



A novel safe and flexible control strategy based on target reaching for the navigation of urban vehicles[☆]



José Vilca^{*}, Lounis Adouane, Youcef Mezouar

Institut Pascal, UBP/IFMA – UMR CNRS 6602, Clermont-Ferrand, France

HIGHLIGHTS

- Novel control strategy based on static/dynamic target reaching for autonomous navigation.
- Control law is synthesized using a Lyapunov function based on a new set of variables.
- Interesting properties of the control law in terms of stability and flexibility.
- Sequential target assignment strategy allows performing safe navigation.
- Experiments using actual vehicles and several simulations show the advantages of the proposal.

ARTICLE INFO

Article history:

Received 21 May 2014
 Received in revised form
 20 January 2015
 Accepted 28 January 2015
 Available online 12 February 2015

Keywords:

Mobile robot navigation
 Control architecture
 Target reaching and following
 Lyapunov stability

ABSTRACT

This paper presents a complete framework for reactive and flexible autonomous vehicle navigation. A human driver reactively guides a vehicle to the final destination while performing a smooth trajectory and respecting the road boundaries. The objective of this paper is to achieve similar behavior in an unmanned ground vehicle to reach a static or dynamic target location. This is achieved by using a flexible control law based on a novel definition of control variables and Lyapunov synthesis. Furthermore, a target assignment strategy to enable vehicle navigation through successive waypoints in the environment is presented. An elementary waypoint selection method is also presented to perform safe and smooth trajectories. The asymptotic stability of the proposed control strategy is proved. In addition, an accurate estimation of the maximum error boundary, according to the controller parameters, is given. With this indicator, the vehicle navigation will be safe within a certain boundaries. Simulations and experiments are performed in different cases to demonstrate the flexibility, reliability and efficiency of the control strategy. Our proposal is compared with different navigation methods from the literature such as those based on trajectory following.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

Autonomous vehicle navigation is a complex problem of major interest to the research community. Systems capable of performing efficient and robust autonomous navigation are unquestionably useful in many robotic applications such as manufacturing technologies [1], urban transportation [2], assistance to disabled or elderly people [3] and surveillance [4]. Although much progress has been made, some specific technologies have to be improved for

effective application in real environments. This paper particularly focuses on the problem of autonomous navigation of vehicles in an urban environment (cf. Fig. 1).

Different strategies for autonomous navigation have been proposed in the literature [5–7]. The most popular approaches are based on following a pre-defined reference trajectory [8,9]. Most of the proposed control laws are dedicated to *trajectory tracking* (to track a time-parametrized reference) [10] and *path following* (to follow a path without explicit temporal references) [11]. These methods link the control to a reference trajectory which could be defined by a combination of path planning and trajectory generation techniques [12].

Typically, to obtain the reference path to be followed by the robot, arc-lines, B-splines or polynomial equations are used over points [13,14,5]. In [7] a feasible path is obtained using a polynomial curvature spiral. In [8], the trajectory generation method provides a smooth path considering the kinodynamic constraints of

[☆] Supported by the French National Research Agency (ANR) through the SafePlatoon project and LABEX IMobS3 (ANR-7107-LABX-716701).

^{*} Corresponding author.

E-mail addresses: Jose.Vilca@univ-bpclermont.fr (J. Vilca),
Lounis.Adouane@univ-bpclermont.fr (L. Adouane),
Youcef.Mezouar@univ-bpclermont.fr (Y. Mezouar).



Fig. 1. Autonomous navigation of an electric vehicle in an urban environment (Clermont-Ferrand, France).

the vehicle. In [15], straight line paths defined by the position and orientation of a single waypoint are considered. In this case, the orientation of the previous waypoint is not taken into account to simplify the implementation of the control law. Nevertheless, trajectory generation presents some drawbacks, such as the necessity of a specific planning method, the proof of guarantee of continuity between different segments of the trajectory and the complexity for replanning. A few works in the literature propose to use only specific set of way-points in the environment to lead the robot toward its final objective. In [16], the authors propose a navigation strategy via assigned static points for a unicycle robot. This strategy does not allow accurate navigation since the kinematic constraints of the robot (maximum velocity and steering), the orientation error and the velocity profile of the robot when it reaches the assigned point are not considered. Harmonic Potential Field (HPF) is used to guide an unmanned aerial vehicle (UAV) to a global waypoint with a position and a direction of arrival in [17]. The author proposes a virtual velocity field which allows to consider the UAV model. Each vector component of the field is treated as an intermediate waypoint with which the robot must comply with it in order to reach the global waypoint. Nonetheless, HPF requires a complex mathematical modeling for different shapes or dimension of the obstacles in the environment. In this paper, we propose a navigation strategy which avoids the pre-generation of any specific reference trajectory. Vehicle movements are obtained according to the proposed control law while considering vehicle kinematic constraints and sequential waypoints to reach (defined by its position, orientation and velocity). The vehicle can thus perform different movements between waypoints without the necessity of replanning any reference trajectory, and it can also add or change the location of the successive waypoints according to the environment configuration or to the task to achieve. Thus, this strategy allows flexible navigation while taking into account appropriate waypoints suitably placed in the environment.

Different control methods for trajectory tracking and path-following dedicated to wheeled mobile robots (unicycle, car-like robot, etc.) have been proposed in the literature [10,18] and [19]. In [9,10] and [20], nonlinear control laws for trajectory tracking are synthesized for a unicycle robot using Lyapunov stability analysis. The Lyapunov functions used in these studies are based only on distance and orientation errors. A trajectory tracking control for a farm vehicle, incorporating sliding in the kinematic model, is proposed in [18]. For the path-following problem, a control law for a tricycle robot is proposed in [19] and [11]. They are based on feedback linearization and chained form representation [21]. The path-following controller thus allows to make the lateral and longitudinal control of the vehicle independent along the reference trajectory. Furthermore, the path-following controller allows smoother convergence to the desired path than the trajectory tracking controller (designed for a time-parametrized trajectory)

[22]. The trajectory tracking controller allows to track the trajectory with a desired velocity profile, while the path-following controller acts only on the orientation to drive it along the path. Both, the path-following and trajectory tracking controllers require the pose of the closest point to the trajectory (w.r.t. robot configuration) and/or the value of curvature at this point (cf. Fig. 15) at each sample time [21] and [10]. Although there exist a multitude of techniques to compute these parameters, they can add an error in certain situations thereby influencing the control of the mobile robot negatively [22] and [18]. In this paper, a control law based on a novel definition of control variables and Lyapunov function is proposed. The synthesized control law can perform either static or dynamic target reaching using only its current pose and velocity. Using dynamic target reaching, trajectory following can also be performed. The control law exhibits good flexibility properties and it could be adapted to different autonomous robotic applications such as multi-robot formations (cf. Section 5.2).

It is not always required to follow a fixed path with high fidelity, specifically in open or low-constrained environments. We will demonstrate in this paper that only few waypoints will be sufficient to guarantee safe and flexible mobile robot navigation. A target assignment strategy is also proposed to perform autonomous navigation through pre-defined waypoints. We will also demonstrate that if we increase the number of these waypoints, the robot control performs as if we had applied common trajectory tracking control.

The rest of the paper is organized as follows: the following section presents the studied problem. Section 3 details the vehicle and target models, the proposed control law, its stability properties and setting the controller parameters. In Section 4 the proposed target assignment strategy and the waypoint selection approach are described. Simulation and experimental results are given in Section 5. A study with a few navigation methods from the literature is also provided. Finally, Section 6 provides a conclusion and perspectives for future studies.

2. Problem statement

An important challenge in the field of autonomous vehicles consists of ensuring safe and flexible navigation in a structured environment (cf. Figs. 1 and 2). In this work, safe navigation consists in not crossing over the road limits and bumping into obstacles while respecting the physical constraints of the vehicle. Flexible navigation consists in allowing different possible movements to achieve the task, while guaranteeing a smooth trajectory of the vehicle. The main idea of the proposed work is to guarantee both criteria simultaneously.

We consider the following scenario (cf. Fig. 2):

- The structured environment is a known road map where the roads have a specific width w_R .
- The vehicle model (kinematic) is known.
- The vehicle starts at the initial pose P_i and it has to reach the final pose P_f (in certain conditions, $P_i = P_f$).

As presented in Section 1 and according to the presented scenario, a safe reference path in static environment can be obtained by different algorithms such as Voronoï diagram [23], potential fields [24] or others [12]. In our case, specific key positions should be defined in the static environment, which we name *waypoints*. Their numbers and configurations in the environment are detailed in Section 4.2. Obviously, the waypoints can also be selected from a pre-defined trajectory if available. Consequently, the navigation problem is simplified to a waypoint tracking problem, i.e., the vehicle is guided by the waypoints (cf. Fig. 8) instead of following a specific fixed path. The vehicle has thus to reach each waypoint with a defined position, orientation and velocity while satisfying

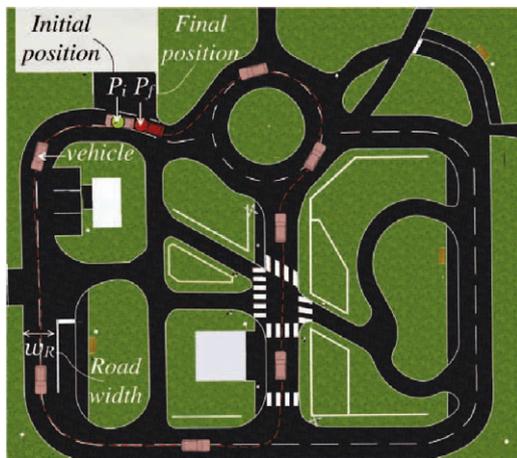


Fig. 2. Nominal scenario with a road map and the task to achieve by the vehicle in its environment.

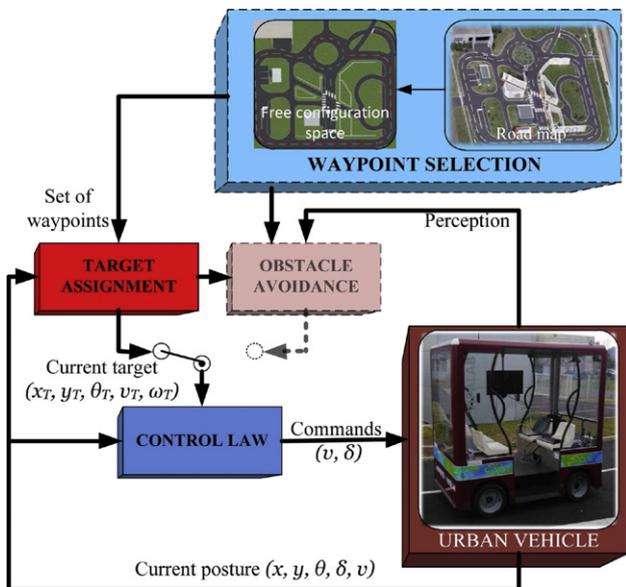


Fig. 3. Overview of the proposed navigation strategy.

distance and orientation error limits (E_{dis} and E_{angle} respectively) to perform safe navigation (cf. Section 3.4).

The proposed navigation strategy can be extended easily in order to deal with dynamic environments, notably using the limit-cycle approach [6] and [20]. This obstacle avoidance approach allows to modify locally the movement of the robot to avoid dangerous static or dynamic obstacles and to come back to its initial plan. Nevertheless, it is not the main focus of this paper.

A single waypoint, defined as a set-point by its position, orientation and velocity, is called a *target*. The control law to reach this target is designed to generate smooth vehicle navigation. Moreover, if the successive waypoints are close to each other then the vehicle tends to perform trajectory tracking navigation (cf. Sections 4 and 5.1.1). Fig. 3 gives an overview of the proposed navigation strategy. In this paper, waypoint selection (dashed blue block in Fig. 3) is simplified (cf. Section 4.2) to focus on navigation strategy (Control law and Target assignment blocks, cf. Section 3 and Section 4.1 respectively).

3. Target-reaching control

This section gives the details of the proposed control law, the proof of its stability and the tuning of controller parameters.

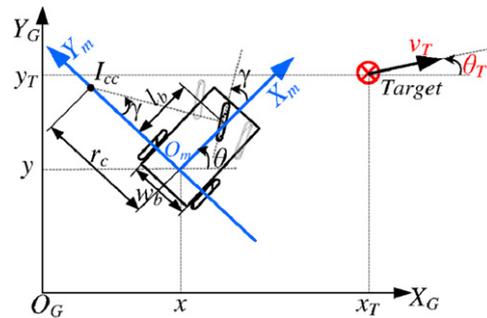


Fig. 4. Vehicle and target configuration in Global ($X_G Y_G$) and Local ($X_m Y_m$) reference frames.

3.1. Vehicle and target modeling

Our experimental vehicle (cf. Fig. 19) is devoted to urban transportation. The vehicle moves on asphalt with a low velocity (less than 3 m/s). Therefore, it appears quite natural to rely on a kinematic model, and to assume pure rolling and non-slipping at wheel ground contact. In such cases, vehicle modeling is commonly performed, for instance relying on a tricycle model as shown below (cf. Fig. 4):

$$\begin{cases} \dot{x} = v \cos(\theta) \\ \dot{y} = v \sin(\theta) \\ \dot{\theta} = v \tan(\gamma) / l_b \end{cases} \quad (1)$$

where O_G and O_m are respectively the origin of the global and local reference frames, (x, y, θ) is the pose (configuration state) at vehicle point O_m , γ is the orientation of the vehicle front wheels, v is the linear velocity at vehicle point O_m . l_b and w_b are respectively the wheelbase and the track width of the vehicle (cf. Fig. 4). I_{cc} is the instantaneous center of curvature of the vehicle trajectory. The radius of curvature r_c is given by:

$$r_c = l_b / \tan(\gamma) \quad (2)$$

and $c_c = 1/r_c$ is the curvature of the vehicle trajectory.

Let us consider a dynamic target modeled as a point with non-holonomic constraints (cf. Fig. 4). This model allows us to use the general model of robot motion and also to simplify the controller equations. Its kinematic characteristics are given by:

$$\begin{cases} \dot{x}_T = v_T \cos(\theta_T) \\ \dot{y}_T = v_T \sin(\theta_T) \\ \dot{\theta}_T = \omega_T \end{cases} \quad (3)$$

where v_T and ω_T are respectively the linear and angular velocities of the target. The radius of curvature is computed by $r_{c_T} = v_T / \omega_T$. An important consideration for target reaching is $v_T \leq v_{max}$ and $r_{c_T} \geq r_{c_{min}}$, where v_{max} and $r_{c_{min}}$ are respectively the maximum linear velocity and the minimum radius of curvature of the vehicle, given by $r_{c_{min}} = l_b / \tan(\gamma_{max})$. For static target reaching (*point stabilization*, i.e., to reach a specific point with a given orientation), ω_T is considered equal to zero and v_T is not necessarily equal to zero; v_T is then considered as a desired velocity value for the vehicle when it reaches the desired target (x_T, y_T, θ_T) .

3.2. Control law

Before presenting the control law, let us introduce the control variables of the system (cf. Fig. 5). The errors with respect to the local frame ($X_m Y_m$) of the vehicle (e_x, e_y, e_θ) between the desired pose (x_T, y_T, θ_T) and the current vehicle pose (x, y, θ) are given by:

$$\begin{cases} e_x = \cos(\theta)(x_T - x) + \sin(\theta)(y_T - y) \\ e_y = -\sin(\theta)(x_T - x) + \cos(\theta)(y_T - y) \\ e_\theta = \theta_T - \theta. \end{cases} \quad (4)$$

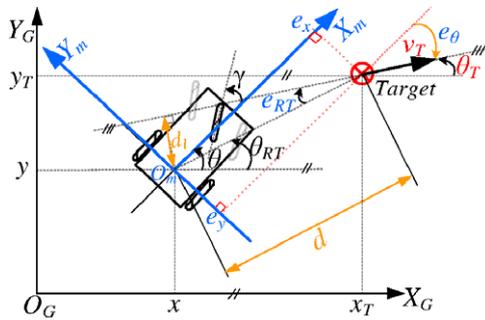


Fig. 5. Control variables according to Lyapunov synthesis.

A new error function e_{RT} is added to the canonical error system (4) (cf. Fig. 5). Let us first define the distance d and the angle θ_{RT} between the target and the vehicle position as (cf. Fig. 5):

$$d = \sqrt{(x_T - x)^2 + (y_T - y)^2} \quad (5)$$

$$\begin{cases} \theta_{RT} = \arctan((y_T - y)/(x_T - x)) & \text{if } d \geq \xi \\ \theta_{RT} = \theta_T & \text{if } d < \xi \end{cases} \quad (6)$$

where ξ is a small positive value ($\xi \approx 0$).

The error e_{RT} is related to the vehicle position (x, y) with respect to the target orientation (cf. Fig. 5). It is defined as:

$$e_{RT} = \theta_T - \theta_{RT}. \quad (7)$$

Furthermore, e_{RT} can be written as a function of e_x , e_y and e_θ as:

$$\begin{aligned} \tan(e_{RT}) &= \tan(e_\theta - (\theta_{RT} - \theta)) \\ &= \frac{\tan(e_\theta) - e_y/e_x}{1 + \tan(e_\theta)e_y/e_x} \\ &= \frac{e_x \tan(e_\theta) - e_y}{e_x + \tan(e_\theta)e_y} \end{aligned} \quad (8)$$

where $\tan(\theta_{RT} - \theta) = e_y/e_x$ (cf. Fig. 5). Hence, e_{RT} allows to consider an additional orientation error w.r.t. e_x , e_y and e_θ , e.g., when $e_\theta = 0$ then $e_{RT} = -e_y/e_x$. The stabilization of this error allows to decrease the lateral distance d_l to zero (18) (cf. Fig. 5), and to always have the robot in the wake of the target.

The derivatives of the errors (4) and (7) can be obtained using (1), (3), (5) and (6):

$$\begin{aligned} \dot{e}_x &= \cos(\theta)(\dot{x}_T - \dot{x}) + \sin(\theta)(\dot{y}_T - \dot{y}) \\ &\quad - \sin(\theta)(x_T - x)\dot{\theta} + \cos(\theta)(y_T - y)\dot{\theta} \\ &= -v + e_y\dot{\theta} + v_T[\cos(\theta_T)\cos(\theta) + \sin(\theta_T)\sin(\theta)] \\ &= -v + e_yv \tan(\gamma)/l_b + v_T \cos(e_\theta) \end{aligned} \quad (9)$$

$$\begin{aligned} \dot{e}_y &= -\sin(\theta)(\dot{x}_T - \dot{x}) + \cos(\theta)(\dot{y}_T - \dot{y}) \\ &\quad - \cos(\theta)(x_T - x)\dot{\theta} - \sin(\theta)(y_T - y)\dot{\theta} \\ &= -e_x\dot{\theta} - v_T \cos(\theta_T)\sin(\theta) + v_T \sin(\theta_T)\cos(\theta) \\ &= -e_xv \tan(\gamma)/l_b + v_T \sin(e_\theta) \end{aligned} \quad (10)$$

$$\begin{aligned} \dot{e}_\theta &= \dot{\theta}_T - \dot{\theta} \\ &= \omega_T - \omega \\ &= \frac{v_T}{r_{cT}} - v \tan(\gamma)/l_b \end{aligned} \quad (11)$$

$$\begin{aligned} \dot{e}_{RT} &= \dot{\theta}_T - \dot{\theta}_{RT} \\ &= \frac{v_T}{r_{cT}} - \frac{d}{dt} \left[\arctan\left(\frac{y_T - y}{x_T - x}\right) \right] \\ &= \frac{v_T}{r_{cT}} - v_T \frac{\sin(\theta_T)(x_T - x) - \cos(\theta_T)(y_T - y)}{d^2} \\ &\quad - \frac{-v \sin(\theta)(x_T - x) + v \cos(\theta)(y_T - y)}{d^2} \\ &= \frac{v_T}{r_{cT}} - \frac{v_T e_x \sin(e_\theta)}{d^2} + \frac{v_T e_y \cos(e_\theta)}{d^2} - \frac{e_y v}{d^2}. \end{aligned} \quad (12)$$

The control law to reach a target (static or dynamic) is obtained using the Lyapunov stability analysis framework. The desired linear velocity v and the front wheel orientation γ of the vehicle which lead the errors (e_x, e_y, e_θ) to converge to zero are chosen such that:

$$v = v_T \cos(e_\theta) + v_b \quad (13)$$

$$\gamma = \arctan(l_b c_c) \quad (14)$$

where v_b and c_c are given by:

$$v_b = K_x [K_d e_x + K_l d \sin(e_{RT}) \sin(e_\theta) + K_o \sin(e_\theta) c_c] \quad (15)$$

$$\begin{aligned} c_c &= \frac{1}{r_{cT} \cos(e_\theta)} + \frac{d^2 K_l \sin(e_{RT}) \cos(e_{RT})}{r_{cT} K_o \sin(e_\theta) \cos(e_\theta)} + K_\theta \tan(e_\theta) \\ &\quad + \frac{K_d e_y - K_l d \sin(e_{RT}) \cos(e_\theta)}{K_o \cos(e_\theta)} + \frac{K_{RT} \sin^2(e_{RT})}{\sin(e_\theta) \cos(e_\theta)} \end{aligned} \quad (16)$$

$\mathbf{K} = (K_d, K_l, K_o, K_x, K_\theta, K_{RT})$ is a vector of positive constants defined by the designer (cf. Section 3.4).

3.3. Stability analysis

In this section, the stability of the error system (9)–(12) is analyzed.

Assumption 1. The subsequent development is based on the assumption that the initial values of e_{RT} and e_θ satisfied:

$$e_{RT} \in]-\pi/2, \pi/2[\quad \text{and} \quad e_\theta \in]-\pi/2, \pi/2[\quad (17)$$

These conditions guarantee that the target is ahead of the vehicle with respect to its orientation.

Theorem 1. The control law given by (13) and (14) ensures that the differential system (9)–(12) is asymptotically stable according to Lyapunov-based analysis and if the Assumption 1 is satisfied [25].

Proof. The stability of the system is analyzed using the Lyapunov method [25]. The proposed Lyapunov function V , given by (18), is a function of three parameters which depend on the distance d between the target and vehicle positions, the distance d_l from the vehicle to the target line (line which passes through the target position with an orientation equal to the target orientation; this term is related to the Line of Flight and Sight of the target [26]), and the orientation error e_θ between the vehicle and the target (cf. Fig. 5).

The candidate Lyapunov function V is a positive-definite function [25] when considering (17). It is given by:

$$\begin{aligned} V &= \frac{1}{2} K_d d^2 + \frac{1}{2} K_l d_l^2 + K_o [1 - \cos(e_\theta)] \\ &= \frac{1}{2} K_d d^2 + \frac{1}{2} K_l d^2 \sin^2(e_{RT}) + K_o [1 - \cos(e_\theta)]. \end{aligned} \quad (18)$$

It can be written with respect to e_x, e_y as follows:

$$V = \frac{1}{2} (e_x^2 + e_y^2) [K_d + K_l \sin^2(e_{RT})] + K_o [1 - \cos(e_\theta)]. \quad (19)$$

To guarantee the stability of the system, \dot{V} must be negative-definite [25]. By taking the derivative of (19) and using (9)–(14), \dot{V} can be written:

$$\begin{aligned} \dot{V} &= (e_x \dot{e}_x + e_y \dot{e}_y) [K_d + K_l \sin^2(e_{RT})] \\ &\quad + K_l d^2 \sin(e_{RT}) \cos(e_{RT}) \dot{e}_{RT} + K_o \sin(e_\theta) \dot{e}_\theta \\ &= (e_x [e_y v c_c - v + v_T \cos(e_\theta)] + e_y [v_T \sin(e_\theta) - e_x v c_c]) \\ &\quad \cdot [K_d + K_l \sin^2(e_{RT})] + K_l d^2 \sin(e_{RT}) \cos(e_{RT}) \end{aligned}$$

$$\begin{aligned} & \cdot \left[\frac{v_T}{r_{cT}} - \frac{v_T e_x \sin(e_\theta)}{d^2} + \frac{v_T e_y \cos(e_\theta)}{d^2} - \frac{e_y v}{d^2} \right] \\ & + K_o \sin(e_\theta) \left(\frac{v_T}{r_{cT}} - v_{cC} \right). \end{aligned} \quad (20)$$

Substituting (13) in (20)

$$\begin{aligned} \dot{V} &= [-e_x v_b + v_T e_y \sin(e_\theta)] [K_d + K_l \sin^2(e_{RT})] \\ & + K_l \sin(e_{RT}) \cos(e_{RT}) \left[d^2 \frac{v_T}{r_{cT}} - v_T e_x \sin(e_\theta) - e_y v_b \right] \\ & + K_o \sin(e_\theta) \left[\frac{v_T}{r_{cT}} - v_T \cos(e_\theta) c_c - v_b c_c \right] \\ & = [e_y (K_d + K_l \sin^2(e_{RT})) - e_x K_l \sin(e_{RT}) \cos(e_{RT})] \\ & \cdot v_T \sin(e_\theta) + \frac{v_T}{r_{cT}} [d^2 K_l \sin(e_{RT}) \cos(e_{RT}) + K_o \sin(e_\theta)] \\ & - v_T K_o \sin(e_\theta) \cos(e_\theta) c_c - v_b [e_x (K_d + K_l \sin^2(e_{RT}))] \\ & - v_b [e_y K_l \sin(e_{RT}) \cos(e_{RT}) + K_o \sin(e_\theta) c_c]. \end{aligned} \quad (21)$$

Using (8) in the first and last terms of (21) and factorizing the common terms, it holds that:

$$\begin{aligned} \dot{V} &= v_T \sin(e_\theta) [K_d e_y - K_l d \sin(e_{RT}) \cos(e_\theta)] \\ & + \frac{v_T}{r_{cT}} [d^2 K_l \sin(e_{RT}) \cos(e_{RT}) + K_o \sin(e_\theta)] \\ & - v_b [K_d e_x + K_l d \sin(e_{RT}) \sin(e_\theta) + K_o \sin(e_\theta) c_c] \\ & - v_T K_o \sin(e_\theta) \cos(e_\theta) c_c. \end{aligned} \quad (22)$$

Finally, using (15) and (16) in (22), we obtain:

$$\begin{aligned} \dot{V} &= -K_x [K_d e_x + K_l d \sin(e_{RT}) \sin(e_\theta) + K_o \sin(e_\theta) c_c]^2 \\ & - v_T K_o K_\theta \sin^2(e_\theta) - v_T K_o K_{RT} \sin^2(e_{RT}) \leq 0. \end{aligned} \quad (23)$$

Eq. (23) shows that the system is stable while the initial conditions (17) are satisfied. To ensure the asymptotic stability of the proposed control law, \dot{V} has to be a negative-definite function. Let us consider two cases, one where $\dot{V} = 0$ with $v_T > \xi$ and another with $v_T = \xi$, where ξ is a constant value ($\xi \approx 0$). Firstly, when $v_T > \xi$ and using the initial assumption $\mathbf{K} > 0$, it is straightforward to show that e_x, e_θ, e_{RT} are equal to zero to satisfy (23); then according to (7), (6) and (17), d is equal to zero ($e_y = 0$). Hence, \dot{V} is equal to zero when $v_T > \xi$, only if $(e_x, e_y, e_\theta) = (0, 0, 0)$.

Secondly, let us consider the case where $v_T = \xi$. The initial assumption is identical. Therefore, the second and third terms of (23) are equal to zero when $v_T = \xi$. Additionally, when $v_T = \xi$ (static case) then we can consider $r_{cT} \rightarrow \infty$ (cf. Section 3.1); consequently the first term of \dot{V} is equal to zero when:

$$K_d e_x + K_l d \sin(e_{RT}) \sin(e_\theta) + K_o \sin(e_\theta) c_c = 0. \quad (24)$$

Replacing (16) with $r_{cT} \rightarrow \infty$ in (24), the following expression is obtained:

$$\begin{aligned} 0 &= K_d e_x + K_l d \sin(e_{RT}) \sin(e_\theta) \\ & + \tan(e_\theta) [K_d e_y - K_l d \sin(e_{RT}) \cos(e_\theta)] \\ & + K_o \sin(e_\theta) \left[K_\theta \tan(e_\theta) + \frac{K_{RT} \sin^2(e_{RT})}{\sin(e_\theta) \cos(e_\theta)} \right] \\ & = K_d [e_x + e_y \tan(e_\theta)] + K_o K_\theta \frac{\sin^2(e_\theta)}{\cos(e_\theta)} + K_o K_{RT} \frac{\sin^2(e_{RT})}{\cos(e_\theta)}. \end{aligned} \quad (25)$$

Using (8) in (25), we finally obtain:

$$K_d d \frac{\cos(e_{RT})}{\cos(e_\theta)} + K_o K_\theta \frac{\sin^2(e_\theta)}{\cos(e_\theta)} + K_o K_{RT} \frac{\sin^2(e_{RT})}{\cos(e_\theta)} = 0. \quad (26)$$

Eq. (26) exhibits quadratic terms. Consequently, considering the initial conditions (17), $\cos(e_{RT})$ and $\cos(e_\theta)$ are greater than zero. Therefore, all the terms of (26) are positive and they must be equal to zero (i.e., $d, e_\theta, e_{RT} = 0$, and if $d = 0$ then $e_x, e_y = 0$). Hence, from (26), \dot{V} is equal to zero when $v_T = \xi$ and $r_{cT} \rightarrow \infty$, only if $(e_x, e_y, e_\theta) = (0, 0, 0)$.

In conclusion, if $v_T > \xi$ or $v_T = \xi$, V is always strictly positive and \dot{V} is always strictly negative while $(e_x, e_y, e_\theta) \neq (0, 0, 0)$. Therefore, the system is asymptotically stable while the initial vehicle conditions (17) are satisfied. \square

3.4. Safe target reaching

Synthesis of the proposed control law using a Lyapunov function enables us to confirm its asymptotic stability. Nonetheless, it does not allow us to obtain immediately the error values when the robot is in the immediate vicinity of the target to reach. The aim of this subsection is to determine a relation between the upper bound of the errors d and e_θ , denoted E_{dis} and E_{angle} (cf. Figs. 5 and 8) and the controller parameters \mathbf{K} . Indeed, according to these dynamics and considering that the vehicle and target localization are always accurate, the values of \mathbf{K} enable the vehicle constraints (maximum velocity, acceleration, steering, etc.) to be satisfied. Our analysis consists in determining the minimum d_i (cf. Fig. 6) which allows to satisfy at the same time, the vehicle physical constraints and the errors (d and e_θ), when the vehicle reaches the target, which must be less or equal to the pre-defined bounds E_{dis} and E_{angle} respectively. Fig. 7 shows the global scheme that we will apply, discussed subsequently, to obtain the minimal initial distance d_i while knowing \mathbf{K} and $e_{\theta 0}$ which will satisfy E_{dis} and E_{angle} .

The subsequent analysis considers a static target ($\dot{x}_T = \dot{y}_T = 0$ and $r_{cT} \rightarrow \infty$) and a limit vehicle configuration, $e_{RT} \approx 0$ and $e_\theta \approx -\pi/2$, i.e., the vehicle has the maximum admissible orientation error with respect to the target and the convergence will be the slowest (cf. Figs. 6 and 5). This analysis will allow reference values for the controller parameters \mathbf{K} to be obtained in the case of a dynamic target, where the distance d between the vehicle and target change more slowly than in the static target case. The control law with the designed parameters \mathbf{K} will have thus more time to converge the system errors to zero (4).

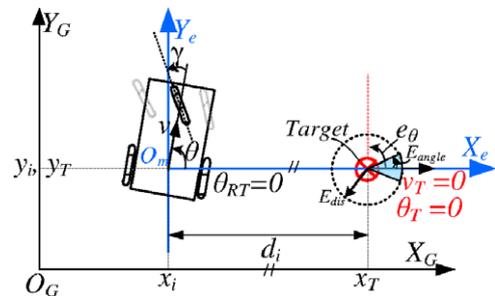


Fig. 6. Limit vehicle configuration for tuning controller parameters.

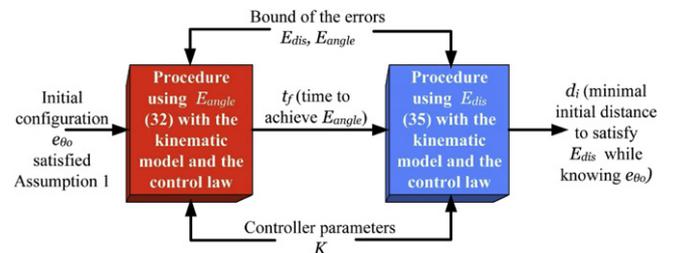


Fig. 7. Block diagram of the analysis.

To simplify controller analysis, the errors in orientation e_θ and distance d are dealt separately. Firstly, the orientation error is computed considering enough initial distance d_i and $e_{RT} \approx 0$. These considerations enable us to estimate the minimum time to attain effectively E_{angle} (cf. Figs. 6 and 8). Using (13)–(15) in (11), \dot{e}_θ can be written as:

$$\begin{aligned} \dot{e}_\theta &= \frac{v_T}{r_{c_T}} - v \tan(\gamma)/l_b \quad (27) \\ &= - \left[\frac{K_x K_d d}{\cos(e_\theta)} + K_x K_o K_\theta \frac{\sin^2(e_\theta)}{\cos(e_\theta)} \right] \\ &\quad \cdot \left[\frac{K_d e_y}{K_o \cos(e_\theta)} + K_\theta \tan(e_\theta) \right] \\ &= - \frac{K_x}{K_o \cos^2(e_\theta)} [K_d d + K_o K_\theta \sin^2(e_\theta)] \\ &\quad \cdot [K_d d \sin(e_\theta) + K_o K_\theta \sin(e_\theta)] \\ &= - \frac{K_x (K_d d + K_o K_\theta)}{K_o \cos^2(e_\theta)} (K_d d + K_o K_\theta \sin^2(e_\theta)) \sin(e_\theta). \quad (28) \end{aligned}$$

To solve the differential equations (28), let us introduce the following notations $A = K_d d$, $B = K_o K_\theta$ and $C = \sqrt{(A/B + 1)}$. Hence, the solution of (28) has the following form:

$$\ln \left[\tan \left(\frac{e_\theta}{2} \right) \left(\frac{C + \cos(e_\theta)}{C - \cos(e_\theta)} \right)^{C/2} \right] \Big|_{e_{\theta_0}}^{e_\theta} = - \frac{K_x A B}{K_o} C^2 t \Big|_0^t. \quad (29)$$

The objective is to compute the time t_f necessary to obtain e_θ value which is less than a given orientation error threshold E_{angle} . Therefore, (29) can be written as:

$$e_\theta = 2f_\theta(t, \mathbf{K}, e_{\theta_0}) \quad (30)$$

where:

$$f_\theta = \tan \left(\frac{e_{\theta_0}}{2} \right) \left[\frac{(C + \cos(e_{\theta_0}))(C - 1)}{(C - \cos(e_{\theta_0}))(C + 1)} \right]^{C/2} e^{-\frac{K_x A B}{K_o} C^2 t_f}. \quad (31)$$

Indeed, for a specific parameter value \mathbf{K} , if the controller can reach the target (with e_θ less than E_{angle} for a time t_f given according to (30)), then for the same \mathbf{K} parameters, e_θ must be less than E_{angle} for a time $t > t_f$.

From (30), the orientation error e_θ depends on the initial orientation error e_{θ_0} and the controller parameters (K_d , K_x , K_o , K_θ). Moreover, K_d can be chosen as a function of the initial distance as $K_d = 1/d_i$, which allows us to obtain $A \leq 1$. The controller parameters are designed to obtain a fast convergence rate, given by

$$t_f = f_\theta^{-1}(E_{angle}, \mathbf{K}, e_{\theta_0}) \quad (32)$$

while taking into account vehicle constraints such as maximum vehicle velocity v_{max} and minimum radius of curvature $r_{c_{min}}$.

Secondly, the time t_f enables the maximum distance covered by the vehicle to be ascertained. With this aim, we can set $e_{RT} = 0$ and $e_\theta = 0$ (straight line to the target). Hence, (15) can be written as

$$v_b = K_x K_d d. \quad (33)$$

Introducing (9), (10), (13) and (33) in the derivative of the distance, we obtain:

$$\begin{aligned} \dot{d} &= (e_x \dot{e}_x + e_y \dot{e}_y)/d \\ &= -e_x v_b \\ &= -K_x K_d d \\ \int_{d_i}^d 1/d \partial d &= \int_0^t -K_x K_d \partial t \Rightarrow d = d_i e^{-K_x K_d t}. \quad (34) \end{aligned}$$

From (34), the convergence of the distance depends on K_x and K_d . Therefore, they are designed to obtain d_i such that:

$$d_i = E_{dis} e^{K_x K_d t_f} \quad (35)$$

while taking into account vehicle constraints v_{max} and $r_{c_{min}}$. Simulations (cf. Section 5.1.1) will validate the above approach. Moreover, (30) and (35) show the relations between initial configuration, controller parameters and error of reaching the target. Therefore, for certain initial configuration and defined error bounds, the controller parameters can be obtained.

4. Navigation strategy based on sequential target assignment

The computation of a time-parametrized path while taking into account different vehicle constraints and environment characteristics is time-consuming [27,28]. Different algorithms that compute a safe path (without temporal reference) [2,12] require less computational time but provide trajectories which do not ensure the safe navigation of the vehicle. In the previous section, we showed that the proposed control law guarantees that the static or dynamic target will be reached. In the following, a strategy to use a finite set of targets (waypoints) to define the desired vehicle route is presented.

4.1. Sequential target assignment

The proposed strategy uses a sequence of sorted waypoints disposed in the environment. The aim of this sequence is to guarantee safe (cf. Sections 3.4 and 4.2) and flexible navigation. An elementary method to select adequate waypoints (target set-points $(x_{T_j}, y_{T_j}, \theta_{T_j}, v_{T_j})$) to perform a safe vehicle navigation in structured environment is presented in Section 4.2. Each waypoint $T_j(x_{T_j}, y_{T_j})$ is defined by the following parameters: D_j the euclidean distance between the last waypoint $T_{j-1}(x_{T_{j-1}}, y_{T_{j-1}})$ and the current waypoint T_j ; θ_{T_j} is the orientation between T_j and $T_{j+1}(x_{T_{j+1}}, y_{T_{j+1}})$:

$$\theta_{T_j} = \arctan((y_{T_{j+1}} - y_{T_j}) / (x_{T_{j+1}} - x_{T_j})). \quad (36)$$

The safety between the specified waypoints is guaranteed by imposing E_{dist} and E_{angle} (cf. Section 3.4). This allows to guide the vehicle when it reaches the target T_j (cf. Fig. 8) in the appropriate conditions to reach the next target T_{j+1} while also guaranteeing safe navigation between T_j and T_{j+1} .

The strategy to assign the target point is shown in Algorithm 1. The parameters of the control law (cf. Section 3.2) enable the vehicle to reach the next target point (cf. Section 5) while ensuring that the vehicle trajectory is always within the road limits (cf. Fig. 8). The error conditions (E_d and E_{angle}) are used to allow switching to the next target, when the vehicle position enters a circle with a radius equal to E_{dis} and center (x_{T_j}, y_{T_j}) . The current target is updated with the following waypoint in the list and the vehicle starts the movement to reach the new target. If the vehicle does not satisfy the error conditions then the perpendicular line L_j

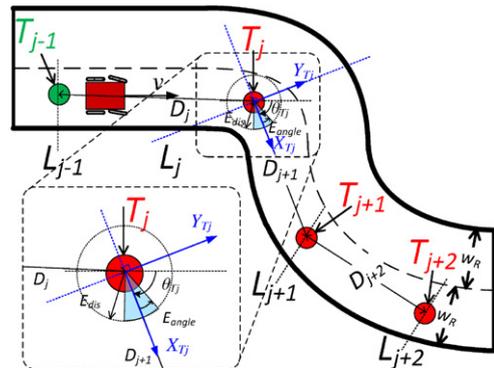


Fig. 8. Description of waypoints and target assignment.

Algorithm 1 Sequential target assignment**Require:** Vehicle pose, current target T_j and a set of N sorted waypoints**Ensure:** Switch between target set-points

- 1: **if** ($(d \leq E_{dis}$ **and** $e_\theta \leq E_{angle})$ **or** $(x^{T_j} \geq 0)$)
 $\{x^{T_j}$ is the coordinate of the vehicle in the local Target frame $X_{T_j}Y_{T_j}$ (cf. Fig. 8) **}** **then**
- 2: Switch from the current target T_j to the next sequential waypoint T_{j+1}
- 3: **end if**

to the line which connects T_j and T_{j+1} is used to switch to the next target when the vehicle crosses this line (cf. Fig. 8).

It can be noticed that the proposed control law, (13) and (14), is well suited to the navigation task. It allows to reach each target with an assigned velocity (13) (which can be different from zero contrary to [16]). The obtained robot movements become smoother and thus appropriate for public transportation.

4.2. Waypoint selection method

Waypoint selection consists in obtaining the minimum number of points (waypoints) on the road to be successively reached by the vehicle to perform safe navigation. These waypoints are selected considering a safe position on the road (as far as possible from the road limits) and the reliability of the obtained vehicle trajectory (smooth changes between the successive points).

In this paper, the proposed waypoint selection (cf. Algorithm 2) is maximally simplified to focus only on the navigation strategy, i.e., target-reaching control (cf. Section 3) and sequential target assignment (cf. Section 4.1). Future studies will focus on the optimization of the waypoints selection in any kind of environment. Therefore, to provide a complete framework to achieve the navigation strategy, in this paper we will consider in the subsequent discussion, the existence of a defined trajectory (infinite number of points); the aim of the method is to select an appropriate number of points (waypoints).

The reference path can be obtained by different algorithms [12] or by using a recorded vehicle trajectory. Different criteria can be considered to obtain the minimum number of straight lines that closely fit the reference path. Criteria such as the euclidean or curvilinear distance, orientation or radius of curvature between waypoints can be used to fix the desired waypoints on the path. The discretized reference path \mathbf{r} is composed of sorted position $r_i = (x_{r_i}, y_{r_i})$ and its tangent orientation θ_{r_i} . The minimum number of straight line segments over the defined path is then computed while considering a constant threshold $\Delta_{\alpha_{max}}$ for the orientation variation of the path Δ_α (cf. Algorithm 2).

Fig. 9 shows one vehicle trajectory and the obtained waypoints using Algorithm 2 with $\Delta_{\alpha_{max}} = 5^\circ$, 15° and 30° respectively. Obviously, the switch between waypoints is smoother with a small value of $\Delta_{\alpha_{max}}$.

Algorithm 2 Waypoint selection based on existing reference path**Require:** Reference path $\mathbf{r} = (\mathbf{x}_r, \mathbf{y}_r)$ and $\Delta_{\alpha_{max}} \in \mathbb{R}^+$ **Ensure:** Set of waypoints S_p

- 1: Init $j = 0$, $r_{w_j} = r_0$ (initial position of \mathbf{r}) and $\theta_{w_j} = \theta_{r_0}$ (tangent of the point along trajectory \mathbf{r})
- 2: **for** $r_i \in \mathbf{r}$ (sorted set of trajectory points) **do**
- 3: Compute $\Delta_\alpha = |\theta_{r_i} - \theta_{w_j}|$
- 4: **if** $\Delta_\alpha \geq \Delta_{\alpha_{max}}$ **then**
- 5: $j = j + 1$
- 6: Set $r_{w_j} = r_i$ and $\theta_{w_j} = \theta_{r_i}$
- 7: Add $w_j(r_{w_j}, \theta_{w_j})$ to S_p
- 8: **end if**
- 9: **end for**

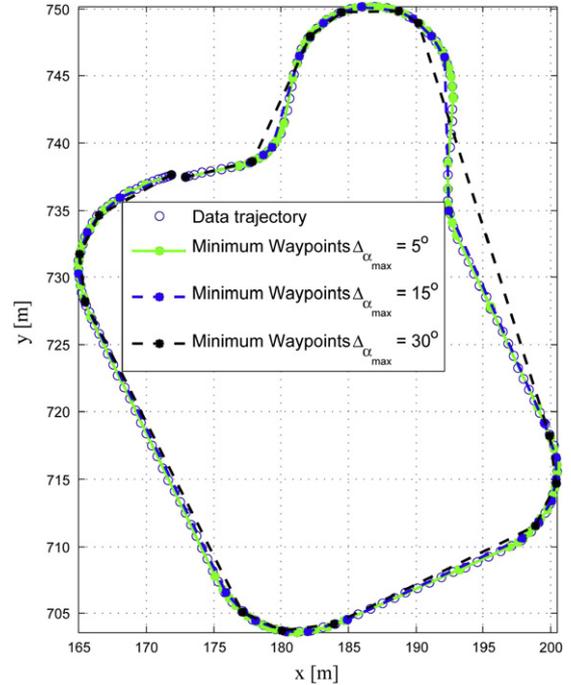


Fig. 9. Example of waypoint selection based on a reference path and Algorithm 2.

5. Proposal validation

This section presents a set of experiments to demonstrate the efficiency of the control law for target reaching and autonomous navigation in a structured environment. Section 5.1 provides simulation results to show the validity of our proposal. Section 5.2 discusses experimental results applied to an urban electric vehicle.

5.1. Simulation results

In these simulations different aspects, such as the stability of the control law to reach a static target, the flexibility of the navigation strategy based on target assignment and the performance of the proposed control law compared with other approaches found in the literature, will be presented. The physical parameters of the urban vehicle VIPALAB (cf. Fig. 19) modeled using tricycle kinematics (1) were considered.

5.1.1. Target reaching

The first simulation shows the performance (safety, smoothness and convergence) of the control law to reach a desired final configuration (pose and velocity). For each simulation, the vehicle starts at the same position but with different initial orientations.

This simulation validates the analysis presented in Section 3.4, where the minimum d_i , obtained for a limit vehicle configuration $e_\theta \approx \pi/2$, allows to satisfy the bound of the errors for other initial configuration. The desired final configuration is $(x_T, y_T, \theta_T) \equiv (15, 4, 0^\circ)$ and $v_T = 1$ m/s. The controller parameters \mathbf{K} are designed considering $E_{dist} \leq 0.1$ m and $E_{angle} \leq 5^\circ$ using (30) and (35). Hence, it is designed such that $\mathbf{K} = (1/d_i, 0.6, 10, 0.1, 0.3, 0.01)$, $d_i = 10.6$ m is the minimum initial euclidean distance to the target. These parameters were chosen to obtain a safe and smooth trajectory, faster response ($t_f \approx 10.5$ s (32)) and velocity value within the limits of the vehicle (cf. Section 3.4), which are $v_{max} = 1.5$ m/s and the minimum radius of curvature $r_{cmin} = 3.8$ m. Fig. 10 shows the trajectory of the vehicle for different initial configurations. The orientation errors are shown in Fig. 11. The Lyapunov function values are shown in Fig. 12.

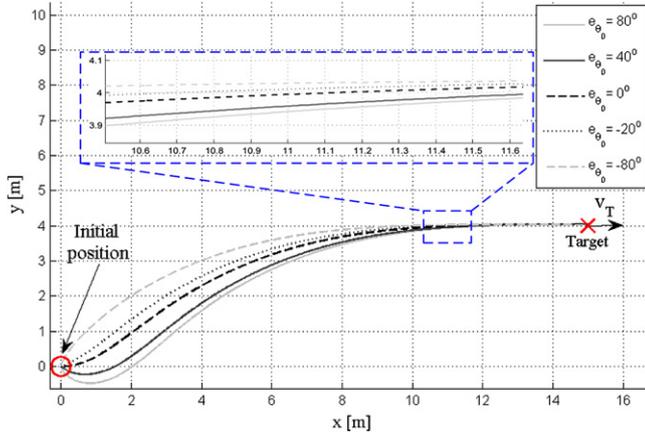


Fig. 10. Trajectories of the vehicle for several initial orientations.

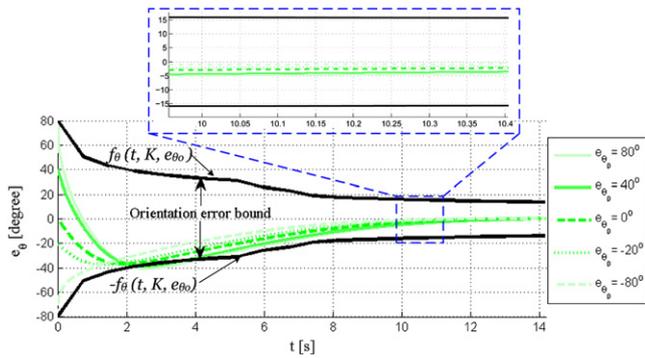


Fig. 11. Orientation errors (e_θ) for several initial vehicle orientations.

Fig. 10 shows that the convergence of the system depends on the initial orientation error. Fig. 11 shows that the system errors are bounded (30) (black line) and converge to zero (cf. Section 3.4). Furthermore, the Lyapunov function shows asymptotic stability (cf. Fig. 12). Fig. 12(a) shows the three terms of the Lyapunov function (18) where the first term is $0.5K_d d^2$, second term is $0.5K_l d_l^2$ and the third term is $K_o[1 - \cos(e_\theta)]$. These figures show that the vehicle satisfies the constraints (velocity, acceleration and steering) presented in Section 3.4.

Moreover, the case of successive target reaching through a set of waypoints is analyzed. Two set of waypoints selected from a reference trajectory are used, one set has a distance between waypoints equal to 2 m and the other equal to 4 m (cf. Fig. 13). Figs. 13 and 14 respectively, show the vehicle trajectories and lateral and angular errors w.r.t. the reference trajectory for two set of waypoints. It can be noted that the obtained vehicle trajectories are close enough from the reference trajectory; and as expected, the lateral and angular errors are smaller when the fixed distance between the waypoints decreases. Therefore, the proposed navigation strategy and control law permits to the vehicle to perform accurate trajectory tracking behavior if the waypoints are close enough.

5.1.2. Comparison study

In this simulation, two common approaches to follow a dynamic target by a tricycle are briefly presented and compared with the proposed control law. The comparison is focused on the position and orientation errors and the convergence time. In the sequel, the dynamic target is assumed to be another identical urban vehicle. These two strategies are briefly described below:

1. **Approach based on a reference path:** In [29], a method for following a vehicle based on a Frenet frame was developed (cf. Fig. 15). It exploits the use of chained systems to separate

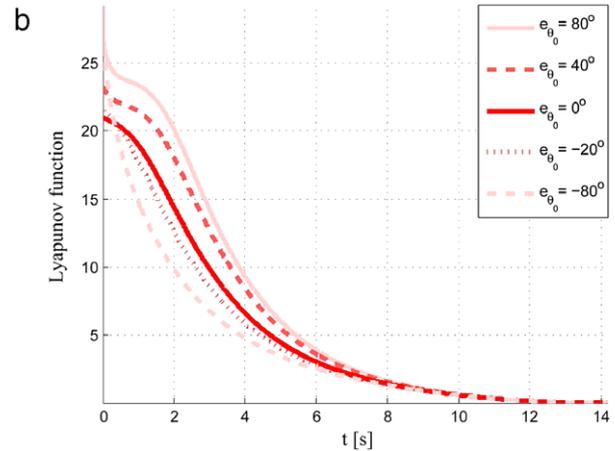
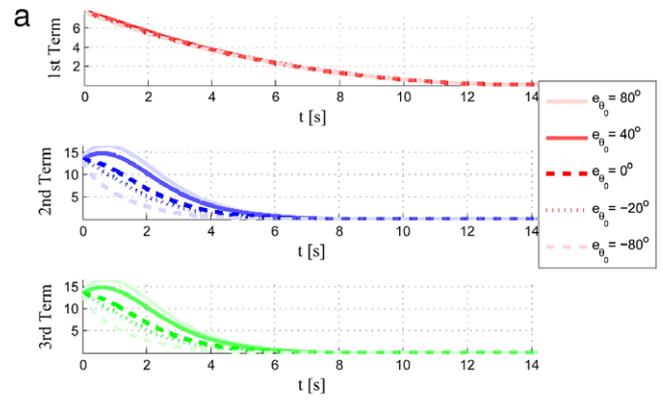


Fig. 12. (a) Different terms of the Lyapunov function (18) and (b) Lyapunov function values for several initial orientations.

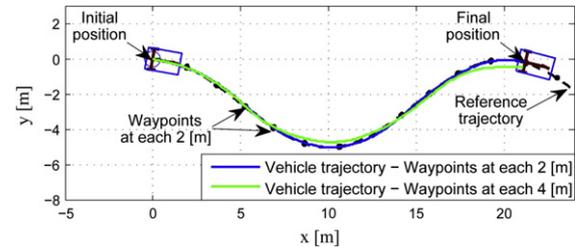


Fig. 13. Vehicle trajectories for different distances between waypoints.

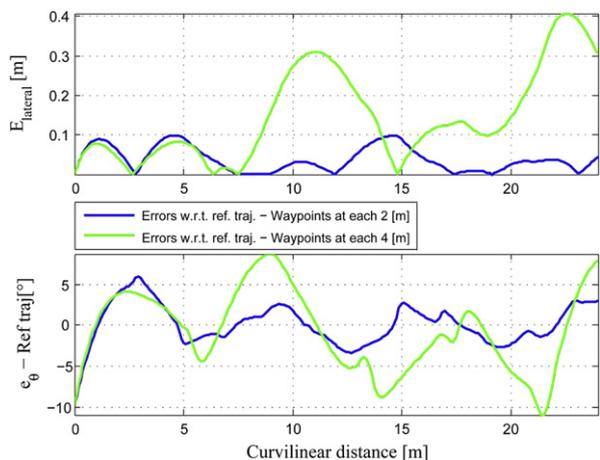


Fig. 14. Errors w.r.t. reference trajectory for different distances between waypoints.

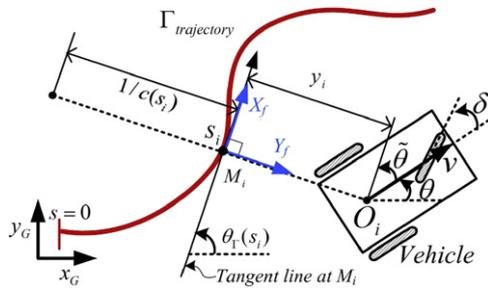


Fig. 15. Vehicle modeling in a Frenet frame.

the lateral and longitudinal control. Therefore, each controller can be designed independently. The lateral control is obtained using chained transformation (more details are given in [11] and [21]). The longitudinal control consists in keeping a specific curvilinear distance d_s between the target and the vehicle. One drawback of this approach is the dependency on a known reference path for the vehicle, i.e., if the vehicle follows a dynamic target then the target trajectory must be accurately known by the vehicle.

2. **Approach based on a target model:** In [21] and [22] a control law to track a reference vehicle (target) is proposed. A variable transformation to obtain the control is applied to the error system and commands. The control law is synthesized using a suitable Lyapunov function (more details can be found in [21]). The desired steering angle is computed by integration. Nevertheless, the control law considers a non-zero linear target velocity, i.e., if the target is static then the commands sent to the vehicle are zero.

The approaches presented above are implemented in simulation and compared to our proposal. In order to do this, the main target (equivalent to a Leader robot in formation control [30]) tracks a sinusoidal trajectory and the followers must maintain a distance of 3 m w.r.t. this first robot, i.e., the secondary target to be reached is located at 3 m (curvilinear distance) from the main target (cf. Fig. 16).

Figs. 16 and 17 show the trajectories and the control output of each vehicle (leader and followers). It can be noticed that the proposed control law has a similar performance to the controller based on a reference trajectory (Frenet control).

Table 1 shows the convergence time to satisfy the error threshold in distance d_{Target} and orientation $e_{\theta_{Target}}$ with regards to the target pose (cf. Fig. 5). The proposed control law has the smallest convergence time to satisfy simultaneously both threshold errors (maximum value between convergence time of d_{Target} and $e_{\theta_{Target}}$). The difference with the Frenet control is equal to 1.81 s and with chained system is equal 2.3 s.

Table 2 shows the convergence time to satisfy the error threshold in distance y_i and orientation $\tilde{\theta}$ with regards to the target trajectory (Frenet reference frame $X_f Y_f$ (cf. Fig. 15)). The Frenet control

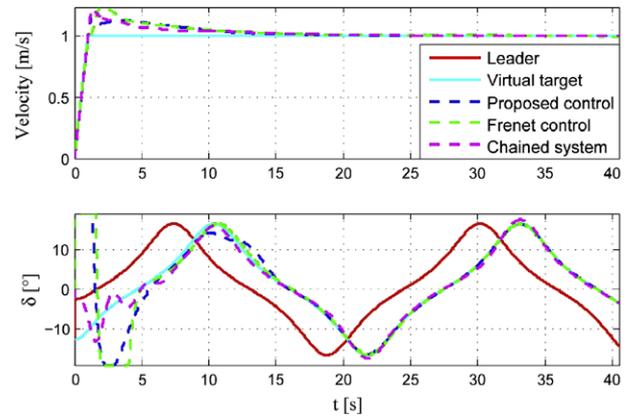


Fig. 17. Control output.

Table 1

Comparison with dynamic target pose (Fig. 5).

	Time [s] to always keep:	
	$d_{Target} < 0.15$ m	$ e_{\theta_{Target}} < 5^\circ$
Proposed control	13.17 s	4.24 s
Frenet control	14.98 s	3.92 s
Chained control	15.47 s	6.62 s

Table 2

Comparison of the errors defined according to Frenet reference frame (Fig. 15).

	Time [s] to always keep:	
	$ y_i < 0.15$ m	$ \tilde{\theta} < 5^\circ$
Proposed control	3.33 s	4.14 s
Frenet control	2.93 s	3.86 s
Chained control	10.22 s	10.57 s

has the smallest convergence time to satisfy simultaneously both threshold errors, since it is dedicated to follow the reference trajectory. However, there is only a small difference (0.28 s) compared to our proposal while our method uses only the current pose of the target (thus more flexible).

The proposed control law was not designed to take into account the reference trajectory, however the obtained results are very close to those designed for trajectory tracking. In addition to its accuracy, the proposed control law has more flexibility (cf. Section 3) to perform the autonomous navigation of the vehicles. Indeed, we need only to know the current pose and dynamics of the target instead of all recorded trajectory.

5.2. Experimental results

The navigation strategy was also experimented with a pair of real urban vehicles to perform dynamic target reaching. The

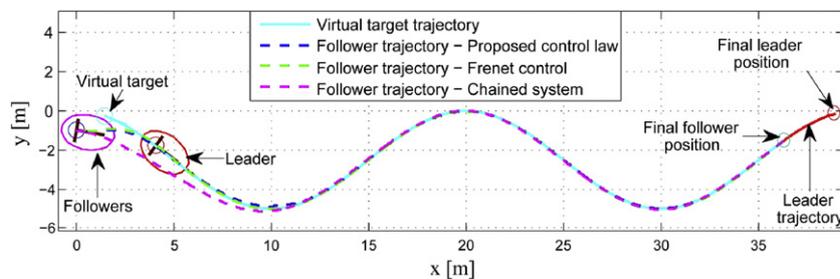


Fig. 16. Trajectory of the leader and followers (proposed control law, Frenet control [11] and Chained system [21]).

Table 3
VIPALAB platform (cf. Fig. 19).

Elements	Description
Chassis	(l, w, h) = (1.96, 1.30, 2.11) m
Computer	Intel Core i7, CPU: 1.73 GHz RAM: 8Go OS(32bits): Ubuntu12.04
RTK-GPS	NacTechGPS, accuracy: 2 cm framerate: 10 Hz
Gyrometer	Xsens MTi, accuracy: 0.2°/s framerate: 2 KHz
Proprioceptive sensors	Steering angle, resolution: 0.02° framerate: 50 Hz Linear speed, resolution: 0.1 m/s framerate: 50 Hz

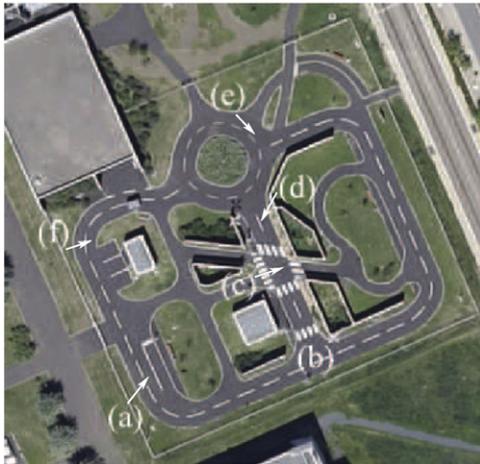


Fig. 18. PAVIN experimental platform. (a)–(f) correspond to the locations shown in Fig. 19.

scenario was built to show different situations, such as: multi-vehicle navigation in formation, static and dynamic target reaching and obstacle avoidance situation.

5.2.1. Testbed and scenario

Navigation was performed in a structured environment named PAVIN (*Plate-forme d'Auvergne pour Véhicules INtelligents*) (cf. Fig. 18). The proposed strategy was implemented using VIPALAB urban vehicles (cf. Fig. 19). This vehicle carry different embedded proprioceptive and exteroceptive sensors such as odometers, gyrometer, steering angle sensor and an RTK-GPS (more details are given in [31]). In these experiments, each vehicle uses a combination of RTK-GPS and gyrometer to estimate its current position and orientation at a sample time of $T_s = 0.1$ s (cf. Table 3). The vehicles

have a range sensor (LIDAR) with a maximum detected range equal to 10 m. These sensors provide enough accurate data w.r.t. the vehicle dynamic. Indeed, in these experiments, the vehicles move at maximum velocity of 1.5 m/s due mainly to the relative short dimensions of the used urban platform (cf. Fig. 18). Moreover, the vehicles communicate by WI-FI, enabling the transmission of the leader's pose data.

5.2.2. Results analysis

Experiments were carried out to show the performance of the proposed control law and target assignment strategy using waypoint selection based on Algorithm 2 (with $\Delta_{\alpha_{max}} = 15^\circ$) on an already defined reference trajectory. The Leader vehicle has to reach successively static waypoints. Moreover, the proposed control law was implemented in another vehicle (Follower) which takes the first vehicle (Leader) as dynamic target to track at a curvilinear distance equal to 5 m (behind the Leader). The tracking of the dynamic target allows to apply the proposed control law to multi-robot systems where the dynamic set-point is given by the leader and the desired geometric formation shape [32]. The configuration of the dynamic target is sent by the Leader to the Follower via WI-FI. This experiment can be found online.¹

Furthermore, to exhibit the flexibility of the proposed navigation strategy, a scenario with the presence of an obstacle is presented (cf. Fig. 19). An obstacle is placed between the waypoints. As mentioned in Section 2, the proposed strategy can easily integrate the obstacle avoidance behavior (cf. Fig. 3). Therefore, the vehicle can perform different maneuvers between waypoints, in this case the obstacle avoidance without the use of any trajectory re-planning method. The used obstacle avoidance method is based on limit-cycles as given in [6,20] and [32]. It was selected because it is a stable and robust method which could use only local information from range sensors. Let us briefly inspect this method (for more details see [6] and [20]). A limit-cycle is a reactive safe trajectory which encloses the hinder obstacle. According to that, the vehicle avoids the obstacle while tracking the direction of the limit-cycle trajectories. The obstacle avoidance is activated as soon as the vehicle detects at least one obstacle which can hinder the future vehicle movements toward the current assigned waypoint [20] and [33].

It can be seen in Fig. 20(a) that the Leader reaches accurately the successive static waypoints successfully and the Follower tracks accurately also the dynamic target (Leader). Moreover, the

¹ <http://maccs.univ-bpclermont.fr/uploads/Profiles/VilcaJM/Navigation.avi>.

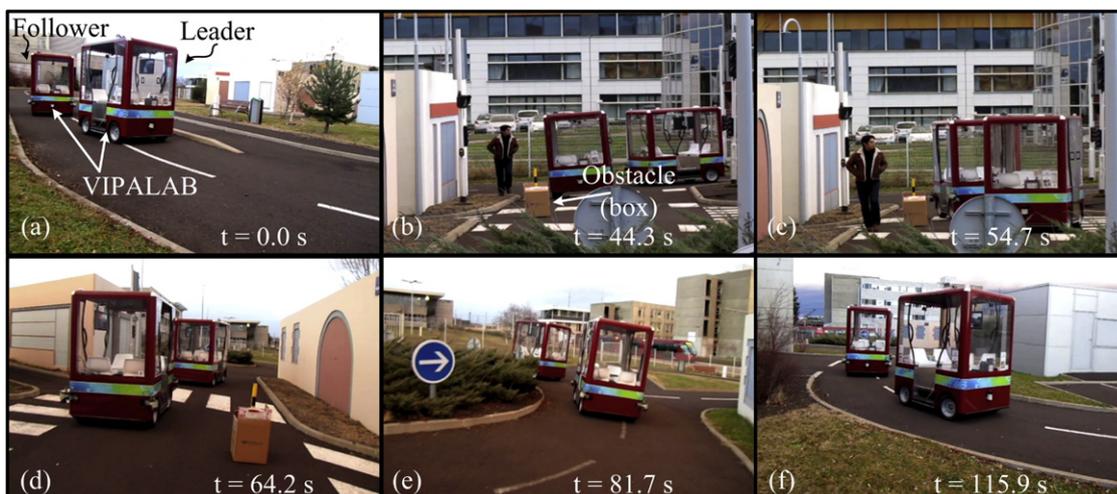


Fig. 19. Some images from the performed experiment.

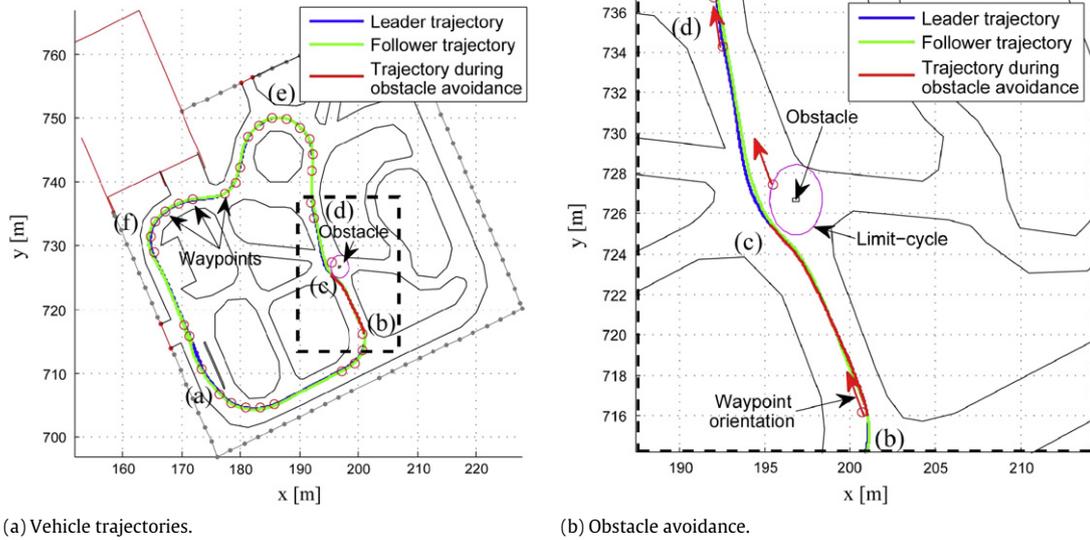


Fig. 20. Vehicle trajectories obtained using GPS and a set of waypoints positioned in the environment using Algorithm 2 ($\Delta\alpha_{max} = 15^\circ$).

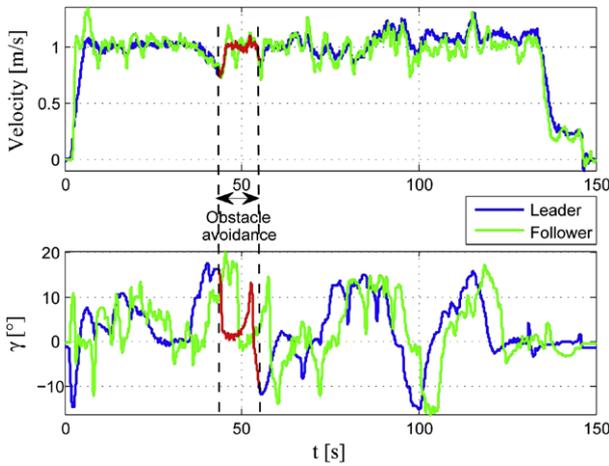


Fig. 21. Control output (real experiment).

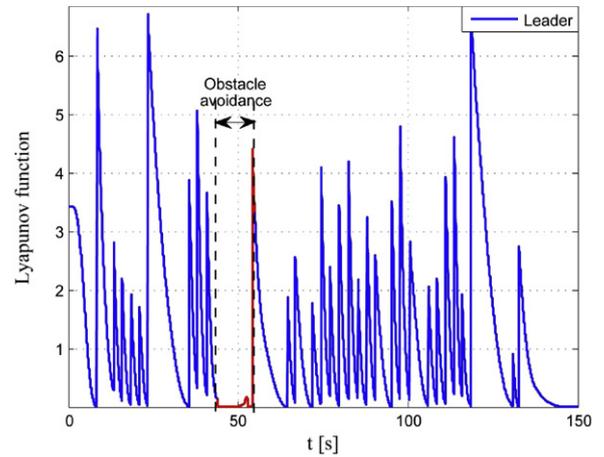


Fig. 22. Lyapunov function of the leader (based on static waypoint reaching).

Follower trajectory using the proposed control law is close to the leader trajectory (cf. Fig. 20(a)). Fig. 20(b) focuses on the vehicles' trajectories when the obstacle avoidance is activated. The Leader detects the hinder obstacle between the waypoints and it applies the reactive limit-cycle method [20,33]. The Follower avoids also the obstacle since it tracks accurately the Leader trajectory. It can be noted that the proposed navigation strategy allows flexible and smooth movements between the waypoints and also to perform different behaviors, such as: obstacle avoidance, emergency stop or waypoint reassignment.

Fig. 21 shows the velocity and steering angle of the vehicles. These actual values have been filtered, during the experimentation, using an Extended Kalman Filter (EKF) to reduce the sensor noise. Figs. 22 and 23 show the Lyapunov function values which highlight that each vehicle is stable and it converges to each static waypoint for the Leader and to the dynamic target for the Follower. Therefore, smooth, flexible and safe trajectories for the vehicles were obtained.

6. Conclusion

This paper has presented a novel safe and flexible control strategy based on target reaching for the navigation of autonomous vehicles in structured environment. A control law was presented

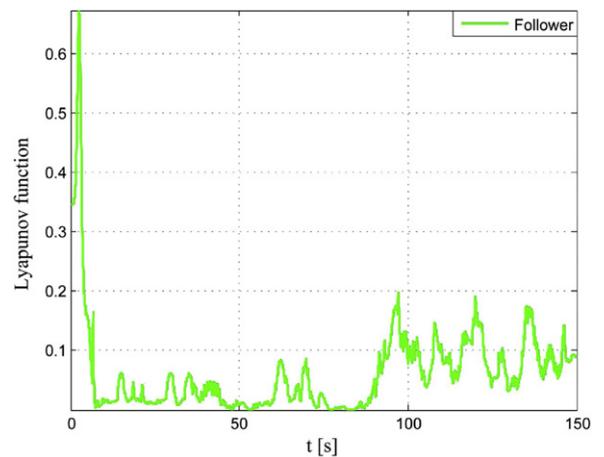


Fig. 23. Lyapunov function of the follower (based on dynamic target tracking).

and synthesized using a suitable Lyapunov function, which takes into account the position, the angle between the robot and the target, and its orientation with respect to set-points. Moreover, it enables static and dynamic target reaching. The stability of the overall control architecture was proved using a suitable Lyapunov

function based on a new set of variables. A sequential target assignment strategy to perform safe navigation was also proposed. It is based on target switching using appropriate reference frames, linked to the current selected waypoint and to the next one in the list. This target assignment strategy enables a smooth and flexible vehicle trajectory while satisfying an upper bound on distance and orientation errors (to not collide with the boundary of the environment). An elementary waypoint selection method was also presented to perform safe and smooth trajectories. Simulations and experiments using real urban vehicles show the efficiency and the flexibility of the proposed control strategy for the navigation of urban vehicles. Furthermore, different target-reaching methods from the literature were presented and their performances were compared with the proposed control law. These comparisons showed the interesting features of the proposed control law in terms of stability and flexibility for different tasks.

In future works, a robust analysis will be developed to determine the sensor inaccuracy effects on our control law. Different methods to obtain the optimal number of waypoints on the road map will be developed. In addition, this strategy will be applied to high dynamic multi-vehicle system based on the combination of virtual structure and Leader–Follower approaches.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at <http://dx.doi.org/10.1016/j.robot.2015.01.008>.

References

- [1] B. Sezen, Modeling automated guided vehicle systems in material handling, *Otomatiklestirilmli Rehberli Arac Sistemlerinin Transport Tekniginde Modellemesi, Dou Universitesi Dergisi* 4 (2) (2011) 207–216.
- [2] P. Avanzini, B. Thuilot, T. Dallej, P. Martinet, J.-P. Dérutin, On-line reference trajectory generation for manually convoying a platoon of automatic urban vehicles, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Saint-Louis, USA, 2009, pp. 1867–1872.
- [3] M.M. Martins, C.P. Santos, A. Frizzera-Neto, R. Ceres, Assistive mobility devices focusing on smart walkers: Classification and review, *Robot. Auton. Syst.* 60 (4) (2012) 548–562.
- [4] S.A. Stoeter, P.E. Rybski, K.N. Stubbs, C.P. McMillen, M. Gini, D.F. Hougen, N. Papanikolopoulos, A robot team for surveillance tasks: Design and architecture, *Robot. Auton. Syst.* 40 (2–3) (2002) 173–183.
- [5] J.-W. Lee, B. Litkouhi, A unified framework of the automated lane centering/changing control for motion smoothness adaptation, in: *15th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2012, pp. 282–287.
- [6] L. Adouane, A. Benzerrouk, P. Martinet, Mobile robot navigation in cluttered environment using reactive elliptic trajectories, in: *18th IFAC World Congress*, 2011.
- [7] T. Gu, J.M. Dolan, On-road motion planning for autonomous vehicles, in: C.-Y. Su, S. Rakheja, H. Liu (Eds.), *Intelligent Robotics and Applications*, Vol. 7508, Springer, Berlin, Heidelberg, 2012.
- [8] M. Bonfè, C. Secchi, E. Scioni, Online trajectory generation for mobile robots with kinodynamic constraints and embedded control systems, in: *10th International IFAC Symposium on Robot Control*, Croatia, 2012.
- [9] Y. Kanayama, Y. Kimura, F. Miyazaki, T. Noguchi, A stable tracking control method for an autonomous mobile robot, in: *Proceedings of the IEEE International Conference on Robotics and Automation*, 1990, pp. 384–389.
- [10] S. Blazic, Four-state trajectory-tracking control law for wheeled mobile robots, in: *10th International IFAC Symposium on Robot Control*, Croatia, 2012.
- [11] C. Samson, Control of chained systems. Application to path following and time-varying point-stabilization of mobile robots, *IEEE Trans. Automat. Control* 40 (1) (1995) 64–77.
- [12] S.M. LaValle, *Planning Algorithms*, Cambridge Univ. Press, 2006.
- [13] J. Connors, G.H. Elkaim, Manipulating b-spline based paths for obstacle avoidance in autonomous ground vehicles, in: *ION National Technical Meeting, ION NTM 2007*, San Diego, CA, USA, 2007.
- [14] J. Horst, A. Barbera, Trajectory generation for an on-road autonomous vehicle, in: *Proceedings of the SPIE: Unmanned Systems Technology VIII*.
- [15] J. Courbon, Y. Mezouar, P. Martinet, Autonomous navigation of vehicles from a visual memory using a generic camera model, *Intell. Transp. Syst.* 10 (2009) 392–402.
- [16] M. Aicardi, G. Casalino, A. Bicchi, A. Balestrino, Closed loop steering of unicycle like vehicles via Iyapunov techniques, *Robotics Automation Magazine, IEEE* 2 (1) (1995) 27–35.
- [17] A.A. Masoud, A harmonic potential approach for simultaneous planning and control of a generic uav platform, *J. Intell. Robot. Syst.* 65 (1) (2012) 153–173.
- [18] H. Fang, R. Lenain, B. Thuilot, P. Martinet, Trajectory tracking control of farm vehicles in presence of sliding, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 58–63.
- [19] A.V. Pesterev, Stabilizing control for a wheeled robot following a curvilinear path, in: *10th International IFAC Symposium on Robot Control*, Croatia, 2012.
- [20] J. Vilca, L. Adouane, Y. Mezouar, Reactive navigation of a mobile robot using elliptic trajectories and effective online obstacle detection, *Gyroscopy and Navigation - Springer Verlag* 4 (2013) 14–25.
- [21] J.-P. Laumond, *La Robotique Mobile, Chapitre 2, Traité IC2 Information-Commande-Communication*, Hermès, 2001.
- [22] B. Siciliano, O. Khatib (Eds.), *Springer Handbook of Robotics, Part E-34*, Springer, 2008.
- [23] J.-C. Latombe, *Robot Motion Planning*, Kluwer Academic Publishers, Boston, MA, 1991.
- [24] O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, *Int. J. Robot. Res.* 5 (1986) 90–99.
- [25] H.K. Khalil, *Nonlinear Systems*, 3rd Edition, Prentice Hall, 2002.
- [26] G.M. Siouris, *Missile Guidance and Control Systems*, Springer-Verlag, 2004.
- [27] H. Choset, K.M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L.E. Kavraki, S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementation*, MIT Press, 2005.
- [28] Y. Kuwata, G.A. Fiore, J. Teo, E. Frazzoli, J.P. How, Motion planning for urban driving using rrt, in: *International Conference on Intelligent Robots and Systems*, 2008, pp. 1681–1686.
- [29] P. Daviet, M. Parent, Platooning for small public urban vehicles, in: O. Khatib, J. Salisbury (Eds.), *Experimental Robotics IV*, in: *Lecture Notes in Control and Information Sciences*, vol. 223, Springer, Berlin, Heidelberg, 1997, pp. 343–354.
- [30] J.-M. Vilca, L. Adouane, Y. Mezouar, Adaptive leader-follower formation in cluttered environment using dynamic target reconfiguration, in: *Springer Tracts in Advanced Robotics*, from *International Symposium on Distributed Autonomous Robotic Systems, DARS 2014*, Daejeon, Korea, 2014.
- [31] IPDS, The Institut Pascal Data Sets (March 2013). <http://ipds.univ-bpclermont.fr>.
- [32] A. Benzerrouk, L. Adouane, P. Martinet, Obstacle avoidance controller generating attainable set-points for the navigation of multi-robot system, in: *IEEE Intelligent Vehicles Symposium (IV)*, Australia, 2013.
- [33] J. Vilca, L. Adouane, Y. Mezouar, Robust on-line obstacle detection using range data for reactive navigation, in: *10th International IFAC Symposium on Robot Control*, Croatia, 2012.



José Vilca is currently a Ph.D. student in the Institut Pascal, UMR 6602, CNRS/Université Blaise Pascal, France. He received his M.Sc. in Dynamic System from University of São Paulo, Brazil, in 2011 and his B.Eng. degree in Electronic Engineering from National University of Engineering, Peru, in 2006. He belongs to the Image, Perception Systems, Robotics Group at Institut Pascal. His research interests include cooperative systems, hybrid control systems, multi-robot coordination, robotics, mobile-robot autonomous navigation and nonlinear control.



Lounis Adouane received his Master of Sciences in 2001 from IRCCyN - ECN Nantes (France), where he worked on the control of legged mobile robotics. In 2005 he obtained the Ph.D. in Automatic Control from LAB - UFC Besançon. During his Ph.D. Lounis Adouane has deeply investigated the field of multi-robot systems, especially those relating to reactive control architectures. After that, he joined in 2006 LAI - INSA Lyon and he studied the hybrid architecture of control applied to cooperative mobile arms robots. Since 2006, he is an Associate Professor at Institut Pascal - Polytech Clermont-Ferrand. His research interests include: Mobile robotics control, Cooperative robotics, Artificial intelligence, Behavioral/Hybrid control architectures and Multi-robot simulation.



Youcef Mezouar received the Ph.D. degree in Computer Science from the Université de Rennes 1, Rennes, France, in 2001 and the “Habilitation à Diriger les Recherches” degree from Université Blaise Pascal, Clermont-Ferrand, France, in 2009. He is currently a Professor at Institut Français de Mécanique Avancée and a member of Institut Pascal, UMR 6602, CNRS/Université Blaise Pascal, France. He is a Coleader of the Image, Perception Systems, Robotics Group, Institut Pascal, and he leads the Modeling, Autonomy and Control in Complex Systems Team at Institut Pascal. His research interests include automatics, robotics, and computer vision, particularly visual servoing and mobile-robot navigation.