ADAPTIVE TRACKING CONTROL OF A MOBILE ROBOT USING NEURAL NETWORKS

Emil PETRE

Department of Automatic Control, University of Craiova
A.I. Cuza Str. No. 13, RO-1100 Craiova, Romania, E-mail: epetre@automation.ucv.ro

Abstract: In this paper we present a design procedure for the motion control of a mobile robot subject to kinematic constraints. The dynamics of the mobile robot is assumed to be completely unknown, and is on-line identified using neural network based estimators. Both the form of the controller and the adaptation laws of neural network weights are derived from a Lyapunov analysis of stability. Under certain conditions, the tracking stability of the closed loop system, and the convergence of the neural network weight updating process are guaranteed. Computer simulations are included to demonstrate the performances of this neural network controller.

Key Words: Nonlinear systems, Mobile robot, Adaptive tracking control, Neural networks.

1. INTRODUCTION

In the recent years, the artificial neural networks (NNs), with their strong learning capability, have proven to be suitable tool for controlling complex nonlinear dynamic systems [1], [5], [6], [7]. The basic idea behind the neural network (NN) based control is to use a NN estimator to identify the unknown nonlinear dynamics and compensate for it. Also, the NN based approach can deal with the control of nonlinear systems that may not be linearly parameterizable, as required in the adaptive approach. With regard to robotic domain, NNs have been widely adopted in the modelling and control of robotic manipulators [4].

In this paper a design procedure for the motion control of a mobile robot subject to kinematic constraints is presented. The dynamics of the mobile robot is assumed to be completely unknown, and is on-line identified using NN based estimators. Both the form of the controller and the adaptation laws of NN weights are derived from a Lyapunov analysis of stability. Under certain conditions, the tracking stability of the closed loop system, and the convergence of the NN weight updating process are guaranteed. No preliminary learning stage of NN weights is required. Computer simulations conducted in the case of a mobile robot with two independently actuated wheels are included to demonstrate the performances of this NN controller by comparison to a classical feedback controller.

2. KINEMATICS AND DYNAMICS OF A MOBILE ROBOT

The dynamics of a mobile robot subject to kinematic constrains has the form [2], [4]:

$$M(q)\ddot{q} + V(q,\dot{q})\dot{q} + G(q,\dot{q}) + A^T(q)\lambda + \delta = B(q)\tau \tag{1}$$

where $q \in \Re^n$ is the vector of generalized coordinates, $\tau \in \Re^p$ is the torque input vector, $\lambda \in \Re^m$ is the vector of constraint forces, $M(q) \in \Re^{n\times n}$ is a symmetric and positive definite inertia matrix, $V(q,\dot{q}) \in \Re^{n\times n}$ is the centripetal and coriolis matrix, $G(q,\dot{q}) \in \Re^n$ is the friction and gravitational vector, $A(q) \in \Re^{m\times n}$ is the matrix associated with constrains, $\delta \in \Re^n$ denotes bounded unknown disturbances including unstructured dynamics, and $B(q) \in \Re^{n\times p}$ is the input transformation matrix.

The m kinematic constrains are described by

$$A(q)\dot{q} = 0 \tag{2}$$

Note that, in the following, the $p = n - m$ case is considered. With respect to the dynamics of a mobile robot (1), the following properties hold [4].

Property 1: $M(q)$ is a bounded symmetric and positive definite matrix.

Property 2: The matrix $\dot{M} - 2V$ is skew symmetric [2], [4], that is

$$\dot{M} - 2V = -\left(\dot{M} - 2V\right)^T \text{ with } \dot{M} = V + V^T \text{ or, } x^T\left(\dot{M} - 2V\right)x = 0, \quad \forall x \in \Re^n \tag{3}$$

Assume that the robot is fully actuated. Let $S(q) \in \Re^{n\times(n-m)}$ denote a full rank matrix formed by $(n-m)$ columns that span the null space of $A(q)$ defined in (2) i.e.,

$$S^T(q)A^T(q) = 0 \tag{4}$$

From (4), one can find an auxiliary vector $\omega(t) \in \Re^{n-m}$ so that for all t,

$$\dot{q} = S(q)\omega(t) \tag{5}$$

This is called the steering system where $\omega(t)$ can be regarded as an angular velocity input vector. Equations (1) and (5) describe the dynamics equations of mobile robot subject to kinematic constrains. Multiplying both sides of (1) by S^T and using (4) we obtain:

$$S^T M(q)\ddot{q} + S^T V(q,\dot{q})\dot{q} + S^T G(q,\dot{q}) + S^T\delta = S^T B(q)\tau \tag{6}$$

Substituting (5) and its time derivative into (6) this can be written in a compact form, as

$$\overline{M}(q)\dot{\omega} + \overline{V}(q,\dot{q})\omega + \overline{G}(q) + \overline{\delta} = \overline{B}(q)\tau \tag{7}$$

where $\overline{M} = S^T MS$, $\overline{V} = S^T(M\dot{S} + VS)$, $\overline{G} = S^T G$, $\overline{\delta} = S^T\delta$, and $\overline{B} = S^T B$.

Property 3: The matrix $\dot{\overline{M}} - 2\overline{V}$ in (7) is skew symmetric.

Proof: $\dot{\overline{M}} - 2\overline{V} = 2S^T M\dot{S} + S^T\dot{M}S - 2S^T(M\dot{S} + VS) = S^T(\dot{M} - 2V)S$. Since $\dot{M} - 2V$ is skew symmetric, therefore, $\dot{\overline{M}} - 2\overline{V}$ is also skew symmetric.

3. NEURAL NETWORK CONTROLLER DESIGN FOR A MOBILE ROBOT

3.1. *Problem statement*

In an application, a mobile robot is required to perform some tasks defined in its task-space. In order to achieve this objective firstly, some reference trajectories $q_d(t)$ are derived. Torque commands τ are then generated by the controller to make the mobile robot tracks the reference trajectories.

In this paper, it is assumed that the reference trajectories are available, i.e. they have already been derived based on desired task-trajectories. The main concern is to provide proper torque inputs that guarantee a stable tracking of reference trajectories in the presence of parameter uncertainty and unknown disturbances.

3.2. *Neural network controller design procedure*

In this section, a NN-based control procedure for a stable tracking of a reference trajectory for the mobile robot described by (5) and (7) is derived. The procedure steps are as follows: a) the robot dynamics is redefined as an error dynamics based on a set of appropriate chosen Lyapunov functions; b) a NN-based estimator is constructed and a NN learning law is proposed; c) a new control law is derived and d) a proof on the tracking stability of the overall closed-loop system and the boundedness on NN weight estimation errors is derived.

From previous section it can be seen that for a mobile robot a tracking error may be defined as

$$\tilde{q} = q_d - q \tag{8}$$

Assume that there exist a Lyapunov function $V_1(\tilde{q}, t)$, a positive continuous function $W_1(t) > 0$ and a reference smooth feedback velocity $\omega_d(t)$, such that [4]:

$$\left.\frac{\partial V_1}{\partial t} + \frac{\partial V_1}{\partial \tilde{q}} \dot{\tilde{q}}\right|_{\dot{q}=S(q)\omega_d} \leq -W_1(t) \text{ when } \tilde{q} \neq 0 \tag{9}$$

Now, the objective is to derive proper torque input τ in (7), such that the angular velocity trajectory $\omega(t)$ defined in (5) tracks the reference velocity $\omega_d(t)$.

Define the robot velocity tracking error $\tilde{\omega}$ as

$$\tilde{\omega} = \omega - \omega_d \tag{10}$$

Differentiating (10), multiplying both side by $\overline{M}$ and substituting (7) into it yields

$$\overline{M}(q)\dot{\tilde{\omega}} + \overline{V}(q,\dot{q})\tilde{\omega} + \overline{G}(q) + \overline{\delta} + \overline{M}(q)\dot{\omega}_d + \overline{V}(q,\dot{q})\omega_d = \overline{B}(q)\tau \tag{11}$$

Equation (11) represents the mobile robot dynamics in term of tracking errors.

Let us choose a Lyapunov function V_2 as

$$V_2 = \frac{1}{2}\tilde{\omega}^T \overline{M}\tilde{\omega} \tag{12}$$

Differentiating (12) along the system trajectories and using (11) and *Property 3* yields

$$\dot{V}_2 = \tilde{\omega}^T \overline{M}\dot{\tilde{\omega}} + \frac{1}{2}\tilde{\omega}^T \dot{\overline{M}}\tilde{\omega} = \tilde{\omega}^T (\overline{B}\tau - \overline{G} - \overline{\delta} - \overline{M}\dot{\omega}_d - \overline{V}\omega_d) \tag{13}$$

To design the robot torque input, we choose a Lyapunov function as

$$V_3 = V_1 + \frac{1}{2}(S\tilde{\omega})^T M(S\tilde{\omega}) = V_1 + V_2 \tag{14}$$

Differentiating (14) yields

$$\dot{V}_3 \leq -W_1(t) + \tilde{\omega}^T (\overline{B}\tau - \overline{G} - \overline{\delta} - \overline{M}\dot{\omega}_d - \overline{V}\omega_d) = -W_1(t) + \tilde{\omega}^T (\overline{B}\tau - \psi) - \tilde{\omega}^T \overline{\delta} \tag{15}$$

with the unknown nonlinear term

$$\psi = \overline{M}\dot{\omega}_d + \overline{V}\omega_d + \overline{G} \tag{16}$$

The nonlinear term ψ in (16) will be identified on-line by using a radial basis function (RBF) NN estimator. It is known that RBF networks have capacity to approximate any smooth function on a compact set $S_x \subset \Re^n$ [5], [6], [7]. If $f(\cdot): S_x \to \Re^n$ is a smooth function and $\{\varphi(x)\}$ is a RBFs basis set, then for each continuous function $f(\cdot)$, there exists a weight matrix W such that

$$f(x) = W^T \varphi(x) + \varepsilon \quad \text{with } \|\varepsilon\| < \varepsilon_N, \; \varepsilon_N > 0 \tag{17}$$

Then, the unknown function ψ in (16) may by identified using a RBF net with sufficiently high number n_n of nodes such that

$$\psi = W^T h(x) + \varepsilon \tag{18}$$

where x is the input pattern to the neural network defined as

$$x = [q_d^T \; \omega_d^T \; \dot{\omega}_d^T \; \tilde{\omega}^T]^T \tag{19}$$

$W \in \Re^{n_n \times (4n-3m)}$ in (18) is the ideal and unknown weight matrix, which is assumed to be constant and bounded by

$$\|W\|_F = \sqrt{tr(W^T W)} \leq W_B \tag{20}$$

with W_B a known positive constant and $\| W \|_F$ the Frobenius norm. The basis functions in vector $h(x)$ can be chosen as Gaussian functions defined as

$$h_i(x) = \exp\left(-\|x - c_i\|^2 / \sigma_i^2\right), \quad i = 1, 2, \ldots, n_n \tag{21}$$

where c_i are centers, and σ_i are widths, which are chosen apriori and kept fixed throughout for simplicity. Then, during the learning process, only the weight matrix W must to be adjusted. The estimates of ψ are given by

$$\hat{\psi} = \hat{W}^T h(x) \tag{22}$$

Thus, the main objective is to design a proper control law and properly NN learning laws, such that the unknown robot dynamics (16) can be compensated for by the NN estimator (22), and the stability of the robot error dynamics (11) and the boundedness on the estimation weights can be guaranteed. In this way we formulate the following theorem.

Theorem. If for the system (7) the control law is chosen as

$$\tau = \overline{B}^{-1}(-k\tilde{\omega} + \hat{\psi}) \tag{23}$$

with $\tilde{\omega}$ given by (10), and the weight updating law for the neural net as

$$\dot{\hat{W}} = -\beta(h\tilde{\omega}^T + \mu\|\omega\|\hat{W}) \tag{24}$$

where $k > 0$ is the control gain, $\beta > 0$ is the learning rate and $\mu > 0$ is a design parameter, then, by properly choosing of k and μ, the tracking errors of error dynamics described by (5) and (11) and the NN estimation weights $\hat{W}$ are all guaranteed to be uniformly ultimately bounded (UUB).

Proof. Assume that the approximation (18) holds, for all x in a compact set S_x. Substituting (23) into (15) yields

$$\dot{V}_3 \le -W_1(t) - k\tilde{\omega}^T\tilde{\omega} - \tilde{\omega}^T\tilde{W}^T h - \tilde{\omega}^T\varepsilon - \tilde{\omega}^T\overline{\delta} \tag{25}$$

where $\tilde{W} = W - \hat{W}$. Since $W_1(t) > 0$, from (25) one obtains

$$\dot{V}_3 \le -k\tilde{\omega}^T\tilde{\omega} - \tilde{\omega}^T\tilde{W}^T h - \tilde{\omega}^T(\varepsilon + \overline{\delta}) \tag{26}$$

If $k^* = \min k$ and it is using the defined boundedness of ε, from (26) it follows that

$$\dot{V}_3 \le -k^*\|\tilde{\omega}\|^2 - \tilde{\omega}^T\tilde{W}^T h + \|\tilde{\omega}\|(\varepsilon_N + \delta_N) \text{ where } \delta_N = \|\overline{\delta}\| \tag{27}$$

Le us chose a Lyapunov function as

$$V = V_3 + \frac{1}{2\beta} tr\{\tilde{W}^T\tilde{W}\} \tag{28}$$

Differentiating (28) and substituting (27) into it yields

$$\dot{V} = \dot{V}_3 + \frac{1}{\beta} tr\{\tilde{W}^T\dot{\tilde{W}}\} \le -k^*\|\tilde{\omega}\|^2 + \|\tilde{\omega}\|(\varepsilon_N + \delta_N) - \frac{1}{\beta} tr\left\{\tilde{W}^T(\dot{\hat{W}} + \beta h\tilde{\omega}^T)\right\} \tag{29}$$

Substituting now (24) into (29) we obtain

$$\dot{V} \le -k^*\|\tilde{\omega}\|^2 + \|\tilde{\omega}\|(\varepsilon_N + \delta_N) + \mu\|\tilde{\omega}\| tr\left\{\tilde{W}^T\hat{W}\right\} \tag{30}$$

Using [4], $tr\left\{\tilde{W}^T\hat{W}\right\} = tr\left\{\tilde{W}^T(W - \tilde{W})\right\} = \left\langle\tilde{W}, W\right\rangle_F - \|\tilde{W}\|_F^2 \le \|\tilde{W}\|_F\|W\|_F - \|\tilde{W}\|_F^2$ (31)

relation (30) can be written as

$$\begin{aligned}\dot{V} &\le -k^*\|\tilde{\omega}\|^2 + \mu\|\tilde{\omega}\|\,\|\tilde{W}\|_F\|W\|_F - \mu\|\tilde{\omega}\|\,\|\tilde{W}\|_F^2 + \|\tilde{\omega}\|(\varepsilon_N + \delta_N) \\ &= -\|\tilde{\omega}\|\left\{k^*\|\tilde{\omega}\| + \mu\left(\|\tilde{W}\|_F - \frac{W_B}{2}\right)^2 - \left(\frac{\mu W_B^2}{4} + \varepsilon_N + \delta_N\right)\right\}\end{aligned} \tag{32}$$

It can be seen that if the parameters μ and k^* are chosen so that

$$k^* > \left(\mu W_B^2 + 4(\varepsilon_N + \delta_N)\right)/4\|\tilde{\omega}\| \tag{33}$$

then $\dot{V} \le 0$. According to Lyapunov theory and to LaSalle principle, this demonstrates the UUB [3] of the tracking error $\tilde{\omega}$ and the NN weight errors $\tilde{W}$ and, subsequently, the weight estimates $\hat{W}$. Therefore, the control torque (23) is also bounded.

4. SIMULATION RESULTS

In this section, an adaptive NN-based tracking controller is designed for the kinematic and the dynamic model corresponding to a mobile robot with two actuated wheels, shown in Fig. 1. The performance of this controller is compared to the performance of a feedback controller designed for the kinematic model like in [2]. The configuration of the mobile robot can be described by five generalized coordinates:

$$q = [x\ y\ \phi\ \theta_r\ \theta_r]^T \tag{34}$$

where (x, y) are the coordinates of the origin P_0, ϕ is the heading angle of the mobile robot, and θ_r and θ_l are the angles of the rights and the left driving wheels. Let denote by m_c and m_w the mass of the robot body and a motor-wheel respectively, and I_c, I_w and I_m the moment of inertia of the body about the vertical axis through P_c (mass center of mobile robot), the motor-wheel about the wheel axis, and the motor-wheel about the wheel diameter, respectively. The kinematic model has the form (5) where

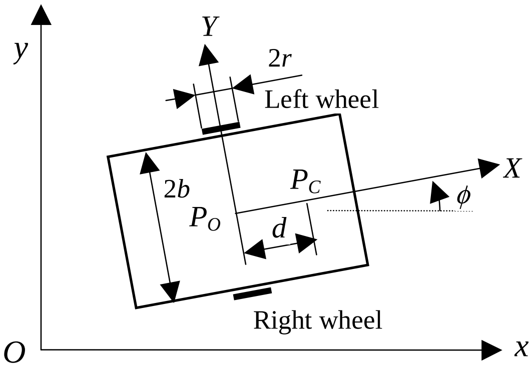

Fig.1. A mobile robot with two actuated wheels

$$S^T(q) = \begin{bmatrix} \frac{r}{2}\cos\phi & \frac{r}{2}\sin\phi & \frac{r}{2b} & 1 & 0 \\ \frac{r}{2}\cos\phi & \frac{r}{2}\sin\phi & \frac{r}{2b} & 0 & 1 \end{bmatrix}, \ \omega = \begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix} \tag{35}$$

If we denote by v_{mr} and ω_{mr} the linear and angular velocities of the mobile robot at the point P_0, the simplest kinematic form of this mobile robot is:

$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ \phi \end{bmatrix} = \begin{bmatrix} \cos\phi & 0 \\ \sin\phi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_{mr} \\ \omega_{mr} \end{bmatrix} \tag{36}$$

The dynamic model has the form (7) where $\overline{\tau}_d = 0$ and $\overline{M}, \overline{V}, \overline{B}$ are expressed as [4]:

$$\overline{M} = \begin{bmatrix} \frac{r^2}{4b^2}(mb^2 + I) + I_w & \frac{r^2}{4b^2}(mb^2 - I) \\ \frac{r^2}{4b^2}(mb^2 - I) & \frac{r^2}{4b^2}(mb^2 + I) + I_w \end{bmatrix}, \ \overline{V} = \begin{bmatrix} 0 & \frac{r^2}{2b} m_c d\dot{\phi} \\ -\frac{r^2}{2b} m_c d\dot{\phi} & 0 \end{bmatrix}, \ \overline{B} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

with $m = m_c + 2m_w$, $I = m_c d^2 + 2m_w b^2 + I_c + 2I_m$. (37)

Let the reference trajectory of the robot be prescribed as

$$\frac{d}{dt}\begin{bmatrix} x_{ref} \\ y_{ref} \\ \phi_{ref} \end{bmatrix} = \begin{bmatrix} \cos\phi_{ref} & 0 \\ \sin\phi_{ref} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_{ref} \\ \omega_{ref} \end{bmatrix} \tag{38}$$

where x_{ref}, y_{ref} and ϕ_{ref} are the configure of the reference robot, and v_{ref} and ω_{ref} are its reference inputs. The tracking errors denoted by e_1, e_2, e_3 are defined as [2]

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{ref} - x \\ y_{ref} - y \\ \phi_{ref} - \phi \end{bmatrix} \tag{39}$$

The input feedback control denoted by v_c and ω_c which make e_1, e_2, e_3 converge asymptotically to zero, are given by [2]:

$$\begin{bmatrix} v_c \\ \omega_c \end{bmatrix} = \begin{bmatrix} v_{ref}\cos e_3 + k_1 e_1 \\ \omega_{ref} + v_{ref} k_2 e_2 + k_3 \sin e_3 \end{bmatrix} \tag{40}$$

where positive constants k_1, k_2, k_3 are control gains.

The values of physical and design parameter are [2]: a = 2, b = 0.75, d = 0.3, r = 0.15, m_c = 30, m_w =1, I_c =15.625, I_w =0.005, I_m = 0.0025, k_1 = k_2 = k_3 = 5. The reference inputs are chosen as follow [2]: $t \in [0, 5)$: $v_{ref} = 0.25(1-\cos(\pi t/5))$, $w_{ref} = 0$; $t \in [5, 20)$: $v_{ref} = 0.5$, $w_{ref} = 0$; $t \in [20, 25)$: $v_{ref} = 0.25(1+\cos(\pi t/5))$, $w_{ref} = 0$; $t \in [25, 30)$: $v_{ref} = 0.15\pi(1-\cos(2\pi t/5))$, $w_{ref} = -v_{ref}/1.5$; $t \in [30, 35)$: $v_{ref} = 0.15\pi(1-\cos(2\pi t/5))$, $w_{ref} = v_{ref}/1.5$; $t \in [35, 40)$: $v_{ref} = 0.25(1+\cos(\pi t/5))$, $w_{ref} = 0$; $t \in [40, 50)$: $v_{ref} = 0.5$, $w_{ref} = 0$.

The simulation results using a RBF-NN controller designed according presented procedure are shown in Figs. 2-5. The network weights were initialized with zero, and the widths of Gaussian functions were chosen 0.025. The values of design parameters used in simulations are: β = 0.025, μ = 0.005, k = 0.02.

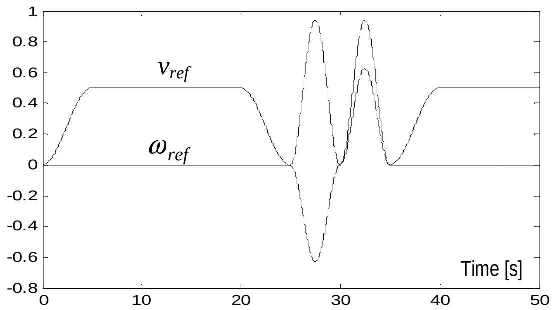

Fig.2. Reference inputs v_{ref}, ω_{ref}

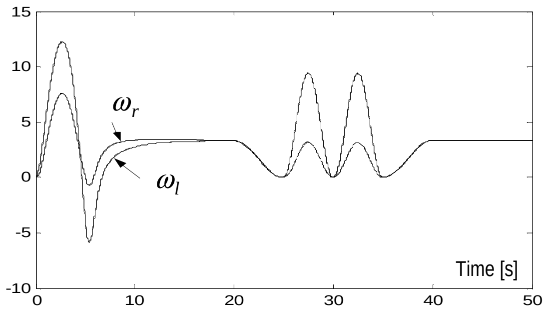

Fig. 3. Angular velocities ω_r, ω_l

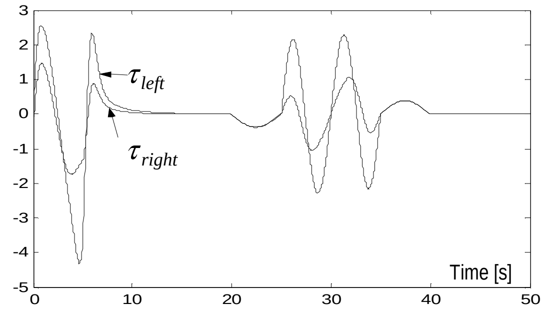

Fig. 4. Torque comands τ_r, τ_l

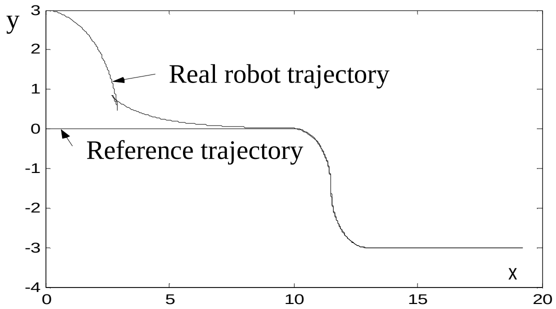

Fig. 5. Mobile robot trajectory

5. CONCLUSIONS

In this paper a design procedure for the motion adaptive control of a mobile robot subject to kinematic constraints was presented. The unknown dynamics of the mobile robot was on-line identified using NN-based estimators. The form of the controller and the adaptation laws of NN weights were derived from a Lyapunov analysis of stability. The simulation results demonstrate a good behaviour of this adaptive NN controller.

REFERENCES

[1]. F.C. Chen, C.C. Liu (1994), Adaptively controlling nonlinear continuous-time systems using multilayer neural networks, *IEEE Trans.Automat.Contr.*, vol.39, no.6, pp.1306-1310.

[2]. T. Fukao, H. Nakagawa, N. Adachi (2000), Adaptive tracking control of a nonholonomic mobile robot, *IEEE Trans. Robot. Automat.*, vol. 16, no. 5, pp. 609-615.

[3]. P. Ioannou, J. Sun (1996), *Robust adaptive Control*, Prentice Hall PTR, 1996.

[4]. S. Lin, A.A.Goldenberg (2001), Neural network control of mobile manipulators, *IEEE Trans. Neural Networks*, vol. 12, no. 5, pp. 1121-1133.

[5]. M.A. Mayosky, G.I.E. Cancelo (1999), Direct adaptive control of wind energy conversion systems using Gaussian networks, *IEEE Trans. Neural Networks*, vol.10, no.4, pp. 898-906.

[6]. R.B. McLain, M.A. Henson, M. Pottmann (1999), Direct adaptive control of partially known nonlinear systems, *IEEE Trans. Neural Networks*, vol. 10, no. 3, pp. 714-721.

[7]. J.T. Spooner, K.M. Passino (1999), Decentralized adaptive control of nonlinear systems using radial basis neural networks, *IEEE Trans.Aut.Contr.*, vol.44, no.11, pp.2050-2057.