Proceedings of the 1996 IEEE International Symposium
on Computer-Aided Control System Design
Dearborn, MI • September 15-18, 1996

Speed, steering angle and path tracking controls for a tricycle robot

André KAMGA & Ahmed RACHID

Laboratoire des Systèmes Automatiques
7 rue du Moulin Neuf - 80000 Amiens - France
Fax : (33) 22 82 76 82. e-mail : Andre.Kamga@sc.u-picardie.fr

Abstract

This paper is devoted to identification and control of the moving velocity of an autonomous vehicle and of the position and velocity of its steering angle. Trajectory tracking is also dealt with by adapting a simple but powerful linearization method.

Keywords : Autonomous vehicle, Mobile robot, Identification, Speed, Position, Tracking control.

1. Introduction

This paper consists of two parts. The first part deals with the velocity and position control of the steering motor as well as the speed control of the driving motor of a tricyle industrial electrical vehicle. This study is necessary to automate the robot. The features of our application are the following :

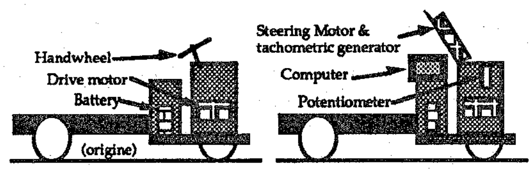
- the driving motor is a DC series one, which is a highly non linear actuator requiring a special control to ensure satisfactory dynamical behavior for low, medium or high speeds of the vehicle ;
- the steering motor is a standard permanent magnet DC motor which can be accuratly modeled by a first order transfer function. Usually, the control of its position and its velocity is done by closing twice a feedback loop and one cannot choose any reference input for the velocity. Here, we propose several control structures which allow the user to choose a reference input for the position and another reference input for the velocity. Therefore, the user can obtain the rotation of the wheel at a desired velocity from a given position angle to a desired one.

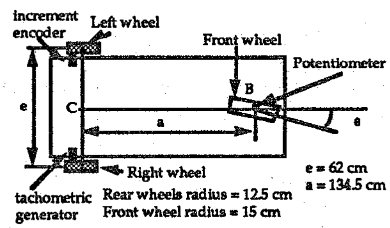
The second part of the paper is devoted to path tracking. We propose a linearization method inspired from that of [Sampei][5], illustrated by simulation and experimental results.

2. Vehicle Description

Our vehicle has its origine in an electrical marketed tricycle (TMST10 of [Charlatte's society][3]) with a driving and steering front wheel and two common axis rear free wheels. The front wheel is steered with a handwheel and moved thanks to a D.C. series motor of 24 V-750 W, controlled with an electronic inverter whose input voltage can vary from 0 to 5 volts. A battery of 24 V D.C., 150 Ah supplies the electrical energy required by the system.

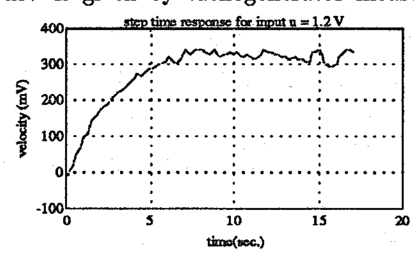
The aim of our work is to automate this manufactured vehicle in order to get an autonomous mobile robot. To this end, many equipments have been added as summarized in the following figures where we distinguish the original configuration:





3. Identification and control of the driving motor

The vehicle is driven by a DC series motor in order to ensure a great starting torque. This type of motor is known to be highly nonlinear. Here, in order to simplify the treatment, we have performed an identification procedure for the vehicle for different speed operating points. Therefore, we don't model the drive motor itself but the dynamics of the vehicle which is interesting because it is very difficult to get by mechanical modelisation. The models for the vehicle speed has been chosen as simple as possible consisting of a first order with delay. The following figure gives an example of forward moving speed response for a step input $u = 1.2$ V. The velocity in mV is given by tachogenerator measurements.



From these experiments, and taking into account that the dead zone of the speed response is 900 mV, we have deduced the following transfer functions for the vehicle speed :

$$H(s) = \frac{ke^{-\tau s}}{1+Ts}; \quad 1.05 \leq k \leq 1.23$$

$0.05 \leq \tau \leq 0.4$, and $2.01 \leq T \leq 5.63$.

Because of the variations of these parameters, the system is nonlinear. Nevertheless we have chosen a linear model in the form :

$$H(s) = \frac{1.08e^{-0.4s}}{1+1.9s}$$

To control this system, one has to design a regulator ensuring stability as well as satisfactory performances despite the variations of the system's model. To this end, and keeping in mind that our regulator must be low cost, we choose a P.I. regulator :

$$R(s) = k_p\left(1+\frac{1}{T_i s}\right) = k_p + \frac{k_i}{s}, \quad k_i = \frac{k_p}{T_i}$$

The closed-loop transfer function is:

$$F(s) = \frac{R(s)H(s)}{1+R(s)H(s)}$$

If we take $T_i = T$, then $F(s)$ simplifies to

$$F(s) = \frac{Z(s)}{1+Z(s)}, \quad Z(s) = \frac{kk_p}{Ts}e^{-\tau s}$$

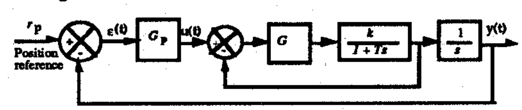
We require a gain margin of -6 db, so that the controller parameters are

$$k_i = \frac{0.8}{k\tau} \text{ and } k_p = \frac{0.8T}{k\tau}.$$

4. Position and velocity control of the steering angle

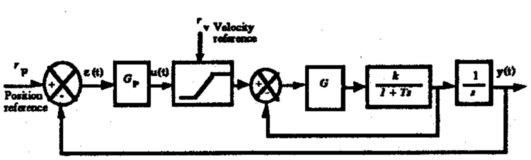
The steering angle is moved by a permanent magnet DC motor (24 V-250 W) which we assume to follow the first order type model $\frac{k}{1+Ts}$.

In general, a position control is performed using the following structure:



Unfortunately, this structure does not allow any reference input for the velocity. In what follows, we present simple controllers to overcome this situation.

A common way to deal with a reference for the velocity is to incorporate a saturation in the above control loop



Another natural way to deal with a velocity reference is to use some obvious kinematics. Let

T_p = time constant for the overall closed-loop

r_p = position reference

r_v = velocity reference

$y(t)$ = measured steering angle

then from $x = vt$, we have the approximation

$$r_p = 5 r_v T_p \qquad (i)$$

which simply says that the position r_p will be reached in $5T_p$ with the velocity r_v.

Given the input references r_p and r_v, equation (i) implies the required time constant T_p for the overall closed-loop. Next, we have to design a controller which ensures such a time constant. To this end, we choose a PD-controller and we determine its parameters using two methods.

Pole placement synthesis :

Let $G_p(s) = k_P + k_D s$, k_P and k_D are the controller parameters to be determined such that the closed-loop denominator is $(1+T_p s)^2$. This gives

$$\begin{cases} k_D = \frac{2T-T_p}{kGT_p} \\ k_P = \frac{T}{kGT_p^2} \end{cases}$$

Compensation synthesis :

Now we take a PD-controller of the form $G_p(s) = k_P(1+T_D s)$; T_D is determined to compensate the closed-loop pole $-1/T_p$. This gives

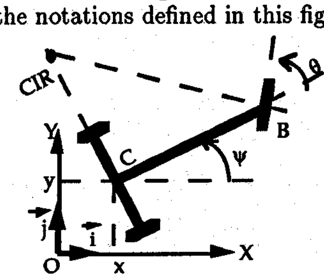
$$T_D = T; \quad k_P = 1/(T_p k G)$$

5. Path tracking control

In this section, we present a control strategy for path tracking and particularly paths which consist of straight lines or circles.

Straight line tracking controller

We use the notations defined in this figure



CIR denotes the intantaneous center of rotation of the vehicle. The following differential equations describe the kinematics of the tricycle robot:

$$\begin{cases} \dot{x} = V_C \cos\psi \\ \dot{y} = V_C \sin\psi \\ \dot{\psi} = \frac{V_C}{a}\tan\theta \end{cases} \qquad (1)$$

with $a = CB$. V_c is the velocity of point C, middle of the rear wheels axis and will be considered as the velocity of the vehicle. Our aim is to design a controller which makes the vehicle follow the X-axis, *i.e.*, $y \to 0, \psi \to 0$ and $\theta \to 0$ while it moves forward.

Following [Sampei], we consider a new scale in x, *i.e.* the distance along the desired path. Then, the differential equations (1) becomes

$$\frac{d}{dx}\begin{pmatrix} y \\ \psi \end{pmatrix} = \begin{pmatrix} \tan\psi \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{a\cos\psi} \end{pmatrix}\tan\theta \qquad (2)$$

while the behaviour of x is given by :

$$\frac{dx}{dt} = V_C\cos\psi \qquad (3)$$

The exact linearization of the state equation (2) is obtained by defining the following state variables

$$\xi^T = (\xi_1 \quad \xi_2) = (y \quad \tan\psi)$$

and $\tan\theta = va\cos^3\psi$ where v is the new input variable. In fact the derivative of the new state vector ξ with respect to the new scale x gives

$$\frac{d}{dx}\begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \begin{pmatrix} \xi_2 \\ \frac{1}{a\cos^3\psi}\tan\theta \end{pmatrix} \qquad (4)$$

Writing that $\frac{d\xi_2}{dx} = v$ we easily see that

$$\tan\theta = va\cos^3\psi \qquad (5)$$

Therefore the linearized system is

$$\frac{d\xi}{dx} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\xi + \begin{pmatrix} 0 \\ 1 \end{pmatrix} v \qquad (6)$$

for which we can design a state feedback controller

$$v = f\xi, \quad f = (f_1 \quad f_2) \qquad (7)$$

using pole placement, for instance, to ensure the stability of the system. If we choose the stable poles p_1 and p_2 then $f_1 = -p_1p_2$ and $f_2 = p_1 + p_2$.

In fact, the characteristic equation of the system (6) is

$$s^2 - f_2 s - f_1 = 0 \qquad (8)$$

where s is the Laplace variable. This also means the linearized closed loop system is a second order differential equation in y writing

$$y'' - f_2 y' - f_1 y = 0 \qquad (9)$$

with y' and y'' the first and the second derivatives of y with respect to the new time scale x. Its cross over pulsation ω_n and its decay coefficient z are expressed as :

$$\omega_n = \sqrt{-f_1} \quad and \quad z = \frac{1}{2}\frac{f_2\sqrt{-f_1}}{f_1} \qquad (10)$$

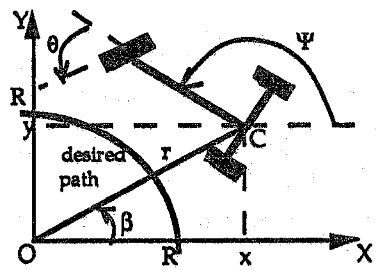
As ω_n has to be positive, f_1 must be negative. If p_1 and p_2 are the poles of the system (6) i.e. the zeros of equation (8) then $f_1 = -p_1p_2$ and $f_2 = p_1 + p_2$. The simulation results of this controller with respect to the time t, are easily obtained thanks to the functions "ODE23" or "ODE45" in MATLAB sofware as follow :

$$\frac{d\xi}{dt} = \frac{d\xi}{dx}\frac{dx}{dt} \quad with \quad \frac{dx}{dt} = V_C\cos\psi \quad and \quad \psi = tan^{-1}\xi_2$$

The steering speed $\dot{\theta}$ is obtained by deriving equation (5) with respect to the time. In the critical case (i.e. $z = 1$) for instance, the poles p_1 and p_2 are real and equal and the system is stable if they are negative. Giving a value to f_1 is then a manner to fix the above poles. The simulations, in critical case, of this controller for $f_1 = -0.25\ m^{-2}$, $f_2 = -1\ m^{-1}$ and a constant speed $V_c = 0.2$ m/s are shown in fig. 1 (see appendix). The path to follow in the global frame is expressed as $Y = (\tan 2\pi/3)X + 4$ *i.e* the orientation of the path with respect to the X-axis of the global frame is 120 degrees. The coordinates of the path's origin in the global frame are $X = 1$ m and $Y = 2.268$ m. Initial conditions are $\xi_1 = -10$ m, $\xi_2 = \tan\frac{\pi}{3}$, and $x = -10$ m.

Circle path tracking controller

The used notations are defined in this figure



Here, the position of the robot is described by the polar coordinates (β, r), the origin O being the center of the desired circle path whose radius is denoted by R. As in [Sampei], we make use of the scale $l = R\beta$ which is the distance along the desired path. With $\Gamma = r - R$ and $\gamma = \psi - \beta$ we obtain the following state equation:

$$\frac{d}{dl}\begin{pmatrix} \Gamma \\ \gamma \end{pmatrix} = \begin{pmatrix} \frac{r\cot\gamma}{R} \\ -\frac{1}{R} \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{r}{Ra\sin\gamma} \end{pmatrix}\tan\theta \qquad (11)$$

The behaviour of the time scale l is given by :

$$\frac{dl}{dt} = V_C\frac{R\sin(\psi - \beta)}{r} \qquad (12)$$

We consider the coordinate transformation

$$\xi^T = (\xi_1 \quad \xi_2) = \left(r - R \quad \frac{r\cot(\psi-\beta)}{R}\right) \quad \text{and}$$

$$\tan\theta = \frac{2a\sin\gamma\cos^2\gamma}{r} + \frac{a\sin^3\gamma}{r} - \frac{R^2a\sin^3\gamma}{r^2}v \qquad (13)$$

where v is the new input. The derivative of the new state ξ with respect to the time scale l gives:

$$\frac{d\xi}{dl} = \begin{pmatrix} \xi_2 \\ \frac{2ra\sin\gamma\cos^2\gamma + ra\sin^3\gamma - r^2\tan\theta}{R^2a\sin^3\gamma} \end{pmatrix} \qquad (14)$$

With $\frac{d\xi_2}{dl} = v$, we obtain (13). The linearized system is then

$$\frac{d\xi}{dl} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \xi + \begin{pmatrix} 0 \\ 1 \end{pmatrix} v \qquad (15)$$

for which we consider the feedback controller (7). Here the system (15) is also a second order differential equation in Γ in the form

$$\Gamma'' - f_2\Gamma' - f_1\Gamma = 0 \qquad (16)$$

with Γ' and Γ'' the first and the second derivatives of Γ with respect to the new time scale l. Equation (16) is easily identified to equation (9). By putting $\Gamma = y$ and $l = x$ the studies done for the choice of the poles of the closed loop system in the straight line controller paragraph, is valid here.

The steering speed $\dot{\theta}$ is obtained by deriving equation (13) with respect to the time t. So, for the critical case for instance, fig.2 (see appendix) shows simulation results for $f_1 = -0.25\ m^{-2}$, $f_2 = -1\ m^{-1}$ and a constant speed $V_C = 0.3$ m/s. The desired circle path has a radius $R = 8$ m and its center is at O, the origin of the global frame. Initial conditions are $\xi_1 = 2$ m, $\xi_2 = 2.165$, $l = 7$ m and $\psi = 7\frac{\pi}{18}$ rad.

6. Experimental results

We have implemented the above controllers on our industrial tricycle, considering only the critical cases. The obtained experimental results are good as shown in Fig. 3, Fig. 4 and Fig. 5 (see appendix). We stress on the fact that these results use only basic sensors: an incremental encoder for the displacement, a tacho-generator for the speed of the robot, a tacho-generator for the steering speed and a potentiometer for the steering angle. In Fig. 3, the vehicle has been programmed to follow a straight line which is the X axis of the global frame, moving forward with $f_1 = -1\ m^{-2}$, $f_2 = -2\ m^{-1}$ and $V_C = 0.2$ m/s. Initial conditions are $\xi_1 = 1$ m, $\xi_2 = \tan 0 = 0$, and $x = 0$. In this case the robot orientation is $\psi = \psi_1$ and its trajectory is $T1$. θ_{C1} and θ_{m1} are respectively the computed (command) and measured or effective steering angles. For $f_1 = -0.25\ m^{-2}$ and $f_2 = -1\ m^{-1}$, while keeping constant all the others parameters we obtain the trajectory $T2$ (see Fig. 3). The others variables have 2 as index. We can then appreciate the important rule of the constants f_1 and f_2 which define the poles of the system and then its rapidity. In fact the biggest their absolute values are, the fastest the vehicle reaches the desired path if it is not initially on it exactly with no error in orientation. This is also valid for circle paths. In Fig. 4, the robot has to follow a circle of radius $R = 3$ m, whose center is at the origin of the global frame. Here $f_1 = -0.25\ m^{-2}$, $f_2 = -1\ m^{-1}$, $V_C = 0.2$ m/s and initial conditions are $\xi_1 = -0.5$ m, $\beta = 0.2618$ rad and $\psi = 1.8326$ rad.

We have also programmed the robot to follow, in critical case ($z = 1$), a path composed with a straight line followed by a circle whose equations in the global frame (O, X, Y) are respectively $Y = \sqrt{3}X$ and $(X - 3.23)^2 + (Y - 1.6)^2 = 4$. The initial position of the vehicle in the global frame is $X = -4$ m, $Y = -2$ m and its orientation with respect to OX axis is $\Phi = 0$. The robot has to follow first the straight line, using the straight line controller, until it reaches the tangent point between the line and the circle, point from which it has to stop tracking the straight line and begin to follow the circle thanks to the circle path controller. The path has to be followed with $f_1 = -4\ m^{-2}$, $f_2 = -4\ m^{-1}$, a constant speed $V_C = 0.2$ m/s and the vehicle has to stop after one turn. Experimental results are shown in figure 5. Note that the initial value of the steering angle θ is negative (-0.4 rad.).We observe an abrupt variation of the steering and orientation angles due to the discontinuity of the curvature of the reference path at the tangent point where the controllers are permuted. One needs reference paths or trajectories without discontinuity in curvature when one passes from a straight line to an arc of circle or from an arc of circle to a straight line. This can be obtained by defining the path with Bezier curves, B-Splines, arcs of clothoids an so on. With so defined paths, the variation of the variables when changing the controllers will be smooth while tracking these paths.

7. Conclusion

Experimental results have been presented for the control of the actuators of an industrial nonlinear vehicle. The obtained dynamical performances are very satisfactory and have been performed using a simple P.I. controller. We have proposed a path tracking method which has been shown very performant by simulation and has been validated by experimental applications. Nevertheless, absolute localization is needed to diminish localization errors while tracking a path. To obtain the smooth variations of the controlled parameters, the reference path will have continue curvature.Because of the definition of the state variables and feedbak (5) the straight line control works only if ψ, θ and y belong respectively to $]-\frac{\pi}{2}, \frac{\pi}{2}[$, $]-\frac{\pi}{2}, \frac{\pi}{2}[$ and $]-\infty, \infty[$. The circle path control works as long as l increases.

8. References

[1] C. Samson, "Control of Chained Systems, Application to Path Following and Time- Varying Point-Stabilization of Mobile robots," *IEEE Trans. Automatic Control, Vol. 40, N 1, January 1995, pp. 64-77.*

[2] B. Steer, "Trajectory Planning for a mobile robot," *The International Journal of Robotics Research, Vol. 8, N 5, October 1989, pp. 3-14.*

[3] Charlatte's society, "Documentation Technique sur le tricycle multiservice TMS T10," *France, 1990.*

[4] W.L. Nelson, I.J. Cox "Local Path Control for an Autonomous Vehicle," *Proc. IEEE Int. Conf. on Rob. and Aut., Vol. 3, 1988, pp. 1504-1510.*

[5] M. Sampei, T. Tamura, T. Kobayashi, and N. Shibui, "Arbitrary Path Tracking Control of Articulated Vehicles Using Nonlinear Control Theory," *IEEE Trans. Control Systems Technology, Vol. 3, N 1, March 1995, pp. 125-131.*

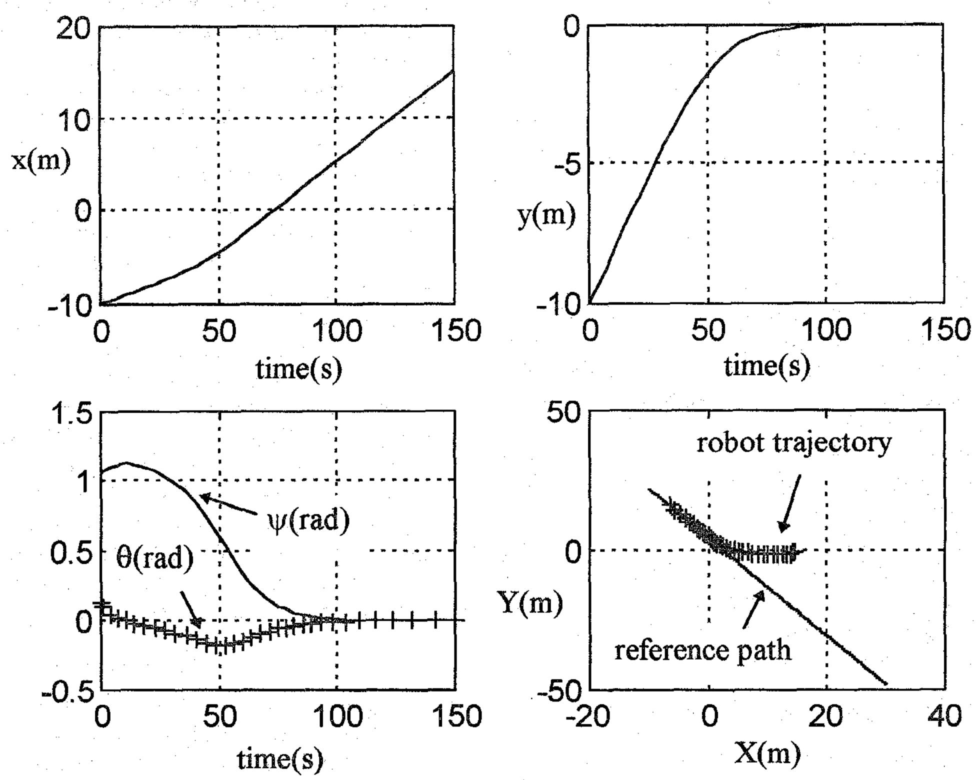


Fig.1 : Simulation results for the straight line tracking controller

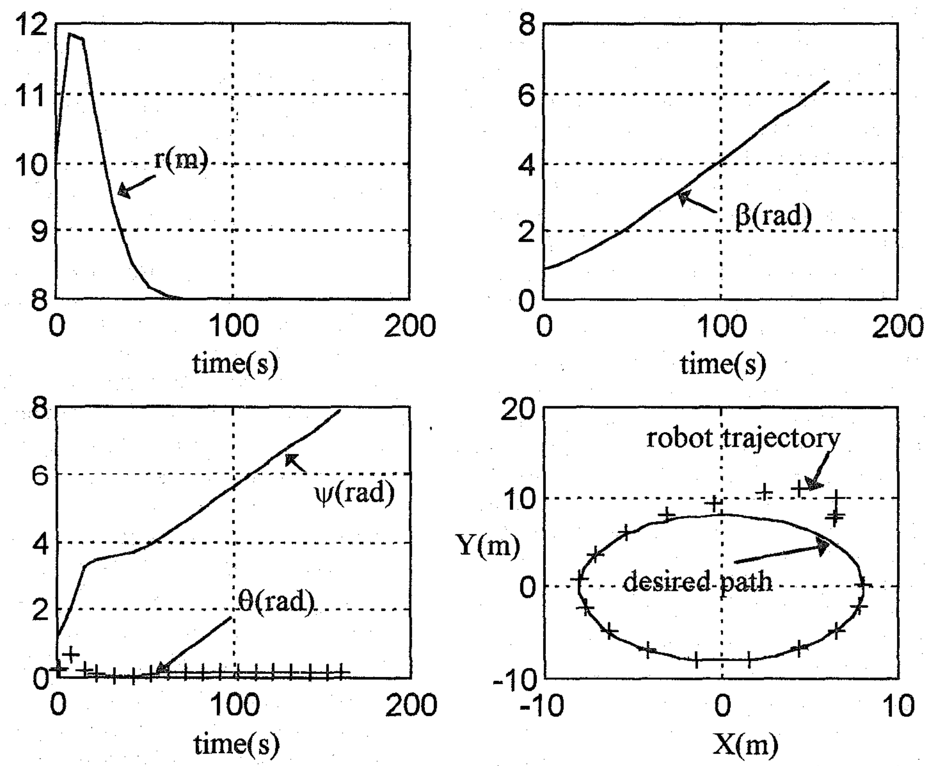


Fig. 2 : Simulation results for the circle path tracking controller

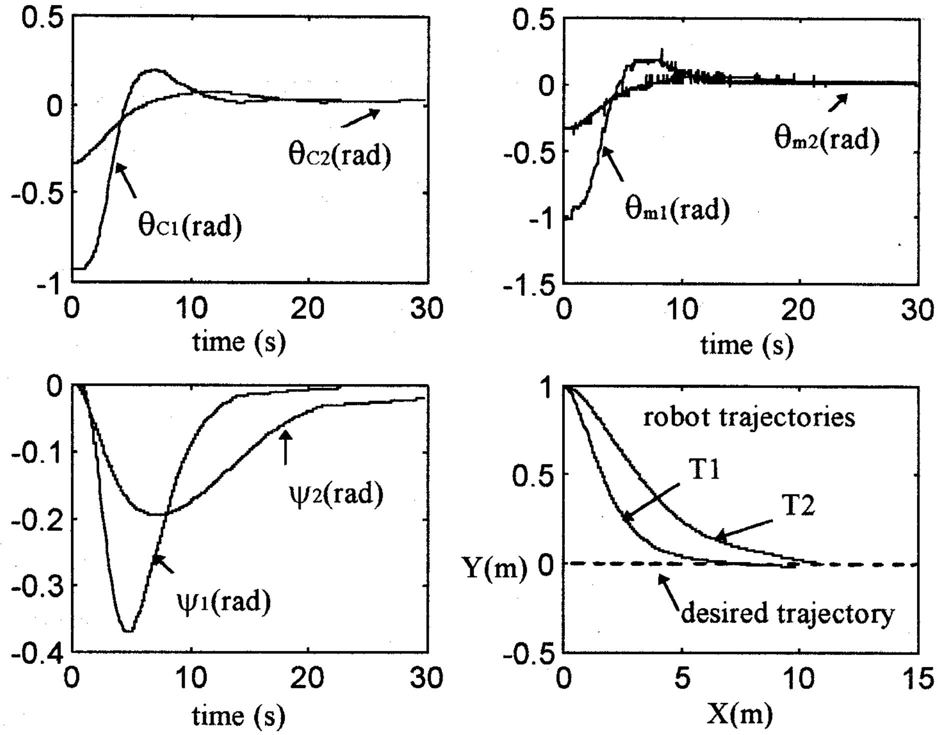


Fig. 3 :Experimental results for the straight line tracking controller

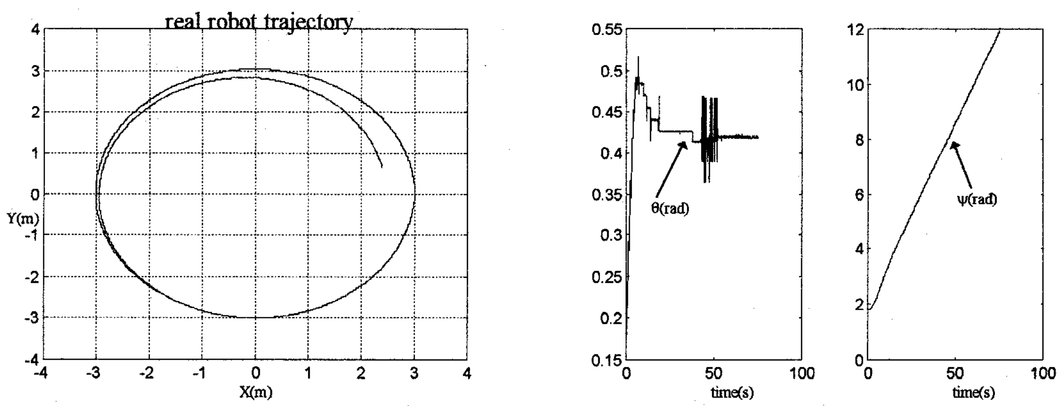


Fig. 4 : Experimental results for a left turn circle path tracking controller

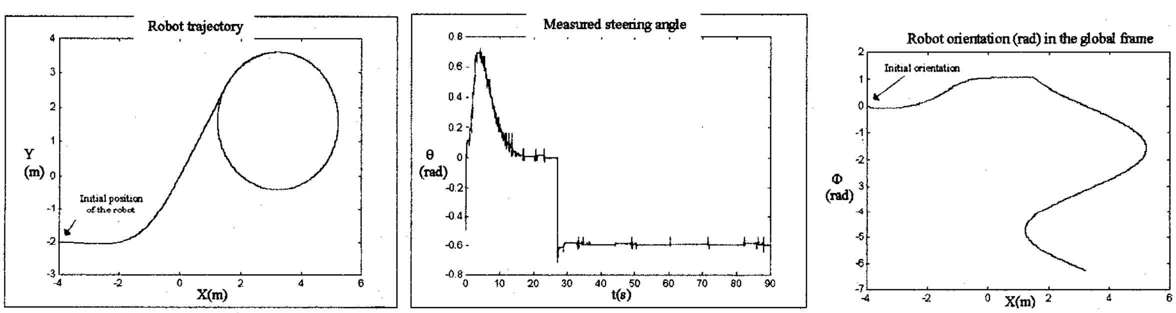


Fig. 5 : Experimental results for the following of a path composed of a straight line and a circle