command described by Eq. (5). This equation is, later on, referred to as "improved curvature following" or "improved CF".

$$\delta_t = s \times tan^{-1}\left(\frac{2Ecos(e_t - tan^{-1}(\frac{d}{e_d}))}{\sqrt{e_d^2 + d^2}}\right) + sin^{-1}\left(\frac{2E}{L} sin^{-1}\frac{\kappa}{2}\right) \quad (5)$$

Fig. 2 represents a vehicle (illustrated by a box of length E) trying to follow the curvilinear trajectory represented below. The relative position and orientation of the vehicle to the target trajectory leads to the definition of several parameters, which are used to compute the desired wheel angle δ (according to Eq. (5)).

The parameters of Eq. (5) (illustrated in Fig. 2) are:

- s a value being 1 if the vehicle is on the left side of the trajectory or -1 if it is on the right side
- e_t the angle between the direction of the vehicle and the tangent to the trajectory

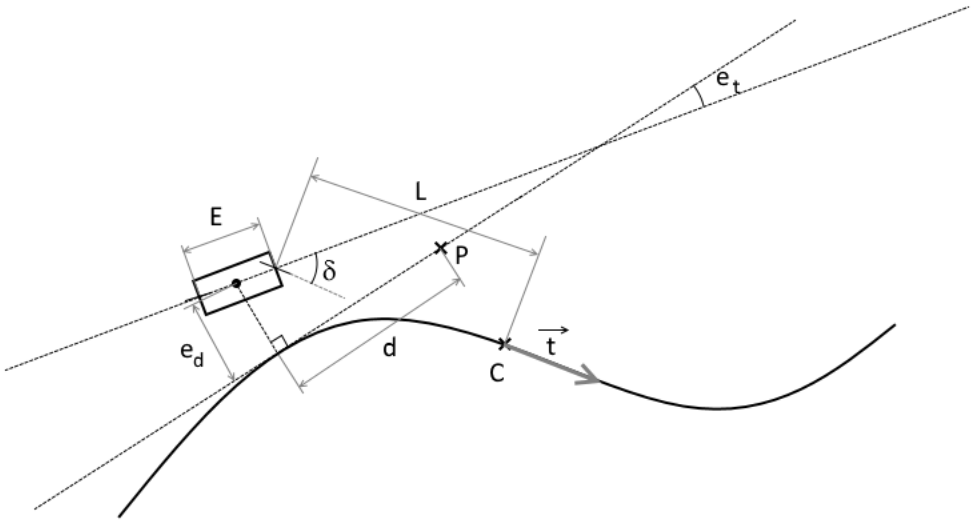

Figure 2: Illustration of the parameters of Eq. (5) - the car is illustrated as a box with two wheels (geometric bicycle model) trying to follow the trajectory below

- d the distance to the target point with $d = 2\tau \times v(t)$, τ being the estimated reaction time of the system (technically the delay between two subsequent acquisitions of a position) and $v(t)$ the velocity of the vehicle ; the choice of this value for d is discussed in 3.2
- e_d the signed distance between the vehicle and the trajectory
- $\vec{t}$ the incoming direction of the trajectory
- L the look-ahead distance for the curvature following part, which can be defined by $L = 2v(t)$ with a minimal value of some meters (like it is commonly done with the pure-pursuit) being at least $E\pi$
- κ the norm of the vector being the difference between the tangent at the target point $\vec{t}$ and the direction of the vehicle, it is a signed value whose sign is the sign of the angle between $\vec{t}$ and the direction of the car

Eq. (5) is formed by the addition of two members. The first member corresponds to the application of the pure pursuit to the point P. The second

member corresponds to the application of the curvature of the trajectory (taken upstream, to take account of the reaction time of the system). Its purpose is to reduce the steady-state error of the pure-pursuit when there is a curvature in the trajectory, as the presence of a curvature will increase the command. This assumption is verified through a numerical analysis of the command in section 3.3.

3.2. Defining the look-ahead distance

The command from Eq. (5) relies on a look-ahead distance. To ensure the convergence of the command, this look-ahead distance d must satisfy the property $d \geq 2v\tau$ (with v being the speed).

The proof for this affirmation is given below, under the following assumptions:

1. The curvature κ_δ from the command satisfies $\kappa_\delta \leq 0.25m^{-1}$
2. The command aims to keep the vehicle on the track, thus e_d and e_t stay within an acceptable limit which is discussed thereafter
3. The track can physically be followed by the vehicle

This being stated, the command can be expressed under the following form:

$$\kappa_\delta = \kappa_c + \hat{\kappa} \tag{6}$$

Where $\hat{\kappa}$ is the curvature of the track to follow and κ_c the correction which can be approximated by:

$$\kappa_c \approx 2\frac{d\sin e_t + e_d \cos e_t}{e_d^2 + d^2} \tag{7}$$

Using numeric integration, the trajectory of the vehicle can be approached by:

$$X_{k+1} = \begin{bmatrix} a \\ x \\ y \end{bmatrix}_{k+1} = \begin{bmatrix} a \\ x \\ y \end{bmatrix}_k + s \begin{bmatrix} -\kappa_\delta \\ \cos\left(\alpha_k - \frac{s\kappa_\delta}{\beta}\right) \\ \sin\left(\alpha_k - \frac{s\kappa_\delta}{\beta}\right) \end{bmatrix} \tag{8}$$

With $\beta \in [1, +\inf]$, α, x, y the angle of the vehicle and its coordinates in a Cartesian coordinate system, and $s = v\tau$ where v is the speed and τ the sampling time.

Replacing κ_δ of (6) into (8) gives:

$$\begin{bmatrix}\hat{\alpha}\\ \hat{x}\\ \hat{y}\end{bmatrix}_{k+1} + \begin{bmatrix}e_t\\ e_x\\ e_y\end{bmatrix}_{k+1} = \begin{bmatrix}\hat{\alpha}\\ \hat{x}\\ \hat{y}\end{bmatrix}_{k} + \begin{bmatrix}e_t\\ e_x\\ e_y\end{bmatrix}_{k} + s\begin{bmatrix}-\kappa_c - \hat{\kappa}\\ \cos\left(\hat{\alpha_k} + e_t - \frac{s(\kappa_c+\hat{\kappa})}{\beta}\right)\\ \sin\left(\hat{\alpha_k} + e_t - \frac{s(\kappa_c+\hat{\kappa})}{\beta}\right)\end{bmatrix} \tag{9}$$

Where $\hat{\alpha}$, $\hat{x}$ and $\hat{y}$ are the ideal coordinates. By developing (9), the following can be written:

$$\begin{cases}\hat{\alpha_{k+1}} + e_{t_{k+1}} = \hat{\alpha_k} + e_{t_k} - s\kappa_c - s\hat{\kappa}\\ Y_{k+1} = \begin{bmatrix}\hat{x}\\ \hat{y}\end{bmatrix}_{k+1} + \begin{bmatrix}e_x\\ e_y\end{bmatrix}_{k+1} = \begin{bmatrix}\hat{x}\\ \hat{y}\end{bmatrix}_{k} + \begin{bmatrix}e_x\\ e_y\end{bmatrix}_{k} + sR(e_t - \frac{s\kappa_c}{\beta})\begin{bmatrix}\cos\left(\hat{\alpha_\kappa} - \frac{s\hat{\kappa}}{\beta}\right)\\ \sin\left(\hat{\alpha_\kappa} - \frac{s\hat{\kappa}}{\beta}\right)\end{bmatrix}\end{cases} \tag{10}$$

Where $R(\Psi)$ is a plane rotation of angle Ψ: $R(\Psi) = \begin{bmatrix}\cos(\Psi) & -\sin(\Psi)\\ sin(\Psi) & \cos(\Psi)\end{bmatrix}$.

From (10) and because it's an orthogonal projection, the following can be deduced:

$$\begin{cases}e_{t_{k+1}} = e_{t_k} - s\kappa_c\\ \begin{bmatrix}e_x\\ e_y\end{bmatrix}_{k+1} = \begin{bmatrix}e_x\\ e_y\end{bmatrix}_{k} + s(R(e_t - \frac{s\kappa_c}{\beta}) - I)\begin{bmatrix}\cos\left(\hat{\alpha_k} - \frac{s\hat{\kappa}}{\beta}\right)\\ \sin\left(\hat{\alpha_k} - \frac{s\hat{\kappa}}{\beta}\right)\end{bmatrix}\end{cases} \tag{11}$$

Where I is the identity matrix. Thus, as e_d is an orthogonal projection on the tangent to the track to follow, this leads to:

$$\begin{bmatrix}e_t\\ e_d\end{bmatrix}_{k+1} = \begin{bmatrix}e_t\\ e_d\end{bmatrix}_{k} + s\begin{bmatrix}-\frac{\kappa_c}{\beta}\\ \sin\left(e_{t_k} - \frac{s\kappa_c}{\beta}\right)\end{bmatrix} \tag{12}$$

So:

$$\begin{bmatrix}e_t\\ e_d\end{bmatrix}_{k+1} = \begin{bmatrix}e_t\\ e_d\end{bmatrix}_{k} + s\begin{bmatrix}-2\frac{d\sin e_{t_k} + e_{d_k}\cos e_{t_k}}{e_{d_k}^2 + d^2}\\ \sin\left(e_{t_k} - s\frac{2}{\beta}\frac{d\sin e_{t_k} + e_{d_k}\cos e_{t_k}}{e_{d_k}^2 + d^2}\right)\end{bmatrix} \tag{13}$$

Under the following hypotheses:

1. $d \geq 2$ and $d > |e_{d_0}|$
2. $e_{d_0}e_{t_0} \leq \frac{\pi}{6}$

3. $s < 5m$
4. $l = \sqrt{e_{d_k}^2 + d^2}$

The system described by (13) can be approximated by the second order Taylor expansion like this:

$$\begin{bmatrix} e_t \\ e_d \end{bmatrix}_{k+1} = \begin{bmatrix} (1 - 2\frac{s}{l}) & -2\frac{s}{l^2} \\ s(1 - \frac{2s}{l\beta}) & (1 - \frac{2s^2}{\beta l^2}) \end{bmatrix} \begin{bmatrix} e_t \\ e_d \end{bmatrix}_k + \begin{bmatrix} o(e_t^2) \\ o(e_d^2) \end{bmatrix} \longrightarrow \begin{bmatrix} e_t \\ e_d \end{bmatrix}_{k+1} \approx A \begin{bmatrix} e_t \\ e_d \end{bmatrix}_k \tag{14}$$

And by diagonalization of A:

$$A = Q \begin{bmatrix} 1 - s\frac{s+\beta^2 l-\varphi}{\beta^2 l^2} & 0 \\ 0 & 1 - s\frac{s+\beta^2 l+\varphi}{\beta^2 l^2} \end{bmatrix} Q^{-1} \tag{15}$$

With $\varphi = \sqrt{-\beta^4 l^2 + (4\beta^3 - 2\beta^2)ls + s^2}$. To ensure the convergence to 0 of the system described by Eq. (14), the constraint $\|1 - s\frac{s+\beta^2 l \pm \varphi}{\beta^2 l^2}\| < 1$ must be satisfied, which implies:

$$l > 2\frac{s}{\beta} \tag{16}$$

Thus:

$$d > 2s \tag{17}$$

3.3. Numerical analysis

A numerical analysis of the Eq. (5) is proposed to observe the following properties of the relation:

- Convergence: is the lateral error asymptotically converging to a real value?
- Steady-state error: what is the difference between the asymptotic value of the lateral error and 0 in steady-state?
- Reaction time: how much time is required to reach the asymptotic value?
- Overshoot: how much is the asymptotic value exceeded before reaching the steady-state?

For this purpose, the relation is applied upon two representative trajectories. They represent a straight line (Eq. (18)) and a circle (Eq. (19)), respectively.

$$T_{line} = \begin{cases} x(t) = t \\ y(t) = 0 \end{cases} \tag{18}$$

$$T_{circle} = \begin{cases} x(t) = R \times cos(t) \\ y(t) = R \times sin(t) \end{cases} \tag{19}$$

The car state S is defined by the position of the rear wheel in $\mathbb{R}^2$, its speed in $\mathbb{R}^+$ and its orientation in $[-\pi, \pi]$, thus $S \in \mathbb{R}^2 \times \mathbb{R} \times [-\pi, \pi]$. The state of the vehicle at a given time is given by $S(t)$ with $t \in \mathbb{R}$.

To compute the next state knowing the current state and the wheel angle δ, the discrete geometric bicycle model is used. Thus, knowing $S(t)$ and the angle of the front wheel $\delta(t)$, the recurrence relation giving $S(t + \Delta t)$ is:

$$S(t + \Delta t) = S(t) + \begin{pmatrix} R(t) \cos \theta(t) \\ R(t)(\sin \theta(t) + 1) \\ a(t)\Delta t \\ \theta(t) \end{pmatrix} \tag{20}$$

Using this relation, the evolution of the error being defined by the distance between the vehicle and the trajectory is computed for both trajectories. The acceleration $a(t)$ is set to 0. The initial state of the vehicle is $S_0 = (x_0, y_0, v_0, \theta_0)$.

Eq. (20) representing the bicycle model is only a description of the movement of the vehicle for a given wheel angle δ. Then, the wheel angle δ must be computed using a control solution. In the following, the Equation 5 is used to compute δ; the pure pursuit and Stanley's command are also used for reference.

The application of the recurrence formula using Eq. (5) to compute the angle of the front wheels δ gives the curve in Fig. 3 and Fig. 4 where the lateral error is measured as a function of the time, considering respectively a straight-line and a circle for the trajectory to follow. The reference curves are computed with the following parameters: the initial velocity is 10 m/s, the distance to the target point is $k \times v(t) = 20m$ (with $k = 2$) for the pure-pursuit, the gain for Stanley's command is $k = 5$.

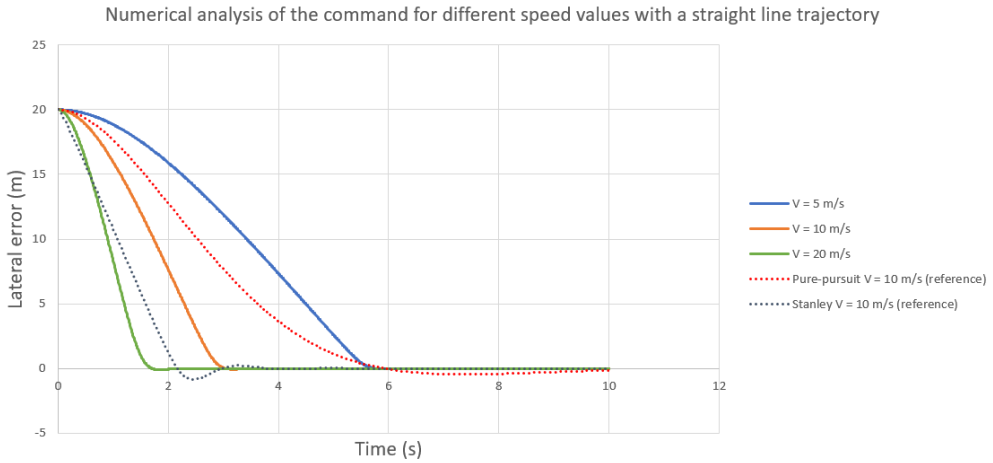

Figure 3: Application of the command at various speeds considering a straight-line trajectory

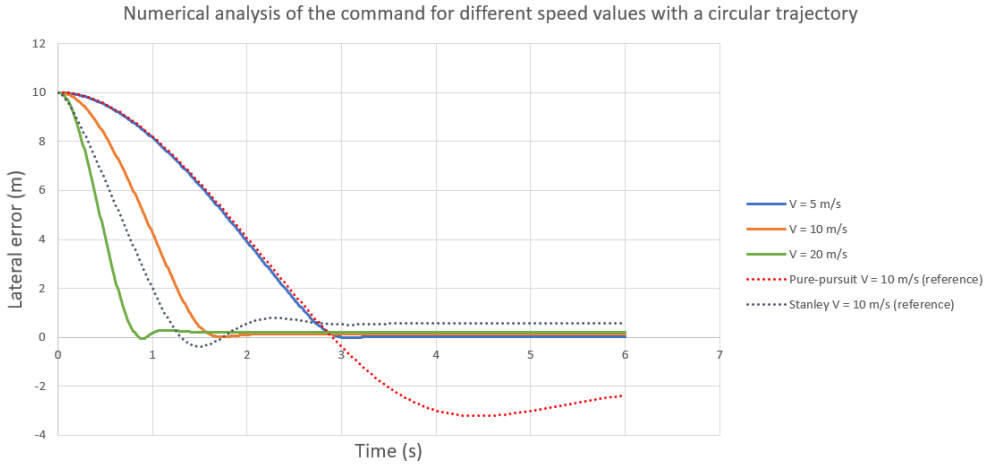

Figure 4: Application of the command at various speeds considering a circular trajectory

From the numerical analysis, the following can be noted about the command given by Eq. (5):

- **Convergence**: In all scenarios (for both trajectories, and the different considered speeds), there is a convergence of the lateral error.
- **Steady-state error**: There is no steady-state error when the trajectory is a straight-line, this result is similar to what can be observed with the pure-pursuit or Stanley's command; a small steady-state error can be observed when the trajectory is a circle, which tends to increase when the speed increases, the measured steady-state error is small compared to the steady-state error observed with the pure-pursuit (16,28x smaller) and Stanley's command (4,17x smaller) (based on the scenario V = 10 m/s)
- **Reaction time**: the considered reaction time is the time $\tau_{5\%}$ required to be under 5% of the difference between the initial lateral error and the final lateral error; according to the chosen initial conditions for the straight-line trajectory it is 1.36 times the reaction time of Stanley's command and 0.54 times the reaction of the pure-pursuit, for the circular trajectory it is 1.23 times the reaction time of Stanley's command and 0.53 times the reaction time of the pure-pursuit.
- **Overshoot**: the considered value for the overshoot is the quantity of overshoot to the final lateral error relative to the total range of the lateral error, considering the case V = 10 m/s for the straight-line it is 0.0% for the command (5), 2.3% for the pure-pursuit and 4.2% for Stanley's command; for the circle, it is 1.1% for the command (5), 10.1% for the pure-pursuit and 9.3% for Stanley's command

Even though, the presented results are interesting, the numerical analysis only focuses on the characteristics of the formula (5) without any consideration for the physical limitations of the car or the sensors. To study the behavior of the command in more realistic scenarios, simulations must be performed.

4. Experiments and results

4.1. Simulations

Simulations have been carried out to compare the proposed solution with the existing strategies. These simulations consist of following a predefined

Strategy	LQR	Linderoth	Improved PP	Improved CF	Stanley
Without latency	3.06E-2	5.08E-1	2.13E-1	2.87E-2	4.01E-2
With latency	4.36E-1	4.89E-1	1.97E-1	8.67E-2	4.81E-2
Latency, error	4.79E-1	5.19E-1	2.02E-1	1.13E-1	1.47E-1

Table 2: Mean lateral error over the entire track (expressed in meters)

track using the different strategies mentioned above. To reproduce the constraints of the real system two perturbations are introduced:

- An inaccuracy on the position of the vehicle
- A latency on the application of the calculated command

Three scenarios are established to evaluate the performance of the vehicle under the following conditions:

- *No latency*: Without imprecision and with low latency
- *Latency*: With latency and without inaccuracy
- *Latency, error*: With latency and inaccuracy

Numerous tests have been carried out, for different reference tracks, different values of inaccuracy on the position and different latency values. For the sake of brevity, the results shown are limited to an asymmetric 8-shaped track, presenting straight-lines and changing radius of curvature, including a sharp one, making it a typical continuous track.

Thus, on the track defined by Fig. 5, the results given by the figures Fig. 6, Fig. 7 and Fig. 8 were obtained, respectively for scenarios 1, 2 and 3. For these measurements latency (when present) was set to 400 ms and the positional inaccuracy was fixed to 10cm for the position (the position used by the vehicle to determine its control was a randomly and uniformly chosen value around the actual value and within a radius of 10cm) and at $\pm 5\,\mathrm{deg}$ for the direction (also uniformly around the actual value).

The command referred to as "improved PP" is an improved version of the pure-pursuit defined in [31], which is basically the pure-pursuit with a gain.

In the figures, the usage of the "improved CF" command (in yellow) shows a reduced lateral error and a better stability compared to the other

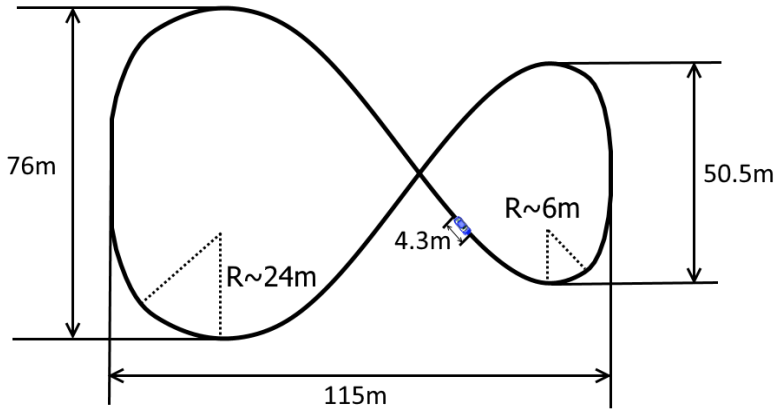

Figure 5: Track used in simulation for the given results

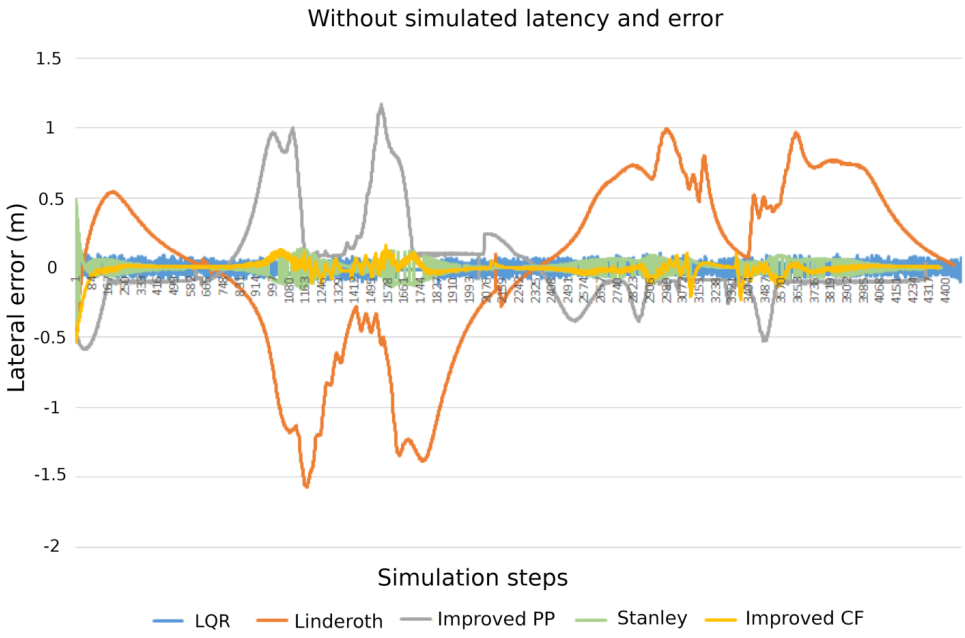

Figure 6: Instantaneous lateral error in the case of a simulation without positional inaccuracy and without latency

commands, with only Stanley's command (in green) and LQR (in blue) approaching the quality of results of "improved CF". However, when latency

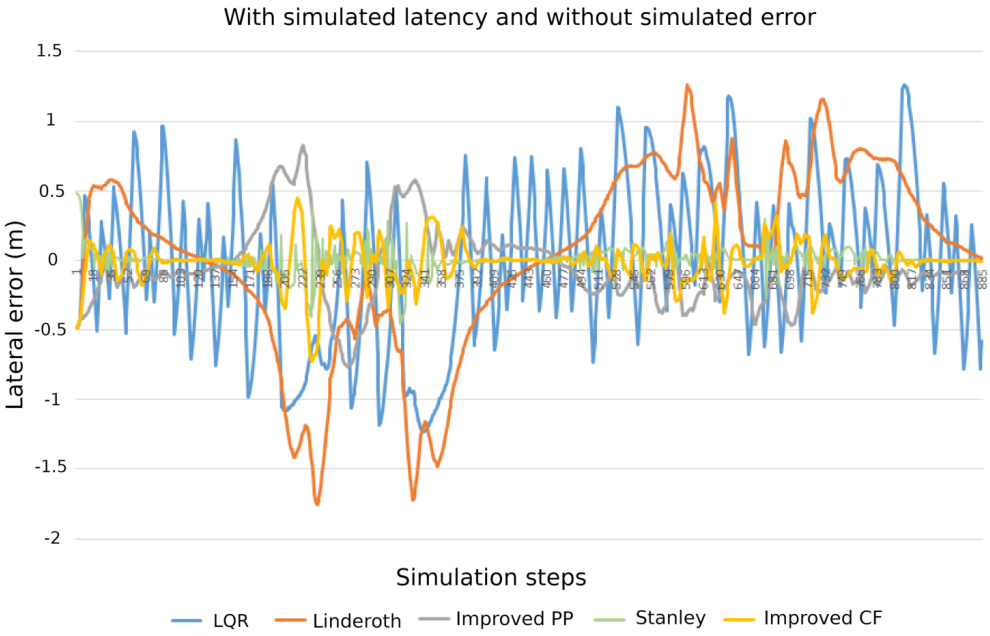

Figure 7: Instantaneous lateral error in the case of a simulation without positional inaccuracies and with latency on the application of the control

and errors are introduced, the LQR's results are far from those given by "improved CF" and Stanley. In this later case, the results provided by "improved CF" are still better than the results provided by Stanley. For clarity, the mean lateral errors and the relative mean lateral error are given by the tables 2 and 3. They show the significant gain provided by the proposed solution, even when there are latency and inaccuracy.

For reference, an implementation of the simulation environment is available online at `https://alexandrelombard.github.io/lateral-control`,

Strategy	LQR	Linderoth	Improved PP	Improved CF	Stanley
Without latency	-6.29%	-94.35%	-86.49%	0.00%	-28.42%
With latency	-80.11%	-82.27%	-56.04%	0.00%	80.18%
Latency, error	-76.43%	-78.21%	-44.16%	0.00%	-23.02%

Table 3: Relative error compared to the proposed solution (5) - The color of the cells goes from red to green, with the greener the cell, the better the gain

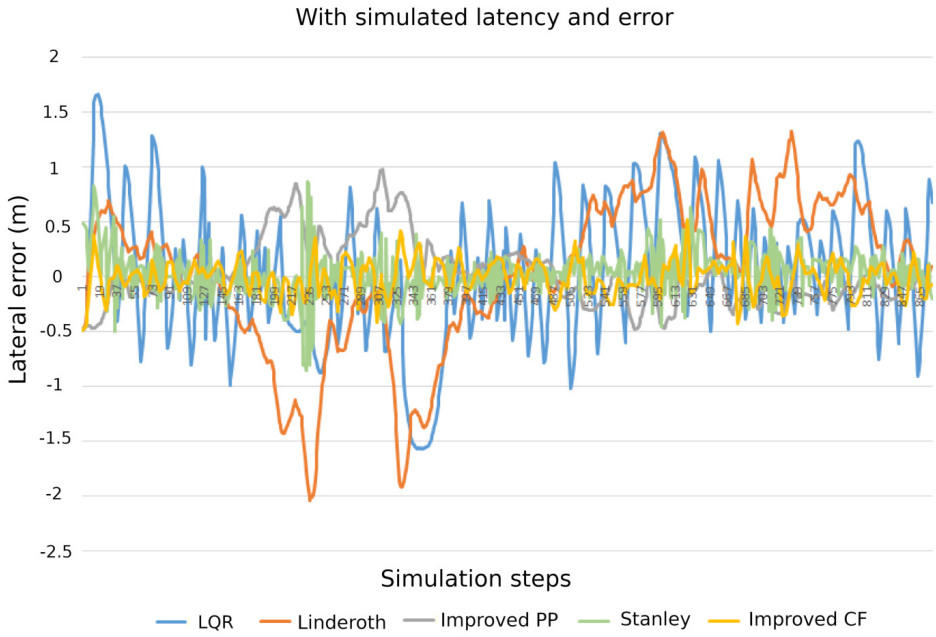

Figure 8: Instantaneous lateral error in the case of a simulation with inaccuracy of positioning and with latency on the application of the control

which allows users to compare different lateral control strategies under several constraints (errors, latency), while allowing them to propose their own.

To highlight the differences between Stanley's command and the proposed improved CF, simulations have been performed on the online simulator with another eight-shaped track. The evolution of the lateral error along the track is given in Fig. 9. In this figure, the average absolute error along the track is $11.2 \times 10^{-2} m$ for improved CF and $12.4 \times 10^{-2} m$ for Stanley (-9.08%). The maximum absolute error is $0.906m$ for Stanley and $0.813m$ for improved CF (-10.2%). Due to longer straight sections (where both Stanley and improved CF show good results) the improvement in the average lateral error seems less important, yet the Fig. 9 shows clearly reduced peaks of the lateral error with improved CF.

Also, according to [32], a weak point of Stanley's command is its poor performance at high-speed, thus simulations have been performed at $50 m.s^{-1}$ (which is not physically realistic given the track, but allow us to study how the commands would theoretically react at higher speeds). The results are

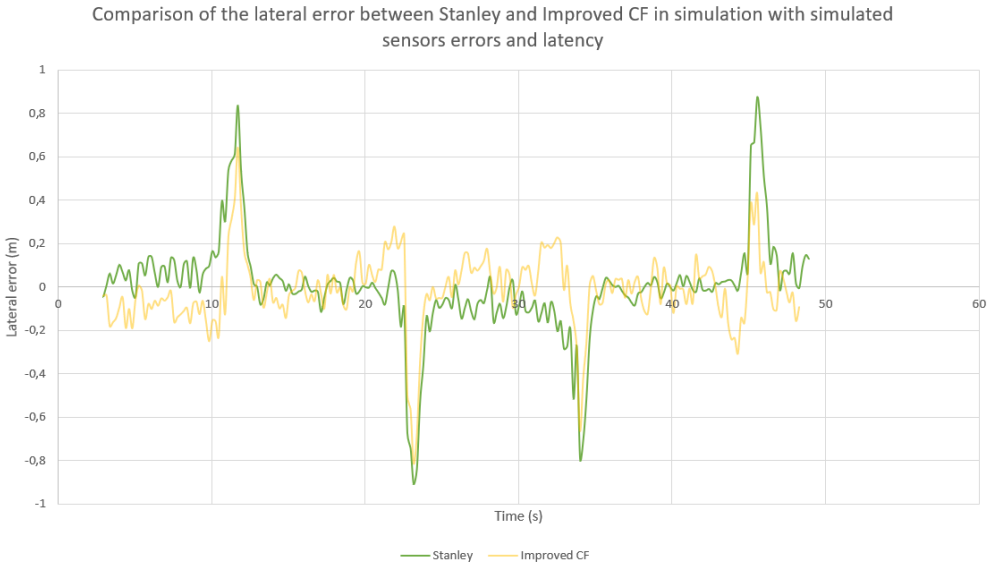

Figure 9: Detailed comparison of the evolution of the lateral error between Stanley and improved CF

presented in Fig. 10 showing the benefits of improved CF: average error of $0.542m$ for improved CF against $0.689m$ for Stanley (-21.4%).

To confirm these results obtained with simulations, real tests have been performed which are described thereafter.

4.2. Real tests

In a second step, the control was tested in real conditions on a robotic vehicle (Renault Scenic III) and equipped with an RTK GPS. The dimensions of the vehicle are given by table 4.

The positioning system used was a GNSS RTK ProFlex 500 to have a precise position (usually below 0.1 meters) every 100 ms. Our experimentation has shown an error less than 5 cm under the condition of a good communication between the RTK base and the RTK rover. For the sake of safety, we managed to make the vehicle brake when the RTK rover had positioning issues or communication issues with the RTK base. Therefore the vehicle was only allowed to move if the mode was "RTK fixedâĂİ or "RTK float" with a maximal communication delay of 4 seconds in this second case. Our experimentation protocol was then close to the one used in the simulations:

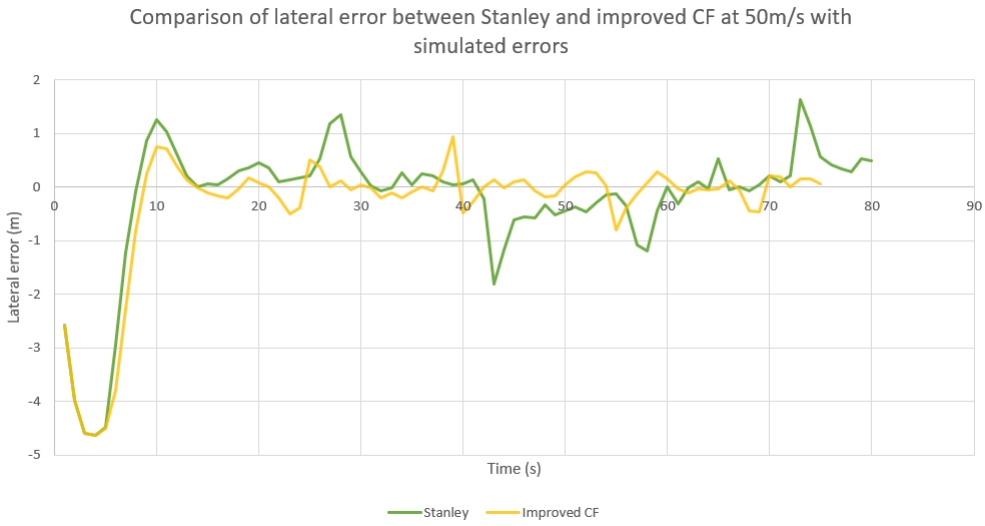

Figure 10: Detailed comparison of the evolution of the lateral error between Stanley and improved CF at high speed (the vehicle position is initialized at around 3m of the track with an angle of -0.6 rad)

the goal was to follow a saved track on an eight-shaped circuit. The first step of the experimentation was to record the points of the track by making manually a first passage on the track. The second step was to let the car try to follow the track without any human intervention.

Fig. 11 refers to the reference circuit used in red and the circuit realized by the vehicle during three turns made using the autonomous lateral control (only acceleration and braking were manually controlled). The minimum radius of the curvature of the reference path is estimated to $8.2m$ according to GPS data.

Parameter	**Value**
Length	4344 mm
Width	1845 mm
Wheelbase	2703 mm
Turning radius	5645 mm
Front track	1546 mm
Rear track	1547 mm

Table 4: Dimensions of the Renault Scenic III used for the tests

Regarding the delay between the reception of the position and the application of the command, they have been estimated by Eq. (21), with δ_{gps} being the age of the position in the worst-case scenario (i.e. 100 ms according to the update frequency of the GPS RTK), δ_{link} the delay of transmission of the position from the GPS to the computer calculating the command, δ_{cpu} the time required to process the data from the GPS, compute and transmit the corresponding command, and $\delta_{command}$ the time required to apply the command on the steering wheel.

$$\Delta = \delta_{gps} + \delta_{link} + \delta_{cpu} + \delta_{command} \tag{21}$$

δ_{link} have been considered insignificant, and $\delta_{cpu} + \delta_{command}$ has been measured as approximately 300 ms. This has lead to an estimated overall delay $\Delta = 400ms$.

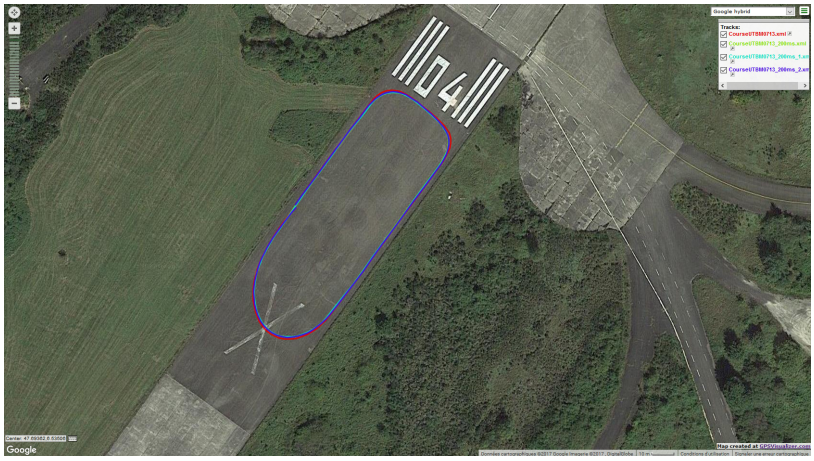

Figure 11: GPS track of the reference trajectory in red, and GPS tracks made according to the autonomous lateral control

Fig. 13 shows the lateral error (orthogonal distance to the reference track) measured as a function of time for each circuit. The speed is controlled manually and varies throughout the circuit and is on average between $4.5m/s$ (in turns) and $10m/s$ (in straight-lines).

An error limited to $1m \pm 0.05m$ can be observed in the corners and a $0.2m \pm 0.05m$ error in straight-lines. Remark: as the GPS RTK is used to measure the distance of the vehicle with the reference track, and that

Figure 12: Screenshot of the video of the experiment: the full video is accessible at `https://youtu.be/uCctBc-eFio`

it suffers from at most 5 cm of error, all the error measurements must be considered with a margin of 5 cm.

The results are slightly worse than those obtained in simulation (for which the error was limited to $0.5m \pm 0.05m$). This is due in part to a shift in the steering wheel control (the 0 *deg* command corresponded to a real steering wheel angle of about 15 deg). This problem was observed during the first two runs, then compensated in the latter, which allowed the error to be limited to $0.8m \pm 0.05m$ in turns and $0.1m \pm 0.05m$ in straight-lines. For the remainder, the additional error is due to an underestimation of the error of the orientation of the vehicle in simulation (± 10 deg in real instead of the ± 5 deg considered in the simulation).

Overall, the error is contained, and the behavior is relatively close to the estimations previously obtained in simulation: low error in the straight-lines, slightly more important in the corners, and no oscillations. And, more importantly, the real tests ensures the feasibility of the implementation of the command.

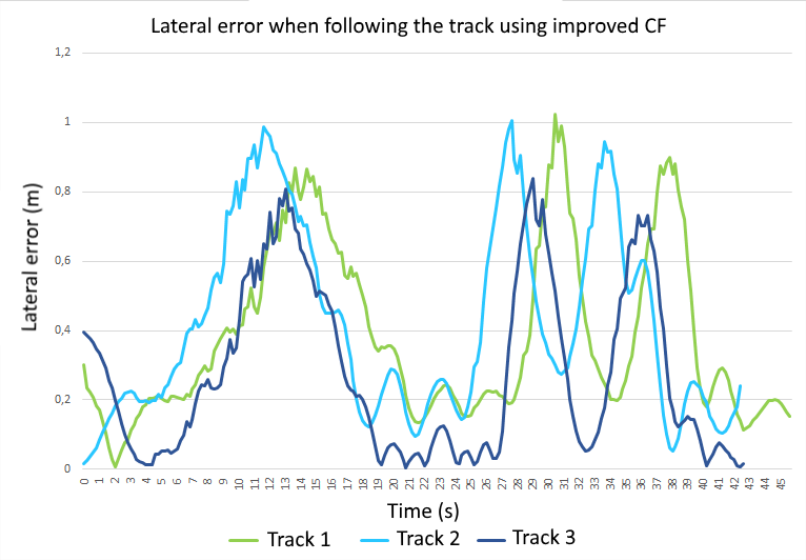

Figure 13: Lateral error measured during the experiment in real conditions with the lateral control

5. Conclusion

This paper presents a contribution to geometric approaches for the lateral control of autonomous cars. The proposed approach benefits from the advantages of the existing geometric approaches (studiability, robustness and low requirements on sensors, etc.) while overcoming some drawbacks of the existing solutions (better results, no specific tuning according to a given track). The results have been checked with a numerical analysis of the proposed command, exposing its main characteristics, simulations with simulated error and latency, and on a real robotized vehicle to ensure its feasibility and to validate the results obtained through simulation.

In the context of the development of autonomous cars, the geometric approaches are interesting as they can be easily used and implemented by manufacturers, deployed on any car, and it requires a small number of dedicated sensors. Moreover, the analysis of the command allows us to accurately expose its behavior in any situation compared to other kind of approaches. This gives better insurance on the behavior of the command, and make it safer than other approaches, so its approval would be easier to obtain. Following this idea, it could also be used as a fallback solution to the more

sophisticated methods, to certify these by providing strong guarantees.

Compared to existing approaches, the simulations shown better results for the proposed solution. Among the other geometric approaches, only the Stanley's command was giving results comparable to the proposed approach, yet the average lateral error of the proposed command was still 23% smaller than the average error of Stanley's command when the simulations included latency and positional error (to reproduce the limitations of the system used in the real tests). Stanley's command was being advantaged only if the simulations were including latency without simulated positional error. It can also be noted that the proposed command does not rely on parameters which may lead to overfitting a specific trajectory, as opposed to other geometric approaches.

The real tests presented in this article are using a GPS RTK as the only required sensor needed to compute the command. In further work, it could be interesting to perform other tests using different positioning systems (like SLAM [33]) to ensure it works in every scenario. Also, the problem of the integration of path planning and path following is still ongoing. The presented tests use a predefined trajectory, but further tests are required to study how the system would work with a dynamically computed trajectory, and eventually a dynamically changing trajectory. Given that the pure-pursuit, which serves as a basis of the proposed command is known to manage well discontinuous tracks, it can be expected to have interesting results. Finally, the lateral control solution has to be associated with a longitudinal control solution to build a complete lane-keeping system.

Acknowledgement

This research work has been conducted with the financial support of the Région Bourgogne-Franche Comté. We also thank the automotive supplier FAAR industry for equipping the cars used for the tests.

References

[1] M. Rajasekhar, A. K. Jaswal, Autonomous vehicles: The future of automobiles, in: 2015 IEEE International Transportation Electrification Conference (ITEC), IEEE, 2015, pp. 1–6.

[2] S. A. Bagloee, M. Tavana, M. Asadi, T. Oliver, Autonomous vehicles: challenges, opportunities, and future implications for transportation policies, Journal of modern transportation 24 (2016) 284–303.

[3] T. SAE, Definitions for terms related to on-road motor vehicle automated driving systems, J3016, SAE International Standard (2014).

[4] I. Barabás, A. Todoruţ, N. Cordoş, A. Molea, Current challenges in autonomous driving, in: IOP Conference Series: Materials Science and Engineering, volume 252, IOP Publishing, 2017, p. 012096.

[5] N. H. Amer, H. Zamzuri, K. Hudha, Z. A. Kadir, Modelling and control strategies in path tracking control for autonomous ground vehicles: a review of state of the art and challenges, Journal of Intelligent & Robotic Systems 86 (2017) 225–254.

[6] G. Antonelli, S. Chiaverini, G. Fusco, A fuzzy-logic-based approach for mobile robot path tracking, IEEE Transactions on Fuzzy Systems 15 (2007) 211–221.

[7] S. Huang, G. Lin, Parallel auto-parking of a model vehicle using a self-organizing fuzzy controller, Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering 224 (2010) 997–1012.

[8] P. Falcone, Nonlinear model predictive control for autonomous vehicles, Ph.D. thesis, Università del Sannio, 2007.

[9] M. A. Abbas, Non-linear model predictive control for autonomous vehicles, Ph.D. thesis, UOIT, 2011.

[10] J. E. Normey-Rico, I. Alcalá, J. Gómez-Ortega, E. F. Camacho, Mobile robot path tracking using a robust pid controller, Control Engineering Practice 9 (2001) 1209–1214.

[11] P. Zhao, J. Chen, Y. Song, X. Tao, T. Xu, T. Mei, Design of a control system for an autonomous vehicle based on adaptive-pid, International Journal of Advanced Robotic Systems 9 (2012) 44.

[12] A. E. Sallab, M. Abdou, E. Perot, S. Yogamani, End-to-end deep reinforcement learning for lane keeping assist, arXiv preprint arXiv:1612.04340 (2016).

[13] K. I. Khalilullah, S. Ota, T. Yasuda, M. Jindai, Development of robot navigation method based on single camera vision using deep learning, in: 2017 56th annual conference of the society of instrument and control engineers of Japan (SICE), IEEE, 2017, pp. 939–942.

[14] R. Sharp, D. Casanova, P. Symonds, A mathematical model for driver steering control, with design, tuning and performance results, Vehicle system dynamics 33 (2000) 289–326.

[15] S. F. Campbell, Steering control of an autonomous ground vehicle with application to the DARPA urban challenge, Ph.D. thesis, Massachusetts Institute of Technology, 2007.

[16] J. M. Snider, et al., Automatic steering methods for autonomous automobile path tracking, Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RITR-09-08 (2009).

[17] K. Jiang, A. C. Victorino, A. Charara, Real-time estimation of vehicle's lateral dynamics at inclined road employing extended kalman filter, in: 2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA), IEEE, 2016, pp. 2360–2365.

[18] A.-T. Nguyen, C. Sentouh, J.-C. Popieul, Fuzzy steering control for autonomous vehicles under actuator saturation: Design and experiments, Journal of the Franklin Institute 355 (2018) 9374–9395.

[19] Z. Chen, J. Wu, B. Zhu, Accurate Pressure Control Strategy of Electronic Stability Program Based on the Building Characteristics of High-Speed Switching Valve, Technical Report, SAE Technical Paper, 2019.

[20] L. Scharf, W. Harthill, P. Moose, A comparison of expected flight times for intercept and pure pursuit missiles, IEEE Transactions on Aerospace and Electronic Systems (1969) 672–673.

[21] R. S. Wallace, A. Stentz, C. E. Thorpe, H. P. Moravec, W. Whittaker, T. Kanade, First results in robot road-following., in: IJCAI, 1985, pp. 1089–1095.

[22] R. C. Coulter, Implementation of the pure pursuit path tracking algorithm, Technical Report, DTIC Document, 1992.

[23] M. Samuel, M. Hussein, M. B. Mohamad, A review of some pure-pursuit based path tracking techniques for control of autonomous vehicle, International Journal of Computer Applications 135 (2016) 35–38.

[24] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, et al., Stanley: The robot that won the darpa grand challenge, Journal of field Robotics 23 (2006) 661–692.

[25] M. Linderoth, K. Soltesz, R. M. Murray, Nonlinear lateral control strategy for nonholonomic vehicles, in: American Control Conference, 2008, IEEE, 2008, pp. 3219–3224.

[26] J. Wit, C. D. Crane, D. Armstrong, Autonomous ground vehicle path tracking, Journal of Robotic Systems 21 (2004) 439–449.

[27] O. Amidi, C. E. Thorpe, Integrated mobile robot control, in: Mobile Robots V, volume 1388, International Society for Optics and Photonics, 1991, pp. 504–523.

[28] A. De Luca, G. Oriolo, C. Samson, Feedback control of a nonholonomic car-like robot, in: Robot motion planning and control, Springer, 1998, pp. 171–253.

[29] Y. H. Cho, J. Kim, Design of optimal four-wheel steering system, Vehicle System Dynamics 24 (1995) 661–682.

[30] R. S. Sharp, Driver steering control and a new perspective on car handling qualities, Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 219 (2005) 1041–1051.

[31] A. Lombard, X. Hao, A. Abbas-Turki, A. El Moudni, S. Galland, et al., Lateral control of an unmaned car using gnss positionning in the context of connected vehicles, Procedia Computer Science 98 (2016) 148–155.

[32] S. Dominguez, A. Ali, G. Garcia, P. Martinet, Comparison of lateral controllers for autonomous vehicle: Experimental results, in: 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2016, pp. 1418–1423.

[33] H. Durrant-Whyte, T. Bailey, Simultaneous localization and mapping: part i, IEEE robotics & automation magazine 13 (2006) 99–110.