Robust trajectory tracking controllers for pose-regulation of wheeled mobile robots

Hector M. Becerra[1], J. Armando Colunga[1] and Jose Guadalupe Romero[2]

***Abstract*— Two robust kinematic controllers for position trajectory tracking of a perturbed wheeled mobile robot are presented. We address a final objective of fixed-time pose-regulation, which means that the robot position and orientation must reach desired final values simultaneously in a user-defined time. To achieve that, we propose the robust tracking of adequate trajectories for position, which drives the robot to get a desired final orientation due to the nonholonomic motion constraint. Hence, the main contribution of the paper is a complete strategy to define adequate reference trajectories as well as robust controllers to track them in order to enforce the pose-regulation of a wheeled mobile robot in a desired time. Realistic simulations show the good performance of the proposed scheme even in the presence of strong disturbances.**

I. Introduction

Control of wheeled mobile robots (WMR) is still an appealing research topic due to the wide application of these robotic systems and the theoretical challenge that represents the control problem [1]. In this paper, we are interested in driving a *perturbed* differential-drive robot, which has nonholonomic motion constraints, to a desired position and orientation (pose) simultaneously. Regarding to related work, only position control of a four degrees of freedom mobile robot has been addressed in [2]. Two separated control loops are proposed to carry out position trajectory tracking.

Many efforts on WMR control have been dedicated to a unifying theory to solve the problems of stabilization and tracking. For instance, a kinematic control law that utilizes a damped dynamic oscillator achieving position/orientation tracking is presented in [3]. The same controller is used for the regulation problem. Recently, a smooth time-varying controller has been proposed in [4] to simultaneously address the problems of stabilization and tracking. In that work, a single controller is able of converting between stabilizer and tracker rather than switching between two controllers.

Motion planning is an essential element in navigation systems and might be an important complement of a control scheme for mobile robots. In [5], a method for steering systems with nonholonomic constraints between arbitrary configurations is proposed. A steering control of mobile robots around pre-planned paths is presented in [6], it proposes simple linear control laws to constrain the demand of the steering controller taken into account the curvature of the pre-planned paths. A combined strategy of motion planning and control has been proposed in [7], this is a complete architecture for autonomous navigation of mobile robots. A module of the strategy generates the trajectories dealing with obstacles and a saturated integral sliding mode controller solves the tracking problem.

Robust controllers for WMR is an important feature to achieve good accuracy in a robotic task. Some efforts have been dedicated to the robust control of WMR at a dynamic level. In [8], an inverse dynamic controller receives the signals of a kinematic controller based on a super-twisting sliding mode control to improve robustness on a trajectory tracking task. A dynamic control scheme for stabilization of robots subject to nonholonomic constraints and external disturbances is proposed in [9]. The method uses the port-Hamiltonian theory to derive smooth time-invariant control laws. At kinematic level, the so-called intelligent PID controller is introduced to control a nonholonomic mobile robot with measurement disturbance in [10]. Recently, a kinematic control scheme oriented to the problem of avoiding obstacles while navigating is proposed in [11]. The controller design considers additive disturbances on the inputs and the solution is based on the supervisory control framework.

In this paper, we present two alternatives of robust kinematic controllers for trajectory tracking of a perturbed differential-drive robot. The control scheme allows us to specify desired trajectories for the position coordinates, whereas the orientation remains as a degree of freedom in the control system. The final goal is the regulation of the robot pose in a desired time, which means that the robot position and orientation must reach desired final values in a user-defined time. We exploit the tracking of adequate trajectories to drive the robot to reach the desired position and also orientation simultaneously, such that pose-regulation is finally achieved. Thus, the main contribution of the paper is a complete strategy to define adequate reference trajectories and robust controllers to track them, which ensures the pose-regulation of a WMR in a desired time. The stability of the controllers, a proportional-integral control and a super-twisting control, is theoretically proved. The complete control strategy is evaluated through realistic simulations introducing additive disturbances.

The paper is structured as follows. Sec. II presents the mathematical modeling and problem statement. Sec. III describes the synthesis of both controllers and presents their stability properties. The definition of desired trajectories is detailed in Sec. IV. Simulation results are shown in Sec. V and conclusions and future work are remarked in Sec. VI.

The first two authors were supported in part by Conacyt [grant 220796].

[1] H. M. Becerra and J. A. Colunga are with Centro de Investigación en Matemáticas (CIMAT), Gto., Mexico. {hector.becerra,jose.colunga}@cimat.mx

[2] J. G. Romero is with Instituto Tecnológico Autónomo de México (ITAM), Mexico City, Mexico. jose.romerovelazquez@itam.mx

II. Modeling and problem formulation

In this paper, we are interested in the problem of *robust pose-regulation* of a differential-drive robot in a desired time, i.e., take the robot *position and orientation* to desired values with respect to a fixed reference frame in a desired time. We consider a *perturbed* kinematic model of the system described by the motion equation

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta+\beta) & 0 \\ \sin(\theta+\beta) & 0 \\ 0 & 1 \end{bmatrix} (\mathbf{u}+\mathbf{d}), \tag{1}$$

where $x, y \in \mathbb{R}$ describe the Cartesian position and $\theta \in \mathbb{R}$ its orientation with respect to a fixed reference frame W (see Fig. 1), $\beta = 0$ for forward motion and $\beta = \pi$ for backward motion, $\mathbf{u} = \text{col}(v, \omega) \in \mathbb{R}^2$ are the translational and rotational input velocities respectively and $\mathbf{d} = \text{col}(d_1, d_2) \in \mathbb{R}^2$ is a *constant* matched disturbance.

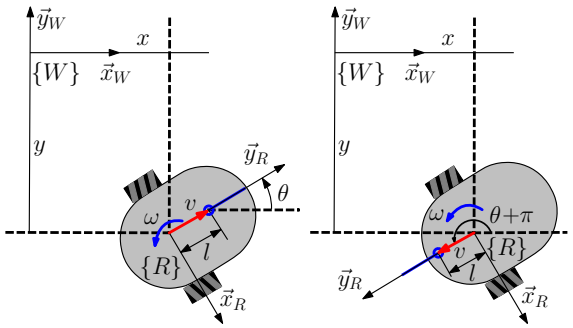

Fig. 1. Kinematic configuration of the robot with a virtual control point at a distance l. Left: Forward motion model. Right: Backward motion model.

A. A transformed kinematic model

In this section, we modify the system (1) via the addition of a virtual point (l) along the longitudinal axis of the robot– see Fig. 1. Hence, the well defined change of coordinates

$$\begin{aligned} x_l &:= x + l\cos(\theta+\beta), \\ y_l &:= y + l\sin(\theta+\beta), \\ \theta &:= \theta, \end{aligned} \tag{2}$$

with $l > 0$ and constant, transforms (1) into

$$\begin{bmatrix} \dot{x}_l \\ \dot{y}_l \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta+\beta) & -l\sin(\theta+\beta) \\ \sin(\theta+\beta) & l\cos(\theta+\beta) \\ 0 & 1 \end{bmatrix} (\mathbf{u}+\mathbf{d}). \tag{3}$$

For controller design and to simplify the notation, we partition and recalled the coordinates as

$$\mathbf{z} = \text{col}(x_l, y_l) \in \mathbb{R}^2,$$

such that the modified model (3) can be written as

$$\begin{aligned} \dot{\mathbf{z}} &:= \mathbf{A}(\theta+\beta)(\mathbf{u}+\mathbf{d}), \\ \dot{\theta} &:= \omega + d_2, \end{aligned} \tag{4}$$

with *full* rank matrix $\mathbf{A}(\theta+\beta)$: $\mathbb{R} \to \mathbb{R}^{2\times 2}$ given by

$$\mathbf{A}(\theta+\beta) := \mathbf{A} = \begin{bmatrix} \cos(\theta+\beta) & -l\sin(\theta+\beta) \\ \sin(\theta+\beta) & l\cos(\theta+\beta) \end{bmatrix}.$$

B. Problem formulation

Consider that we have the perturbed kinematic model of the WMR (1), transformed to system (4) by the change of coordinates (2). The robust pose-regulation problem in a desired time can be enunciated as follows.

Definition 2.1 (Problem statement): Regulate $\mathbf{z}$ and θ of the system (4) to desired constant values $\mathbf{z}^f = \text{col}(z_1^f, z_2^f) \in \mathbb{R}^2$ and θ^f in a user-defined time τ in spite of the disturbance $\mathbf{d}$. Without loss of generality, we will assume $\mathbf{z}^f = \mathbf{0}$.

In order to deal with the time-constraint imposed by τ, we propose to address the regulation problem as a *trajectory tracking problem* for adequate references of the position coordinates x_l and y_l. We will show that by tracking some particular trajectories, the orientation θ can also reach a desired value. By assuming that $\mathbf{z}^f = \mathbf{0}$, we consider that the desired final robot's position will be the origin of $\mathbb{R}^2$.

III. Robust trajectory tracking control

In this section, we present the main result of the paper, where two robust controllers are described to achieve trajectory tracking on coordinates related to the Cartesian position of the robot.

Definition 3.1 (Control objective): Let us define a vector of errors as

$$\tilde{\mathbf{z}}(t) := \mathbf{z} - \mathbf{z}^d(t) := \begin{bmatrix} x_l(t) - x_l^d(t) \\ y_l(t) - y_l^d(t) \end{bmatrix}, \tag{5}$$

with $x_l^d, y_l^d \in \mathbb{R}$ desired trajectories. The objective is to ensure $\lim_{t\to\infty} \tilde{\mathbf{z}} = \mathbf{0}$ to achieve trajectory tracking.

In order to reach the origin of $\mathbb{R}^2$ in the time τ as final robot's position, we assume in the sequel that $\mathbf{z}^d(t) = \mathbf{z}^f = \mathbf{0}$ for $t \geq \tau$. In the following, we denote $|\boldsymbol{x}|^2 := \boldsymbol{x}^\top \boldsymbol{x}$ for $\boldsymbol{x} \in \mathbb{R}^n$. Given a function $f : \mathbb{R}^n \to \mathbb{R}$, we define the differential operator $\nabla_{x_i} f := \left(\frac{\partial f}{\partial \boldsymbol{x}_i}\right)^\top$, where $\boldsymbol{x}_i \in \mathbb{R}^p$ with $p \leq n$, $\boldsymbol{x}_i$ is a subset of components of the vector $\boldsymbol{x}$.

A first controller to achieve the previous control objective makes use of the well known addition of integral action on *actuated* coordinates [12] in order to reject the disturbance $\mathbf{d}$ and it is presented in the following proposition.

Proposition 3.2: Consider the time derivative of (5) and (4) in closed-loop with the controller

$$\begin{aligned} \mathbf{u} &= \mathbf{A}^{-1}\left(-k_p\tilde{\mathbf{z}} + \dot{\mathbf{z}}^d\right) - \boldsymbol{\rho}, \\ \dot{\boldsymbol{\rho}} &= k_i\mathbf{A}^\top\tilde{\mathbf{z}}, \end{aligned} \tag{6}$$

with positive free gains k_p, k_i and $\boldsymbol{\rho} \in \mathbb{R}^2$ an integral action.

I. The closed-loop takes the form

$$\begin{bmatrix} \dot{\tilde{\mathbf{z}}} \\ \dot{\boldsymbol{\rho}} \end{bmatrix} = \begin{bmatrix} -k_p\mathbf{I}_2 & -k_i\mathbf{A} \\ k_i\mathbf{A}^\top & \mathbf{0}_2 \end{bmatrix} \begin{bmatrix} \nabla_{\tilde{z}} H_z \\ \nabla_{\rho} H_z \end{bmatrix}, \tag{7}$$

with total energy function $H_z : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$

$$H_z(\tilde{\mathbf{z}}, \tilde{\boldsymbol{\rho}}) = \frac{1}{2}|\tilde{\mathbf{z}}|^2 + \frac{1}{2}k_i^{-1}|\tilde{\boldsymbol{\rho}}|^2, \tag{8}$$

where $\tilde{\boldsymbol{\rho}} = \boldsymbol{\rho} - \mathbf{d}$.[1]

[1]From the fact that $\mathbf{d}$ is a constant vector we get $\dot{\tilde{\boldsymbol{\rho}}} = \dot{\boldsymbol{\rho}}$.

II. The closed loop system has a globally asymptotically stable equilibrium point at the desired point $(\tilde{\mathbf{z}}, \boldsymbol{\rho}) = (\mathbf{0}, \mathbf{d})$.

III. The closed-loop system has a *zero dynamics* of first order given by

$$\begin{aligned} \dot{\theta} &= \omega + d_2 = 0, \\ \dot{\theta} &= -\tfrac{1}{l}\left(v_1 \sin\theta - v_2 \cos\theta\right) - \rho_2 + d_2 = 0, \end{aligned} \quad (9)$$

where $v_1 = -k_p \tilde{z}_1 + \dot{z}_1^d$ and $v_2 = -k_p \tilde{z}_2 + \dot{z}_2^d$.

Proof: Regarding to point I, replacing the control law (6) in (4) we get the first row of the closed-loop (7). The second row of (7) corresponds to the definition of the integral part given by (6). On the other hand, taking as Lyapunov function (8) and its time derivative along the system (7) yields

$$\dot{H}_z(\tilde{\mathbf{z}}, \tilde{\boldsymbol{\rho}}) = -k_p |\tilde{\mathbf{z}}|^2. \quad (10)$$

This proves that $(\mathbf{0}, \mathbf{d})$ is a stable equilibrium of the closed loop system (7). Furthermore, the trajectories will converge to the largest invariant set included in

$$\mathcal{S} = \left\{ (\tilde{\mathbf{z}}, \tilde{\boldsymbol{\rho}}) \mid \dot{H}_z = 0 \right\} = \left\{ (\tilde{\mathbf{z}}, \tilde{\boldsymbol{\rho}}) \mid \tilde{\mathbf{z}} = \mathbf{0} \right\}. \quad (11)$$

From (7) and (11) we have that

$$\begin{aligned} \mathbf{0} &= -\mathbf{A}\tilde{\boldsymbol{\rho}}, \\ \dot{\boldsymbol{\rho}} &= \mathbf{0}. \end{aligned} \quad (12)$$

Hence, from the fact that $\mathbf{A}$ is a full rank matrix, we can conclude that the largest invariant set $\mathcal{S}$ is

$$\mathcal{M} = \left\{ (\tilde{\mathbf{z}}, \tilde{\boldsymbol{\rho}}) \mid \tilde{\mathbf{z}} = \mathbf{0}, \boldsymbol{\rho} = \mathbf{d} \right\}.$$

This proves that the equilibrium point $(\mathbf{0}, \mathbf{d})$ of the target dynamics (7) is asymptotically stable. Finally, related to point III, the closed-loop system reaches its *zero dynamics* [13] when $\mathbf{z} = \mathbf{z}^d(t) = \mathbf{z}^f = \mathbf{0}$ for $t \geq \tau$. Thus, the outputs z_1 and z_2 can be regulated to zero, but in the system (4), the dynamics of orientation is remaining. Equation (9) is obtained from (6) noting that ω is the second component of the input's vector $\mathbf{u}$. Since $v_1 = 0$, $v_2 = 0$ and $\boldsymbol{\rho} = \mathbf{d}$ when $z_1 \equiv 0$ and $z_2 \equiv 0$, the zero dynamics (9) represents a stable dynamics with bounded constant solution $\bar{\theta}$.

Hence, in the case of the system (4) with outputs $z_1 = x_l$, $z_2 = y_l$, the subset of the state space describing the zero dynamics is given by

$$\begin{aligned} Z^* &= \left\{ [x_l, y_l, \theta]^\top \mid z_1 \equiv 0,\ z_2 \equiv 0 \right\} \\ &= \left\{ \left[0, 0, \bar{\theta}\right]^\top,\ \bar{\theta} \in \mathbb{R} \right\}. \end{aligned}$$

Zero dynamics in this control system means that when the variables related to the robot's position are corrected, the orientation might be different to zero. ■

Remark 3.3: The addition of integral action on actuated coordinates presented in Proposition 3.2 can be equivalent to add an integrator around the passive output presented in [12] for Port Hamiltonian systems, where the task preserves stability in the presence of disturbances or modeling errors.

The following proposition presents a second option of robust controller to achieve the control objective of Definition 3.1. It is based on continuous second order sliding mode control.

Proposition 3.4: Consider the time derivative of (5) and (4) in closed-loop with a super-twisting control

$$\begin{aligned} \mathbf{u} &= \mathbf{A}^{-1}\left(-k_1 \mathbf{S}\text{sign}(\tilde{\mathbf{z}}) + \dot{\mathbf{z}}^d + \boldsymbol{\sigma}\right), \\ \dot{\boldsymbol{\sigma}} &= -k_2 \text{sign}(\tilde{\mathbf{z}}), \end{aligned} \quad (13)$$

with $\mathbf{S} = \begin{bmatrix} |\tilde{z}_1|^{1/2} & 0 \\ 0 & |\tilde{z}_2|^{1/2} \end{bmatrix}$ and positive free gains k_1, k_2.

I. The closed-loop takes the form

$$\begin{aligned} \dot{\tilde{\mathbf{z}}} &= -k_1 \mathbf{S}\text{sign}(\tilde{\mathbf{z}}) + \boldsymbol{\sigma} + \boldsymbol{\varrho}, \\ \dot{\boldsymbol{\sigma}} &= -k_2 \text{sign}(\tilde{\mathbf{z}}), \end{aligned} \quad (14)$$

with $\boldsymbol{\varrho} = \mathbf{A}\mathbf{d}$ a bounded disturbance.

II. The closed-loop system has a globally finite-time stable equilibrium point at the desired point $(\tilde{\mathbf{z}}, \boldsymbol{\sigma}) = (\mathbf{0}, -\boldsymbol{\varrho})$.

III. The closed-loop system has a *zero dynamics* of first order given by

$$\begin{aligned} \dot{\theta} &= \omega + d_2 = 0, \\ \dot{\theta} &= -\tfrac{1}{l}\left(v_1 \sin\theta - v_2 \cos\theta\right) + \sigma_2 + \varrho_2 = 0, \end{aligned} \quad (15)$$

where $v_1 = -k_1 |\tilde{z}_1|^{1/2} \text{sign}(\tilde{z}_1) + \dot{z}_1^d$ and $v_2 = -k_1 |\tilde{z}_2|^{1/2} \text{sign}(\tilde{z}_2) + \dot{z}_2^d$.

Proof: The point I can be directly verified by replacing the control law (13) in (4) to obtain the closed-loop system in (14). Notice that (14) expresses decoupled dynamics for $\tilde{z}_1$ and $\tilde{z}_2$. Thus, the decoupled closed-loop dynamics can be expressed as

$$\begin{aligned} \dot{\tilde{z}}_i &= -k_1 |\tilde{z}_i|^{1/2} \text{sign}(\tilde{z}_i) + \sigma_i + \varrho_i, \\ \dot{\sigma}_i &= -k_2 \text{sign}(\tilde{z}_i), \end{aligned} \quad (16)$$

for $i = 1, 2$. Let us define $\xi_i = \sigma_i + \varrho_i$. Then, (16) can be written as

$$\begin{aligned} \dot{\tilde{z}}_i &= -k_1 |\tilde{z}_i|^{1/2} \text{sign}(\tilde{z}_i) + \xi_i, \\ \dot{\xi}_i &= -k_2 \text{sign}(\tilde{z}_i) + \dot{\varrho}_i, \end{aligned} \quad (17)$$

for $i = 1, 2$. Thus, related to point II, it has been proved in [14] that, for bounded continuously differentiable disturbances, i.e., if $|\varrho_i| < L$ and $|\dot{\varrho}_i| < M$ for some constants $L > 0, M > 0$, the second order systems (17) converge globally to the origin ($z_i = 0$, $\xi_i = 0$) in finite-time if adequate positive control gains k_1 and k_2 are used. Since $\xi_i = 0$, then the unknown terms of the disturbance can be constructed $\varrho_i = -\sigma_i$, so that $\boldsymbol{\varrho} = -\boldsymbol{\sigma}$. Hence, all the trajectories of (14) will converge to the invariant set

$$\mathcal{M} = \left\{ (\tilde{\mathbf{z}}, \boldsymbol{\sigma}) \mid \tilde{\mathbf{z}} = \mathbf{0}, \boldsymbol{\sigma} = -\boldsymbol{\varrho} \right\}.$$

Finally, related to the point III, the derivation of the zero dynamics can be addressed similarly as in Proposition 3.2. Since $v_1 = 0$, $v_2 = 0$ and $\boldsymbol{\sigma} = -\boldsymbol{\varrho}$ when $z_1 \equiv 0$ and $z_2 \equiv 0$, the zero dynamics (15) represents a stable dynamics with bounded constant solution $\bar{\theta}$. ■

Remark 3.5: Both controllers given by (6) and (13) can be implemented for forward and backward motion to track the desired trajectories. Forward motion control is used if $\beta = 0$ and $v = +u_1$, and backward motion is used if $\beta = \pi$ and $v = -u_1$, with u_1 being the first component of $\mathbf{u}$.

IV. Desired trajectories definition

The problem stated in Definition 2.1 requires the robot to reach a desired position and orientation. Up to now, the presented controllers drive to zero the position coordinates $z_1 = x_l$ and $z_2 = y_l$, but the orientation evolves freely as described by the zero dynamics. We propose to define an adequate parabolic trajectory for the robot position in order to achieve also orientation correction for nonholonomic mobile robots. Parabolic trajectories are the simplest ones that can connect two points in $\mathbb{R}^2$ imposing a desired slope at the final point. Let us define an initial robot's pose given by (z_1^i, z_2^i, θ^i) and a final desired pose defined by (z_1^f, z_2^f, θ^f). A parabolic trajectory on the $z_1 - z_2$ plane symmetric to an axis parallel to $\vec{z}_1$ has the form

$$4p(z_1 - h) = (z_2 - k)^2, \tag{18}$$

where $p \in \mathbb{R}$ represents the constant distance between the vertex and the focus and h, $k \in \mathbb{R}$ are the vertex coordinates of the parabola (see Fig. 2).

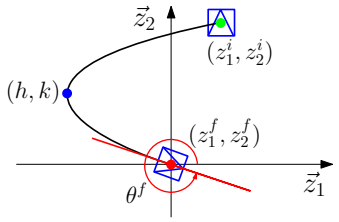

Fig. 2. Parabolic desired trajectory symmetric to an axis parallel to $\vec{z}_1$, whose tangent at the origin of the plane is equal to $\tan(\theta^f)$.

Without loss of generality, consider the final desired position $z_1^d(\tau) = z_1^f = 0$ and $z_2^d(\tau) = z_2^f = 0$. Taking the derivative of (18) evaluated at the desired position and equating to the slope given by $\tan\theta^f$ at the desired position, we have

$$\tfrac{dz_2}{dz_1}\,|_{(0,0)} = \tfrac{-2p}{k} = \tan\theta^f. \tag{19}$$

Moreover, evaluating (18) at the desired position (origin of the plane), we get

$$k^2 = -4ph. \tag{20}$$

Thus, from (19) and (20), the parameters k and h are

$$k = \tfrac{-2p}{\tan\theta^f}, \quad h = \tfrac{-p}{\tan^2\theta^f}. \tag{21}$$

Since we know the initial position (z_1^i, z_2^i) and solving for p in (18), we get $p = \frac{(z_2^i)^2}{4\left(z_1^i - \frac{z_2^i}{\tan\theta^f}\right)}$.

Then, using these results, the proposed trajectory is generalized for an initial pose (z_1^i, z_2^i, θ^i) and a final pose (z_1^f, z_2^f, θ^f) and parametrized in time as follows:

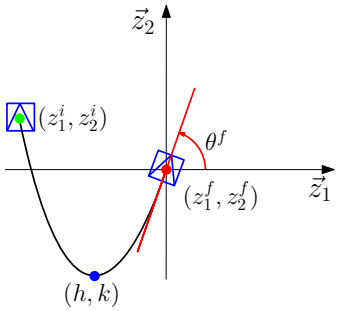

Fig. 3. Parabolic desired trajectory symmetric to an axis parallel to $\vec{z}_2$, whose tangent at the origin of the plane is equal to $\tan(\theta^f)$.

$$\begin{aligned} z_2^d(t) &= \tfrac{z_2^i - z_2^f}{2}\left(1 + \cos\left(\tfrac{\pi t}{\tau}\right)\right) + z_2^f, \;\; 0 \le t \le \tau \\ z_1^d(t) &= \tfrac{\left(z_2^d(t) - z_2^f - k\right)^2}{4p} + z_1^f + h, \quad 0 \le t \le \tau \\ p &= \tfrac{\left(z_2^i - z_2^f\right)^2}{4\left(z_1^i - z_1^f - \frac{z_2^i - z_2^f}{\tan\theta^f}\right)}. \end{aligned} \tag{22}$$

Notice that the computation of the parameters p, k and h for the trajectory (22) has singularities for some values of θ^f. Hence, we propose another complementary option of trajectory that can be used conveniently avoiding these singularities and generating shortest paths depending on the initial and desired final poses. Following a similar procedure to the previous, we can define a parabolic trajectory on the $z_1 - z_2$ plane symmetric to an axis parallel to $\vec{z}_2$, which is given by (see Fig. 3)

$$4p(z_2 - k) = (z_1 - h)^2. \tag{23}$$

By parametrization of the equation (23) in time, we define the following complementary option of desired trajectory:

$$\begin{aligned} z_1^d(t) &= \tfrac{z_1^i - z_1^f}{2}\left(1 + \cos\left(\tfrac{\pi t}{\tau}\right)\right) + z_1^f, \;\; 0 \le t \le \tau \\ z_2^d(t) &= \tfrac{\left(z_1^d(t) - z_1^f - h\right)^2}{4p} + z_2^f + k, \quad 0 \le t \le \tau \\ p &= \tfrac{(z_1^i - z_1^f)^2}{4(z_2^i - z_2^f - (z_1^i - z_1^f)\tan\theta^f)} \\ k &= -p\tan^2\theta^f, \quad h = -2p\tan\theta^f. \end{aligned} \tag{24}$$

Since we have two options of desired trajectories and two ways of motion control to track the trajectory (forward and backward), we propose a strategy to decide for one of the combinations in order to avoid singularities in the trajectory computation and to take into account the length of the generated path. Algorithm 1 presents such strategy for the case when the initial robot position is in the first quadrant of the $z_1 - z_2$ plane. A similar strategy is defined for any initial position in every quadrant of the plane. Notice that exceptions that cannot be directly handled by the proposed strategy are initial positions over any of the axes, i.e., for $z_1^i = 0$ or $z_2^i = 0$. In these cases the robot should be moved away from the initial condition to start inside a quadrant of the plane. Then, the proposed strategy exemplified in Algorithm 1 can be applied. The need of an initial motion for positions over the axes is because in that cases the

Algorithm 1 Trajectory generation for the first quadrant: $z_1^i > 0, z_2^i > 0$.

Input: Initial position (z_1^i, z_2^i), desired pose (z_1^f, z_2^f, θ^f) and desired time τ.
Output: Trajectory to be tracked (22) or (24), and type of motion (see Remark 3.5).

1. Compute $\phi^i = \arctan\left(\frac{z_2^i - z_2^f}{z_1^i - z_1^f}\right)$.
if $0 \leqslant \theta^f \leqslant \frac{\pi}{2}$ **then**
  if $\theta^f < \phi^i$ or $\theta^f = 0$ **then**
    2. Trajectory (24) backward;
  else
    3. Trajectory (22) backward;
  end if
end if
if $\frac{\pi}{2} < \theta^f \leqslant \pi$ **then**
  if $\frac{\pi}{2} < \theta^f < \frac{3}{4}\pi$ **then**
    4. Trajectory (22) backward;
  else
    5. Trajectory (24) forward;
  end if
end if
if $\pi < \theta^f \leqslant \frac{3}{2}\pi$ **then**
  if $\theta^f - \pi < \phi^i$ **then**
    6. Trajectory (24) forward;
  else
    7. Trajectory (22) forward;
  end if
end if
if $\frac{3}{2}\pi < \theta^f < 2\pi$ **then**
  if $\frac{3}{2}\pi < \theta^f < \frac{7}{4}\pi$ **then**
    8. Trajectory (22) forward;
  else
    9. Trajectory (24) backward;
  end if
end if

parabolas become a line along an axis and then an arbitrary desired orientation cannot be reached by tracking that linear trajectory.

The tracking of the proposed parabolic trajectories allow us to define a fixed temporal horizon τ to reach the desired robot pose. Although parabolic trajectories are the simplest ones accomplishing the task constraints, higher order trajectories might benefit in terms of smoothness and path length. Notice that the robot always begins over the desired trajectory, so that the controller has to maintain the tracking of it. Due to the nonholonomic motion of the robot, the tracking controller drives the robot to perform an initial autonomous rotation to get aligned with the path independently of the initial robot orientation. Next, it is proved that orientation correction is also achieved by tracking the proposed trajectories, in such a way that pose-regulation is accomplished.

Proposition 4.1: The robust controllers (6) and (13), tracking the reference trajectory specified by Algorithm 1 (similarly for the other quadrants), drives the virtual control point of the perturbed differential-drive robot (4) to reach the desired pose $(z_1 = z_1^f, z_2 = z_2^f, \theta + \beta = \theta^f)$, i.e., orientation is also corrected.

Proof: It is clear the global robust stability of the error system (7), respectively (14), with the controller (6), respectively (13), which drives the robot position to the desired values $(z_1 = z_1^f,\ z_2 = z_2^f)$ in τ seconds. Thus, it remains to prove that the orientation is also corrected in τ seconds. We assume that the rotational velocity is small when the robot is approaching to the desired position, which is realistic given the construction of the desired trajectory. Thus, the translational velocities of the virtual control point for each axis are given by

$$\dot{z}_1 = v\cos(\theta + \beta), \qquad \dot{z}_2 = v\sin(\theta + \beta).$$

From these relationships we get

$$\tan(\theta + \beta) = \left(\frac{\dot{z}_2}{\dot{z}_1}\right).$$

Let us use the parabolic relationship between Cartesian coordinates (23), according to an option of desired trajectory. A similar analysis follows for the complementary trajectory (18). The time-derivative of (23) provides the relationship

$$\frac{\dot{z}_2}{\dot{z}_1} = \frac{z_1 - h}{2p}.$$

Thus, the orientation angle can be computed as follows when the z_1 and z_2-coordinates track the desired trajectory:

$$\theta + \beta = \arctan\left(\frac{z_1 - h}{2p}\right).$$

As aforementioned, when the robot has tracked the reference trajectory and $t = \tau$ the position reaches the goal $(z_1 = z_1^f,\ z_2 = z_2^f)$. Since the slope of the trajectory (24) for $z_1 = z_1^f$ and $z_2 = z_2^f$ is given by $\tan(\theta^f) = (-h/2p)$ according to the equation for h in (24), we have

$$\theta + \beta = \theta^f = \arctan\left(-\frac{h}{2p}\right).$$

This proves that the desired pose $(z_1 = z_1^f,\ z_2 = z_2^f,\ \theta + \beta = \theta^f)$ is reached in τ seconds. ■

V. Simulation Results

In this section, we present simulation results using indistinctly each controller, (6) the proportional-integral control (PIC) or (13) the super-twisting control (STC). Simulations are implemented using Webots [15] with the robot Pioneer 3-DX. A GPS and a compass provide the required feedback information. The time-derivatives required by the control laws are implemented using the forward Euler approximation. For the disturbance, we have introduced $d_1 = 0.1$ for the translational velocity v and $d_2 = -0.1$ for the rotational velocity ω. We have assumed that the virtual control point is at $l = 0.1$ m. The control gains were fixed to $k_p = 1.5$, $k_i = 1.5$ for the PIC and $k_1 = 0.6$, $k_2 = 0.03$ for the STC. For all the simulations, we have defined that the robot has to reach the desired pose in $\tau = 30$ s. The simulated control cycle runs at 5 ms.

The first part of the results corresponds to the same initial robot pose reaching the origin of the $z_1 - z_2$ plane for 4 experimental runs and reaching a different final desired orientation

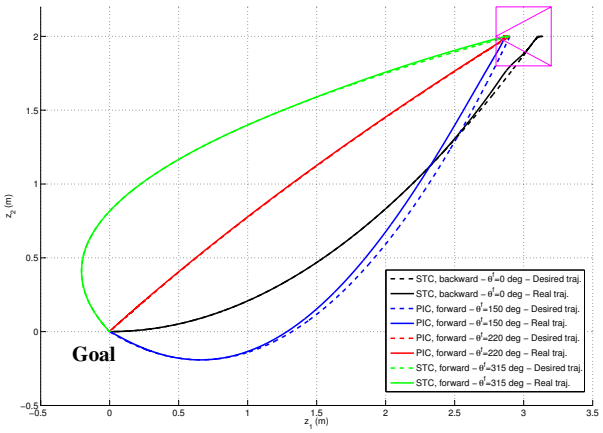

Fig. 4. Executed and desired trajectories starting from the same initial robot pose $\mathbf{P} = (x = 3 \text{ m}, y = 2 \text{ m}, \theta = 180 \text{ deg})$ and reaching the origin of the plane with 4 different desired orientations.

in each case. Fig. 4 shows the executed trajectories and the corresponding references for each experimental run, where the robot is commanded to reach $\theta^f = 0$ deg, $\theta^f = 150$ deg, $\theta^f = 220$ deg and $\theta^f = 315$ deg. Two of the experimental runs are executed using the PIC and the other two using the STC. According to Algorithm 1, the trajectory to reach $\theta^f = 0$ deg is the only one tracked in backward motion. The other trajectories are tracked in forward motion. This is why the trajectory in black in Fig. 4, corresponding to $\theta^f = 0$, starts in a different position than the other ones, since the virtual control point in this case is behind the robot reference frame as defined in Fig. 1(right). For each of the cases evaluated in Fig. 4, the final position error, given by the Euclidean distance between the final position and the origin of the plane, is less than 1 cm for each case. The mean squared tracking error is less than 10 cm^2 for each case. The final orientation errors for the goals $\theta^f = 0$ deg, $\theta^f = 150$ deg, $\theta^f = 220$ deg and $\theta^f = 315$ deg are 2 deg, 5 deg, 0.4 deg and 6.7 deg, respectively.

Two examples, for $\theta^f = 0$ deg with STC and $\theta^f = 220$ deg with PIC, of the evolution of the robot pose with respect to time are shown in Fig. 5. We can see the good tracking of the desired trajectories for position coordinates z_1 and z_2, such that the real evolution of these variables are over imposed to the references. Also, the free variable for the control system θ is taken really close to the corresponding desired value in the fixed time $\tau = 30$ s using either controllers. Fig. 6 presents the evolution of the robot velocities with respect to time for the same two cases $\theta^f = 0$ deg and $\theta^f = 220$ deg. The velocities settle at the contrary sign values of the introduced constant disturbances, such that they are effectively rejected. Notice that although the STC is continuous in contrast to classical sliding mode control, it generates a small high frequency component on the velocities that can be reduced if the control loop is even faster.

The second part of the results corresponds to 4 different initial robot poses to reach the origin of the $z_1 - z_2$ plane with desired orientation $\theta^f = 90$ deg for all the cases. Fig. 7

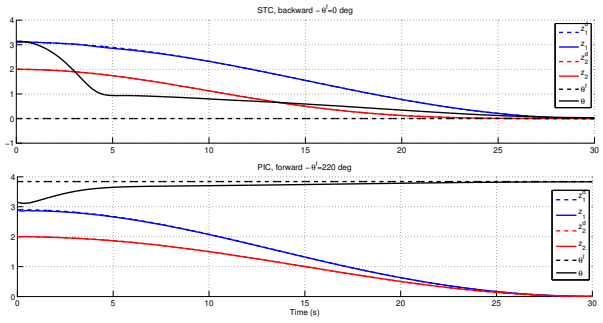

Fig. 5. Evolution of the robot pose with respect to time for the cases of desired orientation $\theta^f = 0$ deg and $\theta^f = 220$ deg.

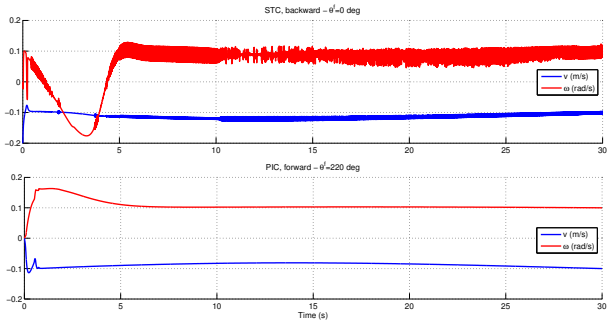

Fig. 6. Evolution of the robot velocities with respect to time for the cases of desired orientation $\theta^f = 0$ deg and $\theta^f = 220$ deg.

presents the executed trajectories and the corresponding references for each experimental run. Two of the experimental runs are executed using the PIC and the other two using the STC. A similar strategy to the one in Algorithm 1 is used to define the desired trajectories and the way to track them (forward or backward). In these cases, the strategy specifies that the trajectories starting in $\mathbf{P}_1$ and $\mathbf{P}_2$ are executed in backward motion and those starting in $\mathbf{P}_3$ and $\mathbf{P}_4$ are executed in forward motion. Similar performance to the first part of the results was obtained in this second part. For each of the cases in Fig. 7, the final position error is less than 1 cm and the mean squared tracking error is less than 10 cm^2. The final orientation errors are 2.6 deg for $\mathbf{P}_1$, 1 deg for $\mathbf{P}_2$, 7.4 deg for $\mathbf{P}_3$ and 8.7 deg for $\mathbf{P}_4$.

Fig. 8 shows the evolution of the robot pose with respect to time for the cases starting in $\mathbf{P}_2$ and $\mathbf{P}_3$. One more time, the real evolution of the position coordinates z_1 and z_2 are practically over imposed to the references, which demonstrate the good performance of both tracking controllers. Also, the robot orientation closely reaches the desired value in the specified time $\tau = 30$ s. We present the corresponding robot velocities for the same two cases starting in $\mathbf{P}_2$ and $\mathbf{P}_3$. Also, the settling values of the velocities are those that effectively cancel the introduced disturbances. In general, we can say that both controllers, PIC and STC, present similar performance for the evaluated conditions.

Finally, we refer to a video included as a complement, where we present the performance of both robust controllers in the presence of more severe disturbances and an obstacle

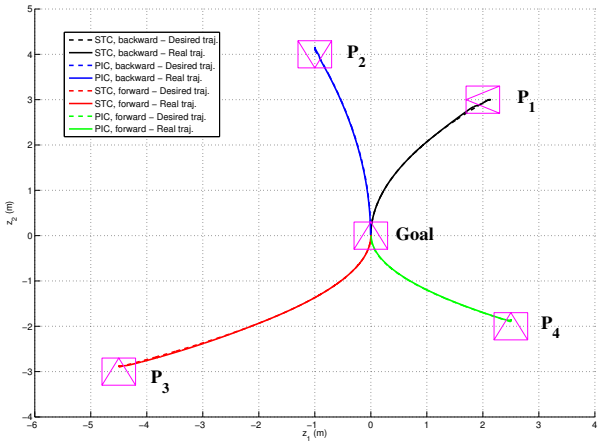

Fig. 7. Executed and desired trajectories starting from 4 different initial robot poses $\mathbf{P}_1 = (x = 2$ m, $y = 3$ m, $\theta = 180$ deg), $\mathbf{P}_2 = (x = -1$ m, $y = 4$ m, $\theta = 270$ deg), $\mathbf{P}_3 = (x = -4.5$ m, $y = -3$ m, $\theta = 90$ deg), $\mathbf{P}_4 = (x = 2.5$ m, $y = -2$ m, $\theta = 90$ deg) and reaching the origin of the plane with desired orientation 90 deg.

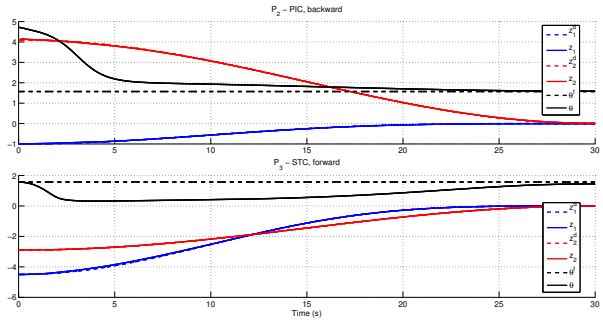

Fig. 8. Evolution of robot pose with respect to time for the cases of initial pose $\mathbf{P}_2$ and $\mathbf{P}_3$.

that the robot crashes, making it deviating from the trajectory for a while. The robot recovers the trajectory to finally reach the desired position and orientation in a desired time.

VI. Conclusions and Future Work

In this paper, we addressed the pose-regulation problem of a perturbed wheeled mobile robot, i.e., drive the robot to reach simultaneously a desired position and orientation regardless of additive disturbances in the control inputs. The pose-regulation problem is faced by tracking adequate trajectories for the position coordinates, such that the robot's orientation gets a final desired value due to the nonholonomic motion constraint. Two alternatives of robust kinematic controllers are presented to ensure an accurate trajectory tracking. Moreover, a complete strategy details how to define the adequate trajectories on the basis of parabolic curves, which are the simplest curves accomplishing the task constraints. The strategy also specifies a convenient motion to track the trajectories in forward or backward motion. Compared to classical stabilization/regulation schemes for wheeled mobile robots, our proposal allow us to define the time of convergence. Realistic simulations have shown good performance of the proposed control scheme.

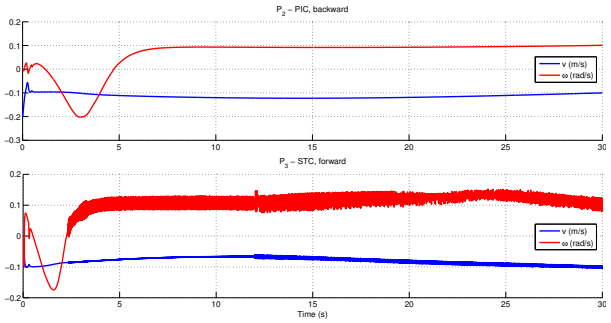

Fig. 9. Evolution of robot velocities with respect to time for the cases of initial pose $\mathbf{P}_2$ and $\mathbf{P}_3$.

As future work, we plan: 1) to implement the complete control strategy in a real robot, 2) to analyze theoretically the behavior of the presented controllers against time-varying disturbances and 3) to contrast the use of other type of trajectories, for instance B-splines, in terms of length of the path and pose-regulation accuracy.

References

[1] I. Kolmanovsky and N. H. McClamroch. Developments in nonholonomic control problems. *IEEE Control Systems*, 15(6):20–36, 1995.

[2] Y. Chung and F. Harashima. A position control differential drive wheeled mobile robot. *IEEE Trans. on Industrial Electronics*, 48:853–863, 2001.

[3] W. E. Dixon, D. M. Dawson, E. Zergeroglu, and F. Zhang. Robust tracking and regulation control for mobile robots. *Int. J. of Robust and Nonlinear Control*, 10(4):199–216, 2000.

[4] Y. Wang, Z. Miao, H. Zhong, and Q. Pan. Simultaneous stabilization and tracking of nonholonomic mobile robots: A Lyapunov-based approach. *IEEE Trans. on Control Systems Technology*, 23(4):1440–1450, 2015.

[5] R. Murray and S. Sastry. Nonholonomic motion planning: steering using sinusoids. *IEEE Trans. on Automatic Control*, 38(5):700–716, 1993.

[6] N. E. Pears. Mobile robot tracking of pre-planned paths. *Advanced Robotics*, 15(1):97–107, 2001.

[7] M. Defoort, J. Palos, A. Kokosy, T. Floquet, W. Perruquetti, and D. Boulinguez. Experimental motion planning and control for an autonomous nonholonomic mobile robot. In *IEEE Int. Conf. on Robotics and Automation*, pages 2221–2226, 2007.

[8] E. S. Elyoussef, N. A. Martins, E. R. De Pieri, and U. F. Moreno. PD-super-twisting second order sliding mode tracking control for a nonholonomic wheeled mobile robot. In *19th IFAC World Congress*, pages 3827–3832, 2014.

[9] A. Donaire, J. G. Romero T. Perez, and R. Ortega. Smooth stabilisation of nonholonomic robots subject to disturbances. In *IEEE Int. Conf. on Robotics and Automation*, pages 4385–4390, 2015.

[10] Y. Ma, G. Zheng, W. Perruquetti, and Z. Qiu. Control of nonholonomic wheeled mobile robots via i-PID controller. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 4413–4418, 2013.

[11] M. Guerra, D. Efimov, G. Zheng, and W. Perruquetti. Finite-time obstacle avoidance for unicycle-like robot subject to additive input disturbances. *Autonomous Robots*, In Press:1–12, 2016.

[12] R. Ortega and E. Garcia-Canseco. Interconnection and damping assignment passivity-based control: A survey. *European Journal of Control*, 10:432–450, 2004.

[13] H. K. Khalil and J. W. Grizzle. *Nonlinear Systems*, volume 3. Prentice Hall Upper Saddle River, 2002.

[14] J. A. Moreno and M. Osorio. Strict Lyapunov functions for the super-twisting algorithm. *IEEE Trans. on Automatic Control*, 57(4):1035–1040, 2012.

[15] O. Michel. Webots: professional mobile robot simulation. *Int. J. of Advanced Robotic Systems*, 1(1):39–42, 2004.