IMA Journal of Mathematical Control and Information (2019) **00**, 1–18
doi:10.1093/imamci/dnz016

Linear algebra-based controller for trajectory tracking in mobile robots with additive uncertainties estimation

G. J. E. Scaglia, M. E. Serrano*, S. A. Godoy and F. Rossomando
National Council for Scientific Research (CONICET) and the National University of San Juan, Argentine

[Received on 28 November 2017; revised on 1 March 2019; accepted on 5 April 2019]

This paper addresses trajectory tracking problem in mobile robots considering additive uncertainties. The controller design method is based on linear algebra theory. Numerical estimation techniques are used to estimate the uncertainty value in each sample time. The controller is calibrated by stochastic way using the Monte Carlo Experiment. In addition, the proof of convergence to zero of the tracking error is included. The theoretical results are validated by simulation and experimental tests. The controller proposed shows that it can be used to reduce the effect of additive uncertainties in the tracking error.

Keywords: tracking control; uncertainties estimation; nonlinear system; linear algebra; mobile robot.

1. Introduction

Mobile robots are widely used due to their simplicity, easiness of control and flexibility. These characteristics have attracted attention of the control community in the past years (Liu *et al.*, 2014; Miah & Gueaieb, 2014; Shen *et al.*, 2014; Toibero *et al.*, 2009). In general, in practical situations to complete an assigned task, it is necessary to solve a path-tracking problem or a trajectory tracking problem. Path tracking is all about following a predefined path, which does not involve time as a constraint. On the contrary, trajectory tracking involves time as a constraint, meaning that the robot has to be at a certain point at a certain time. Sometimes, in real applications, simple goals become complicated tasks to perform due to inevitable uncertainties that appear when controlling mobile robots. The control problem under uncertainties in mobile robots has been steadily considered (Fang & Chao-Li, 2011; Mohareri *et al.*, 2012; Park *et al.*, 2009).

Several control methodologies have been proposed for tracking trajectory, some of which are based on either the kinematic or the dynamic models of vehicles (Do & Pan, 2006; Lee *et al.*, 2001), depending on the operative speed and the precision of the dynamic model. Chwa (2004) proposes two controllers called position and heading controller, respectively. The first controller assures the position tracking and the second one is activated when the tracking error is little enough and tracking reference does not change its position. This reduces the error over the vehicle orientation at the end of the path. Yue *et al.* (2012) presents a double closed-loop control structure with a disturbance observer. This combining structure could globally realize postures-tracking stabilization and has disturbance adaptive capability. On the outer loop, a new function having sinusoidal and inverse tangent characteristics, which are continuously differentiable, is applied to construct the velocity controller by using backstepping technology.

In the literature, classical control methods have been used to control this type of systems. Normey-Rico *et al.* (2001) implements a PID to follow step change references with acceptable results. However, its performance decreases when it follows variable references or when additive uncertainties

appear. An alternative to improve these results is to combine a simple control structure (PID or feedback control) with complex and robust schemes, such as adaptive control and neural networks (Damodaran *et al.*, 2017; Gu & Hu, 2002; Park *et al.*, 2016; Ye, 2008). This improvement compensates these uncertainties and allows variable reference tracking. However, in practical implementation are less feasible to apply.

Furthermore, there are many results that are based on the look-ahead methods (Das & Kar, 2006; Martins *et al.*, 2008), where, instead of the center of mass in the wheeled mobile robots, the intersection point of a straight line passing through the middle of the vehicle and an axis of the two wheels is chosen in the configuration of the posture to make use of the feedback linearization technique. However, this approach has the following problem: as the distance between the center of mass and the intersection point becomes larger, the tracking performance will deteriorate. On the other hand, when it becomes smaller, the control input tends to become much larger as it involves the inverse of almost the singular matrix. Thus, it is also desirable to develop the analytic uncertain kinematic model that adopts the center of mass as the configuration of the posture.

Scaglia *et al.* (2009) proposes a novel trajectory tracking controller. The originality of this control approach is based on the application of linear algebra for trajectory tracking, where the control actions are obtained by solving a system of linear equations. The methodology developed for tracking the desired trajectory (x_{ref}, y_{ref}) is based on determining the trajectories of the remaining state variables when the tracking error tends to zero. These states are determined by analyzing the conditions so that the system of linear equations has an exact solution. This design technique has been applied successfully in several systems (Gandolfo *et al.*, 2014; Rosales *et al.*, 2011; Scaglia *et al.*, 2014; Serrano *et al.*, 2014). In addition, this methodology has been extended to the trajectory tracking problem with velocities limitation (Serrano *et al.*, 2016, 2017). However, in the aforementioned works additive uncertainty has not been considered in the controller design stage.

This paper provides a positive answer to the challenging problem of designing controllers for trajectory tracking in multivariable nonlinear systems considering additive uncertainty. This problem has attracted the interest of many researchers and several solutions are reported in the literature (see for instance Farooq *et al.*, 2014; Lee *et al.*, 2009; Wang, 2012 and many others). However, the proposed controllers are rather complicated to understand and require advanced control knowledge for their implementation.

This work proposes a general and simple control strategy based on linear algebra for trajectory tracking in mobile robots considering additive uncertainties. In order to deal with this issue, a new term representing the uncertainty is added in the mobile robot model. Then, this term is approximated through an interpolation method. In this work is assumed that the additive uncertainty is continuous, with continuous derivatives. Thus, Newton's backward difference is used to estimate the uncertainty value at the next sample time. At each period, the prediction error made in the previous sample time is calculated and then, using this value and the past ones, it is estimated the prediction error at the current instant. Finally, a new control law is obtained analyzing the condition for a system of linear equations has an exact solution. This novel methodology allows us to reject the polynomial uncertainties and calculate the control actions without the need to implement an observer.

The main contribution of this work is that the proposed methodology is based upon easily understandable concepts, and there is no need for complex calculations to attain the control signal. The methodology proposed here yields trajectory tracking controllers with low computational cost, small tracking error and low control effort. Additionally, in comparison with Scaglia *et al.* (2014) and Serrano *et al.* (2017), our approach considers additive uncertainty in the problem formulation in order to reduce its effect in the tracking error. The proof of the zero convergence of the tracking error under uncertainty is

another main contribution of this work. In addition, our approach can also be implemented embedded (it does not compute higher order derivate or trigonometric exponentiation). Thus, the developed algorithm is easier to implement in a real system because the use of discrete equations allows direct adaptation to any computer system or programmable device running sequential instructions at programmable clock speed. Another contribution of this paper is the application of Monte Carlo (MC)-based sampling experiment in the simulations for controller tuning. The theoretical results are compared and validated by simulations and experimentation in a PIONEER 3AT (P3-AT).

The paper is organized as follows. Section 2 presents the kinematic model of the mobile robot. Section 3 describes the formulation of the proposed control algorithm. In Sections 4 and 5, simulation, experimental results and their discussion are presented. Finally, conclusions are detailed in Section 6.

2. Kinematic model of the mobile robot

A nonlinear kinematic model for a mobile robot will be used; this model has been used in several recent papers by other authors (Jung *et al.*, 2012; Scaglia *et al.*, 2009; Toibero *et al.*, 2009; Yang *et al.*, 2016), represented by (2.1). In (2.1) V is the linear velocity of the mobile robot, W is the angular velocity of the mobile robot, (x, y) represents the Cartesian position and θ is the orientation of the mobile robot.

$$\begin{cases} \dot{x} = V\cos\theta \\ \dot{y} = V\sin\theta \\ \dot{\theta} = W \end{cases} \tag{2.1}$$

Note that the dynamics of the mobile, as well as those from the actuators, are not initially considered. They will be taken into account later on as uncertainties in the model (2.1).

3. Control design

Then the aim is to find the values of V and W so that the mobile robot may follow a pre-established trajectory (x_d, y_d) with a minimum error. Through the Euler's approximation of the kinematic model of the mobile robot (2.1), the following set of equations are obtained:

$$\mathbf{z}_{n+1} = \begin{bmatrix} x_{n+1} \\ y_{n+1} \\ \theta_{n+1} \end{bmatrix} = \begin{bmatrix} x_n + T_s V_n \cos(\theta_n) \\ y_n + T_s V_n \sin(\theta_n) \\ \theta_n + T_s W_n \end{bmatrix} = \hat{\mathbf{f}}\left(\mathbf{z}_n, \mathbf{u}_n\right), \tag{3.1}$$

where (x_n, y_n) is the Cartesian position of the mobile robot at time nT_s, V_n and W_n are the linear velocity and angular velocity at the instant $n \in \{1, 2, 3...\}$ and θ_n is the orientation of the mobile robot. The sample time is T_s. Consider a real plant $\mathbf{z}_{n+1} = \mathbf{f}(\mathbf{z}_n, \mathbf{u}_n)$, the additive uncertainty can be expressed by $\mathbf{E}_n = \mathbf{f}(\mathbf{z}_n, \mathbf{u}_n) - \hat{\boldsymbol{f}}(\mathbf{z}_n, \mathbf{u}_n)$, where $\hat{\boldsymbol{f}}(\mathbf{z}_n, \mathbf{u}_n)$ is the discrete-time nonlinear model of the system. Note that if, as it will be assumed, $\mathbf{z}$ and $\mathbf{u}$ are bounded and $\mathbf{f}$ is Lipschitz, then $\mathbf{E}_n$ can be modeled as a bounded uncertainty (Mayne *et al.*, 2000; Michalska & Mayne, 1993). Now, an additive uncertainty is

incorporated into the system model

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \\ \theta_{n+1} \end{bmatrix} = \begin{bmatrix} x_n \\ y_n \\ \theta_n \end{bmatrix} + T_s \begin{bmatrix} \cos(\theta_n) & 0 \\ \sin(\theta_n) & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_n \\ W_n \end{bmatrix} + \underbrace{\begin{bmatrix} E_{x,n} \\ E_{y,n} \\ E_{\theta,n} \end{bmatrix}}_{\mathbf{E_n}} = f(\mathbf{z_n}, \mathbf{u_n}), \tag{3. 2}$$

where $\mathbf{E_n}$ is the additive uncertainty. The uncertainty $\mathbf{E_n}$ results from sampling the uncertainty at the n instant. Here is assumed that uncertainty is a continuous function; thus, $\mathbf{E_n}$ can be used to model perturbed systems as well as a wide class of model mismatches, which are the most common source of uncertainty in the bibliography. Then as $\mathbf{E_n}$ is a continuous function, admits a Taylor series development around a known value that in this work is the uncertainty in the previous instant. With these considerations, the first-order difference of $\mathbf{E_n}$ is defined as $\nabla\mathbf{E}_n = \mathbf{E}_n - \mathbf{E}_{n-1}$, the second-order difference as $\nabla^2\mathbf{E}_n = \nabla\left(\nabla\mathbf{E}_n\right) = \nabla\left(\mathbf{E}_n - \mathbf{E}_{n-1}\right) = \mathbf{E}_{\mathrm{n}} - 2\mathbf{E}_{n-1} + \mathbf{E}_{\mathrm{n\text{-}2}}$ and as a rule, the q-th order difference is defined as $\nabla^q\mathbf{E}_n = \nabla\left(\nabla^{q-1}\mathbf{E}_n\right)$. Then the q-th difference of a $q-1$-order polynomial is zero.

The value at the n instant of the uncertainty cannot be known; however, the values of the uncertainty at the previous sampling times can be calculated. Thus, according to Taylor, it is possible to estimate the value of the uncertainty at the current time considering the previous values. Next, an approach to eliminate the influence of the additive uncertainty on the tracking error is proposed.

This paper aims to estimate the uncertainty $\mathbf{E_n}$ using $\hat{\mathbf{E}}_\mathbf{n}$ defined by (3.3); this is calculated considering the Newton's backward difference formula of $\mathbf{E_n}$ (see Hildebrand, 1987). Note that (3.4) is obtained considering the same order of the uncertainty $\mathbf{E_n}$ (m-order).

$$\hat{\mathbf{E}}_\mathbf{n} = \begin{bmatrix} \hat{E}_{x,n} \\ \hat{E}_{y,n} \\ \hat{E}_{\theta,n} \end{bmatrix} \tag{3}$$

where

$$\begin{aligned} \hat{E}_{x,n} &= E_{x,n-1} + \tfrac{(E_{x,n-1} - E_{x,n-2})}{1!} + \ldots + \sum_{i=0}^{m} \binom{m}{i} (-1)^i \tfrac{E_{x,n-m-i-1}}{m!} \\ \hat{E}_{y,n} &= E_{y,n-1} + \tfrac{(E_{y,n-1} - E_{y,n-2})}{1!} + \ldots + \sum_{i=0}^{m} \binom{m}{i} (-1)^i \tfrac{E_{y,n-m-i-1}}{m!} \\ \hat{E}_{\theta,n} &= E_{\theta,n-1} + \tfrac{(E_{\theta,n-1} - E_{\theta,n-2})}{1!} + \ldots + \sum_{i=0}^{m} \binom{m}{i} (-1)^i \tfrac{E_{\theta,n-m-i-1}}{m!} \end{aligned} \tag{3.4}$$

and

$$\begin{aligned} \hat{E}_{x,n} &= \sum_{j=0}^{m} \sum_{i=0}^{j} \binom{j}{i} (-1)^i \tfrac{E_{x,n-j-i-1}}{j!} \\ \hat{E}_{y,n} &= \sum_{j=0}^{m} \sum_{i=0}^{j} \binom{j}{i} (-1)^i \tfrac{E_{y,n-j-i-1}}{j!} \\ \hat{E}_{\theta,n} &= \sum_{j=0}^{m} \sum_{i=0}^{j} \binom{j}{i} (-1)^i \tfrac{E_{\theta,n-j-i-1}}{j!}. \end{aligned} \tag{3.5}$$

The estimated variables $\hat{x}_n$, $\hat{y}_n$ and $\hat{\theta}_n$ represent the estimated values of the variables x_n, y_n and θ_n and are calculated according to (3.6),

$$\begin{cases} \hat{x}_n = x_{n-1} + T_s V_{n-1} \cos(\theta_{n-1}) \\ \hat{y}_n = y_{n-1} + T_s V_{n-1} \sin(\theta_{n-1}) \\ \hat{\theta}_n = \theta_{n-1} + T_s W_{n-1} \end{cases}, \tag{3.6}$$

where x_{n-1}, y_{n-1} and θ_{n-1} correspond to the position and orientation of the robot at time $n-1$, and V_{n-1} and W_{n-1} correspond to the values of the control actions calculated at time $n-1$. It is easy to demonstrate that if $\nabla^m \mathbf{E}_n = 0$ and $\hat{\mathbf{E}}_\mathbf{n}$ is calculated according to (3.3) and (3.4) $\Rightarrow \mathbf{E}_n = \hat{\mathbf{E}}_\mathbf{n}$.

Now, the estimation of additive uncertainties $\hat{\mathbf{E}}_\mathbf{n}$ (computed by (3.3)) is incorporated into the system model

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \\ \theta_{n+1} \end{bmatrix} = \begin{bmatrix} x_n \\ y_n \\ \theta_n \end{bmatrix} + T_s \begin{bmatrix} \cos(\theta_n) & 0 \\ \sin(\theta_n) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V_n \\ W_n \end{bmatrix} + \underbrace{\begin{bmatrix} \hat{E}_{x,n} \\ \hat{E}_{y,n} \\ \hat{E}_{\theta,n} \end{bmatrix}}_{\hat{E}_n}. \tag{3.7}$$

Below, a control law that allows position tracking is proposed.

If the desired trajectory is known $x_{d,n+1}$ and $y_{d,n+1}$, then x_{n+1}, y_{n+1} and θ_{n+1} in (3.1) may be replaced by $x_{d,n+1}$, $y_{d,n+1}$ and $\theta_{ez,n+1}$, respectively, where θ_{ez} is calculated according to (3.14) and represents the necessary orientation so that the tracking error tends to zero (see Proof of Theorem 3.1). But, in order to obtain a smooth position tracking, the desirable next state $(x_{d,n+1}, y_{d,n+1}, \theta_{ez,n+1})$ is not necessarily the new reference state value $(x_{ref,n+1}, y_{ref,n+1}, \theta_{ez,n+1})$. These states are replaced assuming an approach proportional to the error

$$\begin{bmatrix} x_{d,n+1} \\ y_{d,n+1} \\ \theta_{d,n+1} \end{bmatrix} = \begin{bmatrix} x_{ref,n+1} - k_v(x_{ref,n} - x_n) \\ y_{ref,n+1} - k_v(y_{ref,n} - x_n) \\ \theta_{ez,n+1} - k_w(\theta_{ez,n} - \theta_n) \end{bmatrix}; \quad \mathbf{e}_n = \begin{bmatrix} e_{x,n} \\ e_{y,n} \\ e_{\theta,n} \end{bmatrix} = \begin{bmatrix} x_{ref,n} - x_n \\ y_{ref,n} - y_n \\ \theta_{ref,n} - \theta_n \end{bmatrix}, \tag{3.8}$$

where k_v and k_w are design positive parameters, $(0 < k_v < 1)$ and $(0 < k_w < 1)$. Note that

- if $k_v = 0$, $(x_{d,n+1} = x_{ref,n+1})$, the goal is to reach the reference trajectory in one step;
- if $0 < k_v < 1$, the error will decrease.

Thus, the approach proposed in (3.8) is applied in order to get a smooth trajectory. The same analysis can be applied to $y_{d,n+1}$ and $\theta_{d,n+1}$.

In addition, we define

$$\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta\theta \end{bmatrix} = \begin{bmatrix} x_{ref,n+1} - k_v(x_{ref,n} - x_n) - x_n \\ y_{ref,n+1} - k_v(y_{ref,n} - y_n) - y_n \\ \theta_{ez,n+1} - k_w(\theta_{ez,n} - \theta_n) - \theta_n \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} \cos(\theta_n) & 0 \\ \sin(\theta_n) & 0 \\ 0 & 1 \end{bmatrix}. \tag{3.9}$$

REMARK 3.1 The value of the difference between the reference and the real trajectory shall be called tracking error. It is given by $e_{x,n} = x_{ref,n} - x_n$ and $e_{y,n} = y_{ref,n} - y_n$. The tracking error is represented by $\|e_n\| = \sqrt{e_{x,n}^2 + e_{y,n}^2}$.

Then $(x_{n+1} - x_n, y_{n+1} - y_n, \theta_{n+1} - \theta_n)$ in (3.7) are replaced by the corresponding terms according to (3.9), and the system (3.10) is obtained.

$$\mathbf{B}\underbrace{\begin{bmatrix} V_n \\ W_n \end{bmatrix}}_{u} = \underbrace{\frac{1}{T_s}\left(\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta\theta \end{bmatrix} - \hat{\mathbf{E}}_{\mathbf{n}}\right)}_{b} \tag{3.10}$$

System (3.10) is a system with three equations and two unknowns. Note that the first two rows in (3.10) involve a unique variable V_n. In order that system (3.10) has an exact solution, vector $\mathbf{b}$ must be contained in the space formed by the columns of $\mathbf{B}$ $C(\mathbf{B})$; thus, $\mathbf{b}$ must be a linear combination of the columns of $\mathbf{B}$. Then, (3.11) the real constants a_1 and a_2 must exist.

$$a_1\begin{bmatrix} \cos(\theta_n) \\ \sin(\theta_n) \\ 0 \end{bmatrix} + a_2\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta\theta \end{bmatrix} - \hat{\mathbf{E}}_{\mathbf{n}}; \quad a_1, a_2 \in \Re \tag{3.11}$$

By separating the first two rows it is possible to find the condition to be fulfilled by orientation θ such that the system (3.11) has exact solution

$$a_1\begin{bmatrix} \cos(\theta_n) \\ \sin(\theta_n) \end{bmatrix} = \begin{bmatrix} (\Delta x - \hat{E}_{x,n}) \\ (\Delta y - \hat{E}_{y,n}) \end{bmatrix} \tag{3.12}$$

Thence,

$$\frac{\sin(\theta_n)}{\cos(\theta_n)} = \frac{\Delta y - \hat{E}_{y,n}}{\Delta x - \hat{E}_{x,n}}. \tag{3.13}$$

In order to get a single solution, the direction $\theta_{ez,n+1}$ must be

$$\theta_{ez,n+1} = \operatorname{atan}\frac{y_{ref,n+1} - k_v(y_{ref,n} - y_n) - y_n - \hat{\boldsymbol{E}}_{y,n}}{x_{ref,n+1} - k_v(x_{ref,n} - x_n) - x_n - \hat{\boldsymbol{E}}_{x,n}}, \tag{3.14}$$

where θ_{ez} represents the necessary orientation such that the mobile robot reaches and follows the reference trajectory. As can be seen in comparison with the previous linear algebra controllers (Scaglia *et al.*, 2009; Serrano *et al.*, 2017), the column space of matrix $\mathbf{b}$ is modified to include uncertainty estimation $\hat{\mathbf{E}}_{\mathbf{n}}$. This new approach allows obtaining a control law that decreases the uncertainty influence on the tracking error. This desired orientation for the mobile robot guarantees that the nonholonomic system reaches the pre-established trajectory.

In order to find the value of the control actions to be applied in each sampling time, system (3.10) must be solved. The optimum solution of it from mean squares is obtained from normal equation

($\mathbf{B^T Bu} = \mathbf{B^T b}$) (see Strang, 2005). Then considering (3.10) and (3.14) the controller proposed for the mobile robot is given in (3.15)

$$\begin{bmatrix} V_n \\ W_n \end{bmatrix} = \begin{bmatrix} \left(\frac{\Delta x - \hat{E}_{x,n}}{T_s}\right)\cos(\theta_{ez,n}) + \left(\frac{\Delta y - \hat{E}_{y,n}}{T_s}\right)\sin(\theta_{ez,n}) \\ \frac{\Delta\theta - \hat{E}_{\theta,n}}{T_s} \end{bmatrix}. \tag{3.15}$$

THEOREM 3.1 The mobile robot described by (3.1) controlled by the approach described above leads to an asymptotic stable system, where an appropriate sampling period T_s is considered and taking $\mathbf{E_n}$ unknown and each component represented by an m-order polynomial.

The values of V_n and W_n represent the control actions necessary to meet the control goal. The proof of Theorem 3.1 and the convergence to zero of tracking errors are shown in the Appendix.

4. Simulation results

The simulation results for the performance evaluation of the position tracking controller proposed in the previous section are presented here. As already discussed, the controlled system behavior depends on the parameters k_v and k_w. Thus, in this work, and in order to determine values for the parameters of the controller for $m = 0, 1, 2$, the MC method is applied.

4.1. *MC randomized algorithm*

In the field of systems and control, probabilistic methods have been found useful especially for problems related to the robustness of uncertain systems (Tempo & Ishii, 2007). One of these methods, the MC randomized algorithm, is widely used in many fields such as the radioactive decay, systems of interacting atoms, the traffic on roads, etc. (Barat *et al.*, 2006). In the control area, MC methods allow one to estimate an expectation value and they provide effective tools for the analysis of probabilistically robust control schemes.

Because of its nature, these types of algorithms can give an erroneous result with a non-zero probability. So, it could be posed the natural question of how many simulations must be performed to be sure of finding the correct answer. Under a sufficiently large sample size N, a probabilistic statement can be made as shown below.

THEOREM 4.1 (Tempo & Ishii (2007)). Let $\varepsilon, \delta \in (0, 1)$, where ε is an *a priori* specified accuracy, and $1 - \delta$, the confidence interval. If

$$N \geqslant \left\lceil \frac{\log \frac{1}{\delta}}{\log \frac{1}{1-\varepsilon}} \right\rceil \tag{4.1}$$

then, the empirical maximum satisfies the following inequality with probability greater than $1 - \delta$:

$$\text{Prob}_{\Delta}\left\{J(\Delta) \leqslant \hat{J}_{\max}\right\} \geqslant 1 - \varepsilon. \tag{4.2}$$

That is,

$$\text{Prob}_{\Delta(1,\ldots,N)}\left\{\text{Prob}_{\Delta}\left\{J(\Delta) \leqslant \hat{J}_{\max}\right\} \geqslant 1 - \varepsilon\right\} > 1 - \delta, \tag{4.3}$$

where J is the performance function and $\hat{J}_{\max}$, the *empirical maximun*. For further details, see Tempo & Ishii (2007).

The theorem says that the empirical maximum is an estimate of the true value within an *a priori* specified accuracy ε with confidence, δ if the sample size N satisfies (4.1). The algorithm may not produce an approximately correct answer, but the probability of this event is no greater than δ. It is worthy to emphasize that, in Theorem 4.1, the sample size N is finite and moreover is not dependent on the size of the uncertain set $\mathbf{B}$, the structured set of uncertainty matrices and the probability density function $f_{\Delta}(\Delta)$, but only on ε and δ. In the next section, (4.1) is used to estimate the number of simulations.

4.2. *Monte Carlo experiment*

In this subsection, the MC method is applied to select an appropriate set of controller parameters. Although the optimum is not guaranteed, the Monte Carlo Experiment (MCE) provides an approximate solution based on a large number of trials (M). In this paper, it is adopted a confidence value (δ) of 0.01 and an accuracy of 0.007 (ε). Then from (4.1), it is necessary to make 1,000 simulations. Hence, 1,000 values of each parameter ranging from 0 to 1 were simulated. This parameter range ensures convergence to zero tracking error (see proof of Theorem 3.1).

The aim of MCE is to find the parameter values (k_v and k_w) optimizing a defined cost function. An idea widely used in the literature is to consider the cost incurred by the tracking error and the control effort (Blažič, 2011; Cheein & Scaglia, 2013). Let Φ be a desired trajectory, where $\#\Phi$ is the number of points of such trajectory. Let $\sum_{i=0}^{\#\Phi} \frac{1}{2}\left(x_{ref,i} - x_i\right)^2$ the quadratic error in the *x-coordinate*; $\sum_{i=0}^{\#\Phi} \frac{1}{2}\left(y_{ref,i} - y_i\right)^2$ the quadratic error in the *y-coordinate*; $\sum_{i=0}^{\#\Phi} \frac{1}{2}\left(V_i\right)^2$ the effort of the traction control; and $\sum_{i=0}^{\#\Phi} \frac{1}{2}\left(W_i\right)^2$ the effort in the heading control. Thus, the cost function can be represented by the combination of all of them weighted with the aim of reducing the tracking error avoiding high control actions values

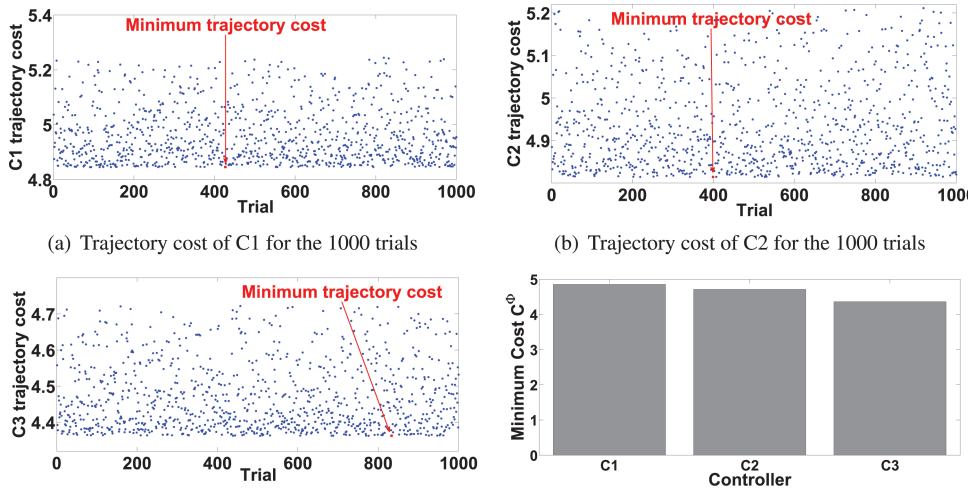
$$C^{\Phi} = \frac{1}{2}\sum_{n=0}^{\#\Phi}\left(\mathbf{e}_n{}^T P \mathbf{e}_n + \mathbf{u}_n{}^T Q \mathbf{u}_n\right) \tag{4.4}$$

with $\mathbf{e}_n{}^T = \begin{bmatrix} e_{x,n} & e_{y,n} \end{bmatrix}$, $\mathbf{u}_n{}^T = \begin{bmatrix} V_n & W_n \end{bmatrix}$, $P = 0.9$ and $Q = 0.1$.

Thus, the objective is to find k_v and k_w in such way that C^{Φ} is minimized. In order to do so, in this work the MCE is carried considering $m = 0, 1, 2$ in (3.4). The MC experiment allows finding empirically the parameter values minimizing the cost function.

The MCE considerations are as follows:

- The model mismatch between (2.1) and the Pioneer actual behavior is represented by the uncertainty $\mathbf{E}_n$, with high (unknown) order difference.
- The simulations are performed using MatLab software platform and Mobile Sim program provided by the manufacturer Pioneer Mobile Robot. The simulations are performed with $m = 0, 1, 2$ and will be called C1, C2 and C3, respectively. For each controller 1,000 simulations are run.
- All simulations are implemented with the same desired trajectory Φ. In this section, a sinusoidal trajectory is considered. The sampling time used is $T_s = 0.1s$.



(a) Trajectory cost of C1 for the 1000 trials

(b) Trajectory cost of C2 for the 1000 trials

(c) Trajectory cost of C3 for the 1000 trials

(d) Minimum cost obtained for each controller

FIG. 1. Results of MCE.

- For each simulation, the controller parameters are chosen in a random way, such that $0 < k_v < 1$ and $0 < k_w < 1$. The lower and upper bounds are chosen such that tracking errors tend to zero (see proof of Theorem 3.1).
- The cost C^{Φ} for each controller is calculated when the robot reaches the desired trajectory. Due to the unknown character of the uncertainty, the steady-state error will be affected by the higher-order uncertainty differences.

Figure 1(a) shows the results of the 1,000 simulations when $m = 0$, C1 in the sequel. The results show the values taken by the cost function for each simulation; scattered values are obtained due to the randomness with which the parameters were chosen in each simulation. The minimum cost obtained for C1 is $C^{\Phi} = 4.8449$. Figure 1(b) shows that the lowest cost obtained by C2, with $m = 1$, corresponds to $C^{\Phi} = 4.7514$. Figure 1(c) shows the results of the cost function for 1,000 trials when using the controller C3 proposed in this work, $m = 2$; this assumes that uncertainty $\mathbf{E}_n$ is of order 2. For this controller, the lowest cost obtained is $C^{\Phi} = 4.3623$. By inspection, it can be seen that, in general, all the cost obtained by C2 is lower than the minimum value obtained for $m = 0$ (C1). In addition, the costs obtained by C1 and C2 are higher than those obtained by C3. The minimum cost obtained for each controller is resumed in Fig. 1(d).

The analysis of the results shows that the performance of the controller improves as m increases. Thus, the results obtained by the MCE to choose the controller parameters verify the theoretical results obtained in the previous section. Table 1 shows the summary of the results obtained with each controller.

5. Experimental results

In this section, two experiments are reported to demonstrate the operation of the proposed controller. The experiments were performed using a P3-AT mobile robot. The P3-AT mobile robot includes an estimation system based on odometric-based positioning system. This system uses 100 tick encoders

TABLE 1 *Simulations summary of the MCE*

Controller	Minimum cost	Controller's parameters
C1	$C^{\Phi} = 4.8449$	$k_v = 0.932$ and $k_w = 0.928$
C2	$C^{\Phi} = 4.7514$	$k_v = 0.939$ and $k_w = 0.91$
C3	$C^{\Phi} = 4.3623$	$k_v = 0.912$ and $k_w = 0.943$

FIG. 2. The pionner 3AT mobile robot and the laboratory facilities.

with inertial correction recommended for dead reckoning to compensate for skid steering. Updating through external sensors is necessary. The P3-AT has a PID velocity controller used to maintain the velocities of the mobile robot at the desired value. This problem is separated from the strategy of trajectory tracking and it is not considered in this paper (Normey-Rico *et al.*, 2001, 1999).

In order to compare the approaches proposed in this work (C1, C2 and C3) the original controller developed in Scaglia *et al.* (2009) is also implemented in the P3-AT (C4 hereinafter). Figure 2 shows the P3-AT and the laboratory facilities where the experiments were carried out. The controller parameters obtained in the previous section (Table 1) are used.

The designing details of the controller C4 can be found in its respective reference, and only the experimental results without a theoretical analysis of the properties of each controller are shown here. For those, Scaglia *et al.* (2009) offer a deep insight into the controller design.

The controllers implemented for comparison are the following:

- Controller proposed in this paper when $m = 0$ in (3.4), C1 in the sequel (sky blue line in plots).
- Controller proposed in this paper when $m = 1$ in (3.4), C2 in the sequel (brown line in plots).
- Controller proposed in this paper when $m = 2$ in (3.4), C3 in the sequel (green line in plots).
- The controller developed in Scaglia *et al.* (2009), C4 in the sequel (blue line in plots).

In order to quantify the performance of each controller in every test, the mean and variance of the tracking errors are calculated. In addition, the trajectory cost (C^{Φ}) will be computed and plotted in order to compare the results, and it will be calculated according to (4.4). The mean and error variance will be

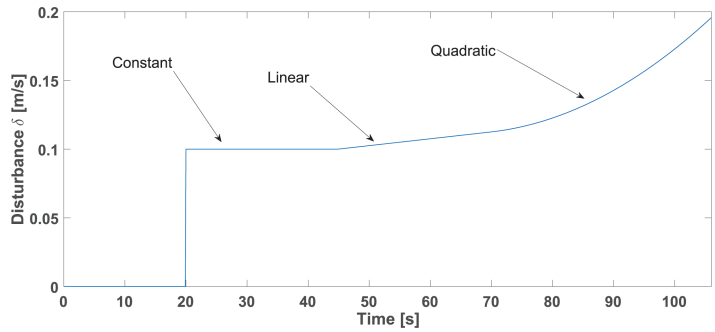


FIG. 3. Kinematic disturbance.

calculated according to (5.1) and (5.2), respectively (Jazwinski, 2007).

$$\bar{X} = \frac{\sum_{i=1}^{N} e_i}{N} \tag{5.1}$$

$$\sigma^2 = \frac{\sum_{i=1}^{N} \left(e_i - \bar{X}\right)^2}{N}, \tag{5.2}$$

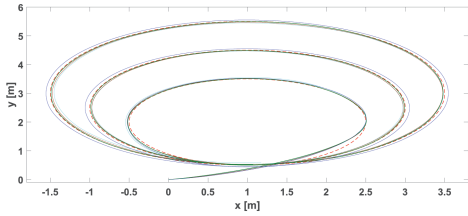
where N is the number of sampling periods performed during the experience and e_i represents the tracking error at the i instant (see Remark 3.1).

5.1. *Curvature test*

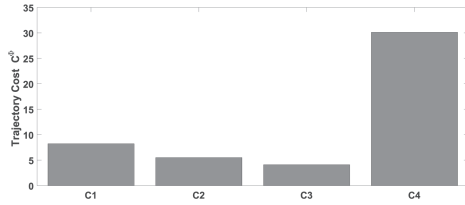
The first one is a curvature test in which the performance of each controller, using different circle-shaped trajectories, is evaluated as recommended in Batavia *et al.* (n.d.). Three circle-trajectories are used in this work, with different radius. The internal trajectory has a radius of $r = 1.5\,m$, the medium one $r = 2.5\,m$ and the last one $r = 3.5\,m$. The initial robot position is the system origin and the reference trajectory begins in the position $(x_{ref}, y_{ref}) = (1\,m, 0.5\,m)$. The sample time used is $T_s = 0.1\,s$. In this test, a kinematic disturbance in (2.1) is assumed to exist (Chwa, 2012). The disturbance δ is introduced according to (5.3), its time variation can be show in Fig. 3.

$$\begin{cases} \dot{x} = (V + \delta)\cos\theta \\ \dot{y} = (V + \delta)\sin\theta \\ \dot{\theta} = W \end{cases} \tag{5.3}$$

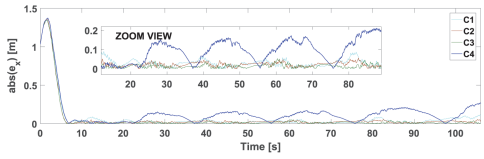
The reference trajectory and the results of the controllers are shown in Fig. 4(a). As can be seen, all controllers reach and follow the desired trajectory. Figure 4(c and d) and shows the plots of the value of the tracking errors in the *x-coordinate* and *y-coordinate* according to each controller used in the test for the three curvatures shown in Fig. 4(a). In Fig. 4(c and d) and, it is possible to appreciate that both errors remain bounded and close to zero when the robot reaches the reference trajectory. The lowest cost is obtained by C3 as it can be seen in Fig. 4(b).



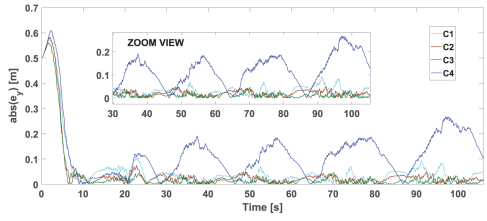
(a) Desired Trajectory and robot position for C1, C2, C3 and C4



(b) Cost of the trajectory tracking



(c) Tracking error in x



(d) Tracking error in y

FIG. 4. Results of curvature test.

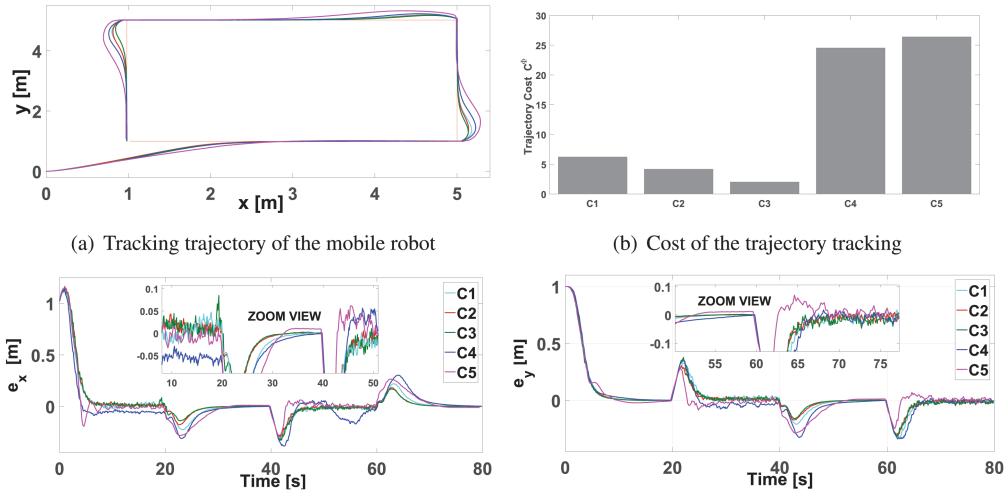
TABLE 2 *Results of the curvature test*

Controller	**Curvature test**		
	$\bar{X}[m]$	$\sigma^2[m^2]$	C^{Φ}
C1	0.1054	0.0609	7.29
C2	0.0795	0.0610	5.72
C3	0.0601	0.0601	4.86
C4	0.2066	0.0061	31.91

Table 2 shows, quantitatively, the results of the experience. The C3 controller has the lowest mean with a value of 1.9 cm which is smaller compared to the size of the robot. Furthermore, the tracking error variance is very low for the proposed controllers (C1, C2 and C3).

5.2. *Squared trajectory*

In different applications, the trajectory to be followed by the robot is usually re-planned. This strategy can be used in applications such as obstacle avoidance and contour-following. So if the danger of collision is large, the trajectory to be followed by the robot is modified abruptly and the robot must follow that path to avoid a collision. Thus, the controller performance when the trajectory changes abruptly will be analyzed. In addition, in order to test the limits of our formulation, and as recommended by Roth & Batavia (2002), a square trajectory was chosen. Specifically, the robot has to track a square trajectory. The square reference trajectory is generated with a constant linear velocity of $V = 0.3\,m/s$. The initial position of the robot is at the system origin and the trajectory begins in the position $(x_{ref}(0), y_{ref}(0)) = (1\,m, 1\,m)$.



(a) Tracking trajectory of the mobile robot

(b) Cost of the trajectory tracking

(c) Tracking error in x

(d) Tracking error in y

FIG. 5. Results of square trajectory.

In order to extend the comparison with another approach of the bibliography, the controller developed in Michałek & Kozłowski (2012) was implemented in the P3-AT; these will be called C5. The C5 structure shown in Michałek & Kozłowski (2012) results from simple geometrical interpretations related to the unicycle kinematics, from the introduction of the so-called convergence vector field and from the decomposition of the control process into the orienting and pushing subprocesses and it is shown below.

$$
\begin{aligned}
h_x &= k_p e_x + \dot{x} \\
h_y &= k_p e_y + \dot{y} \\
r &= h_x \cos\psi + h_y \sin\psi \\
u &= k_\psi(\psi_{ref} - \psi) + \dot{\psi}_{ref}.
\end{aligned}
$$

The controller's parameters used in the experimentation(k_p, k_ψ) are obtained from Michałek & Kozłowski (2012).

Figure 5(a) shows the trajectory followed by the P3-AT mobile robot on the plane $x - y$ for each controller. The initial position of the mobile robot was $x = 0\,m, y = 0\,m$. It can be seen from Fig. 5(a) that the robot tends to the desired trajectory with all controllers, following it in a precise way. Figure 5(b) shows the quadratic error incurred by each controller. As can be seen, C3 presents the best performance; however, C1 an C2 have a good tracking compared with C4 and C5. Figure 5(c and d) and shows how the tracking errors remain close to zero after the robot reaches the trajectory. Figure 6 shows the reference orientation and the robot orientations for each controller. In addition, the values taken by θ_{ez} at each sampling time are shown. As it can be seen, first θ_{ez} takes different values to the reference values to ensure that the robot converges to the reference position. Then, when the robot reaches the reference position, the variable θ_{ez} takes the reference values.

Finally, Table 3 summarizes the results obtained by each controller in the square trajectory experience. As before, the proposed controllers have a better performance than C4 and C5. Moreover,

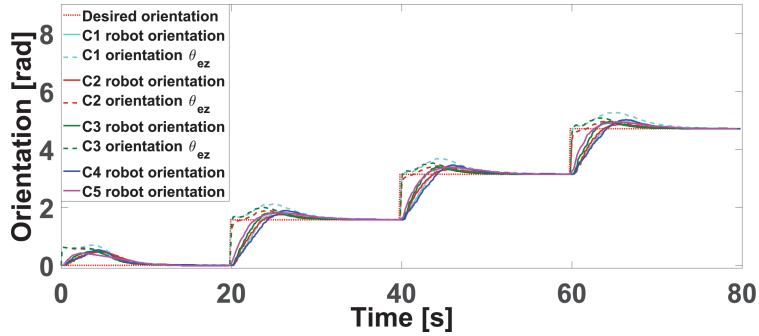


FIG. 6. Orientation of the mobile robot.

TABLE 3 *Results of the square trajectory*

Controller	Square trajectory		
	$\bar{X}[m]$	$\sigma^2[m^2]$	C^{Φ}
C1	0.026	0.00019	6.21
C2	0.018	0.00018	4.16
C3	0.014	0.00009	2.01
C4	0.132	0.0031	24.53
C5	0.1857	0.0904	26.39

it is observed that the variance values obtained are small and the mean of the errors does not exceed 2.6 cm.

6. Conclusions

A methodology based on numerical methods and linear algebra to design control algorithms for mobile robots when a multivariable nonlinear model represents the system with additive uncertainty have been presented. Novel estimation terms have been added, which are chosen based on the variation hypothesis of $\mathbf{E}_n$. The estimation $\hat{\mathbf{E}}_\mathbf{n}$ is calculated considering the same order of the uncertainty $\mathbf{E}_\mathbf{n}$ in order to avoid the influence of the modeling error to tracking error.

A contribution of this work involves the application of a MC method that successfully found the controller parameters. Several tests were carried out to demonstrate the effectiveness of the proposed methodologies. The obtained results prove the good performance of the proposed methodology, even when compared with a controller of the literature. The presented approach reduces the effect of disturbances and modeling errors in the tracking error without increasing the controller complexity in an excessive way. These results show that the tracking error decreases when the order m of $\hat{\mathbf{E}}_\mathbf{n}$ increases.

In comparison with other previous published control laws (Blažič, 2011; Damodaran *et al.*, 2017; Farooq *et al.*, 2014; Gu & Hu, 2002; Lee *et al.*, 2009; Park *et al.*, 2016; Wang, 2012; Ye, 2008) the proposed controller presents the advantages of being easy to design and more realizable in practical implementation. Thus, the algorithm can be implemented directly on the robot's microcontroller without the need to implement it on an external computer, because the calculations are simple to perform.

In addition, the method proposed here does not need a model transformation compared with Blažič (2011). In the straight line section, the reference and robot velocities, in the square trajectory test, remain constant when our approach is applied. Thus, our controller does not present the disadvantage of Blažič (2011) where the reference velocities must meet the condition of persistent excitation. In addition, our approach does not need to develop the analytic uncertain kinematic model in comparison with the methodology proposed in Das & Kar (2006) and Martins *et al.* (2008).

The mentioned characteristics make that the proposed methodology can be applied to many nonlinear systems, making it a promising technique for its application on several systems of the industry.

Acknowledgements

The authors thank the Institute of Chemical Engineering of the National University of San Juan, Argentine.

Funding

This work was partially funded by the Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET - National Council for Scientific Research), Argentina.

References

Barat, A., Ruskin, H. J. & Crane, M. (2006) Probabilistic models for drug dissolution. Part 1. Review of Monte Carlo and stochastic cellular automata approaches. *Simul. Model. Pract. Theory*, **14**, 843–856.

Batavia, P. H., Roth, S. A. & Singh, S. n.d., Autonomous coverage operations in semi-structured outdoor environments. *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. **1**. IEEE, pp. 743–749.

Blažič, S. (2011) A novel trajectory-tracking control law for wheeled mobile robots. *Rob. Auton. Syst.*, **59**, 1001–1007.

Cheein, F. A. & Scaglia, G. (2013) Trajectory tracking controller design for unmanned vehicles: a new methodology. *J. Field Robot.*, **31**, 861–887.

Chwa, D. (2004) Sliding-mode tracking control of nonholonomic wheeled mobile robots in polar coordinates. *IEEE Trans. Control Syst. Technol.*, **12**, 637–644.

Chwa, D. (2012) Fuzzy adaptive tracking control of wheeled mobile robots with state-dependent kinematic and dynamic disturbances. *IEEE Trans. Fuzzy Syst.*, **20**, 587–593.

Damodaran, S., Kumar, T. & Sudheer, A. (2017) Design and implementation of ga tuned pid controller for desired interaction and trajectory tracking of wheeled mobile robot. *Proceedings of the Advances in Robotics*. New York, NY, USA: ACM, p. 34.

Das, T. & Kar, I. N. (2006) Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots. *IEEE Trans. Control Syst. Technol.*, **14**, 501–510.

Do, K. & Pan, J. (2006) Global output-feedback path tracking of unicycle-type mobile robots. *Robot. Comput. Integr. Manuf.*, **22**, 166–179.

Fang, Y. & Chao-Li, W. (2011) Adaptive stabilization for uncertain nonholonomic dynamic mobile robots based on visual servoing feedback. *Acta Automat. Sinica*, **37**, 857–864.

Farooq, U., Hasan, K., Hanif, A., Amar, M. & Asad, M. U. (2014) Fuzzy logic based path tracking controller for wheeled mobile robots. *Int. J. Comput. Electr. Eng.*, **6**, 145–150.

Gandolfo, D., Rosales, C., Patiño, D., Scaglia, G. & Jordan, M. (2014) Trajectory tracking control of a pvtol aircraft based on linear algebra theory. *Asian J. Control*, **16**, 1849–1858.

Gu, D. & Hu, H. (2002) Neural predictive control for a car-like mobile robot. *Rob. Auton. Syst.*, **39**, 73–86.

Hildebrand, F. B. (1987) Introduction to numerical analysis. Chelmsford, MA: Courier Corporation.

Jazwinski, A. H. (2007) *Stochastic processes and filtering theory*. Courier Corporation.

Jung, E.-J., Yi, B.-J. & Yuta, S. 2012, Control algorithms for a mobile robot tracking a human in front, in '*Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*', IEEE, pp. 2411–2416.

Lee, J. H., Lin, C., Lim, H. & Lee, J. M. (2009) Sliding mode control for trajectory tracking of mobile robot in the rfid sensor space. *Int. J. Control. Autom. Syst.*, **7**, 429–435.

Lee, T.-C., Song, K.-T., Lee, C.-H. & Teng, C.-C. (2001) Tracking control of unicycle-modeled mobile robots using a saturation feedback controller. *IEEE. Trans. Control. Syst. Technol.*, **9**, 305–318.

Liu, R., Huskic, G. & Zell, A. (2014) Dynamic objects tracking with a mobile robot using passive uhf rfid tags. *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, Chicago, IL, USA: IEEE, 4247–4252.

Martins, F. N., Celeste, W. C., Carelli, R., Sarcinelli-Filho, M. & Bastos-Filho, T. F. (2008) An adaptive dynamic controller for autonomous mobile robot trajectory tracking. *Control Eng. Pract.*, **16**, 1354–1363.

Mayne, D. Q., Rawlings, J. B., Rao, C. V. & Scokaert, P. O. (2000) Constrained model predictive control: stability and optimality. *Automatica J. IFAC*, **36**, 789–814.

Miah, M. S. & Gueaieb, W. (2014) Mobile robot trajectory tracking using noisy RSS measurements: an RFID approach. *ISA Trans.*, **53**, 433–443.

Michałek, M. & Kozłowski, K. (2012) Feedback control framework for car-like robots using the unicycle controllers. *Robotica*, **30**, 517–535.

Michalska, H. & Mayne, D. Q. (1993) Robust receding horizon control of constrained nonlinear systems. *IEEE Trans. Automat. Contr.*, **38**, 1623–1633.

Mohareri, O., Dhaouadi, R. & Rad, A. B. (2012) Indirect adaptive tracking control of a nonholonomic mobile robot via neural networks. *Neurocomputing*, **88**, 54–66.

Normey-Rico, J. E., Alcalá, I., Gómez-Ortega, J. & Camacho, E. F. (2001) Mobile robot path tracking using a robust pid controller. *Control Eng. Pract.*, **9**, 1209–1214.

Normey-Rico, J. E., Gómez-Ortega, J. & Camacho, E. F. (1999) A smith-predictor-based generalised predictive controller for mobile robot path-tracking. *Control Eng. Pract.*, **7**, 729–740.

Park, B. S., Yoo, S. J., Park, J. B. & Choi, Y. H. (2009) Adaptive neural sliding mode control of nonholonomic wheeled mobile robots with model uncertainty. *IEEE Trans. Control Syst. Technol.*, **17**, 207–214.

Park, S.-H., Choi, W.-H. & Jie, M.-S. (2016) Type-2 fuzzy self-tuning pid controller design and steering angle control for mobile robot turning. *J. Adv. Navig. Technol.*, **20**, 226–231.

Rosales, A., Scaglia, G., Mut, V. & Di Sciascio, F. 2011, Formation control and trajectory tracking of mobile robotic systems—a linear algebra approach. *Robotica* **29**, 335–349. Cited By (since 1996):3 Export Date: 18 November 2013 Source: Scopus.

Roth, S. A. & Batavia, P. (2002) Evaluating path tracker performance for outdoor mobile robots. *Automation Technology for Off-Road Equipment*. St. Joseph, Michigan, EEUU: American Society of Agricultural and Biological Engineers.

Scaglia, G., Aballay, P. M., Serrano, M. E., Ortiz, O. A., Jordan, M. & Vallejo, M. D. (2014) Linear algebra based controller design applied to a bench-scale oenological alcoholic fermentation. *Control Eng. Pract.*, **25**, 66–74.

Scaglia, G., Montoya, L. Q., Mut, V. & di, F. (2009) Numerical methods based controller design for mobile robots. *Robotica*, **27**, 269–279.

Scaglia, G., Rosales, A., Quintero, L., Mut, V. & Agarwal, R. (2010) A linear-interpolation-based controller design for trajectory tracking of mobile robots. *Control Eng. Pract.*, **18**, 318–329.

Serrano, M. E., Godoy, S. A., Mut, V. A., Ortiz, O. A. & Scaglia, G. J. (2016) A nonlinear trajectory tracking controller for mobile robots with velocity limitation via parameters regulation. *Robotica*, **34**, 2546–2565.

Serrano, M. E., Godoy, S. A., Rómoli, S. & Scaglia, G. J. (2017) A numerical approximation-based controller for mobile robots with velocity limitation. *Asian J. Control*, **19**, 2165–2177.

Serrano, M. E., Scaglia, G. J. E., Godoy, S. A., Mut, V. & Ortiz, O. A. (2014) Trajectory tracking of underactuated surface vessels: a linear algebra approach. *IEEE Transactions on Control Systems Technology*, **22**, 1103–1111.

Shen, D., Sun, W. & Sun, Z. (2014) Adaptive PID formation control of nonholonomic robots without leader's velocity information. *ISA Trans.*, **53**, 474–480.

Strang, G. (2005) Linear algebra and its applications. thomsonbrooks. *Cole, Belmont, CA, USA*.

Tempo, R. & Ishii, H. (2007) Monte cCarlo and Las Vegas randomized algorithms for systems and control*: an introduction. *Eur. J. Control*, **13**, 189–203.

Toibero, J. M., Roberti, F. & Carelli, R. (2009) Stable contour-following control of wheeled mobile robots. *Robotica*, **27**, 1–12.

Wang, K. (2012) Near-optimal tracking control of a nonholonomic mobile robot with uncertainties. *Int. J. Adv. Rob. Syst.*, **9**, 66–76.

Yang, H., Fan, X., Xia, Y. & Hua, C. (2016) Robust tracking control for wheeled mobile robot based on extended state observer. *Adv. Robot.*, **30**, 68–78.

Ye, J. (2008) Adaptive control of nonlinear pid-based analog neural networks for a nonholonomic mobile robot. *Neurocomputing*, **71**, 1561–1565.

Yue, M., Tang, F., Liu, B. & Yao, B. (2012) Trajectory-tracking control of a nonholonomic mobile robot: backstepping kinematics into dynamics with uncertain disturbances. *Appl. Artif. Intell.*, **26**, 952–966.

Appendix A. Proof of Theorem 3.1

Proof. If the system behavior is ruled by (3.2) and the controller is designed by (3.3), (3.14) and (3.15), then if $n \to \infty$ thus $e_n \to 0$, when position tracking problems are considered and the controller parameters fulfill $0 < kv < 1$ and $0 < kw < 1$.

The proof of convergence to zero of the tracking error starts with the variable θ. Considering the orientation from (3.2) and the control action from (3.15),

$$\theta_{n+1} = \theta_n + T_s \left(\frac{\theta_{ez,n+1} - k_w(\theta_{ez,n} - \theta_n) - \theta_n - \hat{E}_{\theta,n}}{T_s} \right) + E_{\theta,n}. \tag{A.1}$$

By simple mathematical operations,

$$\theta_{n+1} = \theta_{ez,n+1} - k_w(\theta_{ez,n} - \theta_n) - \hat{E}_{\theta,n} + E_{\theta,n}. \tag{A.2}$$

Then, if $E_{\theta,n}$ is unknown and each component is an m-order polynomial the uncertainty $E_{\theta,n}$ and their estimated value $\hat{E}_{\theta,n}$ can be eliminated from (A.2),

$$\theta_{ez,n+1} - \theta_{n+1} = k_w(\theta_{ez,n} - \theta_n). \tag{A.3}$$

Remark A.1 Consider the next geometric progression

$$\begin{aligned} a_1 &= ka_0 \\ a_2 &= ka_1 = k^2 a_0 \\ &\vdots \\ a_{n+1} &= ka_n = k^n a_0. \end{aligned} \tag{A.4}$$

Then, if $0 < k < 1$ and $n \to \infty$ (with $n \in N$), then $a_n \to 0$.

Thus, by considering the orientation error and (A.3) the equation (A.5) is given,

$$e_{\theta,n+1} = k_w e_{\theta,n} \tag{A.5}$$

Then, if $0 < k_w < 1$ and $n \to \infty$ (with $n \in N$), then $e_{\theta,n+1} \to 0$, (see Remark A.1).

Now, the convergence analysis of e_x and e_y is developed below. From the corresponding equation of the system (3.2) and considering the control action V_n from (3.15),

$$x_{n+1} = x_n + T_s \cos(\theta_n) \left(\left(\frac{\Delta x - \hat{E}_{x,n}}{T_s} \right) \cos(\theta_{ez,n}) + \left(\frac{\Delta y - \hat{E}_{y,n}}{T_s} \right) \sin(\theta_{ez,n}) \right) + E_{x,n} \tag{A.6}$$

$$x_{n+1} = x_n + \left(\left(\Delta x - \hat{E}_{x,n} \right) \cos(\theta_{ez,n}) \cos(\theta_n) + \left(\Delta x - \hat{E}_{x,n} \right) \sin^2(\theta_{ez,n}) \right) + E_{x,n}. \tag{A.7}$$

By using the Taylor interpolation rule, the functions $\cos(\theta_n)$ can be expressed as

$$\cos\theta_n = \cos\theta_{ez,n} + \sin \underbrace{\left(\theta_{ez,n} + \lambda \left(\theta_{ez,n} - \theta_{n-1} \right) \right)}_{\theta_{\lambda,n}} e_{\theta,n} \tag{A.8}$$

with $0 < \lambda < 1$.

Considering (A.7) and (A.8) and operating (A.9) is obtained,

$$x_{n+1} = x_n + \left(\left(\Delta x - \hat{E}_{x,n} \right) \cos^2(\theta_{ez,n}) + \left(\Delta x - \hat{E}_{x,n} \right) \sin\left(\theta_{\lambda,n} \right) e_{\theta,n} + \left(\Delta x - \hat{E}_{x,n} \right) \sin^2(\theta_{ez,n}) \right) + E_{x,n} \tag{A.9}$$

$$x_{n+1} = x_n + \left(\Delta x - \hat{E}_{x,n} \right) + f_{\lambda,n} e_{\theta,n} + E_{x,n} \tag{A.10}$$

with $f_{\lambda,n} = \left(\Delta x - \hat{E}_{x,n} \right) \sin\left(\theta_{\lambda,n} \right)$.

If $E_{x,n}$ is unknown and each component is an m-order polynomial, the uncertainty $E_{x,n}$ and their estimated value $\hat{E}_{x,n}$ can be eliminated from (A.10)

$$\begin{aligned} x_{n+1} = {} & x_n + x_{ref,n+1} - k_v (x_{ref,n} - x_n) - x_n - \\ & - \left(E_{x,n-1} + E_{x,n-1} - E_{x,n-2} + \frac{E_{x,n-1} - 2E_{x,n-2} + E_{x,n-3}}{2} \right) + f_{\lambda,n} e_{\theta,n} + E_{x,n}. \end{aligned} \tag{A.11}$$

Thence,

$$x_{ref,n+1} - x_{n+1} = k_v (x_{ref,n} - x_n) - f_{\lambda,n} e_{\theta,n} \tag{A.12}$$

$$e_{x,n+1} = k_v e_{x,n} - f_{\lambda,n} e_{\theta,n}. \tag{A.13}$$

Now, applying the same reasoning to the *y-coordinate* it yields,

$$e_{y,n+1} = k_v e_{y,n} - g_{\varsigma,n} e_{\theta,n} \tag{A.14}$$

with

$$g_{\varsigma,n} = \left(\Delta y - \hat{E}_{y,n} \right) \sin\left(\theta_{\varsigma,n} \right); 0 < \varsigma < 1. \tag{A.15}$$

Taking into account (A.13) and (A.14) the system (A.16) is obtained,

$$\begin{bmatrix} e_{x,n+1} \\ e_{y,n+1} \end{bmatrix} = \begin{bmatrix} k_v & 0 \\ 0 & k_v \end{bmatrix} \begin{bmatrix} e_{x,n} \\ e_{y,n} \end{bmatrix} - \begin{bmatrix} f_{\lambda,n} e_{\theta,n} \\ g_{\varsigma,n} e_{\theta,n} \end{bmatrix} e_{\theta,n}. \tag{A.16}$$

REMARK A.2 Equation (A.16) is a linear system with a nonlinearity that tends to zero. It can be shown that the nonlinearity is bounded in the same manner as shown for other functions in Scaglia *et al.* (2010). If $0 < k_v < 1$ then $e_n \to 0$ when $n \to \infty$, see Appendix A ((A31), (A35)–(A41)) in Scaglia *et al.* (2010).