# Efficient trajectory of a car-like mobile robot

*Francisco Valero, Francisco Rubio and Antonio José Besa*
Universitat Politecnica de Valencia, Valencia, Spain, and
*Carlos Llopis-Albert*
CITV-Universitat Politècnica de València, Valencia, Spain

**Abstract**

**Purpose** – The purpose is to create an algorithm that optimizes the trajectories that an autonomous vehicle must follow to reduce its energy consumption and reduce the emission of greenhouse gases.

**Design/methodology/approach** – An algorithm is presented that respects the dynamic constraints of the robot, including the characteristics of power delivery by the motor, the behaviour of the tires and the basic inertial parameters. Using quadratic sequential programming with distributed and non-monotonous search direction (Quadratic Programming Algorithm with Distributed and Non-Monotone Line Search), an optimization algorithm proposed and developed by Professor K. Schittkowski is implemented.

**Findings** – Relations between important operating variables have been obtained, such as the evolution of the autonomous vehicle's velocity, the driving torque supplied by the engine and the forces acting on the tires. In a subsequent analysis, the aim is to analyse the relationship between trajectory made and energy consumed and calculate the reduction of greenhouse gas emissions. Also this method has been checked against another different methodology commented on in the references.

**Research limitations/implications** – The main limitation comes from the modelling that has been done. As greater is the mechanical systems analysed, more simplifying hypotheses should be introduced to solve the corresponding equations with the current computers. However, the solutions are obtained and they can be used qualitatively to draw conclusions.

**Practical implications** – One main objective is to obtain guidelines to reduce greenhouse gas emissions by reducing energy consumption in the realization of autonomous vehicles' trajectories. The first step to achieve that is to obtain a good model of the autonomous vehicle that takes into account not only its kinematics but also its dynamic properties, and to propose an optimization process that allows to minimize the energy consumed. In this paper, important relationships between work variables have been obtained.

**Social implications** – The idea is to be friendly with nature and the environment. This algorithm can help by reducing an instance of greenhouse gases.

**Originality/value** – Originality comes from the fact that we not only look for the autonomous vehicle's modelling, the simulation of its motion and the analysis of its working parameters, but also try to obtain from its working those guidelines that are useful to reduce the energy consumed and the contamination capability of these autonomous vehicles or car-like robots.

**Keywords** Optimization, Obstacle avoidance, Car-like robot navigation, Robot dynamics

**Paper type** Research paper

## 1. Introduction

For a mobile robot to be able to move autonomously, it must have an efficient navigation system that allows it to carry out its task during transit from an initial to a final configuration.

Part of the function assigned to a robot's navigation system consists of planning the trajectory, which considers three fundamental aspects:

1. the locomotion system;
2. its dynamic behaviour, which includes the consideration of the driving forces, resistance forces and the inertial characteristics of the robot; and
3. the environment in which it will move and its representation, considering the possible existence of obstacles.

Normally, in the process of obtaining the trajectory, it is worth minimizing certain working variables such as the time or energy consumed.

In autonomous car-like robots, the behaviour of the locomotion system is highly conditioned by the wheels and their interaction with the terrain. It is a key determinant of the car-like robot's dynamic response. As for the modelling of the environment and the representation of the space through which the robot will move, it is a topic that has been investigated for decades.

A brief summary of the techniques used for modelling the environment and for the representation of space from the 1980s to the present day includes the generalized cones method (Brooks, 1983), graph search techniques, roadmaps, Voronoi diagrams (Eberhart and Shi, 2001), and visibility graphs. A recent example of visibility graphs can be found in Glavaski *et al.* (2009). In Buniyamin *et al.* (2011), the authors introduce

The current issue and full text archive of this journal is available on Emerald Insight at: **www.emeraldinsight.com/0143-991X.htm**

Industrial Robot: the international journal of robotics research and application
46/2 (2019) 211–222

[DOI 10.1108/IR-10-2018-0214]

Received 18 October 2018
Revised 25 January 2019
12 February 2019
27 February 2019
Accepted 28 February 2019

a bug algorithm that solves the problem of navigation without using any map or model of the environment, using only the data from the sensors.

The Quadtree representation technique and decomposition in cells (Yahja *et al.*, 1998; Singh *et al.*, 2000) must also be cited. A summary of the techniques for modelling the environment applied to path planning can be found in Sariff and Buniyamin (2006), where some of the strengths and weaknesses of the presented methods are discussed. An application of some of the algorithms (techniques based on searches in graphs like the A* algorithm, the greedy search or the uniform cost search) are shown in Rubio *et al.* (2009). And a summary of the state of the art and future lines of research in relation to motion planning for autonomous robots can be found in Katrakazas *et al.* (2015). In Pala *et al.* (2013), the main objective of the algorithm is to find, if it exists, an efficient path between cells in a given binary map, using the grid occupation matrix, as discussed in Gonzalez-Arjona *et al.* (2011).

As for the dynamic behaviour of the vehicle, several trends can be observed in the literature. One is to consider or reduce the car-like robot to a point as in Tokekar *et al.* (2014), where the authors analyse the planning of trajectories, imposing restrictions on energy during its motion. This limits the speed and acceleration values given by the actuators, namely, an electric DC brush motor. In fact, the dynamic characteristics of the car-like robot, such as masses and inertias, are not taken into account, nor are considerations of the contact forces between tire and ground. Neither do they include any considerations about changes in the direction of motion. However, these are aspects that are essential for good trajectory planning.

Another tendency is to use the full car model of the car-like robot as in Labakhua *et al.* (2005), where the authors present a kinematic model of the vehicle; however, the kinematic considerations are limited to imposing restrictions on speeds and accelerations without considering dynamic restrictions associated with masses and inertia of the vehicle or the behaviour of the tire. In this work the paths are obtained by introducing kinematic constraints to follow curves of the cubic polynomial type, trigonometric splines and clothoids.

In Ghita and Kloetzer (2012) and Li *et al.* (2017), the vehicle is also modelled as a full car. In Ghita and Kloetzer (2012), the same authors state that they do not use a dynamic model, as at low speeds, the kinematic model is sufficient to obtain computationally feasible results. When velocities are important or masses are not negligible, the dynamic model must be introduced.

A novel integrated local trajectory planning and tracking control (ILTPTC) framework for autonomous vehicles driving along a reference path with obstacle avoidance is presented in Li *et al.* (2017). However, they do not consider the interactions between the ground and the tire and, in fact, the trajectory planning is limited to the generation of an optimum path with a specific profile of speeds without dynamic considerations. They limit the velocity of the vehicle, taking into account considerations such as the state of the road and traffic rules. And they limit the lateral acceleration so as not to have to take into account the effects of drift and rolling on the stability of the vehicle. They also limit linear acceleration to limit maximum velocity.

Another trajectory planner is presented for a full car-like mobile robot in Li and Shao (2015), where the kinematic principles are accurately described by differential equations, and the constraints are strictly expressed using algebraic inequalities. Although it tries to describe the trajectory in terms of the corresponding differential equations, it does not take into account the driving forces, interaction with the terrain or inertial characteristics of the system.

In the cited works, the trajectory of the vehicle is determined on the basis of kinematic considerations, limiting the velocity values due to the characteristics of the actuators, especially those of the electric motors, or other issues. They do not really take into account the inertial characteristics of the mobile robot or use any model of tire.

A third trend in trajectory planning is represented by authors who work with the simplified dynamic model of the robot, considering a system with few degrees of freedom. In Wang and Qi (2001), the authors use the bicycle model of a four-wheel-steering (4WS) vehicle. The limits of vehicle mechanism, drive and brake torque are taken into account and dynamic constraints are replaced by velocity kinematics and acceleration based on inertial and friction parameters. However, it does not include lateral tire friction limits.

In this paper, the authors present a planner for obtaining trajectories for mobile robots with wheels, which considers the basic dynamic properties of the robot, including the lateral friction limit of the tires, motor and brake torques, obtaining feasible and efficient trajectories for the robot based on the recursive resolution of optimization problems.

It is a global planner that makes it possible to obtain trajectories considering the constraints associated with the dynamics of the robot in an environment with stationary obstacles. The procedure is based on the determination of passing configurations between which the path is adjusted by polynomial interpolation functions whose coefficients are determined to minimize the time while respecting the dynamic constraints of the vehicle, thus defining the trajectory.

To minimize the time, a Quadratic Programming Algorithm with Distributed and Non-Monotone Line Search (NLPQLP) is used, which was proposed and developed by Schittkowski (2010, 2015).

This approach marks a clear difference from planners that only include kinematic constraints, as in Simba *et al.* (2014, 2016), or that are conservative or do not guarantee the feasibility of trajectories, as in Li and Shao (2015) and Tokekar *et al.* (2014). Also it is different from Chong and Byung (2007), in which the authors analyse the minimum-energy translational trajectory generation for a two-wheeled mobile robot The authors simplify too much the model so that they only consider a WMR moving in a straight line. Also the dynamics of the whole vehicle are neglected. In Gokhan Bayar, 2013, using a car-like robot similar to this study's robot, the dynamics are also neglected. The need to deep into autonomous car driving is highlighted in Anderson and Anderson (2015).

The article is organized as follows: Section 2 introduces the concepts and definitions used in modelling the trajectory. Section 3 presents the procedure for obtaining the efficient trajectory without collisions. Section 4 details the kinematic and dynamic restrictions used in optimization problems, as well as the modelling of the car-like robot from which the

constraints are obtained. Three examples of application of the trajectory planning algorithm are illustrated in Section 5, and the results obtained are discussed. Section 6, the conclusions are summarized.

## 2. Modelling the trajectory

### 2.1 Definitions

*2.1.1 Local reference system*

It is sited on the centre of gravity of the robot and an ISO reference system is associated with it.

*2.1.2 Position*

The position of the robot is defined by the location of the origin of the local reference system, $p\ (x, y)$.

*2.1.3 Configuration*

The configuration of the robot is defined by the position and orientation of the local reference system, $c\ (p,\ \theta)$. A configuration is said to be feasible when it belongs to a trajectory and has no collision.

*2.1.4 Adjacent position*

Given a feasible configuration of the robot $c_j$, it is said that $p_k$ is adjacent to it if it has been obtained by increasing the coordinates corresponding to the position of $c_j$.

*2.1.5 Obstacle*

It is static and is defined by a combination of pattern obstacles, circles and polygons that determine forbidden zones for the robot.

*2.1.6 Interval*

Given two positions of the robot $p_j$ and $p_k$, the interval $I_j$ will be defined by the polynomials of the form:

$$\begin{aligned} &\forall t \in [0,\quad t_j]; \\ &x_j = a_{xj} + b_{xj}t + d_{xj}t^2 + e_{xj}t^3 \\ &y_j = a_{yj} + b_{yj}t + d_{yj}t^2 + e_{yj}t^3 \end{aligned} \tag{1}$$

where $t$ is the variable time associated with the motion of the robot, and the following must be fulfilled:

$$\left.\begin{aligned} x_j(0) &= x_j \\ y_j(0) &= y_j \end{aligned}\right\} \text{and} \left.\begin{aligned} x_j(t_j) &= x_k \\ y_j(t_j) &= y_k \end{aligned}\right\}$$

*2.1.7 Trajectory*

Given a sequence of $m$ robot positions, $P = \{p_1, p_2, \ldots, p_m\}$, a trajectory $T$ is defined by a sequence of $m - 1$ intervals between the positions of $P$ that satisfies:

- Continuity in positions.

$$\left.\begin{aligned} x_j(0) &= x_j \\ y_j(0) &= y_j \end{aligned}\right\} \text{and} \left.\begin{aligned} x_j(t_j) &= x_{j+1} \\ y_j(t_j) &= y_{j+1} \end{aligned}\right\}$$

  $(4\ (m - 1))$ equations are set.

- Continuity in velocities.

  The initial and final velocities of the trajectory must be zero,

$$\left.\begin{aligned} \dot{x}_1(0) &= 0 \\ \dot{y}_1(0) &= 0 \end{aligned}\right\} \text{and} \left.\begin{aligned} \dot{x}_{m-1}(t_m) &= 0 \\ \dot{y}_{m-1}(t_m) &= 0 \end{aligned}\right\}$$

  four equations are set.

  The initial velocity of each interval must be equal to the final velocity of the previous one,

$$\left.\begin{aligned} \dot{x}_j(0) &= \dot{x}_{j-1}(t_{j-1}) \\ \dot{y}_j(0) &= \dot{y}_{j-1}(t_{j-1}) \end{aligned}\right\}$$

  $(2\ (m - 2))$ equations are set.

- Continuity in accelerations.

  The initial acceleration of each interval must be equal to the end of the previous one:

$$\left.\begin{aligned} \ddot{x}_j(0) &= \ddot{x}_{j-1}(t_{j-1}) \\ \ddot{y}_j(0) &= \ddot{y}_{j-1}(t_{j-1}) \end{aligned}\right\}$$

  $(2\ (m - 2))$ equations are set.

A linear system of $[8(m - 1)]$ equations is available, which makes it possible to obtain the coefficients of the polynomials (1) that define the intervals so that the trajectory is fully determined.

*2.1.8 Minimum time trajectory $T^{min}$*

It is a trajectory that takes the minimum time to go from the initial position to the target configuration.

The minimum time and the corresponding trajectory are obtained from an optimization procedure in which the objective function is:

$$f(t) = \sum_{j=1}^{m} t_j \text{ (see equation 2)}$$

subject to the kinematic and dynamic restrictions. See Section 2.2 for more details about the optimization procedure. See also Rubio *et al.* (2010, 2016).

*2.1.9 Offspring trajectory*

It is said that a trajectory $T_k^{min}$ is an offspring from another $T_j^{min}$with a sequence of m positions when the sequence of positions of $T_k^{min}$ is equal to that of $T_j^{min}$ plus one, provided that the position added is not the first or the last one:

$$P^k = P^j \bigcup p_n, \text{for } n \neq 1 \text{ and } n \neq m + 1.$$

The offspring trajectories from several generations will have different numbers of intervals, but they will always maintain the same initial and final positions.

*2.1.10 Trajectory space*

For a robot with a given initial position $p_i$ and a final position $p_f$, the configuration space $TS$ is defined as the set of minimum time trajectories between $p_i$ and $p_f$. When the robot operates in an environment with obstacles, the subspace of $TS$ formed by the trajectories without collisions will be represented as $TS^c$.

### 2.2 Generation of a minimum time trajectory

For a car-like robot (with kinematic and dynamic constraints as described below), given the initial configuration $c_i$, the final position $p_f$ and a series of $(m-1)$ passing positions $p_j$ with $j = 1 \ldots m-1$, an optimization problem to obtain the minimum time trajectory $T^{min}$ associated with the sequence of positions $P = \{p_i, \ldots, p_j, \ldots, p_f\}$ is set.

*2.2.1 Objective function*

The trajectory will consist of $m$ intervals between the $m+1$ positions of $P$, where $t_j$ for $j = 1, m$ are the times associated with the intervals that comply with the equations of type (1) and the conditions associated with the definition in Section 2.1.7, so that the objective function is:

$$f(t) = \sum_{j=1}^{m} t_j \tag{2}$$

Constraints:

- Initial orientation $\theta_i$, corresponding to the initial configuration $c_i$.
- The steering wheel angle does not exceed a specified value $\delta_{max}$.
- The maximum speed of the vehicle cannot exceed $V_{max}$.
- The driving force is limited by the torque curve of the engine.
- The adhesion of the tires to the terrain is limited.

This is an optimization problem with nonlinear constraints, whose solution is obtained by the NLPQLP created and proposed by Professor Klaus Schittkowski. It should be considered that, in each iteration, the linear system associated with obtaining the coefficients of the equations of type (1) will be solved using the normalized time method (Suñer *et al.*, 2007) so as not to penalize the computation times.

## 3. Generation of the trajectory without collisions

The problem is to obtain an efficient and collision-free trajectory for a mobile robot in an environment with static obstacles. An efficient trajectory is understood to be one that is near the minimum time with a low computational cost and which respects the restrictions imposed on the robot in Section 2.2.2. Collision detection is specific for each type of standard obstacle, considering the mobile robot as a rectangular shape that is delimited by four segments. For circles, the distance from each segment to the centre of the circle is calculated and if it exceeds the radius there is no collision. For polygons, it is verified that there is no intersection between the segments corresponding to mobile robot and those of the obstacles (Rubio *et al.*, 2009, 2010).

The initial data are:

- Information about the robot that is needed for its modelling, as described in Section 4.
- Information about obstacles and their locations.
- Initial configuration and final position of the mobile robot.

The steps followed to generate the trajectory are similar to those employed by Rubio *et al.* (2016), on a PUMA robot with fixed base, but adapted to the needs of the mobile robot, resulting:

- Calculation of the minimum time initial trajectory.
- The trajectory $T_i^{min}$ is obtained from a single interval with $P^i = \{p_i, p_f\}$.
- Search for collisions.
- On the trajectory $T_i^{min}$, the first configuration with collision $c_c$ is identified, as is the one previous to it $c_a$ [See Figure 1(a) and 1(b)].
- *Generation of adjacent positions*: Four adjacent positions are generated from $c_a$ according to the definition in Section 2.1.4, $(p_{a,j}, j = 1, \ldots, 4)$ [see Figure 1(c)] by choosing the positions that are far enough from any obstacle $p_{ak}$ $(0 \leq k \leq 4)$; if none exists $(k = 0)$, a configuration in the previous trajectory $c_{a-1}$ is searched for, and the algorithm works recursively until it finds a configuration that results in $(k \neq 0)$.
- *Generation of offspring trajectories*: For each of the adjacent positions generated in c. that are not contained within an obstacle, an offspring trajectory $T_k^{min}(0 \leq k \leq 4)$ associated with $P^k = P^i \cup p_{a,k}$ is generated. $P^i$ has the initial configuration, the target and the crossing points of the trajectory where the collision has been located [see Figure 1(d)].
- *Selection of the trajectory*: The trajectories generated in point d. are placed in a set of trajectories ordered by time $TS_t = \left\{T_1^{min} \ldots T_p^{min}\right\}$. The minimum time trajectory within the set $T_1^{min}$ is selected, taken out of $TS_t$ and also checked for collisions. If there are any collisions, the algorithm returns to point c. This process is repeated (iterating) until a solution $T_1^{min}$ without a collision is reached.

## 4. Modelling the constraints

The proposed optimization problem requires expressions of reduced complexity that allow iterative calculations to be performed efficiently. The use of dynamic constraints in this type of applications is highly conditioned by computational times, which is why simplified, efficient models are used.

### 4.1 Robot modelling

The RBK robot is an electric vehicle for internal transport powered by a hydrogen fuel cell and batteries with autonomous operation capacity (Figure 2). Its main features are rear-wheel drive, steering on the front wheels, power 3.3 kW, mass 690 kg, top speed 32 km/h, length 2.66 m, width 1.23 m, height 1.70 m, wheelbase $L$ = 1.65 m, height of the centre of gravity ($G$) $h$ = 0.50 m, distance from $G$ to the front axle $L_a$ = 1.10 m, distance from $G$ to the front axle $L_b$ = 0.55 m (Figure 3). The model used is based on the well-known "bicycle model", which gives rise to the following simplifying assumptions:

- no roll and pitch motions;
- no side-load transfer;
- no aerodynamic effects;
- a plane model with three degrees of freedom and a restriction associated with the steering angle;
- the front wheels are simplified into one that will exert the force corresponding to both, and the same simplification applies to the rear wheels;
- the steering angle corresponds to that of the single front wheel of the model; and
- the sideslip and steering angles are small.

**Figure 1** Generation of offspring trajectories.

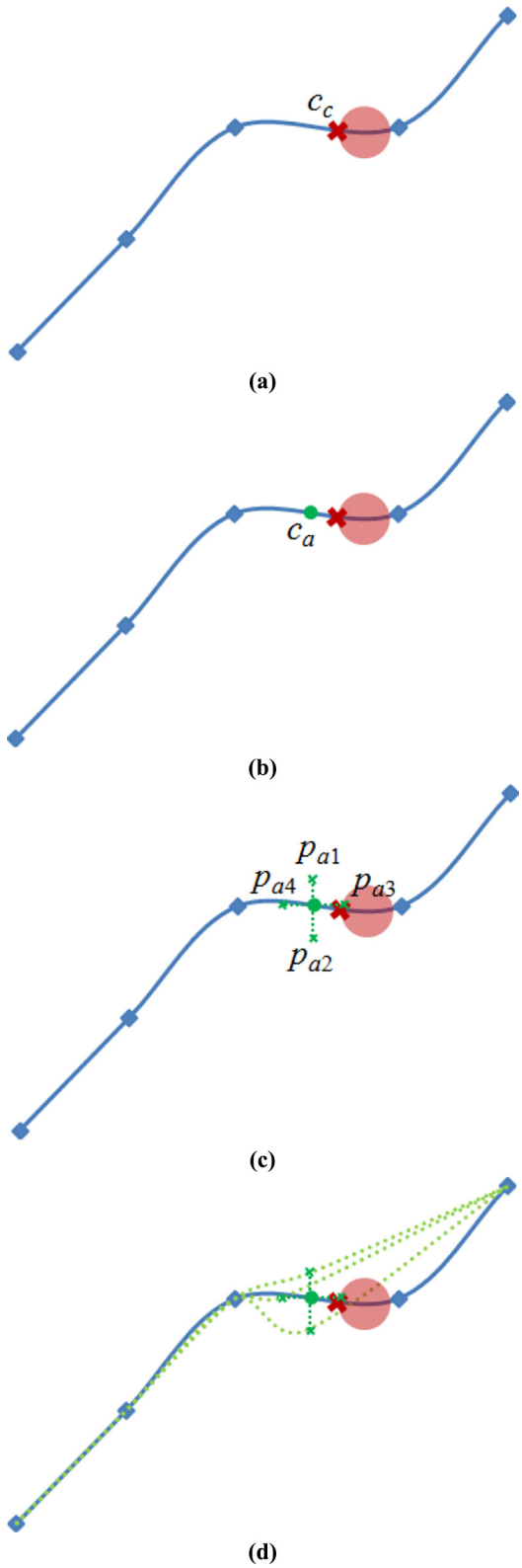
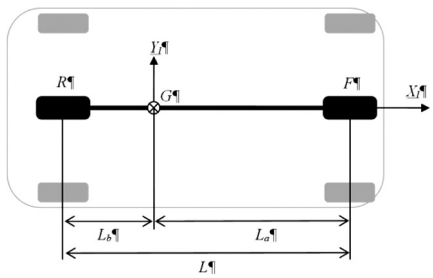



**Figure 2** Car-like mobile robot


**Figure 3** Bicycle model




The kinematics of the centre of gravity of the vehicle on the trajectory are known, so in the global reference system, the following expressions are met:

Position of the centre of gravity:

$$\begin{aligned} x_G &= a_x + b_x t + d_x t^2 + e_x t^3 \\ y_G &= a_y + b_y t + d_y t^2 + e_y t^3 \end{aligned} \tag{3}$$

Velocity of the centre of gravity:

$$\overrightarrow{V_G} = \dot{x}_G \vec{i} + \dot{y}_G \vec{j}$$

and its magnitude:

$$\left|\overrightarrow{V_G}\right| = \sqrt{\dot{x}_G^2 + \dot{y}_G^2} \tag{4}$$

Orientation of the velocity of the centre of gravity:

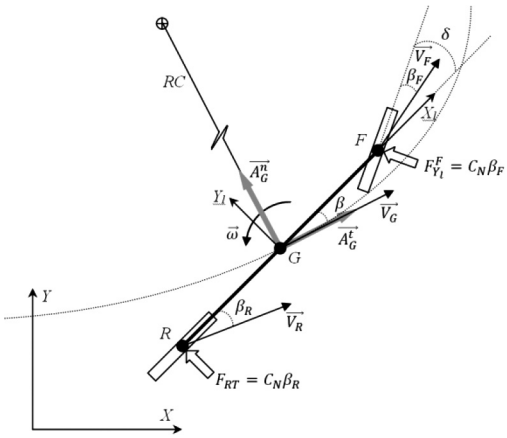
$$\theta = \tan^{-1} \frac{\dot{y}_G}{\dot{x}_G} \tag{5}$$

Angular velocity, where $\beta$ is the sideslip of the mobile robot (see Figure 4), which is assumed to be small according to the hypothesis:

$$\omega = \dot{\theta} + \dot{\beta}$$

$\dot{\beta}$ is considered negligible, resulting in:

**Figure 4** Kinematics and sideslip angles




$$\omega = \frac{\dot{x}_G\ddot{y}_G - \dot{y}_G\ddot{x}_G}{\dot{x}_G^2 + \dot{y}_G^2} \tag{6}$$

In a local reference system that is linked to the vehicle according to the ISO (International Organization for Standardization) convention, the velocity of the rear axle is:

$$\overrightarrow{V_R^l} = \overrightarrow{V_G^l} + \vec{\omega} \wedge \overrightarrow{r_{GR}}$$

so that:

$$\overrightarrow{V_R^l} = \left|\overrightarrow{V_G}\right|\cos\beta\,\vec{i}_l + \left(\left|\overrightarrow{V_G}\right|\sin\beta - L_b\omega\right)\vec{j}_l$$

and considering the small sideslip hypothesis:

$$\overrightarrow{V_R^l} \approx \left|\overrightarrow{V_G}\right|\vec{i}_l + \left(\left|\overrightarrow{V_G}\right|\beta - L_b\omega\right)\vec{j}_l \tag{7}$$

based on the velocity components, the sideslip of the rear axle is obtained:

$$\beta_R = \tan^{-1}\frac{\left(\left|\overrightarrow{V_G}\right|\beta - L_b\omega\right)}{\left|\overrightarrow{V_G}\right|}$$

and using the small sideslip hypothesis:

$$\beta_R \approx \beta - \frac{(\dot{x}_G\ddot{y}_G - \dot{y}_G\ddot{x}_G)L_b}{\left(\dot{x}_G^2 + \dot{y}_G^2\right)^{3/2}} \tag{8}$$

with an approach similar to that used to obtain equation (7), for the front axle:

$$\overrightarrow{V_F^l} \approx \left|\overrightarrow{V_G}\right|\vec{i}_l + \left(\left|\overrightarrow{V_G}\right|\beta + L_a\omega\right)\vec{j}_l \tag{9}$$

as $\delta$ the steering angle is small according to the hypotheses, similar to equation (8), the forward sideslip is:

$$\beta_F \approx \delta - \beta - \frac{(\dot{x}_G\ddot{y}_G - \dot{y}_G\ddot{x}_G)L_a}{\left(\dot{x}_G^2 + \dot{y}_G^2\right)^{3/2}} \tag{10}$$

The normal acceleration for the trajectory in $G$ is:

$$A_G^n = -\ddot{x}_G\sin\theta + \ddot{y}_G\cos\theta \tag{11}$$

and the tangential acceleration:

$$A_G^t = \ddot{x}_G\cos\theta + \ddot{y}_G\sin\theta \tag{12}$$

In the local reference system that is linked to the vehicle, the lateral acceleration (direction $Y_l$) is:

$$A_G^{Y_l} = A_G^n\cos\beta - A_G^t\sin\beta$$

as the angle $\beta$ is small, the equation can be written as:

$$A_G^{Y_l} \approx A_G^n - A_G^t\beta \tag{13}$$

Under the small sideslip hypothesis, it is usual to consider the lateral behaviour of the tires linearly, so as the front and rear tires are equal, the lateral forces are:

$$\begin{aligned} F_{RT} &= -C_T\beta_R \\ F_{FT} &= -C_T\beta_F \end{aligned} \tag{14}$$

with a direction normal to the rim and opposite to the sideslip (Figure 4).

Setting the Newton–Euler equations for the lateral forces and the moments, the following expression is met:

$$F_{RT} + F_{FT}\cos\delta = m\,A_G^{Y_l}$$

considering $\delta$ small:

$$F_{RT} + F_{FT} = m\,A_G^{Y_l} \tag{15}$$

The equation of moments is:

$$F_{FT}\cos\delta\,L_a - F_{RT}L_b = I_z\dot{\omega}$$

where $I_z$ is the moment of inertia of the vehicle around an axis parallel to $Z$ passing through $G$, taking the usual simplifications and performing the following operation:

$$C_T(-L_a\beta_F + L_b\beta_R) = I_z\dot{\omega} \tag{16}$$

Substituting in equations (15) and (16), $\beta$ and $\delta$ are obtained by solving the linear system.

From equations (8) and (10), $\beta_F$ and $\beta_R$ are obtained.

### 4.2 Constraint associated with the initial orientation

The robot must start moving from an initial configuration $c_i$ ($p_i$, $\theta_i$) and from zero velocity, so the initial acceleration must have the orientation $\theta_i$:

$$\tan\theta_i = \frac{d_{y1}}{d_{x1}}$$

$d_{x1}$ and $d_{y1}$ being coefficients of the polynomials (1) corresponding to the first interval of the trajectory, so the corresponding constraint is:

$$\tan\theta_i - \frac{d_{y1}}{d_{x1}} = 0 \tag{17}$$

### 4.3 Constraint of the steering angle

For each interval $i$ of the trajectory, $\delta(t_{ij})$ is obtained in a discrete number of points $j$, so that:

$$\delta_i = \max\left(\delta\left(t_{ij}\right)\right)$$

and for each interval, the imposed constraint is:

$$\delta_{max}^2 - \delta_i^2 > 0, \forall i \tag{18}$$

### 4.4 Constraint of maximum velocity

For each interval $i$ of the trajectory, $\left|\overrightarrow{V_G}\left(t_{ij}\right)\right|$ is obtained from equation (4) in a discrete number of points $j$, so that:

$$V_{Gi}^2 = \max\left(\left|\overrightarrow{V_G}\left(t_{ij}\right)\right|^2\right)$$

and for each interval, the imposed constraint is:

$$V_{max}^2 - V_{Gi}^2 > 0, \forall i \tag{19}$$

### 4.5 Constraints associated with forces on tires

Tire forces owing to contact and adherence to the terrain can be written in local coordinates, as:

$$\left.\begin{aligned}\overrightarrow{F}_F &= F_F^{x_l}\overrightarrow{i}_l + F_F^{y_l}\overrightarrow{j}_l + F_F^{z_l}\overrightarrow{k}_l \\ \overrightarrow{F}_R &= F_R^{x_l}\overrightarrow{i}_l + F_R^{y_l}\overrightarrow{j}_l + F_R^{z_l}\overrightarrow{k}_l\end{aligned}\right\} \tag{20}$$

In $X_l$ direction, assuming a small steering angle, the following equilibrium equation is set:

$$F_F^{x_l} + F_R^{x_l} = mA_{x_l} \tag{21}$$

where $A_{x_l}$ is the acceleration:

$$A_{x_l} = m(\ddot{x}_G\cos(\beta + \theta) + \ddot{y}_G\sin(\beta + \theta))$$

The force on the front wheel is:

$$\left.\begin{aligned}A_{x_l} > 0 \rightarrow \quad & F_F^{x_l} = 0 \\ A_{x_l} \leq 0 \rightarrow \quad & F_F^{x_l} = 0.6\ mA_{x_l} + F_{F_r}^{x_l}\end{aligned}\right\} \tag{22}$$

with losses due to rolling motion:

$$F_{F_r}^{x_l} = \mu_r F_F^{z_l} \tag{23}$$

where $\mu_r$ is considered constant because the velocity and sideslip angles are small.

The force on the rear wheel is:

$$\left.\begin{aligned}A_{x_l} > 0 \rightarrow F_R^{x_l} &= mA_{x_l} + F_{R_r}^{x_l} + F_{F_r}^{x_l} \\ A_{x_l} \leq 0 \rightarrow F_R^{x_l} &= 0.4\ mA_{x_l} + F_{R_r}^{x_l}\end{aligned}\right\} \tag{24}$$

with:

$$F_{R_r}^{x_l} = \mu_r F_R^{z_l} \tag{25}$$

In $Y_l$ direction, considering small steering and sideslips angles, the forces are:

$$\left.\begin{aligned}F_F^{y_l} &= -C_n\beta_F \\ F_R^{y_l} &= -C_n\beta_R\end{aligned}\right\} \tag{26}$$

where $C_n$ is a characteristic of the tire.

In $Z_l$ direction, considering the load transfer due to the acceleration $A_{x_l}$, the force is:

$$\left.\begin{aligned}F_F^{z_l} &= \frac{m}{L}(L_b\ g - \ A_{x_l}\ h) \\ F_R^{z_l} &= \frac{m}{L}(L_a\ g - \ A_{x_l}\ h)\end{aligned}\right\} \tag{27}$$

Considering a friction circle to limit the maximum force that can be transmitted between the tires and the ground due to the coefficient of friction $\mu_t$, the following condition is obtained:

$$\left.\begin{aligned}\sqrt{\left(F_F^{x_l}\right)^2 + \left(F_F^{y_l}\right)^2} &< \mu_t F_F^{z_l} \\ \sqrt{\left(F_R^{x_l}\right)^2 + \left(F_R^{y_l}\right)^2} &< \mu_t F_R^{z_l}\end{aligned}\right\} \tag{28}$$

Each interval $i$ of the trajectory is discretized into a discrete number of points, obtaining for each point $j$ the force $F_{F,R_{ij}}^{z_l}$ from equation (27), $F_{F,R_{ij}}^{x_l}$ considering equations (22)-(25) and $F_{F,R_{ij}}^{y_l}$ from equation (26). For each interval and each wheel, the boundary condition established in equation (28) is considered:

$$RT_{Fi} = \min\left[\left(\mu_t F_{F_{ij}}^{z_l}\right)^2 - \left(F_{F_{ij}}^{x_l{}^2} + F_{F_{ij}}^{y_l{}^2}\right)\right]$$
$$RT_{Ri} = \min\left[\left(\mu_t F_{R_{ij}}^{z_l}\right)^2 - \left(F_{R_{ij}}^{x_l{}^2} + F_{R_{ij}}^{y_l{}^2}\right)\right]$$

The following two constraints are obtained for each interval of the trajectory:

$$\left.\begin{aligned}RT_{Fi} > 0 \\ RT_{Ri} > 0\end{aligned}\right\} \forall i \tag{29}$$

### 4.6 Constraint of the driving force

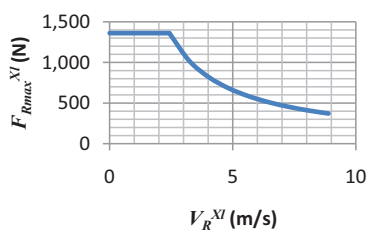
The traction of the vehicle is achieved by means of an electric motor and a gear reduction of velocity on the rear wheels, resulting in a maximum driving force on the rear wheel of the simplified model as shown in the graph in Figure 5.

This behaviour is adjusted by the following expressions:

$$\left.\begin{aligned}0 < V_R^{x_l} \leq 2,42\frac{m}{s} &\rightarrow F_{R_{max}}^{x_l} = 1361\ N \\ 2,42 < V_R^{x_l} \leq 8,89\frac{m}{s} &\rightarrow F_{R_{max}}^{x_l} = \frac{3300}{V_R^{x_l}}\ N\end{aligned}\right\} \tag{30}$$

For each interval $i$ of the trajectory, $F_R^{x_l}(t_{ij})$ and $F_{Rmax}^{x_l}(t_{ij})$ are obtained in a discrete number of points $j$, and considering.

**Figure 5** Driving force in relation to velocity



$RF_i = min\left(F^{x_j}_{Rmax}(t_{ij}) - F^{x_j}_{R}(t_{ij})\right)$, the constraints being:

$$RF_i > 0, \forall\ i \tag{31}$$

## 5. Results

Four examples are presented to illustrate the behaviour and the quality of the algorithm. In the first two, the same initial configuration is used to reach different points, and in the third one, very distant points within the chosen space are used to work with. The work area is 7,200 $m^2$on the ground floor of the UPV buildings where the laboratory is located.

Finally, the example proposed in Dai *et al.* (2018), will be solved and the results obtained will be compared. The vehicles used (in this paper and in Dai *et al.*, 2018) are similar and have been characterized using the bicycle model.

### 5.1 First example

The initial configuration is $c_i$ (81.5 m, 21.7 m, 3.14 rad) and the target point is $p_f$ (70.0 m, 64.4 m). The solution required a computation time of $T_c$ = 281.25 ms, obtaining a trajectory of $m$ = 6 intervals with a value of the objective function of$f(t)$ = 33.91 s.

Figure 6 shows the trajectory (in blue) followed by the vehicle to avoid obstacles, as can be seen in the detail.

Figure 7 shows the evolution of the vehicle velocity throughout the trajectory. It is possible to observe how it increases with the radius of curvature. Figure 8 shows the evolution of the torque applied to the wheels to achieve the motion, where positive values are supplied by the engine through transmission and negative values by means of the brake system.

**Figure 6** (a) Trajectory (in blue) followed by the robot and (b) detail in the area approaching an obstacle (measured in metres)

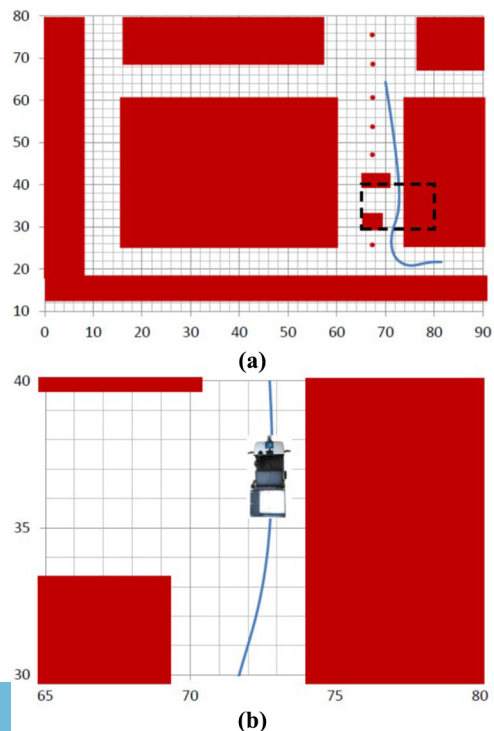



**Figure 7** Evolution of the vehicle velocity throughout the trajectory

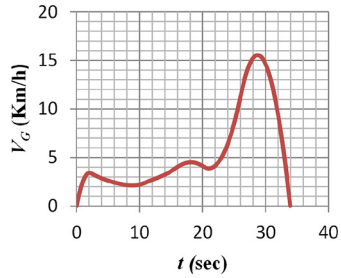
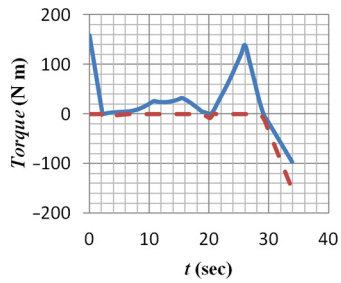



**Figure 8** Evolution of the torque applied to the rear (solid line) and front (dashed line) wheels




### 5.2 Second example

The initial configuration is $c_i$ (81.5 m, 21.7 m, 3.14 rad) and the target point $p_f$ (63.5 m, 73.0 m). The solution required a computation time of $T_c$ = 921.87 ms, obtaining a trajectory of $m$ = 6 intervals with a value of the objective function of $f(t)$ = 32.71 s.

Figure 9 shows how the trajectory obtained has greater curvatures compared to Example 1. Thus, a higher speed is achieved and a lower time is taken (Figure 10), despite less torque being applied at the start (Figure 11), to reach a point close to the previous example.

**Figure 9** Trajectory (in blue) followed by the robot in Example 2 (measured in metres)

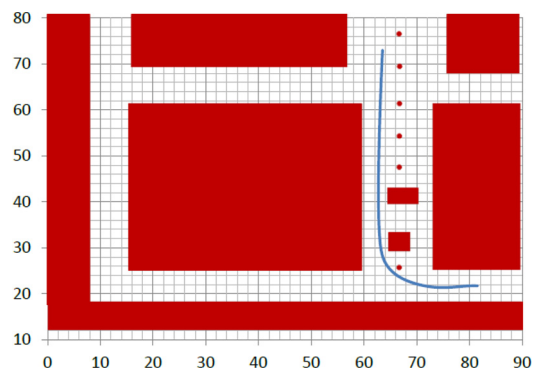


**Figure 10** Evolution of the vehicle velocity throughout the trajectory

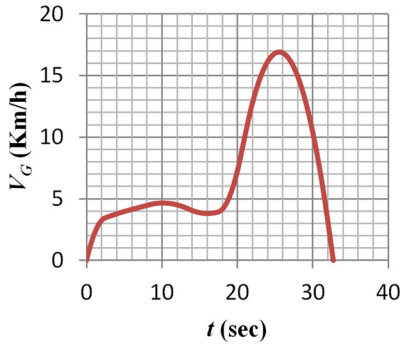
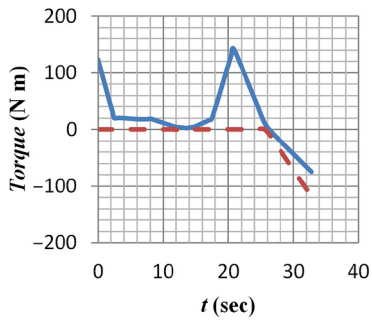



**Figure 11** Evolution of the torque applied to the rear wheels (solid line) and front wheels (dashed line)




### 5.3 Third example

The initial configuration is $c_i$ (12.0 m, 75.0 m, 4.71 rad) and the target point $p_f$ (81.5 m, 21.7 m). The solution required a computation time of $T_c$ = 437.50 ms, obtaining a trajectory of $m$ = 6 intervals with a value of the objective function of $f(t)$ = 59.34 s.

Figure 12 shows the trajectory travelled, its continuity and smoothness.

Figure 13 shows the values associated with the limitations imposed by the torque and steering angle that are used in calculating the constraints using equations (31) and (18), respectively.

Figure 14 shows the values associated with the adhesion forces and their limits, which are used to calculate the constraints using equation (29).

Figure 15(a) shows the evolution of the velocity that is used in the constraint formulated in equation (19), and Figure 15(b) shows the evolution of the sideslip angles in the tires, which verify the validity of the hypotheses associated with them.

### 5.4 Fourth example

In this example, to compare results, the dimensional and inertial characteristics of the RBK have been adapted to those of the 4WS4WD vehicle described in Dai *et al.* (2018). The main differences between the two vehicles are that 4WS4WD has four driving and steering wheels and it can consider a different coefficient of friction for each section in the path, while RBK has rear driving wheels, and steering on the front wheels and it uses the same coefficient of friction between the tires and the terrain for the entire trajectory (in this example, 0.3). The trajectory, generated using the passing points described in Dai *et al.* (2018), and without velocities constraints, can be seen in Figure 15.

**Figure 12** (a) Trajectory (in blue) followed by the robot and (b) detail in the area approaching an obstacle (measured in metres)

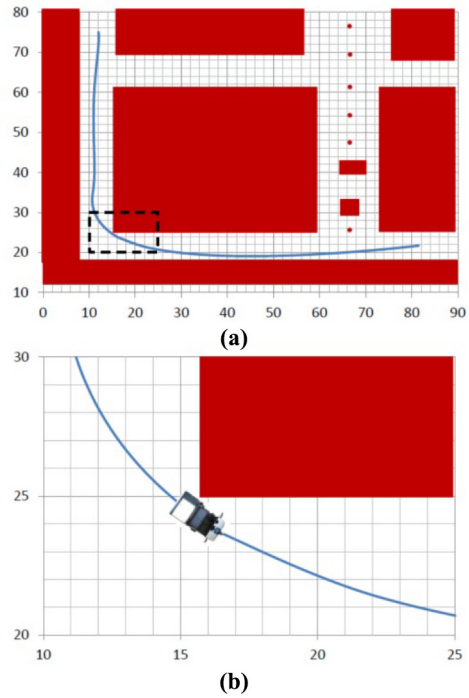



**Figure 13** (a) Longitudinal force on the rear wheel (solid line) and force limit available through the transmission (dashed line) and (b) steering angle (solid line) and its limit (dashed line)

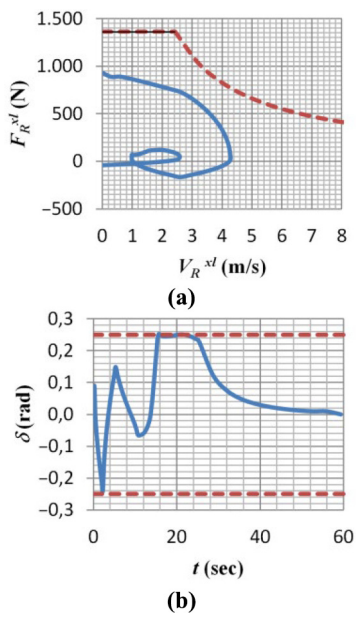


**Figure 14** Square of the resultant of the forces in the plane of the track mark (solid line) and square of the adhesion limit (dashed line): (a) for the front wheel and (b) for the rear wheel

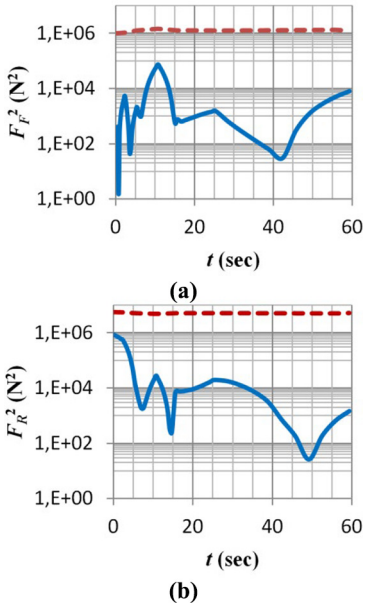



**Figure 15** (a) Velocity and (b) sideslip of the front wheels (long dashed line) and the rear wheels (short dashed line)

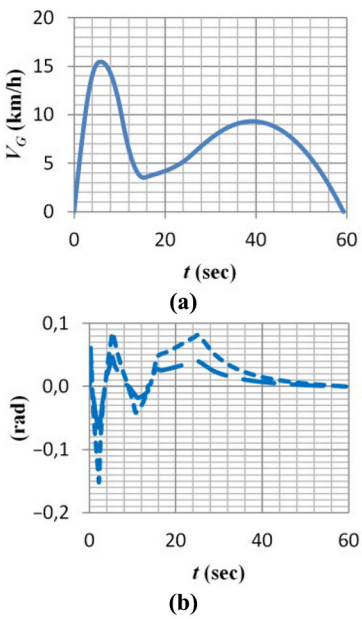



Figure 16 shows the trajectory obtained under the conditions described above. It can be seen how the resulting curvature is softer than the reference one.

The vehicle spends 36.5 sin performing the trajectory with the velocity profile shown in Figure 17. Remember that there are no constraints imposed on the velocity at the passing points.

**Figure 16** Trajectory followed by the vehicle

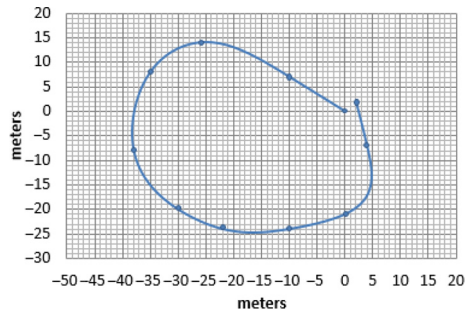



**Figure 17** Longitudinal velocity

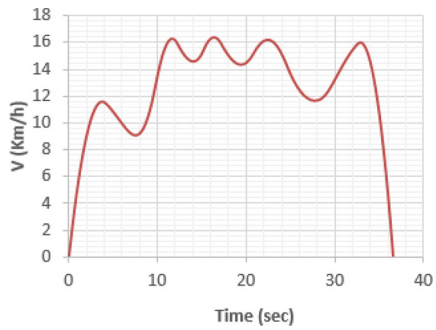



Figure 18 shows how the friction constraints are working in the front wheels along the trajectory, and it is possible to check how the limit is reached in different points.

Figure 19 shows the evolution of the driving torque applied on the rear wheels. It can be seen how it takes negative values at some points given that it has braking capacity, and how the maximum values are consistent with those of the reference, considering that it is a vehicle with rear-wheel drive.

## 6. Conclusions

The planner obtains safe and efficient trajectories in environments with stationary obstacles. Safety means not only the absence of collisions but also the feasibility of the trajectory that is guaranteed based on compliance with the dynamic

**Figure 18** Friction limit, $F_{lim}$ and magnitude of the resultant longitudinal and lateral forces, $F_f$, on the front wheels

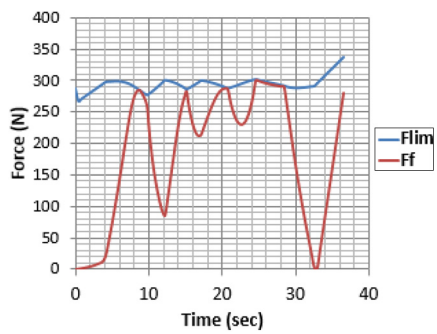
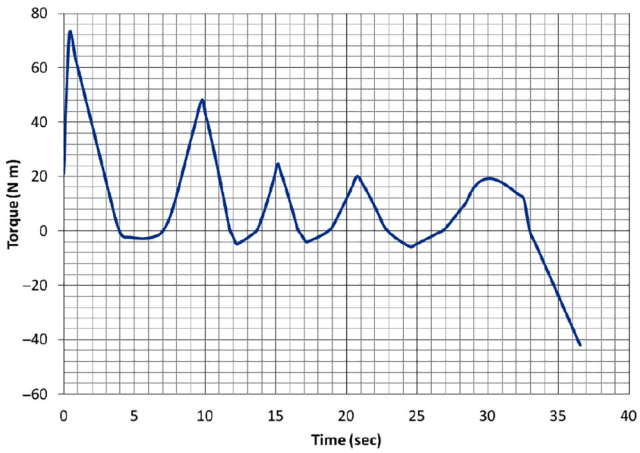


**Figure 19** Driving torque applied on the rear wheels




constraints of the mechanical system, which allows us to consider both the performance of the powertrain and the effects of the behaviour of the tires or the load of the vehicle, among others parameters. The evolution of those parameters such as autonomous vehiclés velocity, the driving torque supplied by the engine, the forces acting on the tires, etc.

With respect to the example of reference, trajectories with softer radii of curvature are obtained.

Limitations come from the models used to define the path and the dynamic behaviour of the robot, since low-complexity algorithms are sought to reduce the computing times. These limitations have led to conservative results for the trajectories obtained.

Future work is associated with reduction of energy consumed by the vehicle and how it affects to the greenhouse gas emissions.

Also, it is necessary to quantify the discrepancy between the real model and the theoretical model, that is to say, how the theoretical model fits the real model.

## References

Anderson, M. and Anderson, S.L. (2015), "Toward ensuring ethical behavior from autonomous systems: a case-supported principle-based paradigm", *Industrial Robot: An International Journal*, Vol. 42 No. 4.

Bayar, G. (2013), "Long distance autonomous trajectory tracking for an orchard vehicle", *Industrial Robot: An International Journal*, Vol. 40 No. 1.

Brooks, R.A. (1983), "Solving the find path problem by good representation of free space", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 13 No. 2, pp. 190-197.

Buniyamin, N., Wan Ngah, W.A.J., Sariff, N. and Mohamad, Z. (2011), "A simple local path planning algorithm for autonomous mobile robots", *International Journal of Systems Applications, Engineering & Development*, pp. 2-5.

Chong, H.K. and Byung, K.K. (2007), "Minimum-energy translational trajectory generation for differential-driven wheeled mobile robots", *Journal of Intelligent and Robotic Systems*, Vol. 49 No. 4, pp. 367-383, doi: 10.1007/s10846-007-9142-0.

Dai, P., Taghia, J., Lam, S. and Katupitiya, J. (2018), "Integration of sliding mode based steering control and PSO based drive force control for a 4WS4WD vehicle", *Autonomous Robots*, Vol. 42 No. 3, pp. 553-568.

Eberhart, R.C. and Shi, Y. (2001), "Particle swarm optimization: developments, applications and resources, in evolutionary computation", *Proceedings of the 2001 Congress on Evolutionary Computation*.

Ghita, N. and Kloetzer, M. (2012), "Trajectory planning for a car-like robot by environment abstraction", *Robotics and Autonomous Systems*, Vol. 60 No. 4, pp. 609-619.

Glavaski, D., Volf, M. and Bonkovic, M. (2009), "Mobile robot path planning using exact cell decomposition and potential field methods", *WSEAS Transactions on Circuits and Systems*, Vol. 8 No. 9.

Gonzalez-Arjona, D., Sanchez, A., de Castro, A. and Garrido, J. (2011), "Occupancy-grid indoor mapping using FPGA-based mobile robots", *Proceedings of the Conference on Design of Circuits and Integrated Systems, Albufeira, Portugal, 16-18 November 2011*, pp. 345-350.

Katrakazas, C., Quddus, M., Chen, W.H. and Deka, L. (2015), "Real-time motion planning methods for autonomous on-road driving: state-of-the-art and future research directions", *Transportation Research Part C*, Vol. 60, pp. 416-442.

Labakhua, L., Nunes, U., Rodrigues, R. and Leite, F. (2005), "Trajectory planning methods for autonomous car-like vehicles", *The International Symposium on System Theory, Automation, Robotics, Computers, Informatics, Electronics and Instrumentation*, available at: ace.ucv.ro/sintes12/sintes12_2005/mechatronics/m7.pdf

Li, B. and Shao, Z. (2015), "Simultaneous dynamic optimization: a trajectory planning method for nonholonomic car-like robots", *Advances in Engineering Software*, Vol. 87, pp. 30-42.

Li, X., Sun, Z., Cao, D., Liu, D. and He, H. (2017), "Development of a new integrated local trajectory planning and tracking control framework for autonomous ground vehicles", *Mechanical Systems and Signal Processing*, Vol. 87, pp. 118-137.

Pala, M., Eraghi, N.O., López-Colino, F., Sanchez, A., de Castro, A. and Garrido, J. (2013), "HCTNav: a path planning algorithm for low-cost autonomous robot navigation in indoor environments", *ISPRS International Journal of Geo-Information*, Vol. 2 No. 3, pp. 729-748, doi: 10.3390/ijgi2030729.

Rubio, F., Llopis-Albert, C., Valero, F. and Suñer, J.L. (2016), "Industrial robot efficient trajectory generation without collision through the evolution of the optimal trajectory", *Robotics and Autonomous Systems*, Vol. 86, pp. 106-112.

Rubio, F., Valero, F., Sunyer, J.L. and Mata, V. (2009), "Direct step-by-step method for industrial robot path planning", *Industrial Robot: An International Journal*, Vol. 36 No. 6, pp. 594-607.

Rubio, F., Sunyer, J.L., Garrido, A. and Valero, F. (2010), "The simultaneous algorithm and the best interpolation function for trajectory planning", *Industrial Robot: The International Journal of Robotics Research and Application*, doi: 10.1108/01439911011063263.

Sariff, N. and Buniyamin, N. (2006), "An overview of autonomous mobile robot path planning algorithms", *4th Student Conference on Research and Development*, doi: 10.1109/SCORED.2006.4339335

Schittkowski, K. (2010), "NLPQLPA Fortran implementation of a sequential quadratic programming algorithm with distributed and non-monotone line search", Report, Department of Computer Science, University of Bayreuth.

Schittkowski, K. (2015), NLPQLP: a Fortran implementation of a sequential quadratic programming algorithm with distributed and non-monotone line search, User's Guide, Version 5.0.

Simba, K.R., Uchiyama, N. and Sano, S. (2014), "Real-time obstacle-avoidance motion planning for autonomous mobile robots", *Australian Control Conference*.

Simba, K.R., Uchiyama, N. and Sano, S. (2016), "Real-time smooth trajectory generation for nonholonomic mobile robots using Bézier curves", *Robotics and Computer-Integrated Manufacturing*, Vol. 41, pp. 31-42.

Singh, S., Simmons, R., Smith, T., Stentz, A., Verma, V., Yahja, A. and Schwehr, K. (2000), "Recent progress in local and global traversability for planetary rovers", *IEEE Conference on Robotics and Automation*.

Suñer, J.L., Valero, F., Ródenas, J.J. and Besa, A. (2007), *Comparación Entre Procedimientos de Solución de la Interpolación Por Funciones Splines Para la Planificación de Trayectorias de Robots Industriales*, ISBN: 978-9972-2885-31.

Tokekar, P., Karnad, N. and Isler, V. (2014), "Energy-optimal trajectory planning for car-like robots", *Autonomous Robots*, Vol. 37 No. 3, pp. 279-300, doi: 10.1007/s10514-014-9390-3.

Wang, D. and Qi, F. (2001), "Trajectory planning for a four-wheel-steering vehicle", *Proceedings of the 2001 IEEE International Conference on Robotics & Automation*, available at: www.ntu.edu.sg/home/edwwang/confpapers/wdwicar01.pdf

Yahja, A., Singh, S. and Brumitt, B.L. (1998), "Framed-quadtree path planning for mobile robots operating in sparse environments", *IEEE Conference on Robotics and Automation (ICRA)*, *Leuven*.

## Corresponding author

**Francisco Rubio** can be contacted at: frubio@mcm.upv.es