J Intell Robot Syst (2010) 60:395–414
DOI 10.1007/s10846-010-9420-0

PDC Control Design for Non-holonomic Wheeled Mobile Robots with Delayed Outputs

El-Hadi Guechi · Jimmy Lauber · Michel Dambrine · Gregor Klančar · Sašo Blažič

Received: 30 March 2009 / Accepted: 29 March 2010 / Published online: 23 April 2010

Abstract This paper presents a new technique for tracking-error model-based Parallel Distributed Compensation (PDC) control for non-holonomic vehicles where the outputs (measurements) of the system are delayed and the delay is constant. Briefly, this technique consists of rewriting the kinematic error model of the mobile robot tracking problem into a TS fuzzy representation and finding a stabilizing controller by solving LMI conditions for the tracking-error model. The state variables are estimated by nonlinear predictor observer where the outputs are delayed by a constant delay. To illustrate the efficiency of the proposed approach a comparison between the TS fuzzy observer and the nonlinear predictor observer is shown. For this study the reference trajectory is built by taking into account the acceleration limits of the mobile robot. All experiments are implemented on simulation and the real-time platform.

E.-H. Guechi (✉) · J. Lauber · M. Dambrine
LAMIH FRE CNRS 3304, Université de Valenciennes et du Hainaut-Cambrésis,
Le Mont Houy, 59313 Valenciennes Cedex 9, France
e-mail: el-hadi.guechi@univ-valenciennes.fr

J. Lauber
e-mail: jimmy.lauber@univ-valenciennes.fr

M. Dambrine
e-mail: michel.dambrine@univ-valenciennes.fr

G. Klančar · S. Blažič
Laboratory of Modeling, Simulation and Control, Faculty of Electrical Engineering, Tržaška 25,
1000 Ljubljana, Slovenia

G. Klančar
e-mail: gregor.klancar@fe.uni-lj.si

S. Blažič
e-mail: saso.blazic@fe.uni-lj.si

Keywords PDC control · TS fuzzy model · Non-holonomic mobile robot · TS fuzzy observer · Non-linear predictor observer · Delay system · Trajectory tracking

1 Introduction

Control of mobile robots is a well-known challenge in autonomous robot control community. In the last years, numerous applications of mobile robots have appeared especially in the aeronautical space exploration, inspection of nuclear power plants, automated agriculture and mobile-robot games such as robot soccer [1], motivating the increasing interest in mobile robotics. Such applications need to find an adequate control law for the mobile robot even in hard cases such as existence of moving obstacles to avoid [2] and where the outputs of the system are delayed. Finding a control law for tracking problem remains also a remarkable challenge and has opened a research area in the control of mobile robots. Many approaches are available in the literature to study this area. In [3], a fuzzy regulator based on inference motor for mobile robot control is presented, the drawback of this control method is that the inference matrix of the fuzzy regulator is filled arbitrarily and moreover the robot may overlap its final position. Flatness properties of many types of mobile robots have also been exploited for trajectory tracking [4, 5]. This approach is powerful to solve the problem of motion planning, but it turns out to be more problematic if asymptotic tracking is needed because of the existence of a singularity when the robot velocity reaches zero [4].

A lot of work has been done on reference trajectory tracking where the tracking error-model is used [6]. If the Brockett's condition is not satisfied, the tracking error model cannot be stabilized by a continuously differentiable, static feedback [7]. However, we can stabilize it by time-varying feedback [8] or a discontinuous one such as integral sliding mode [9]. The alternative is to use the combination of a nonlinear feed-forward control and a linear feedback control [6, 10, 11]. The stability of the tracking-error model is proven by using a specific Lyapunov function and the stabilizing gains are computed by linearizing the tracking-error model around zero. In [9], the authors propose a control design by integral sliding mode for trajectory tracking in order to achieve robustness with respect to some matching perturbations.

The approach proposed in our paper is based on Takagi–Sugeno fuzzy modeling, introduced in [12], obtained from the nonlinear model of the kinematic error. Then, a classical Parallel Distributed Compensation (PDC) law [13] is computed using LMI techniques [14]. The proposed architecture guarantees the stability in a compact set of the error states for all the trajectories with linear and angular velocities bounded. The important feature of the proposed approach is that it also takes into account the delay in the outputs of the system where the stability of the proposed observer is treated formally. Moreover, with this approach control law can be implemented easily in real time since it is possible to find stabilizing gains that can operate for several trajectories when their linear and angular velocities are bounded and the stability property is proven for any initial condition in a pre-specified compact set of the state space.

This paper is organized as follows. Section 2 is devoted to modeling and trajectory generation issues of mobile wheeled robots. In Section 3, the transformation of the

nonlinear model into Takagi–Sugeno representation and the synthesis of a PDC law are presented. The filtering issue of the measurements by a TS fuzzy observer is presented in Section 4. The control design taking into account the constant delay in the outputs of the system is presented in Section 5. Simulation and experimental results are presented in Sections 6 and 7, respectively.

2 Mobile Robot Modeling and Trajectory Generation

2.1 Kinematic Model

By taking into account the non slipping condition, the kinematic model of the mobile robot in the X-Y plane (see Fig. 1) can be written as follows:

$$\begin{cases} \dot{x} = v\ \cos\theta \\ \dot{y} = v\ \sin\theta \\ \dot{\theta} = w \end{cases} \tag{1}$$

where the considered control inputs of the mobile robot v and w are the linear and the angular speed of the robot, respectively. The output variables are x and y (the robot gravity-center position) and θ (the angle between the speed vector and the x-axis, i.e., the robot orientation).

2.2 Kinematic Error-model of Trajectory Tracking

Figure 2 illustrates the definition of the posture error $e = \begin{bmatrix} e_x & e_y & e_\theta \end{bmatrix}^T$ expressed in the frame of the real robot and determined—using the actual posture $q = \begin{bmatrix} x & y & \theta \end{bmatrix}^T$ of the real robot and the reference posture $q_r = [x_r \ \ y_r \ \ \theta_r]^T$ of a virtual reference robot—by the following equation

$$\begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} (q_r - q) \tag{2}$$

From Eqs. 1 and 2 and assuming that the virtual robot has a kinematic model similar to Eq. 1, the posture error model can be written as follows:

$$\begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_\theta \end{bmatrix} = \begin{bmatrix} \cos(e_\theta) & 0 \\ \sin(e_\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_r \\ w_r \end{bmatrix} + \begin{bmatrix} -1 & e_y \\ 0 & -e_x \\ 0 & -1 \end{bmatrix} u \tag{3}$$

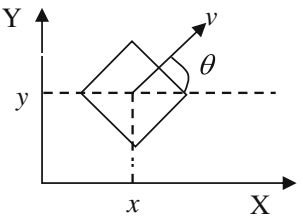
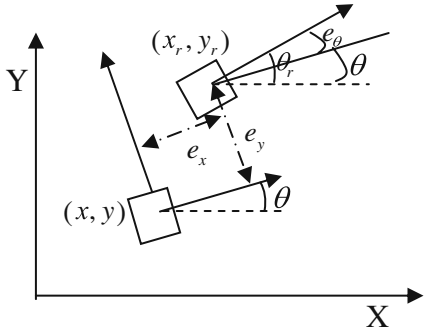


Fig. 1 Mobile robot

Fig. 2 Posture error



where v_r is the linear reference velocity and w_r is the angular reference velocity. The control law is then defined as $u^T = [v \;\; w]$. In [10] the authors proposed to set $u = u_F + u_B$ with $u_F = [v_r \;\cos(e_\theta) \;\; w_r]^T$ a feed-forward control action vector and u_B a feedback control action to be defined later. Inserting the control expression in Eq. 3, the resulting model is given by:

$$\begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_\theta \end{bmatrix} = \begin{bmatrix} 0 & w_r & 0 \\ -w_r & 0 & v_r \text{sinc}(e_\theta) \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} + \begin{bmatrix} -1 & e_y \\ 0 & -e_x \\ 0 & -1 \end{bmatrix} u_B \tag{4}$$

2.3 Trajectory Generation with Limited Acceleration

The goal of the trajectory generation is to meet the following objectives:

- The initial posture of the robot is (x_i, y_i, θ_i) and its initial linear velocity v_i
- The final posture of the robot is (x_f, y_f, θ_f) and its final linear velocity is v_f.
- The acceleration is limited as discussed later.

Without loss of generality we can assume that the linear reference velocity of the robot is always non-negative.

2.3.1 C^3 Bézier Curve Generation

The above goal is achieved by using the following C^3 Bézier curve as a reference trajectory which is expressed by the following equations:

$$x_r(\eta) = x_i(1-\eta)^3 + 3x_1\eta(1-\eta)^2 + 3x_2\eta^2(1-\eta) + x_f\eta^3; \tag{5}$$

$$y_r(\eta) = y_i(1-\eta)^3 + 3y_1\eta(1-\eta)^2 + 3y_2\eta^2(1-\eta) + y_f\eta^3; \tag{6}$$

where the parameter $\eta \in [0,1]$, the coordinates (x_1, y_1) and (x_2, y_2) are respectively equal to $(x_i + d_i\cos(\theta_i), y_i + d_i\sin(\theta_i))$, and $\left(x_f - d_f\cos\left(\theta_f\right), y_f - d_f\sin\left(\theta_f\right)\right)$. This ensures that the robot starts with its initial orientation θ_i and that it reaches its final position with orientation θ_f. The length d_i is the distance between the initial and the first intermediate control point. The length d_f is the distance between the final and the second intermediate control point. Those distances are computed to obtain the optimal trajectory where the initial velocity v_i and the final velocity v_f of the robot are satisfied [15]. In the forward movement the reference linear velocity $v_r(\eta)$,

the angular velocity $w_r(\eta)$, and the curvature $\kappa(\eta)$ are obtained by using a flatness principle [2, 4], as follows:

$$v_r(\eta) = \sqrt{x_r'(\eta)^2 + y_r'(\eta)^2}$$

$$w_r(\eta) = \frac{x_r'(\eta).y_r''(\eta) - y_r'(\eta).x_r''(\eta)}{x_r'(\eta)^2 + y_r'(\eta)^2} \tag{7}$$

$$\kappa(\eta) = \frac{x_r'(\eta).y_r''(\eta) - y_r'(\eta).x_r''(\eta)}{\left(x_r'(\eta)^2 + y_r'(\eta)^2\right)^{3/2}} \tag{8}$$

2.3.2 Acceleration Limitations

The tangential and radial accelerations are given by the expressions:

$$a_t = \frac{dv}{dt}, \quad a_r = v\,w = v^2\kappa \tag{9}$$

In order to respect the maximal radial and tangential acceleration, the two accelerations of the trajectory must be inside the ellipse given by [16]:

$$\frac{a_t^2}{(a_t^{\max})^2} + \frac{a_r^2}{\left(a_r^{\max}\right)^2} = 1 \tag{10}$$

The maximal tangential acceleration $a_t^{\max}$ and the radial acceleration $a_r^{\max}$ can be obtained experimentally according to the friction force and the mass of the robot. In our case the maximal tangential acceleration $a_t^{\max} = 0.5\ \mathrm{m/s^2}$ and the maximal radial acceleration $a_r^{\max} = 1\ \mathrm{m/s^2}$ were taken.

The reference trajectory for this study is shown in Fig. 3. It is composed of two C^3 Bezier curves (shown in Fig. 3 by a dashed line and a full line, respectively). The

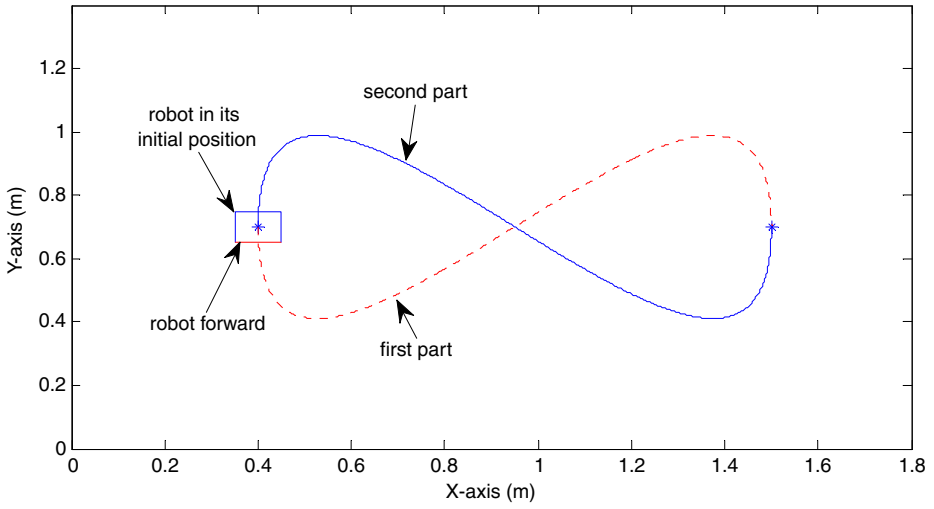


Fig. 3 Reference trajectory of the robot

first part of the reference trajectory starts from ($x_i = 0.4$ m; $y_i = 0.7$ m; $\theta_i = -90°$) with initial linear velocity $v_i = 0.02$ m/s and reaches a final position ($x_f = 1.5$ m; $y_f = 0.7$ m; $\theta_f = -90°$) with linear velocity $v_f = 0.2$ m/s.

Figure 4 shows the curvature of the first part of the reference trajectory. In the turning points TP1 and TP2, where the absolute value of curvature is maximum, the radial acceleration is maximal [16]. The position of these two turning points depends on the one of the intermediate control points of the Bezier curve (x_1, y_1) and (x_2, y_2). Using the radial acceleration formula presented in Eq. 9, the maximal allowable speed in each turning point is given by the following equation:

$$v(\eta) = \sqrt{\frac{a_r^{\max}}{|\kappa(\eta)|}}$$

Before and after the turning points TP1 and TP2, the robot can accelerate or decelerate. In this case, the maximum velocity profile (Fig. 5) for each point is determined and the tangential acceleration is computed using the formula 10 [16]. The maximal allowable velocity of the trajectory is equal to the minimum velocity of all velocity profiles (see Fig. 5). The arc length can be computed by:

$$ds(\eta) = \sqrt{x_r'(\eta)^2 + y_r'(\eta)^2}\, d\eta, \tag{11}$$

where $x_r'(\eta)$ and $y_r'(\eta)$ are derivatives of x and y with respect to η.

The travel time required to follow the trajectory from the initial point to the final point can be calculated from:

$$t = \int_0^1 \frac{\sqrt{x_r'(\eta)^2 + y_r'(\eta)^2}}{v_r(\eta)} d\eta \tag{12}$$

This can be integrated by using a numerical quadrature scheme. The solution for the reference trajectory is only obtained for the first part (Fig. 5). The results for the

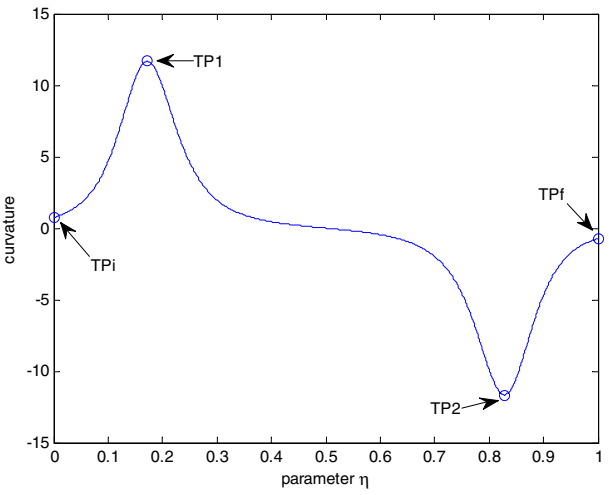


Fig. 4 Curvature of the first part of the reference trajectory

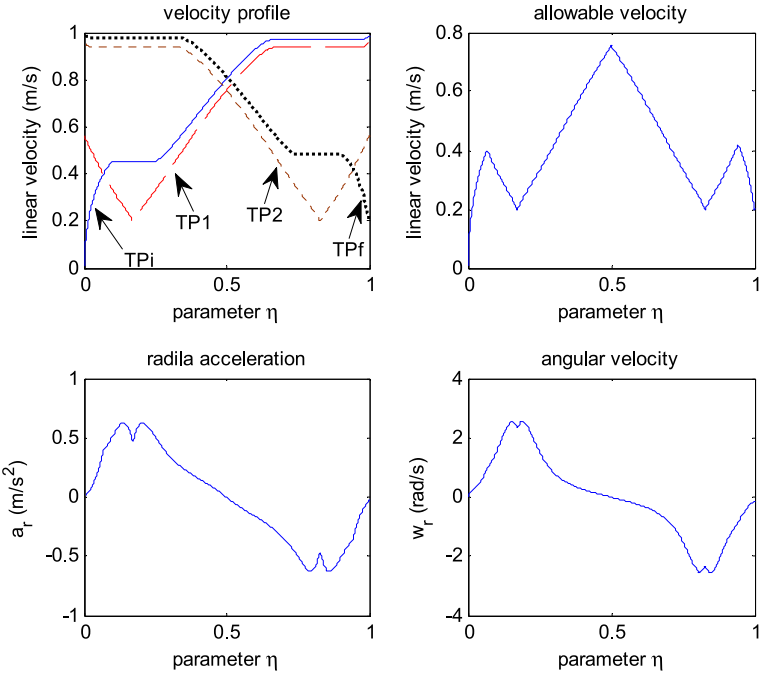


Fig. 5 Velocity profile (*upper left*), allowable velocity (*upper right*), radial acceleration (*lower left*) and angular velocity (*lower right*)

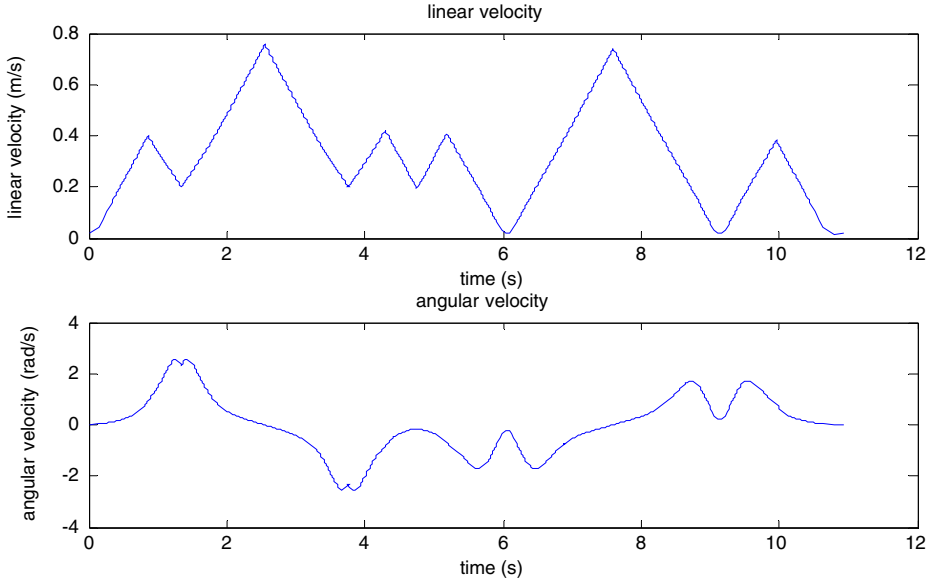


Fig. 6 Reference linear and angular velocity of the robot

second part follow in a straight-forward manner. The travel time and the length of the first part of the reference trajectory are 4.76 s and 1.73 m, respectively.

The second part of the reference trajectory starts from ($x_i = 1.5$ m; $y_i = 0.7$ m; $\theta_i = -90°$) with initial linear velocity $v_i = 0.2$ m/s and reaches a final position ($x_f = 0.4$ m; $y_f = 0.7$ m; $\theta_f = -90°$) with linear velocity $v_f = 0.02$ m/s. The final reference trajectory composed by these two C^3 Bezier curves (Fig. 3) has a travel time of 11.7 s and the length of 3.47 m. The linear and the angular velocities of the final reference trajectory are presented in Fig. 6. The minimum and the maximum reference linear velocity are 0.0138 and 0.755 m/s, respectively, while the minimum and the maximum reference angular velocity are −2.552 and 2.553 rad/s, respectively.

3 TS Fuzzy Modeling and Control Design

The TS models are represented through the following polytopic form [17]:

$$\begin{cases} \dot{\xi}(t) = \sum_{i=1}^{r} h_i(z(t))\left(A_i\xi(t) + B_i u(t)\right) \\ \upsilon(t) = \sum_{i=1}^{r} h_i(z(t))\left(C_i\xi(t)\right) \end{cases} \tag{13}$$

where $\xi(t) \in \mathrm{R}^n$ is a state vector, $\upsilon(t) \in \mathrm{R}^p$ is an output vector and $z(t) \in \mathrm{R}^q$ is the premise vector depending on the state vector, A_i, B_i, C_i are constant matrices. The number of rules r is related to the number of the nonlinearities of the model as shown later. And finally, the nonlinear weighting functions $h_i(z(t))$ are all non negative and such that $\sum_{i=1}^{r} h_i = 1$. For any nonlinear model, one can find an equivalent fuzzy TS model in a compact region of the state space variable using the sector nonlinearity approach which consists in decomposing each bounded nonlinear terms in a convex combination of its bounds [17]. Considering the nonlinear tracking error model (Eq. 4), four bounded nonlinear functions appear: $n_1 = w_r$, $n_2 = v_r \operatorname{sinc}(e_\theta)$, $n_3 = e_y$ and $n_4 = e_x$ leading to $r = 16$ rules. In order to avoid loss of controllability, we will assume the following assumptions:

$$|e_x| \leq 0.1 \text{ m}, \ |e_y| \leq 0.1 \text{ m}, \ |e_\theta| \leq \frac{\pi}{2} \text{ rad}, \tag{14}$$

giving the TS model:

$$\dot{e}(t) = A_{z(t)}e(t) + B_{z(t)}u_B(t) \tag{15}$$

where $A_{z(t)} = \sum_{i=1}^{r} h_i(z(t))A_i$, $B_{z(t)} = \sum_{i=1}^{r} h_i(z(t))B_i$, $n = 16$ and

$$A_i = \begin{bmatrix} 0 & -\varepsilon_i^1 w_{r,\max} & 0 \\ \varepsilon_i^1 w_{r,\max} & 0 & \mu_i \\ 0 & 0 & 0 \end{bmatrix}, \quad B_i = \begin{bmatrix} -1 & \varepsilon_i^3 e_{\max} \\ 0 & \varepsilon_i^4 e_{\max} \\ 0 & -1 \end{bmatrix}$$

$w_{r,\max} = 2.5529$ (rad/s), $e_{\max} = 0.1$ (m), and

$$\varepsilon_i^1 = \begin{cases} +1 & \text{for } 1 \le \text{i} \le 8 \\ -1 & \text{else} \end{cases}, \quad \varepsilon_i^3 = \begin{cases} +1 & \text{for i} \in \{3,\ 4,\ 7,\ 8,\ 11,\ 12,\ 15,\ 16\} \\ -1 & \text{else} \end{cases},$$

$$\varepsilon_i^4 = (-1)^{i+1}, \quad \mu_i = \begin{cases} 0.0088 & \text{for } 1 \le \text{i} \le 4 \text{ and } 9 \le \text{i} \le 12 \\ 0.755 & \text{else} \end{cases},$$

In order to stabilize the TS model (15), a PDC (Parallel Distributed Compensation) control law [13]:

$$u_B(t) = -\sum_{i=1}^{r} h_i(z(t))\, F_i e(t) = -F_{z(t)} e(t) \tag{16}$$

is used. Its gains are computed according to the following lemmas that combines Theorem 2.2 from [18] and a decay rate performance measure from [19]. The proof follows exactly the same lines as the proofs in original works.

Lemma 1 [18] *The TS model* (15) *can be stabilized via the PDC control law* (16) *with the decay rate performance of* $\gamma > 0$ [19] *if there exist matrices* M_i $(i = 1, 2, ..., r)$ *and* $X > 0$ *such that the following LMI conditions hold:*

$$\begin{cases} \Upsilon_{ii} < 0, \quad i = 1, 2, ..., r \\ \dfrac{2}{r-1}\Upsilon_{ii} + \Upsilon_{ij} + \Upsilon_{ji} < 0, \quad i, j = 1, 2, ..., r,\ i \neq j \end{cases}$$

with $\Upsilon_{ij} = XA_i^T + A_iX - M_j^T B_i^T - B_i M_j + \gamma X$ *where* $\gamma > 0$. *The gains* F_i *of* (16) *are given by* $F_i = M_i X^{-1}$.

Remark If $V(e(t))$ is a Lyapunov function, the decay rate γ is defined by the following property: $\dot{V}(e(t)) \le -\gamma \cdot V(e(t))$, ensuring temporal performances.

In our case, the LMI conditions given in *Lemma 1* are feasible.
According to the constraint on the control input, the following LMIs are added.

Lemma 2 [19] *Assume that the initial condition* $e(0)$ *is known. The constraint* $v^2(t) + w^2(t) \le \sigma^2$ *is enforced at all times* $t \ge 0$ *if the following LMIs hold:*

$$\begin{bmatrix} 1 & e(0)^T \\ e(0) & X \end{bmatrix} \ge 0, \quad \begin{bmatrix} X & M_i^T \\ M_i & \sigma^2 I \end{bmatrix} \ge 0$$

4 Filtering by TS Fuzzy Observer

Generally, if the measurement of the full state vector is not available, an observer is added to the control structure to derive an utput feedback law. In our case, all the

states are measured, but these measurements of the robot position are noisy. So, for filtering purpose, a TS fuzzy observer is introduced:

$$\dot{\hat{e}}(t) = A_{z(t)}\hat{e}(t) + B_{z(t)}u_B(t) + K_{z(t)}\left(e(t) - \hat{e}(t)\right) \tag{17}$$

where $K_{z(t)} = \sum_{i=1}^{r} h_i(z(t))K_i$, $\hat{e}(t)$ is the estimated posture error. The control law changes to:

$$u_B(t) = -F_{z(t)}\hat{e}(t) \tag{18}$$

The dynamics of the estimation error $\tilde{e}(t) = e(t) - \hat{e}(t)$ are given by:

$$\dot{\tilde{e}}(t) = \left(A_{z(t)} - K_{z(t)}I\right)\tilde{e}(t) \tag{19}$$

Lemma 3 [18] *The estimation error* (19) *converges to zero if there exists matrices* $N_i (i = 1, 2, \ldots, r)$ *and* $P_o > 0$ *such that the following LMI conditions hold:*

$$\begin{cases} \Upsilon_{ii} < 0, \quad i = 1, 2, ..., r \\ \dfrac{2}{r-1}\Upsilon_{ii} + \Upsilon_{ij} + \Upsilon_{ji} < 0, \quad i, j = 1, 2, ..., r,\ i \neq j \end{cases}$$

with $\Upsilon_{ij} = A_i^T P_o + P_o A_i - N_j C_i - C_i^T N_j^T$. *The gains* K_i *are given by* $K_i = P_o^{-1} N_i$.

Using the control law (Eq. 18), the closed-loop system (Eq. 15) can be put in the following form:

$$\dot{e}(t) = \left(A_{z(t)} - B_{z(t)}F_{z(t)}\right)e(t) + B_{z(t)}F_{z(t)}\tilde{e}(t) \tag{20}$$

The tracking error model (Eq. 20) and the estimation error model (Eq. 19) can be combined into the augmented model as follows:

$$\begin{bmatrix} \dot{e} \\ \dot{\tilde{e}} \end{bmatrix} = \begin{bmatrix} A_z - B_z F_z & B_z F_z \\ 0 & A_z - K_z I \end{bmatrix} \begin{bmatrix} e \\ \tilde{e} \end{bmatrix}, \tag{21}$$

And so, the controller gains F_i and the observer gain K_i can be designed independently according to the separation principle [20]. Note that for the calculation of n_3 and n_4 one must still use e_x and e_y (not $\hat{e}_x$ and $\hat{e}_y$) for the separation principle to hold.

We obtain a good tracking using fuzzy observer if the measurements are not delayed. In practice, for example in the robot soccer game where the work space is supervised by the camera, the posture of the robot is obtained after a delay τ which makes the tracking task more difficult. For solving this problem, we propose a nonlinear predictor observer which is the subject of the next section.

5 Control Design with Nonlinear Predictor Observer

For the visual control where the outputs feedback is delayed, it is possible to make the combination of observation and prediction. This idea proposed by [21, 22] in a

case of a linear system will be developed here for the prediction of the mobile robot state. For this, consider the following nonlinear predictor observer:

$$\begin{cases} \dot{\hat{x}}(t) = v(t)\cos\left(\hat{\theta}(t)\right) - L_1\left(x(t-\tau) - \hat{x}(t-\tau)\right) \\ \dot{\hat{y}}(t) = v(t)\sin\left(\hat{\theta}(t)\right) - L_2\left(y(t-\tau) - \hat{y}(t-\tau)\right) \\ \dot{\hat{\theta}}(t) = w(t) - L_3\left(\theta(t-\tau) - \hat{\theta}(t-\tau)\right) \end{cases} \tag{22}$$

where L_1, L_2, L_3 are constant gains, and τ is the value of the measurement delay assumed to be constant. The block diagram of the whole closed-loop control system using nonlinear predictor observer (Eq. 22) and PDC control law is shown in Fig. 7 where $\hat{q}(t) = [\hat{x}(t)\ \ \hat{y}(t)\ \ \hat{\theta}(t)]^T$ is the estimated state.

Let $\phi_x(t) = x(t) - \hat{x}(t)$, $\phi_y(t) = y(t) - \hat{y}(t)$ and $\phi_\theta(t) = \theta(t) - \hat{\theta}(t)$ be the estimations of the errors. Using the kinematic model (Eq. 1) and the nonlinear predictor observer (Eq. 22), the dynamics of the estimation errors can be written as follows:

$$\begin{cases} \dot{\phi}_x(t) = e_1(t) + L_1\phi_x(t-\tau) \\ \dot{\phi}_y(t) = e_2(t) + L_2\phi_y(t-\tau) \\ \dot{\phi}_\theta(t) = L_3\phi_\theta(t-\tau) \end{cases} \tag{23}$$

where $e_1(t) = (\cos(\theta(t)) - \cos(\hat{\theta}(t)))v(t)$ and $e_2(t) = (\sin(\theta(t)) - \sin(\hat{\theta}(t)))v(t)$. If the gains L_i are such that $L_i\,\tau < \pi/2$ for $i \in \{1, 2, 3\}$, then the estimation errors converge asymptotically towards 0. Indeed, from the Nyquist's stability criterion, $\phi_\theta(t)$ converges exponentially to 0. Since the functions *sin* and *cos* are both contractive, we have $|e_i(t)| \leq |\phi_\theta(t)|\,|v(t)| \leq |\phi_\theta(t)|\,v_{\max}$, for $i \in \{1, 2\}$. So the two first equations of (Eq. 23) may be seen as asymptotically stable, linear time-invariant systems with exponentially vanishing inputs $e_i(t)$.

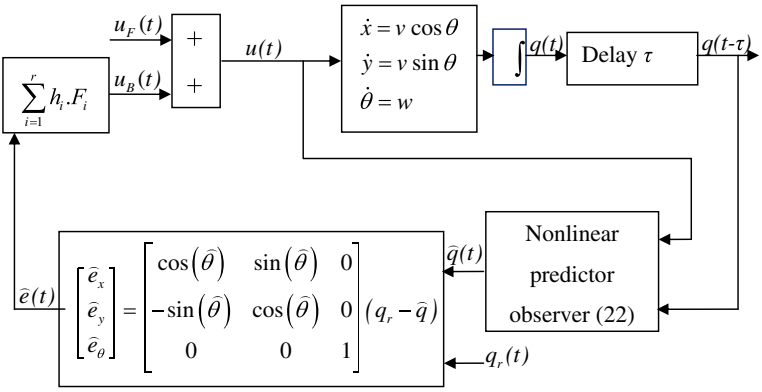


Fig. 7 Block diagram of the whole closed-loop system with nonlinear predictor observer

6 Simulation Results

In order to check the efficiency of the proposed approach, a simulation study was done. The proposed PDC control was used for trajectory tracking. The gains F_i were obtained by using Lemma 1 with $\gamma = 0.05$ and Lemma 2 with $\sigma^2 = 5$ and $e(0) = [0.01\,(\mathrm{m}) - 0.02\,(\mathrm{m}) - 45^\circ]^T$:

$$F_1 = \begin{bmatrix} -13.9877 & -16.5821 & -2.9276 \\ 0.6140 & -2.6436 & -0.5775 \end{bmatrix}, \quad F_2 = \begin{bmatrix} -14.0373 & -15.7158 & -2.8102 \\ 0.1187 & -1.3585 & -0.2857 \end{bmatrix},$$

$$F_3 = \begin{bmatrix} -13.6341 & -18.5365 & -3.3632 \\ -0.5503 & -2.6484 & -0.5785 \end{bmatrix}, \quad F_4 = \begin{bmatrix} -13.9523 & -16.2994 & -2.9271 \\ -0.0731 & -1.3551 & -0.2851 \end{bmatrix},$$

$$F_5 = \begin{bmatrix} -14.8533 & -7.7818 & -1.2700 \\ 1.9132 & -33.3688 & -7.9544 \end{bmatrix}, \quad F_6 = \begin{bmatrix} -14.9139 & -5.8723 & -0.9079 \\ 1.1836 & -31.5206 & -7.8053 \end{bmatrix},$$

$$F_7 = \begin{bmatrix} -14.5437 & -12.4382 & -2.2881 \\ 3.7322 & -32.6158 & -7.7841 \end{bmatrix}, \quad F_8 = \begin{bmatrix} -14.9061 & -6.0653 & -1.0266 \\ 1.4352 & -31.4754 & -7.7925 \end{bmatrix},$$

$$F_9 = \begin{bmatrix} -13.6333 & 18.5412 & 3.3641 \\ 0.5516 & -2.6467 & -0.5782 \end{bmatrix}, \quad F_{10} = \begin{bmatrix} -13.9519 & 16.3020 & 2.9275 \\ 0.0739 & -1.3546 & -0.2850 \end{bmatrix},$$

$$F_{11} = \begin{bmatrix} -13.9871 & 16.5869 & 2.9284 \\ -0.6128 & -2.6419 & -0.5772 \end{bmatrix}, \quad F_{12} = \begin{bmatrix} -14.0369 & 15.7184 & 2.8107 \\ -0.1178 & -1.3580 & -0.2856 \end{bmatrix},$$

$$F_{13} = \begin{bmatrix} -14.5430 & 12.4464 & 2.2897 \\ -3.7353 & -32.6168 & -7.7840 \end{bmatrix}, \quad F_{14} = \begin{bmatrix} -14.9060 & 6.0663 & 1.0267 \\ -1.4354 & -31.4754 & -7.7925 \end{bmatrix},$$

$$F_{15} = \begin{bmatrix} -14.8532 & 7.7843 & 1.2705 \\ -1.9140 & -33.3686 & -7.9543 \end{bmatrix}, \quad F_{16} = \begin{bmatrix} -14.9138 & 5.8733 & 0.9081 \\ -1.1839 & -31.5206 & -7.8053 \end{bmatrix},$$

In a real environment, the measurements obtained by a camera are delayed. For this simulation the outputs of the system are delayed by a constant delay of 0.132 s. First

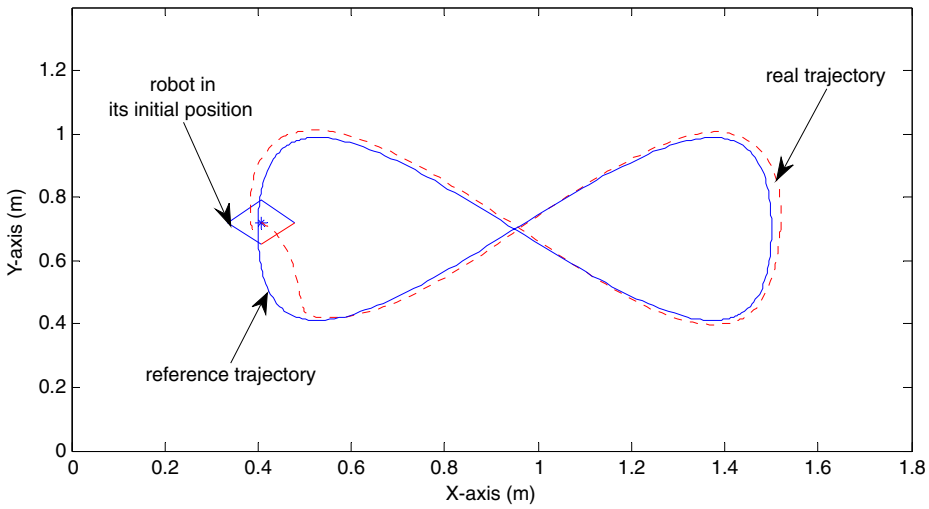


Fig. 8 Trajectory tracking with the PDC control and the TS observer

we tested the approach with the PDC control and the TS fuzzy observer (Eq. 17) although the latter does not take into account the delay. Then a combination of (Eq. 17) and the proposed nonlinear predictor observer is used to compensate for the delay. The reference trajectory from Fig. 3 is used in all experiments.

6.1 PDC and Fuzzy Observer

The states are estimated by a TS fuzzy observer that can be derived by using Lemma 3. The observer gains K_i ($i = 1, 2, \ldots, 16$) are given as follows:

$$K_1 = K_2 = K_3 = K_4 = \begin{bmatrix} 0.2502 & -0.0233 & -0.0216 \\ -0.0241 & 0.2551 & 0.0100 \\ -0.0208 & 0.0095 & 0.2365 \end{bmatrix},$$

$$K_5 = K_6 = K_7 = K_8 = \begin{bmatrix} 0.2502 & -0.0233 & -0.0216 \\ -0.0241 & 0.2610 & 0.3774 \\ -0.0208 & 0.3506 & 0.2305 \end{bmatrix},$$

$$K_9 = K_{10} = K_{11} = K_{12} = \begin{bmatrix} 0.2502 & 0.0233 & 0.0216 \\ 0.0242 & 0.2551 & 0.0099 \\ 0.0208 & 0.0095 & 0.2365 \end{bmatrix},$$

$$K_{13} = K_{14} = K_{15} = K_{16} = \begin{bmatrix} 0.2502 & 0.0234 & 0.0216 \\ 0.0242 & 0.2610 & 0.3774 \\ 0.0208 & 0.3506 & 0.2305 \end{bmatrix},$$

The approach results in a stable behavior although quite large tracking errors can be observed. Figure 8 shows the trajectory of the robot (dashed line) and the reference trajectory (full line). Figure 9 shows the robot velocities (dashed line) and the reference velocities (full line) where the difference among them is noticeable. The posture error is shown in Fig. 10.

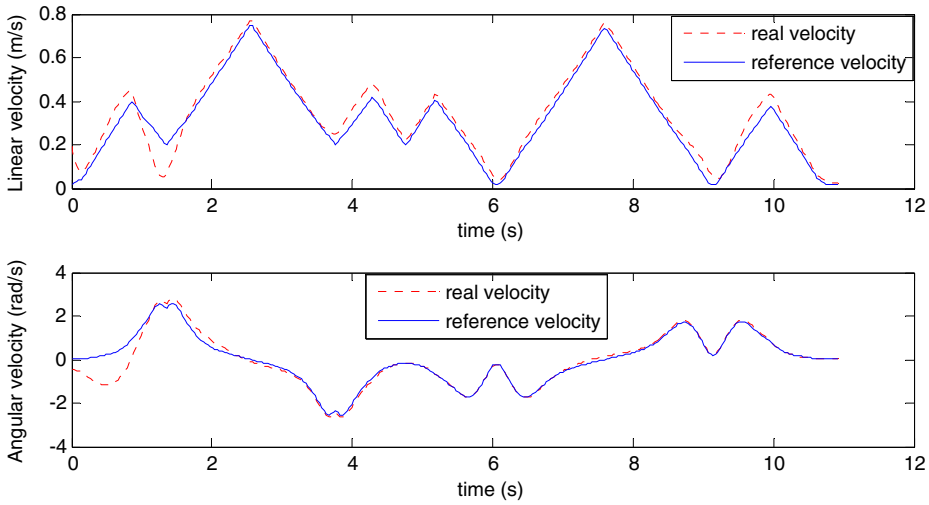


Fig. 9 Comparison of simulated and reference velocities

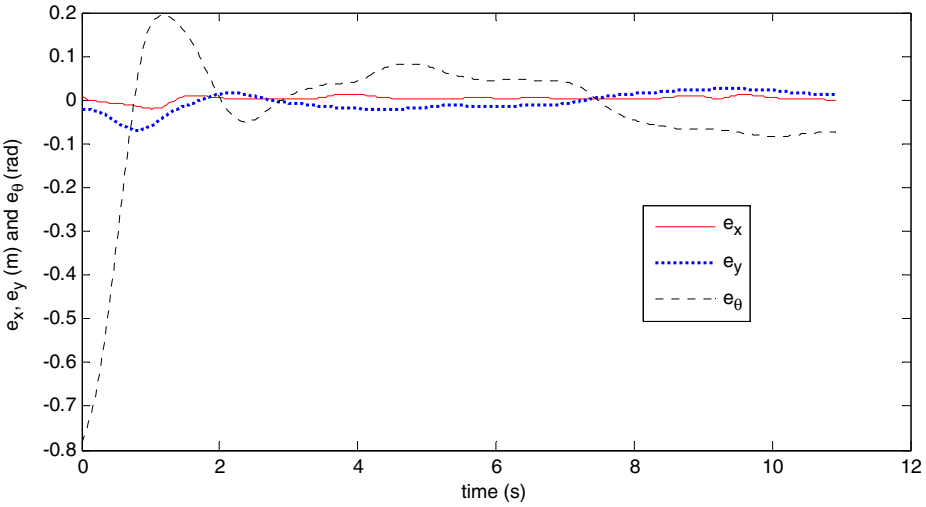


Fig. 10 Error of the robot posture

6.2 PDC and Nonlinear Predictor Observer

The states are estimated by a nonlinear predictor observer that can be derived by using the stability demonstration presented in Section 5 with the constant delay $\tau = 0.132\ s$. With delayed outputs, the proposed control structure with nonlinear predictor observer allows a better following of a given trajectory (Fig. 11) compared to the TS fuzzy observer performance (Fig. 8). Figure 11 shows that the trajectory of the robot (dashed line) converges towards the reference one (full line). The linear

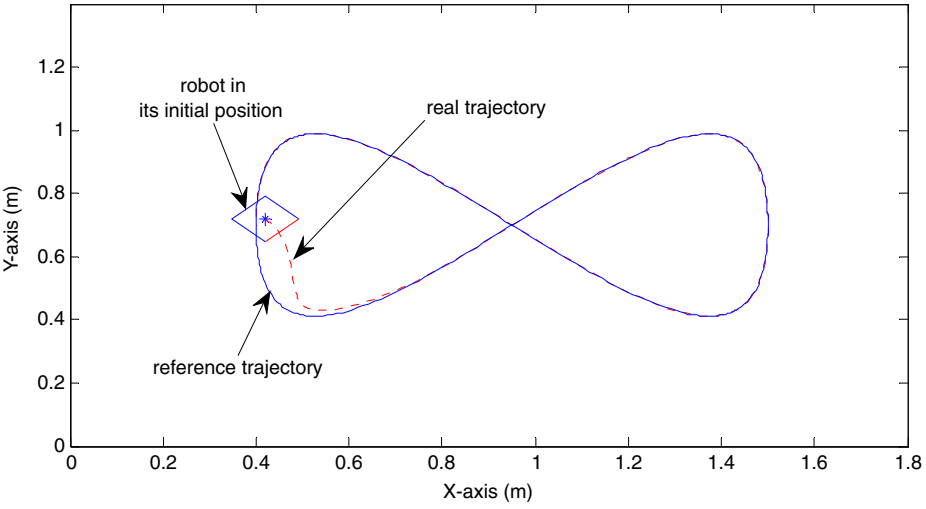


Fig. 11 Trajectory tracking with PDC control and nonlinear predictor observer

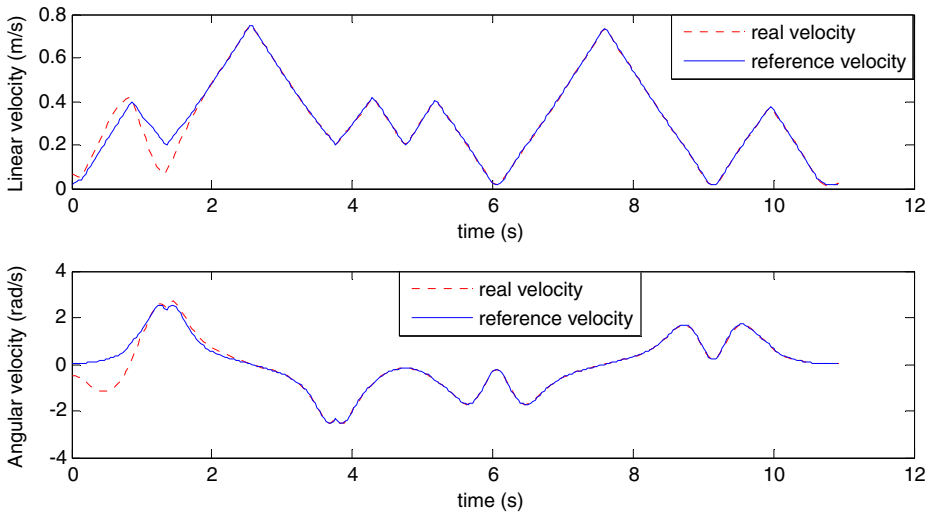


Fig. 12 Comparison of simulated and reference velocities

velocity and the angular velocity of the robot are given in Fig. 12. Figure 13 represents the convergence of the robot position error towards 0 using the proposed control law.

7 Experimental Results

The same experiments as in the previous section were applied to the trajectory tracking for a MIABOT mobile robot. The robot positions are measured via a

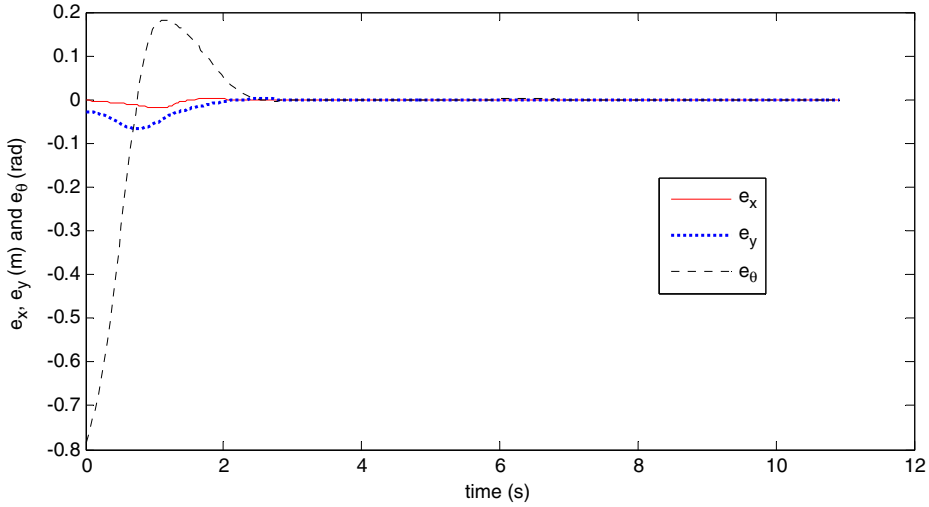


Fig. 13 Error of the robot posture

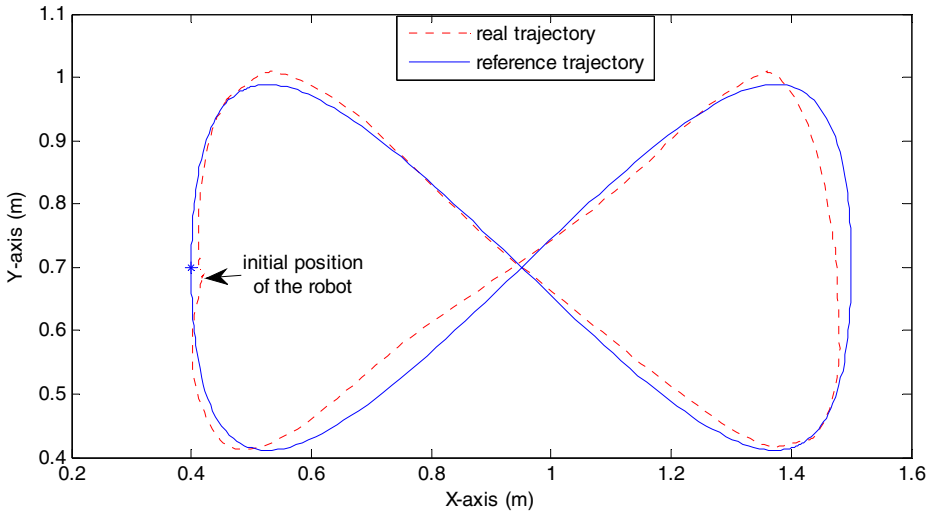


Fig. 14 Trajectory tracking with PDC control and TS observer

Basler (A311fc) camera whose frame rate is 30 images per second and the delay produced by this camera is about 66 ms. The control is sent to the robot via wireless communication. The control code is programmed in C++ which is adequate for the real-time application. The robot initial position is $x_i = 0.41$ m; $y_i = 0.68$ m; $\theta_i = 248°$.

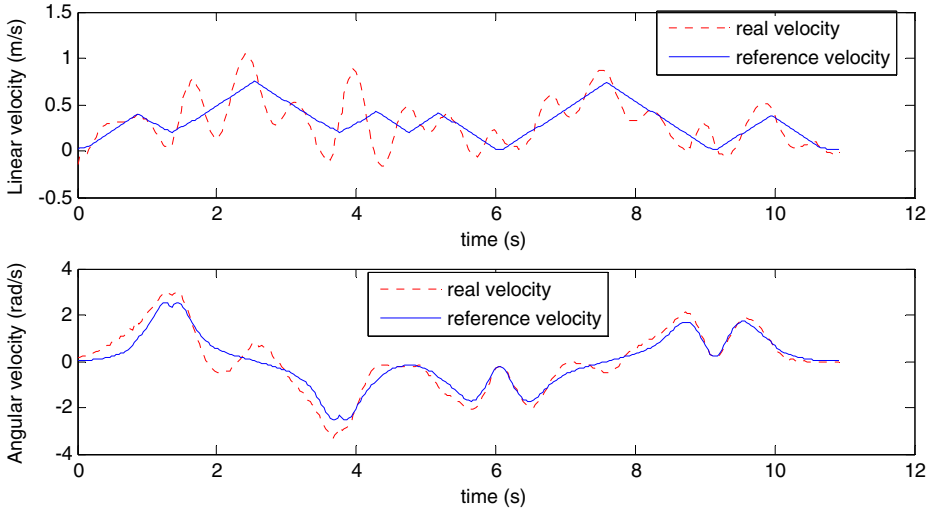


Fig. 15 Comparison of real and reference velocities

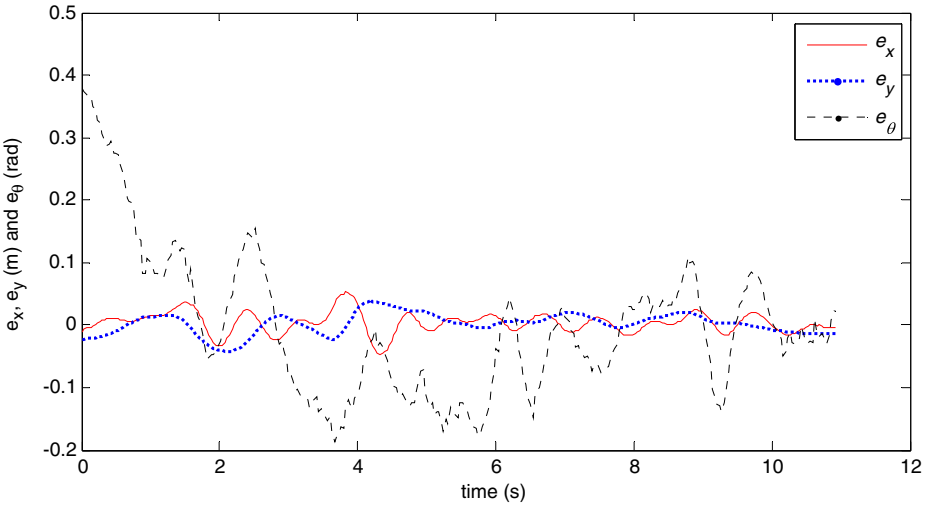


Fig. 16 Error of the robot posture

7.1 Experimental Results with TS Observer

Figure 14 shows the trajectory tracking. The robot (dashed line) follows the reference trajectory (full line) with a big error. Figure 15 shows the real velocities of the robot (dashed line) and the reference velocities (full line), where the differences are quite big. Figure 16 shows the error posture.

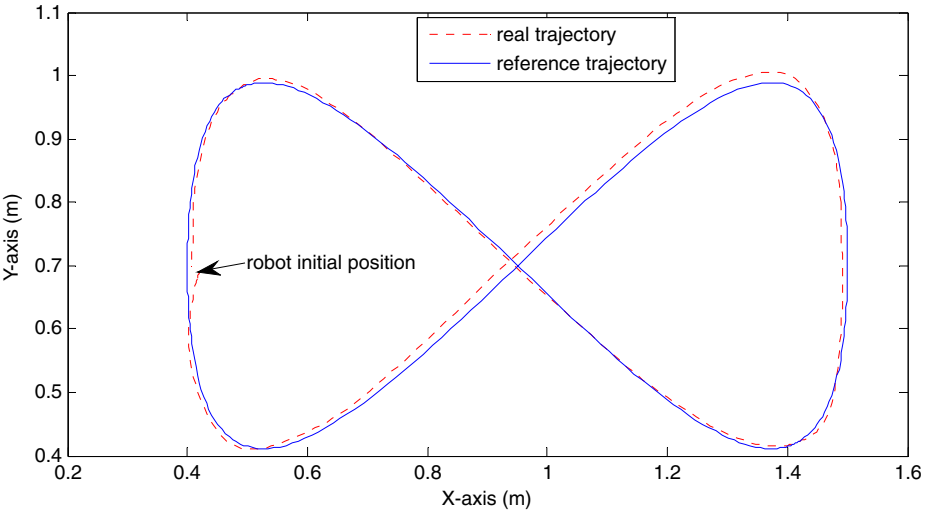


Fig. 17 Trajectory tracking with PDC control and nonlinear predictor observer

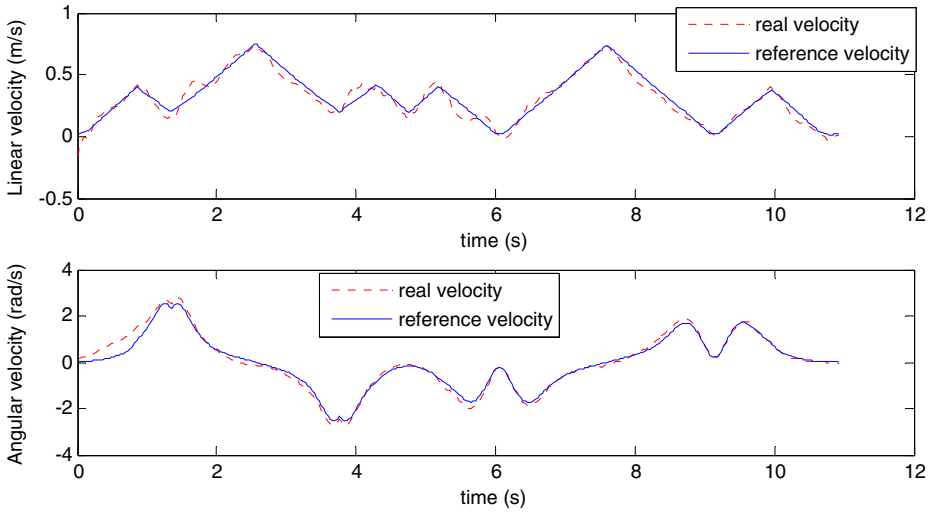


Fig. 18 Comparison of real and reference velocities

7.2 Experimental Results with Nonlinear Predictor Observer

The experimental results obtained using nonlinear predictor observer show the robot (dashed line) follows the reference trajectory (full line) better (Fig. 17) compared to the trajectory tracking using PDC control with TS observer (Fig. 14). Figure 18 shows the real velocities of the robot (dashed line) and the reference velocities (full line), where better coincidence than in the previous case (Fig. 15) is observed. Figure 19 shows the posture error. Note that the signals do not converge to 0. Rather they

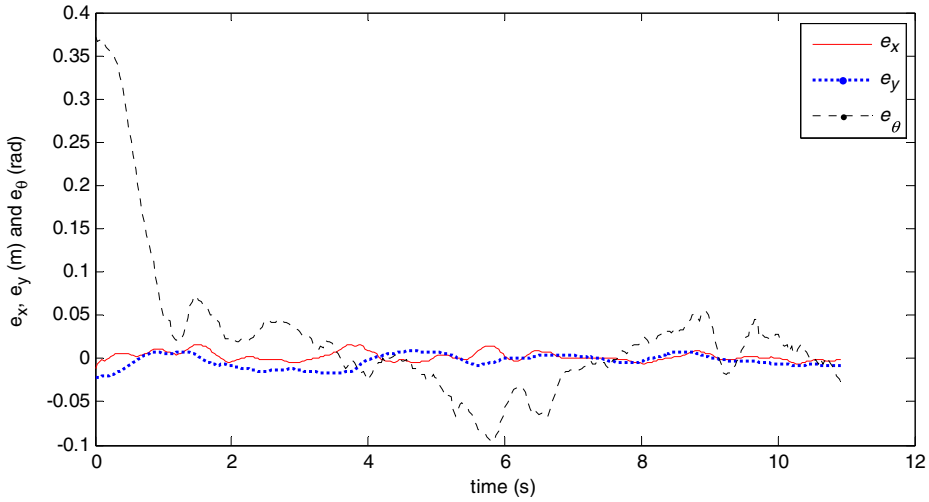


Fig. 19 Error of the robot posture

converge to some residual set that depends on the size of the disturbances. This is due to the fact that a persistent noise is present in measurements. Note also that Figs. 14, 15, 16 and 17 do not show the actual position of the robot, but the measured one.

8 Conclusion

The present article proposes a new approach of trajectory tracking for non-holonomic mobile robots with taking into account the delay in the output measurements. Two approaches were proposed and compared, namely PDC control using TS fuzzy observer and PDC control using nonlinear predictor observer. PDC control is based on the theory of TS fuzzy control. The stabilizing gains are obtained using LMIs and the stability in a compact region of the error space is proven. Good simulation and experimental results have been obtained. They show the efficiency of the proposed approach. They also show that the nonlinear predictor observer can cope with delay much better than a TS fuzzy observer as expected.

The main advantages of the proposed method are the existence of the control law for all the trajectories that respect the bounds on the linear and the angular velocities, the ability to compensate for the delay in measurements, and easy real time implementation. Future work concerns the implementation of this control law in real time for moving obstacle avoidance.

Acknowledgements The present research work has been supported by International Campus on Safety and Intermodality in Transportation the European Community, the Délégation Régionale à la Recherche et à la Technologie, the Ministère de l'Enseignement supérieur et de la Recherche, the Région Nord Pas de Calais and the Centre National de la Recherche Scientifique: the authors gratefully acknowledge the support of these institutions.

References

1. Vadakkepat, P., Peng, X., Kiat, Q.B., Heng, L.T.: Evolution of fuzzy behaviors for multi-robotic system. Robot. Auton. Syst. **55**(2), 146–161 (2007)
2. Guechi, E.-H., Lauber, J., Dambrine, M.: On-line moving-obstacle avoidance using piecewise Bézier curves with unknown obstacle trajectory. In: 16th Mediterranean Conference on Control and Automation. IEEE Med., vol. 08, pp. 505–510 (2008)
3. Yen, J., Pfluger, N.: A Fuzzy logic based extension to Payton and Rosenblatt's command fusion method for mobile robot navigation. IEEE Trans. Syst. Man Cybern. **25**(6), 971–978 (1995)
4. Fliess, M., Levine, J., Martin, P., Rouchon, P.: Flatness and defect of nonlinear systems; introductory theory and examples. Int. J. Control **61**(6), 1327–1361 (1995)
5. Laumond, J.-P.: Robot Motion Planning and Control. LNICS, vol. 229. Springer, New York (1998)
6. Kanayama, Y., Kimura, Y., Miyazaki, F., Noguchi, T.: A stable tracking control method for a non-holonomic mobile robot. In: Proceedings of the IEEE/RSJ International Workshop on Intelligent Robots and Systems IROS'91, vol. 3, pp. 1236–1241. Osaka, Japan (1991)
7. Brockett, R.W.: Asymptotic stability and feedback stabilization. In: Millman, R.S., Brockett, R.W., Sussmann, H.J. (eds.) Differential Geometric Control Theory. Birkauser, Boston (1983)
8. Samson, C.: Time-varying feedback stabilization of car-like wheeled mobile robots. Int. J. Rob. Res. **12**(1), 55–64 (1993)
9. Defoort, M., Floquet, T., Kökösy, A., Perruquetti, W.: Integral sliding mode control for trajectory tracking of a unicycle type mobile robot. Integr. Comput.-Aided Eng. **13**(3), 277–288 (2006)

10. Klanèar, G., Škrjanc, I.: Tracking-error model-based predictive control for mobile robots in real time. Robot. Auton. Syst. **55**(3), 460–469 (2007)
11. Maalouf, E., Saad, M., Saliah, H.: A higher level path tracking controller for a four-wheel differentially steered mobile robot. Robot. Auton. Syst. **54**, 23–33 (2005)
12. Tanaka, K., Ikeda, T., Wang, H.O.: Robust stabilization of a class of uncertain nonlinear system via fuzzy control: quadratic stabilizability, H∞ control theory and linear matrix. IEEE Trans. Fuzzy Syst. **4**, 1–13 (1996)
13. Wang, H.O., Tanaka, K., Griffin, M.F.: An approach to fuzzy control of nonlinear systems; stability and design. IEEE Trans. Fuzzy Syst. **4**(1), 14–23 (1996)
14. Boyd, S., El Ghaoui, L., Feron, E., Balakrishnan, V.: Linear Matrix Inequalities in System and Control Theory. Studies in Applied Mathematics, vol. 15. SIAM, Philadelphia (1994)
15. Jolly, K.G., Sreerama Kumar, R., Vijayakumar, R.: A Bezier curve based path planning in a multi-agent robot soccer system without violating the acceleration limits. Robot. Auton. Syst. **57**(1), 23–33 (2009)
16. Lepetič, M., Klančar, G., Škrjanc, I., Matko, D., Potočni, B.: Time optimal path planning considering acceleration limits. Robot. Auton. Syst. **45**(3–4), 199–210 (2003)
17. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its application to modeling and control. IEEE Trans. Syst. Man Cybern. **15**(1), 116–132 (1985)
18. Tuan, H.D., Apkarian, P., Narikiyo, T., Yamamoto, Y.: Parameterized linear matrix inequality techniques in fuzzy control system design. IEEE Trans. Fuzzy Syst. **9**, 324–332 (2001)
19. Tanaka, K., Wang, H.O.: Fuzzy Control Systems Design and Analysis: A Linear Matrix Inequality Approach. Wiley-Interscience, New York (2001)
20. Yoneyama, J., Noshikawa, M., Katayama, H., Ichikawa, A.: Design of output feedback controllers for Takagi–Sugeno fuzzy systems. Fuzzy Sets Syst. **121**, 127–148 (2001)
21. Chamroo, A., Seuret, A., Vasseur, C., Richard, J.-P., Wang, H.P.: Observing and controlling plants using their delayed and sampled outputs. In: IMACS Multiconference on Computational Engineering in Systems Applications, China, pp. 851–857 (2006)
22. Seuret, A., Michaut, F., Richard, J.-P., Divoux, T.: Networked control using GPS synchronization. In: American Control Conference, pp. 4195–4200 (2006)