PRACTICAL POINT STABILIZATION OF A NONHOLONOMIC MOBILE ROBOT USING NEURAL NETWORKS

R. Fierro and F. L. Lewis

Automation and Robotics Research Institute. The University of Texas at Arlington
7300 Jack Newell Blvd. South. Fort Worth, Texas 76118-7115
rfierro@arri.uta.edu, flewis@arri.uta.edu

Abstract

A control structure that makes possible the integration of a kinematic controller and a neural network (NN) computed-torque controller for nonholonomic mobile robots is presented. This control algorithm is applied to the practical point stabilization problem *i.e.*, stabilization to a small neighborhood of the origin. The NN controller proposed in this work can deal with unmodelled bounded disturbances and/or unstructured unmodelled dynamics in the vehicle. On-line NN weight tuning algorithms that do not require off-line learning yet guarantee small tracking errors and bounded control signals are utilized.

1. Introduction

Much has been written about solving the problem of motion under nonholonomic constraints using the kinematic model of a mobile robot, little about the problem of integration of the nonholonomic kinematic controller and the dynamics of the mobile robot. Moreover, the literature on robustness and control in presence of uncertainties in the dynamical model of such systems is sparse [9]. Some preliminary results on nonholonomic system with uncertainties are given in [3], [8].

Another intensive area of research has been neural networks applications in closed-loop control. In contrast to classification applications, in feedback control the NN becomes part of the closed-loop system. Therefore, it is desirable to have a NN control with on-line learning algorithms that do no require preliminary off-line tuning [11]. Several groups by now are doing rigorous analysis of NN controllers using a variety of techniques [4], [11], [12], [13]. In this paper, we present a robust-adaptive kinematic/neuro-controller based on the universal approximation property of NN [7]. The NN learns the full dynamics of the mobile robot *on-line*, and the kinematic controller stabilizes the state of the system in a small neighborhood of the origin.

In the literature, the nonholonomic navigation problem is simplified by neglecting the vehicle dynamics and considering only the steering system. The approach proposed in this paper corrects this omission by means of a NN controller. It provides a rigorous method of taking into account the specific vehicle dynamics to convert a steering system command into control inputs for the actual vehicle.

This paper is organized as follows. In Section 2, we present some basics of nonholonomic systems and NN. Section 3 discusses the nonlinear kinematic-NN controller as applied to the point stabilization problem. Stability is proved by Lyapunov theory. Section 4 presents some simulation results. Finally, Section 5 gives some concluding remarks.

2. Preliminaries

A Nonholonomic System A generalized mechanical system having an n-dimensional configuration space $\mathcal{C}$ with generalized coordinates $(q_1,...,q_n)$ and subject to m nonholonomic constraints can be described in the d'Alembert-Lagrange form [10],

$$\mathbf{M}(q)\ddot{q} + \mathbf{V_m}(q,\dot{q})\dot{q} + \mathbf{F}(\dot{q}) + \mathbf{G}(q) + \tau_d = \mathbf{B}(q)\tau - \mathbf{A}^T(q)\lambda, \quad (1)$$

$$\mathbf{A}(q)\dot{q} = 0, \quad (2)$$

where $\mathbf{M}(q) \in \mathfrak{R}^{n \times n}$ is a symmetric, positive definite inertia matrix, $\mathbf{V_m}(q,\dot{q}) \in \mathfrak{R}^{n \times n}$ is the centripetal and coriolis matrix, $\mathbf{F}(\dot{q}) \in \mathfrak{R}^{n \times 1}$ denotes the surface friction, $\mathbf{G}(q) \in \mathfrak{R}^{n \times 1}$ is the gravitational vector, τ_d denotes bounded unknown disturbances including unstructured unmodelled dynamics, $\mathbf{B}(q) \in \mathfrak{R}^{n \times r}$ is the input transformation matrix, $\tau \in \mathfrak{R}^{r \times 1}$ is the input vector, $\mathbf{A}(q) \in \mathfrak{R}^{m \times n}$ is the matrix associated with the constraints, and $\lambda \in \mathfrak{R}^{m \times 1}$ is the vector of constraint forces.

Let $\mathbf{S}(q)$ be a full rank matrix $(n-m)$ formed by a set of smooth and linearly independent vector fields spanning the null space of $\mathbf{A}(q)$, *i.e.*,

$$\mathbf{S}^T(q)\mathbf{A}^T(q) = 0. \quad (3)$$

According to (2) and (3), it is possible to find an auxiliary vector time function $v(t) \in \mathfrak{R}^{n-m}$ such that, for all t

$$\dot{q} = \mathbf{S}(q)v(t). \quad (4)$$

Equations (1)-(4) are well-known results in nonholonomic system theory, see for instance [1], [9], [14]. A car-like mobile robot is a typical example of a nonholonomic mechanical system [6].

The system (1) is now transformed into a more appropriate representation for controls purposes [15].

$$\dot{q} = \mathbf{S}v, \quad (5)$$

$$S^T M S \dot{v} + S^T (M \dot{S} + V_m S) v + \bar{F} + \bar{\tau}_d = S^T B \tau, \qquad (6)$$

where $v(t) \in \Re^{n-m}$ is a velocity vector. By appropriate definitions we can rewrite equation (6) as follows

$$\bar{M}(q) \dot{v} + \bar{V}_m (q, \dot{q}) v + \bar{F}(v) + \bar{\tau}_d = \tau, \qquad (7)$$

where $\bar{M}(q) \in \Re^{2 \times 2}$ is a symmetric, positive definite inertia matrix, $\bar{V}_m(q, \dot{q}) \in \Re^{2 \times 2}$ is the centripetal and coriolis matrix, $\bar{F}(v) \in \Re^{2 \times 1}$ is the surface friction, $\bar{\tau}_d$ denotes bounded unknown disturbances including unstructured unmodelled dynamics, and $\bar{\tau} \in \Re^{2 \times 1}$ is the input vector.

Feedforward Neural Networks [11] A 'three-layer' feedforward NN has two layers of adjustable weights. The neural network output y is a vector with m components that are determined in terms of the n components of the input vector x by the formula

$$y_i = \sum_{j=1}^{N_h} \left[w_{ij} \sigma \left(\sum_{k=1}^{n} v_{jk} x_k + \theta_{vj} \right) + \theta_{wi} \right]; \quad i = 1, \dots, m \qquad (8)$$

where $\sigma(\cdot)$ are the activation functions and N_h is the number of hidden-layer neurons. The first-to-second-layer interconnection weights are denoted by v_{jk} and the second-to-third-layer interconnection weights by w_{ij}. The threshold offsets are denoted by θ_{vj}, θ_{wi}.

By collecting all the NN weights v_{jk}, w_{ij} into matrices of weights V^T, W^T, one can write the NN equation is terms of vectors as

$$y = W^T \sigma (V^T x), \qquad (9)$$

The thresholds are included as the first columns of the weight matrices. Any tuning of W and V then includes tuning of the thresholds as well.

The main property of a NN we shall be concerned with for controls purposes is the *function approximation property* [5], [7]. Let $f(x)$ be a smooth function from $\Re^n$ to $\Re^m$. Then, it can be shown that, as long as x is restricted to a compact set U_x of $\Re^n$, for some number of hidden layer neurons N_h, there exist weights and thresholds such that one has

$$f(x) = W^T \sigma (V^T x) + \varepsilon. \qquad (10)$$

This equation means that a NN can approximate any function in a compact set. The value of ε is called the *NN functional approximation error*. In fact, for any choice of a positive number ε_N, one can find a NN such that $\varepsilon < \varepsilon_N$ in U_x.

For controls purposes, all one needs to know is that, for a specified value of ε_N these *ideal* approximating NN weights exist. Then, an estimate of $f(x)$ can be given by

$$\hat{f}(x) = \hat{W}^T \sigma (\hat{V}^T x), \qquad (11)$$

where $\hat{W}$, $\hat{V}$ are estimates of the ideal NN weights that are provided by some on-line weight tuning algorithms.

3. Control Design

Many approaches exist to selecting a velocity control $v(t)$ for the steering system (4). In this section, we desire to convert such a prescribed control $v(t)$ into a torque control $\tau(t)$ for the actual physical cart. Therefore, our objective is to design a NN control algorithm so that (5), (6) exhibits the desired behavior motivating the specific choice of the velocity $v(t)$.

The smooth steering system control, denoted by v_c, can be found by any technique in the literature. Using the algorithm to be derived and proved in Section 3.3, the practical point stabilization problem is solved as follows:
Point Stabilization as an Extension of the Tracking Problem The trajectory tracking problem for nonholonomic vehicles is posed as follows:
Let there be prescribed a reference cart

$$\dot{x}_r = v_r \cos \theta_r, \quad \dot{y}_r = v_r \sin \theta_r, \quad \dot{\theta}_r = w_r,$$
$$q_r = [x_r \ y_r \ \theta_r]^T, \quad v_r = [v_r \ w_r]^T. \qquad (12)$$

As in [5] it is assumed that the reference cart moves along the x-axis, *i.e.*,

$$\dot{x}_r = v_r, \quad q_r = [x_r \ 0 \ 0]^T, \quad v_r = [v_r \ 0]^T. \qquad (13)$$

Therefore, the point stabilization problem consists of finding a smooth time-varying velocity control input $v_c(t)$ such that $\lim_{t \to \infty}(q_r - q) = 0$ and $\lim_{t \to \infty} x_r = 0$. Then compute the torque input $\tau(t)$ for (7), such that $v \to v_c$ as $t \to \infty$.

$\square$

3.1 NN Control Design for Tracking a Trajectory

The structure for the point stabilization system to be derived in Section 3.3 is presented in Fig. 1. In this figure, no knowledge of the dynamics of the cart is assumed. The function of the NN is to reconstruct the dynamics (7) by learning it on-line.
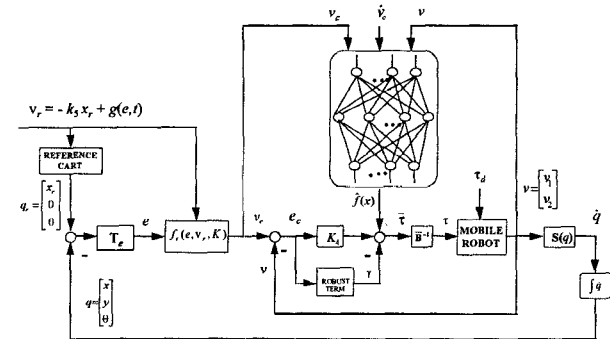

Fig. 1. Practical point stabilization using NN.

The contribution of this paper lies in deriving a suitable $\tau(t)$ from a specific $v_c(t)$ that controls the steering system (5). In the literature, the nonholonomic point stabilization problem is simplified by neglecting the vehicle dynamics (6) and considering only the steering

system (5). To compute the vehicle torque $\tau(t)$, it is assumed that there is 'perfect velocity tracking' so that $v = v_c$, then (6) is used to compute $\tau(t)$. There are three problems with this approach: first, the perfect velocity tracking assumption does not hold in practice, second, the disturbance τ_d is ignored, and, finally, complete knowledge of the dynamics is needed. A better alternative to this unrealistic approach is the adaptive NN controller now developed.

To be specific, it is assumed that the solution to the steering system point stabilization problem in [2] is available. This is denoted as $v_c(t)$. Then, a control $\tau(t)$ for (5), (6) is found that guarantees robust practical point stabilization despite unknown dynamical parameters and bounded unknown disturbances $\overline{\tau}_d(t)$. The position error is expressed in the basis of a frame linked to the mobile platform as

$$e = \mathbf{T}_e(q_r - q),$$

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ \sin\theta & -\cos\theta & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x_r - x \\ y \\ \theta \end{bmatrix}. \qquad (14)$$

and the derivative of the error is

$$\dot{e} = \begin{bmatrix} v_2 e_2 - v_1 + v_r \cos e_3 \\ -v_2 e_1 + v_r \sin e_3 \\ -v_2 \end{bmatrix}. \qquad (15)$$

An auxiliary velocity control input that achieves point stabilization for (5) is given by [2]

$$v_c = f_c(e, \mathbf{v}_r, K) = \begin{bmatrix} v_r \cos e_3 + k_1 e_1 \\ k_2 v_r \dfrac{\sin e_3}{e_3} e_2 + k_3 e_3 \end{bmatrix}. \qquad (16)$$

The derivative of v_c becomes

$$\dot{v}_c = \begin{bmatrix} \dot{v}_r \cos e_3 \\ k_2 \dot{v}_r \dfrac{\sin e_3}{e_3} e_2 \end{bmatrix}$$

$$+ \begin{bmatrix} k_1 & 0 & -v_r \sin e_3 \\ 0 & k_2 v_r \dfrac{\sin e_3}{e_3} & k_2 v_r \dfrac{e_3 \cos e_3 - \sin e_3}{e_3^2} e_2 + k_3 \end{bmatrix} \dot{e}, \qquad (17)$$

where

$$\mathbf{v}_r = -k_5 x_r + g(e, t), \qquad (18)$$

and

$$g(e, t) = \|e\|^2 \sin t, \qquad (19)$$

where k_1, k_2, k_3, $k_5 > 0$ are design parameters. They can be chosen to satisfy certain performance criteria. If the mobile robot is able to follow the desired velocity (16), then the position error e converges asymptotically to a neighborhood of the origin [6]. Different time-varying functions $g(e,t)$ are available in the literature, see [9] and the references therein. We have chosen the approach presented in [2].

Given the desired velocity $v_c(t) \in \Re^2$, define now the auxiliary velocity tracking error as

$$e_c = v_c - v. \qquad (20)$$

Differentiating (20) and using (7), the mobile robot dynamics may be written in terms of the velocity tracking error as

$$\overline{\mathbf{M}}(q)\dot{e}_c = -\overline{\mathbf{V}}_m(q,\dot{q})e_c - \overline{\tau} + f(x) + \overline{\tau}_d, \qquad (21)$$

where the important *nonlinear mobile robot function* is

$$f(x) = \overline{\mathbf{M}}(q)\dot{v}_c + \overline{\mathbf{V}}_m(q,\dot{q})v_c + \overline{\mathbf{F}}(v). \qquad (22)$$

The vector x required to compute $f(x)$ can be defined as

$$x \equiv [v^T \quad v_c^T \quad \dot{v}_c^T]^T, \qquad (23)$$

which can be measured.

Function $f(x)$ contains all the mobile robot parameters such as masses, moments of inertia, friction coefficients, and so on. These quantities are often imperfectly known and difficult to determine.

3.2 Mobile Robot Controller Structure

In applications the nonlinear robot function $f(x)$ is at least partially unknown. Therefore, a suitable control input for velocity following is given by the computed-torque like control

$$\overline{\tau} = \hat{f} + K_4 e_c - \gamma, \qquad (24)$$

with K_4 a diagonal, positive definite gain matrix, and $\hat{f}(x)$ an *estimate* of the robot function $f(x)$ that is provided by the neural network. The robustifying signal $\gamma(t)$ is required to compensate the unmodelled unstructured disturbances. Using this control in (21), the closed-loop system becomes

$$\overline{\mathbf{M}}\dot{e}_c = -(K_4 + \overline{\mathbf{V}}_m)e_c + \tilde{f} + \overline{\tau}_d + \gamma, \qquad (25)$$

where the velocity tracking error is driven by the *functional estimation error*

$$\tilde{f} = f - \hat{f}. \qquad (26)$$

In computing the control signal, the estimate $\hat{f}$ can be provided by several techniques, including *adaptive control*. The robustifying signal $\gamma(t)$ can be selected by several techniques, including sliding-mode methods and others under the general aegis of *robust control* methods.

3.3 Neural Net Controller

By using the controller (24), there is no guarantee that the control $\overline{\tau}$ will make the velocity tracking error small. Thus, the control design problem is to specify a method of selecting the matrix gain K_4, the estimate $\hat{f}$, and the robustifying signal $\gamma(t)$ so that both the error $e_c(t)$ and the control signals are bounded. It is important to note that the latter conclusion hinges on showing that the estimate $\hat{f}$ is bounded. Moreover, for good performance, the bound on $e_c(t)$ should be in some sense 'small enough' because it will affect directly the position error $e(t)$. In this section we shall use a NN to compute

the estimate $\hat{f}$. A major advantage is that this can always be accomplished, due to the NN approximation property (10).

Some definitions are required in order to proceed:

d.1) We say that the solution of a nonlinear system with state $x(t) \in \Re^n$ is uniformly ultimately bounded (UUB) if there exists a compact set $U_x \subset \Re^n$ such that for all $x(t_0) = x_0 \in U_x$, there exists a $\delta > 0$ and a number $T(\delta, x_0)$ such that $\|x(t)\| < \delta$ for all $t \geq t_0 + T$.

d.2) Given $\mathbf{A} = [a_{ij}]$, $\mathbf{B} \in \Re^{m \times n}$ the Frobenius norm is defined by

$$\|\mathbf{A}\|_F^2 = tr\{\mathbf{A}^T \mathbf{A}\} = \sum_{i,j} a_{ij}^2, \qquad (27)$$

with $tr\{\cdot\}$ the trace. The associated inner product is $\langle \mathbf{A}, \mathbf{B} \rangle_F = tr\{\mathbf{A}^T \mathbf{B}\}$.

d.3) For notational convenience we define the matrix of all the NN weights as $\hat{\mathbf{Z}} \equiv diag\{\hat{\mathbf{W}}, \hat{\mathbf{V}}\}$, and the weight estimation errors as $\tilde{\mathbf{V}} = \mathbf{V} - \hat{\mathbf{V}}$, $\tilde{\mathbf{W}} = \mathbf{W} - \hat{\mathbf{W}}$, $\tilde{\mathbf{Z}} = \mathbf{Z} - \hat{\mathbf{Z}}$.

d.4) Define the hidden layer output error for a given x as

$$\tilde{\sigma} = \sigma - \hat{\sigma} = \sigma(\mathbf{V}^T x) - \sigma(\hat{\mathbf{V}}^T x). \qquad (28)$$

The Taylor series expansion for a given x is

$$\sigma(\mathbf{V}^T x) = \sigma(\hat{\mathbf{V}}^T x) + \sigma'(\hat{\mathbf{V}}^T x)\tilde{\mathbf{V}}^T x + O(\tilde{\mathbf{V}}^T x), \qquad (29)$$

where $O(\tilde{\mathbf{V}}^T x)$ denotes the higher-order terms in the Taylor series.

The following mild assumptions always hold in practical applications:

a.1) The ideal weights are bounded by known positive values so that $\|\mathbf{Z}\|_F \leq Z_M$ with Z_M known.

a.2) The desired velocity control input is bounded in the sense, for instance, that for some known constant $V_c \in \Re$, $\|v_c\|, \|\dot{v}_c\| \leq V_c$.

The following are facts, and do not restrict the class of systems susceptible to our approach:

f.1) For each time t, $x(t)$ in (23) is bounded by

$$\|x\| \leq c_1 + c_2 \|e_c\|, \qquad (30)$$

for computable positive constants c_i.

f.2) For the sigmoid activation functions the higher-order terms in the Taylor series (29) are bounded by

$$\|O(\tilde{\mathbf{V}}x)\| \leq c_3 + c_4 V_c \|\tilde{\mathbf{V}}\|_F + c_5 \|\tilde{\mathbf{V}}\|_F \|e_c\|, \qquad (31)$$

for computable positive constants c_i.

We will use a neural net to approximate $f(x)$ for computing the control in (24). By placing into (24) the neural network approximation equation given by (11), the control input then becomes

$$\bar{\tau} = \hat{\mathbf{W}}^T \sigma(\hat{\mathbf{V}}^T x) + K_4 e_c - \gamma, \qquad (32)$$

and the velocity error dynamics is given by

$$\bar{\mathbf{M}}\dot{e}_c = -(K_4 + \bar{\mathbf{V}}_\mathbf{m})e_c + \mathbf{W}^T \sigma(\mathbf{V}^T x) - \hat{\mathbf{W}}^T \sigma(\hat{\mathbf{V}}^T x) + (\varepsilon + \bar{\tau}_d) + \gamma. (33)$$

Adding and subtracting some terms and using the previously given definitions, assumptions and facts, the error system becomes

$$\bar{\mathbf{M}}\dot{e}_c = -(K_4 + \bar{\mathbf{V}}_\mathbf{m})e_c + \tilde{\mathbf{W}}^T(\hat{\sigma} - \hat{\sigma}'\hat{\mathbf{V}}^T x) + \hat{\mathbf{W}}^T \hat{\sigma}'\tilde{\mathbf{V}}^T x + w + \gamma \ (34)$$

where the disturbance terms are

$$w(t) = \tilde{\mathbf{W}}^T \hat{\sigma}'\mathbf{V}^T x + \mathbf{W}^T O(\tilde{\mathbf{V}}^T x) + \varepsilon + \bar{\tau}_d. \qquad (35)$$

f.3) The disturbance term (35) is bounded according to

$$\|w(t)\| \leq C_0 + C_1 \|\tilde{\mathbf{Z}}\|_F + C_2 \|\tilde{\mathbf{Z}}\|_F \|e_c\|, \qquad (36)$$

with C_i known positive constants. Details are discovered in [11].

It remains now to show how to select the tuning algorithms for the NN weights $\hat{\mathbf{Z}}$, and the robustifying term $\gamma(t)$ so that robust stability and tracking performance are guaranteed.

Theorem 1 Given a nonholonomic mobile robot (5), (6). Let the following assumptions hold:

a.3) A smooth time-varying auxiliary velocity control input $v_c(t)$ is prescribed that solves the point stabilization problem for the steering system (5), neglecting the dynamics (6). A sample v_c [2] is given by (16).

a.4) $K = [k_1 \ k_2 \ k_3 \ k_5]^T$ is a vector of positive constants.

a.5) $K_4 = k_4 \mathbf{I}$, where k_4 is a sufficiently large positive constant.

Take the control $\bar{\tau} \in \Re^2$ for (7) as (32) with robustifying term

$$\gamma(t) = -K_z(\|\hat{Z}\|_F + Z_M)e_c - e_c \qquad (37)$$

and gain

$$K_z > C_2 \qquad (38)$$

with C_2 the known constant in (36). Let NN weight tuning be provided by (39). Then, the velocity tracking error $e_c(t)$, the position error $e(t)$, and the NN weight estimates $\hat{v}, \hat{w}$ are UUB. Moreover, the velocity tracking error may be kept as small as desired by increasing the gain K_4.

$$\dot{\hat{\mathbf{W}}} = \mathbf{F}\hat{\sigma}e_c^T - \mathbf{F}\hat{\sigma}'\hat{\mathbf{V}}^T x e_c^T - \kappa \mathbf{F}\|e_c\|\hat{\mathbf{W}}, \ (39.\text{a})$$

$$\dot{\hat{\mathbf{V}}} = \mathbf{G}x(\hat{\sigma}'^T \hat{\mathbf{W}}e_c)^T - \kappa \mathbf{G}\|e_c\|\hat{\mathbf{V}}, \qquad (39.\text{b})$$

where $\mathbf{F}$, $\mathbf{G}$ are positive definite design parameter matrices, $\kappa > 0$ and the hidden-layer gradient $\hat{\sigma}'$ is easily computed—for the sigmoid activation function it is given by

$$\hat{\sigma}' \equiv diag\{\sigma(\hat{\mathbf{V}}^T x_d)\}[\mathbf{I} - diag\{\sigma(\hat{\mathbf{V}}^T x_d)\}]. (40)$$

Proof: See the Appendix.
□

Remarks The first terms of (39) are nothing but the standard backpropagation algorithm. The last terms correspond to the e-modification [12] from adaptive control theory; they must be added to ensure bounded NN weights estimates. The middle term in (39.a) is a <u>novel term</u> needed to prove stability.

4. Simulation Results

We should like to illustrate the NN control scheme presented in Fig. 1. Note that the NN controller <u>does not</u>

require knowledge of the dynamics, not even their structure. The controller gains were chosen so that the closed-loop system exhibits a critical damping behavior: $K = [10 \ 5 \ 4 \ 1]^T$, $K_4 = diag\{25, 25\}$. For the NN, we selected the sigmoid activation functions with $N_h = 10$ hidden-layer neurons, $\mathbf{F} = \mathbf{G} = diag\{10,10\}$ and $\kappa = 0.1$.

It is well-known that the rates of convergence provided by smooth periodic velocity inputs are at most $1/\sqrt{t}$ [14] *i.e.*, nonexponential. In order to have an acceptable closed-loop performance, we may use feedback laws which are smooth everywhere except at the boundary of a small neighborhood of the origin. The following choice has been proposed in [2]

$$g(e,t) = \begin{cases} \sin t & \text{if } \|e\| \geq \varepsilon > 0 \\ 0 & \text{otherwise.} \end{cases} \quad (41)$$

Although asymptotic convergence of the mobile robot cannot be guaranteed, the reference cart can be proven to be asymptotically stable. Therefore, the mobile robot can be stabilized to an arbitrarily small neighborhood of the origin. Simulation results that verify the validity of the combined kinematic/NN controller are depicted in Fig. 2-5.
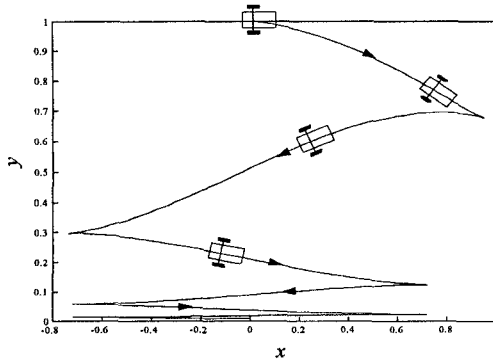

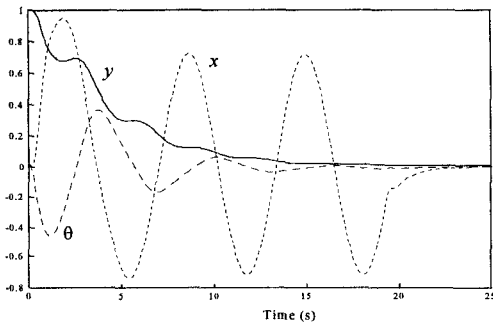**Fig. 2.** Trajectory of the mobile robot.

Fig. 3. Position and orientation.

The mobile base <u>maneuvers</u>, *i.e.*, exhibits forward and backward motions to reach the origin. Note that there is no path planning involved —the mobile base naturally describes a path that satisfies the nonholonomic constraints.
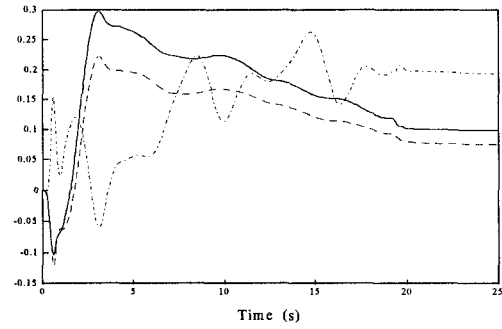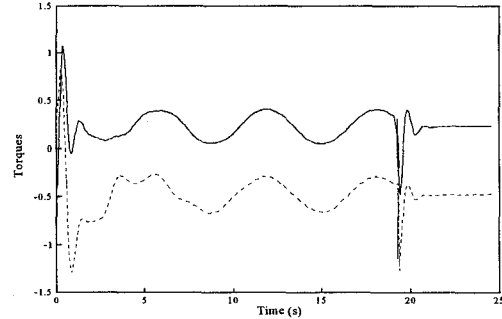

Fig. 4. Some NN weights.

Fig. 5. Applied Torques: Right and (- -) Left Wheels.

5. Conclusions

A stable control algorithm for practical point stabilization of a nonholonomic mobile robot, and that does not require knowledge of the cart dynamics has been derived using a NN approach. This feedback servo-control scheme is valid as long as the velocity control inputs are smooth and bounded, and the disturbances acting on actual cart are bounded.

In fact, perfect knowledge of the mobile robot parameters is unattainable, *e.g., friction* is very difficult to model by conventional techniques. To confront this, a *neural network* controller with guaranteed performance has been derived. There is not need of *a priori* information of the dynamic parameters of the mobile robot, because the NN learns them on-the-fly.

APPENDIX: PROOF OF THEOREM 1

Let the approximation property (10) hold with a given accuracy ε_N for all x in the compact set U_x.

Consider the following Lyapunov function candidate

$$V = \frac{k_2}{2}(e_1^2 + e_2^2) + \frac{e_3^2}{2} + V_1, \quad (42)$$

where

$$V_1 = \frac{1}{2}[e_c^T \overline{\mathbf{M}} e_c + tr\{\widetilde{\mathbf{W}}^T \mathbf{F}^{-1} \widetilde{\mathbf{W}}\} + tr\{\widetilde{\mathbf{V}}^T \mathbf{G}^{-1} \widetilde{\mathbf{V}}\}], \quad (43)$$

Differentiating yields

$$\dot{V} = k_2(e_1 \dot{e}_1 + e_2 \dot{e}_2) + e_3 \dot{e}_3 + \dot{V}_1. \quad (44)$$

Differentiating V_1, and using the skew symmetry property and the tuning rules give

$$\dot{V}_1 = -e_c^T K_4 e_c + \kappa \|e_c\| tr\{\widetilde{\mathbf{Z}}^T (\mathbf{Z} - \widetilde{\mathbf{Z}})\} + e_c^T(w + \gamma) \quad (45)$$

since

$$tr\{\widetilde{\mathbf{Z}}^T(\mathbf{Z}-\widetilde{\mathbf{Z}})\} = \langle\widetilde{\mathbf{Z}},\mathbf{Z}\rangle_F - \left\|\widetilde{\mathbf{Z}}\right\|_F^2 \le \left\|\widetilde{\mathbf{Z}}\right\|_F \|\mathbf{Z}\|_F - \left\|\widetilde{\mathbf{Z}}\right\|_F^2 \quad (46)$$

there results

$$\dot{V}_1 \le -\|e_c\|[K_{4\min}\|e_c\| + \kappa\left\|\widetilde{\mathbf{Z}}\right\|_F(\left\|\widetilde{\mathbf{Z}}\right\|_F - Z_M) - C_0 - C_1\left\|\widetilde{\mathbf{Z}}\right\|_F] - \|e_c\|^2, \quad (47)$$

where $K_{4\min}$ is the minimum singular value of K_4 and the last inequality holds due to (38). By substituting (47) and (15) in (44), we obtain

$$\dot{V} \le k_2 e_1(v_2 e_2 - v_1 + v_r\cos e_3) + k_2 e_2(-v_2 e_1 + v_r\sin e_3)$$

$$+ e_3(-v_2) - \|e_c\|^2 - \|e_c\|[K_{4\min}\|e_c\| + \kappa\left\|\widetilde{\mathbf{Z}}\right\|_F(\left\|\widetilde{\mathbf{Z}}\right\|_F - Z_M) \quad (48)$$

$$- C_0 - C_1\left\|\widetilde{\mathbf{Z}}\right\|_F].$$

Defining $k_1 > \dfrac{k_2}{4}$, $k_2 > 0$, $k_3 > \dfrac{1}{4}$ yield

$$\dot{V} \le -k_2(k_1 - \frac{k_2}{4})e_1^2 - (k_3 - \frac{1}{4})e_3^2 - (e_4 - \frac{k_2}{2}e_1)^2$$

$$-(e_5 - \frac{1}{2}e_3)^2 - \|e_c\|[K_{4\min}\|e_c\| + \kappa\left\|\widetilde{\mathbf{Z}}\right\|_F(\left\|\widetilde{\mathbf{Z}}\right\|_F - Z_M) \quad (49)$$

$$- C_0 - C_1\left\|\widetilde{\mathbf{Z}}\right\|_F].$$

Since the first four terms in (49) are negative, there results

$$\dot{V} \le -\|e_c\|[K_{4\min}\|e_c\| + \kappa\left\|\widetilde{\mathbf{Z}}\right\|_F(\left\|\widetilde{\mathbf{Z}}\right\|_F - Z_M) - C_0 - C_1\left\|\widetilde{\mathbf{Z}}\right\|_F]. \quad (50)$$

Thus, $\dot{V}$ is guaranteed negative as long as either

$$\|e_c\| > \frac{\kappa C_3^2 + C_0}{K_{4\min}} \equiv b_c \quad (51)$$

or

$$\left\|\widetilde{\mathbf{Z}}\right\|_F > C_3 + \sqrt{C_3^2 + \frac{C_0}{\kappa}} \equiv b_z \quad (52)$$

where

$$C_3 = \tfrac{1}{2}(Z_M + \tfrac{C_1}{\kappa}). \quad (53)$$

Therefore, $\dot{V}$ is negative outside a compact set. According to a standard Lyapunov theory extension, this demonstrates the *UUB* of both $\|e_c\|$ and $\left\|\widetilde{\mathbf{Z}}\right\|_F$.

Finally from (18) the reference cart equation is given by

$$\dot{x}_r = -k_5 x_r + g(e,t). \quad (54)$$

Since $g(e,t)$ tends to zero for the ideal case, and $g(e,t) = 0$ if $\|e\| < \varepsilon$ for the practical case, it can be shown that $x_r \to 0$ as $t \to \infty$.

□

References

[1] A. M. Bloch, M. Reyhanoglu and N. H. McClamroch, "Control and stabilization of nonholonomic dynamic systems," *IEEE Trans. on Automatic Control*, vol. 37, no. 11, pp. 1746-1757, November 1992.

[2] C. Canudas de Wit, H. Khennouf, C. Samson and O. J. Sørdalen, "Nonlinear control design for mobile robots," in *Recent Trends in Mobile Robots*, Y. F. Zheng ed., pp. 121-156, World Scientific, 1993.

[3] C. Canudas de Wit and H. Khennouf, "Quasi-continuous stabilizing controllers for nonholonomic systems: design and robustness consideration," *Proceedings of the European Control Conference*, 1995.

[4] F.-C. Chen and C.-C. Liu, "Adaptively controlling nonlinear continuous-time systems using multilayer neural networks," *IEEE Trans. Automat. Control*, vol. 39, no. 6, pp. 1306-1310, June 1994.

[5] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Controls Signals and Systems*, vol. 2, no. 4, pp. 303-314, 1989.

[6] R. Fierro and F. L. Lewis, "Control of a nonholonomic mobile robot: backstepping kinematics into dynamics," *Proc. 34th IEEE Conference on Decision and Control*, pp. 3805-3810, New Orleans, December 1995.

[7] K. Hornik, M. Stinchcombe and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, pp. 359-366, 1989.

[8] Z. P. Jiang and J.-B. Pomet, "Combining backstepping and time-varying techniques for a new set of adaptive controllers," *Proc. 33th IEEE Conference on Decision and Control*, pp. 2207-2212, 1994.

[9] I. Kolmanovsky and N. H. McClamroch, "Developments in nonholonomic control problem," *IEEE Control Systems*, pp. 20-36, December 1995.

[10] F. L. Lewis, C. T. Abdallah and D. M. Dawson, *Control of Robot Manipulators*, MacMillan, New York, 1993.

[11] F. L. Lewis, A. Yesildirek and K. Liu, "Neural net robot controller: structure and stability proofs," *Proc. 31th IEEE Conference on Decision and Control*, pp. 2785-2791, San Antonio, December 1993.

[12] K. S. Narendra, "Adaptive control using neural networks," in *Neural Networks for Control*, W. T. Miller, R. S. Sutton, P. W. Werbos editors, pp. 115-142, Cambridge: MIT press, 1991.

[13] M. M. Polycarpou and P. O. Ioannu, "Neural networks as on-line approximators of nonlinear systems," *Proc. 30th IEEE Conference on Decision and Control*, pp. 7-12, Tucson, December 1992.

[14] C. Samson, "Velocity and torque feedback control of a nonholonomic cart," in *Lecture Notes in Control and Information Science*, C. Canudas de Wit ed., pp. 125-151, Springer-Verlag, 1991.

[15] Y. Yamamoto and X. Yun, "Coordinating locomotion and manipulation of a mobile manipulator," in *Recent Trends in Mobile Robots*, Y. F. Zheng ed., pp. 157-181, World Scientific, 1993.