Wheeled Mobile Robot Control Using Sliding Modes and Neural Networks

Vinícius M. de Oliveira[1], Edson R. de Pieri[2], Walter F. Lages[3]

vinicius@ieee.org, edson@das.ufsc.br, w.fetter@ieee.org

[1]Department of Mathematics, Federal University of Rio Grande

Av. Italia, km 8 s/n, 96201-900, Rio Grande, RS, Brazil

[2]Department of Automation and Systems, Federal University of Santa Catarina

CP 476, 88040-900, Florianópolis, SC, Brazil

[3]Electrical Engineering Department, Federal University of Rio Grande do Sul

Av. Oswaldo Aranha, 103, 90035-190 Porto Alegre, RS, Brazil

Abstract

The complete model of a mobile robot can be divided into kinematics and dynamics. To take advantage from this fact, a combined controller of a sliding mode kinematic controller with a neural network computed-torque dynamic controller is proposed. The proof of stability is based on the Lyapunov theory. Concerning with Brockett's theorem, the kinematic controller fulfills the requirements and the robot can be stabilized in a desired posture. Experimental real-time results are presented.

1 Introduction

A few years ago the term **robotics** was used to remember only robotic manipulator due to its widespread application in industry. Nowadays this term also brings to mind service robots, also known as mobile robots.

Mobile robots can be applied in a large variety of applications. The most seen applications are mail delivering, bomb disposal, helping physical deficient (e. g. wheelchairs) [1, 2], spatial exploration [3] and inhospitable environment (such as volcanoes) and in underwater activities (e. g. petroleum exploration, substituting divers in dangerous tasks) [4]. Besides that, there are some application of autonomous guided vehicles in the reduction of traffic congest, pollution and accidents caused by human inability.

Due to this wide variety of mobile robots applications, the control of such systems starts to play an important role in technical literature. In the opposite of the commonly used robotic manipulators, mobile robots are nonholonomic systems, i. e., they have constraints that cannot be integrated. For mobile robots with differential drive it is possible to consider the kinematics isolated from the dynamics, while analysing the system. The majority of technical work consider only the control of the kinematics of the system, once the nonholonomic constrainsts are kinematic constraints [5, 6, 7, 8].

When controlling a mobile robot you can intend the robot tracks a desired trajectory, the robot moves toward a desired posture (desired position and orientation) or the robot tracks a desired trajectory parking in a desired location. Brockett has shown that to park a nonholonomic mobile robot in a reference posture it is necessary to apply a non-smooth or time-variant feedback control law, i. e., the system is not stabilizable with a smooth time-invariant feedback control. Thus, the researchers focused their works to develop new control techniques to overcome this limitation.

Once the control challenge is in the kinematics of the robot, the dynamics wasn't considered to design control laws. Some kinemaitc control techniques are: fuzzy control [9], discontinuous control [10], time-variant state feedback control [11, 12] and predictive control [13, 14]. Some techniques to control the dynamics of the system are linearizing feedback [15], backstepping [16], adaptive linearizing control [17] and neural networks [18, 19].

Nowadays, when implementing mobile robots, there are some problems related with the mathematical modelling of the kinematics and of the dynamics, some difficulties to estimate the orientation and the position of the robot, some complexity in the control design and also when planning a path to be tracked. When executing computational simulation the robot models require a special attention, because if these models are accurate, the parameters will not require to be adjust (or at least they will be aproximate) when used in the real environment, and the result will be, or should be, very close that obtained in the simulation environment.

Another problem to be considered when executing a practical experience is concerned with the information available from the robot. While in simulation all variables involved are available and ready to be used at any instant, in practice, it is possible to have only some variables available. To deal with it the estimation of the variables not available is indicated.

As the mobile robot navigates autonomously, it is necessary to the robot the knowledge of its actual position and orientation to execute the task in a safe way, avoiding to get lost. To have this information is necessary to apply some methods to estimate the position and orientation, using for that special devices such as encoders, sonars, video camera, digital compass, GPS and others.

Differently of [20], this work presents a kinematic control loop based on sliding modes technique and a dynamic control loop based on neural network technique. The task imposed to the robot is to move torward a reference posture (origin of the system), and the control law satisfies the restriction showed by Brockett. The neural network is choosen due to its capability to *learn* and to the fact the accurate measurement of the parameters involved in the dynamics is very difficult.

2 Mobile Robot Modelling

The mobile robot used in this work (see figure 1) is a circular platform with 4 wheels, with 2 of them mounted on the same axle with a DC motor attached to each one [21]. This is a differential-driver robot. See [6] for some insights on structural properties of this class of robot.

A mobile robot system having an n-dimensional configuration space $\mathcal{C}$ with generalized coordinates $\vec{q} = [q_1 \cdots q_n]^T$ and with m constraints can be described as follows [18], [22]:

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

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

Considering the time-independence of all kinematic equality constraints one can write:

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

Let $\mathbf{S}(q)$ be a full rank matrix $(n - m)$ belonging to the null space of $\mathbf{A}^T(q)$, such that:
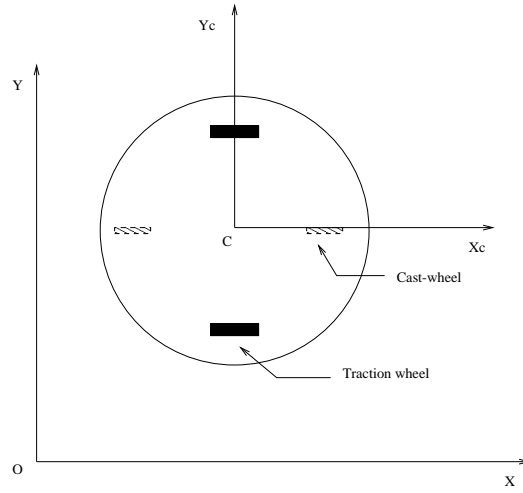
$$\mathbf{S}^T(q)\mathbf{A}^T(q) = 0 \tag{3}$$

2

Figure 1: Configuration of the wheels of the mobile robot.

Based on 2 and 3 its possible to find out $\vec{v}(t) \in \mathcal{R}^{n-m}$, such that, for all t:

$$\dot{q} = \mathbf{S}(q)\eta(t) \qquad (4)$$

where η is a column vector with the linear and angular velocities as its elements.

The system 1 will be now transformed into a suitable representation to the control perspective. By differentiating 4 and replacing the result in 1, then pre-multiplying by $\mathbf{S}^T(q)$ and using 2 and 3 it is possible to eliminate the constraint matrix $\mathbf{A}^T(q)\lambda$, resulting in:

$$\mathbf{S}^T\mathbf{M}\mathbf{S}\dot{\eta} + \mathbf{S}^T(\mathbf{M}\dot{\mathbf{S}} + V_m\mathbf{S})\eta + \mathbf{S}^TF + \mathbf{S}^T\tau_d = \mathbf{S}^T\mathbf{B}\tau \qquad (5)$$

Which can be rewritten as:

$$\overline{\mathbf{M}}\dot{\eta} + \overline{\mathbf{V}}_{\mathbf{m}}\eta + \overline{F} + \overline{\tau}_d = \overline{\tau} \qquad (6)$$

where $\overline{\mathbf{M}}(q) \in \mathcal{R}^{r \times r}$ is the inertia matrix (symmetric and positive definite), $\overline{\mathbf{V}}_{\mathbf{m}}(q, \dot{q}) \in \mathcal{R}^{r \times r}$ is the centripetal and coriolis terms matrix, $\overline{F}(v) \in \mathcal{R}^{r \times 1}$ is the friction terms, $\overline{\tau}_d$ denotes bounded unknown disturbances including unstructured unmodeled dynamics. $\overline{\tau}$ is the input vector ($\overline{\mathbf{B}}$ is a constant nonsingular matrix that depends on geometric parameter of the robot). The matrix $\overline{\mathbf{M}}$ and the norm of the $\overline{\mathbf{V}}_{\mathbf{m}}$ are bounded and the matrix $\dot{\overline{\mathbf{M}}} - 2\overline{\mathbf{V}}_{\mathbf{m}}$ is skew-symmetric.

3   Sliding Mode Controller

The purpose of the control law obtained by sliding mode technique is to track the nonlinear trajectory of the system to a pre-specified surface (defined by the designer) in the state space and maintain it in this surface for all subsequent time. This surface is called *switching surface*. When the system trajectory is above the switching surface, the feedback presents one gain, and when the system trajectory is below the surface, the feedback uses a different gain.

This switching surface is also known as *sliding manifold* because, at least in theory, once this surface is intercepted by the system trajectory, the control law would impose to the system trajectory to track the surface for all the future time (the trajectory will slide over the surface).

3

The dynamics of the process limited to this surface denotes the behavior of the controlled system. The first step is to design the sliding surface according to the desired behavior of the closed-loop system, such as convergence to the origin, parametric variation robustness [23, 24, 25, 26].

For instance, it is shown a simple example to illustrate such technique. Consider the first order system:

$$\dot{x}(t) = u(x, t) \tag{7}$$

with

$$u(x, t) = sign(x) = \begin{cases} -1 & if \ x > 0 \\ 0 & if \ x = 0 \\ +1 & if \ x < 0 \end{cases} \tag{8}$$
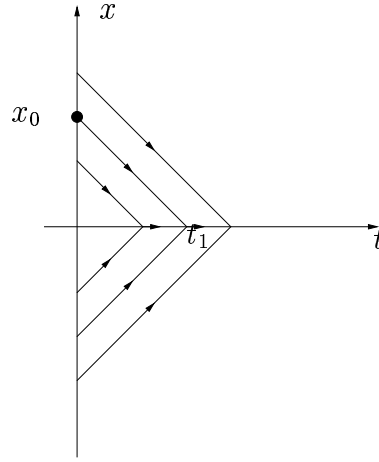

PSfrag replacements

Figure 2: Trajetories of the system 7.

The trajectory of the system 7, with the control 8 applied, is shown in figure 2. It is possible to observe the control law changes from -1 to $+1$ around the surface $\sigma(x, t) = x = 0$. Thus, for any initial state x_0 exists a finite time t_1 such that $x(t) = 0 \ \forall t \geq t_1$.

It is important to note that to design such sliding surface one has to consider the control law is able to switch from one value to the other in an infinitesimal time [27]. As in the real it is impossible to have a null switching time, it causes an effect known as *chattering* [28, 4]. This fenomenum is present, usually as a high frequency oscilation in the equilibrium point, and can excite high frequency modes of the dynamics.

The delay in switching from one value to another can be caused by many reasons, such as data measurement, compute of the control law, actuator limits. In [25] we have some methods to elimininate, or to reduce, the effects of the *chattering*.

3.1 Kinematic Loop Control

The controller proposed in this section has a simple functional structure, as depicted in the block diagram in figure 3. The system inputs are the reference posture and the state vector of the kinematic model.

Lemma 3.1 *Let the work space of a robot be defined as* $\mathcal{T} \subset \mathcal{R}^2$ *and its configuration space given by* $\mathcal{C} = \mathcal{T} \times [-\pi, \pi[\in \mathcal{R}^3$.
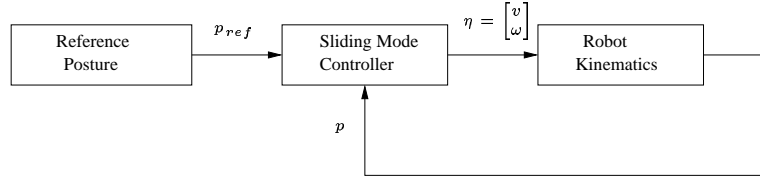
4

Figure 3: Block diagram of the kinematic loop control.

In this control scheme a Lyapunov function V is designed to navigate the system to the origin of the work space $\mathcal{T} \subset \mathcal{R}^2$. The robot navigation to the origin of the configuration space $\mathcal{C}$ is guaranteed by the associated gradient $\varepsilon = \nabla V$. Once this gradient posses some necessary properties, the robot navigation will happen.

The control law is designed to keep the system trajectory along to the $\varepsilon(x, y)$ gradient. Thus, the invariance and order reduction properties of the sliding mode technique are applied [28]. The objective is to keep the linear velocity vector of the vehicle colinear to the gradient; the velocity of the movement along the gradient can be determined independently. The trajectory is obtained solving the following equation:

$$\frac{dy}{dx} = \frac{\varepsilon_y(x, y)}{\varepsilon_x(x, y)} \tag{9}$$

The trajectory must be smooth and continuous, and the first derivatives of the gradient associated to the Lyapunov function $\frac{\partial \varepsilon_x}{\partial x}$, $\frac{\partial \varepsilon_x}{\partial y}$, $\frac{\partial \varepsilon_y}{\partial x}$ and $\frac{\partial \varepsilon_y}{\partial y}$ are limited.

Let the orientation error be given by:

$$\Delta\theta = \theta_\varepsilon - \theta \tag{10}$$

The error dynamics is obtained taking the first time-derivative of the equation 10, resulting:

$$
\begin{aligned}
\Delta\dot\theta &= \frac{\partial\theta_\varepsilon}{\partial x}\dot{x} + \frac{\partial\theta_\varepsilon}{\partial y}\dot{y} - \omega \\[2mm]
&= \nabla\theta_\varepsilon \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} - \omega \\[2mm]
&= \nabla\theta_\varepsilon \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} v - \omega \\[2mm]
&= \left[\frac{1}{1+\left(\frac{\varepsilon_y}{\varepsilon_x}\right)^2}\frac{\partial}{\partial x}\left(\frac{\varepsilon_y}{\varepsilon_x}\right) \quad \frac{1}{1+\left(\frac{\varepsilon_y}{\varepsilon_x}\right)^2}\frac{\partial}{\partial y}\left(\frac{\varepsilon_y}{\varepsilon_x}\right) \right] \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} v - \omega \\[2mm]
&= \left[\frac{\varepsilon_x^2}{\varepsilon_x^2+\varepsilon_y^2}\frac{\partial}{\partial x}\left(\frac{\varepsilon_y}{\varepsilon_x}\right) \quad \frac{\varepsilon_x^2}{\varepsilon_x^2+\varepsilon_y^2}\frac{\partial}{\partial x}\left(\frac{\varepsilon_y}{\varepsilon_x}\right) \right] \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} v - \omega \\[2mm]
&= \left[\frac{\varepsilon_x\frac{\partial\varepsilon_y}{\partial x}-\varepsilon_y\frac{\partial\varepsilon_x}{\partial x}}{\|\varepsilon\|^2}\cos(\theta) \quad \frac{\varepsilon_x\frac{\partial\varepsilon_y}{\partial y}-\varepsilon_y\frac{\partial\varepsilon_x}{\partial y}}{\|\varepsilon\|^2}\sin(\theta) \right] v - \omega
\end{aligned} \tag{11}
$$

This equation can be rewriten to a simpler form:

$$\Delta\dot\theta = F(x, y, \theta)v - \omega \tag{12}$$

where

$$F(x, y, \theta) = \left[\frac{\varepsilon_x\frac{\partial\varepsilon_y}{\partial x}-\varepsilon_y\frac{\partial\varepsilon_x}{\partial x}}{\|\varepsilon\|^2}\cos(\theta) \quad \frac{\varepsilon_x\frac{\partial\varepsilon_y}{\partial y}-\varepsilon_y\frac{\partial\varepsilon_x}{\partial y}}{\|\varepsilon\|^2}\sin(\theta) \right] \tag{13}$$

5

Defining the control input ω as:

$$\omega \triangleq F(x, y, \theta)v + \xi sign(\Delta\theta)\sqrt{|\Delta\theta|} \tag{14}$$

ensures the convergence of $\Delta\theta$ to zero will be in a finite time, with ξ as a positive and finite scalar constant [29].

To ensure the existence of the sliding mode with a finite control is necessary that the term $F(x, y, \theta)v$, in 14, be finite. The velocity control used to give a finite $F(x, y, \theta)v$ is defined as follow:

$$v(t) \triangleq -\|\varepsilon\|v^*(t) \tag{15}$$

where $v^*(t)$ is a limited auxiliar control input. It is possible to observe that $\varepsilon(0, 0)$ and, consequently, $v(t)$ and $F(x, y, \theta)v$ also converge to zero at the origin. When the sliding mode occurs along $\Delta\theta = 0$, the gradient field ε is followed by the system, reducing the kinematic system 4, under the control 14 and 15, to the following system:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \end{bmatrix} \frac{v(t)}{\|\varepsilon\|} = -\begin{bmatrix} \varepsilon_x \\ \varepsilon_y \end{bmatrix} v^*(t) \tag{16}$$

At the first sight is possible to think the reduced system is still under the restrictions of Brockett's theorem [30], once the system has a state dimension higher than control input dimension. But, in a more detailed analisys, one can observe that due to the sliding mode technique, the gradient of the Lyapunov function is exactly tracked, reducing the order of the system by the restriction of the movement to the resulting manifold of 9.

3.2 Control Law

Let $V(x, y)$ be candidate to the Lyapunov function, expressed by:

$$V(x, y) = \frac{1}{2}\left(\frac{x^2}{a} + y^2\right) > 0 \tag{17}$$

where a is a positive scalar constant and V a positive definite function. Let the associated gradient be given as:

$$\varepsilon(x, y) = -\nabla V = \begin{bmatrix} -\frac{x}{a} \\ -y \end{bmatrix} \tag{18}$$

According to the expression 9 and with the associated gradient 18, we have trajectories of the type:

$$y = \gamma|x|^a \tag{19}$$

where γ depends on the initial conditions, without importance to the control.

The control input responsible by the co-linear orientation of the robot to the gradient $\varepsilon(x, y)$ is obtained substituting equation 18 in 14, resulting in:

$$\omega = -\frac{x\sin\theta - y\cos\theta}{\sqrt{x^2 + (ay)^2}}v^* + \xi sign(\Delta\theta)\sqrt{|\Delta\theta|} \tag{20}$$

with the orientation error expressed by equation 10 and velocity control given by:

$$v(t) = -\sqrt{x^2 + (ay)^2}\, v^*(t) \tag{21}$$

6

While the sliding mode exists, we have $\theta = \theta_\varepsilon$ and the robot movement is dictate by the reduced system 16 with the restriction 9. Taking:

$$v^*(t) = -v_0 \tag{22}$$

the position error has an exponential convergence. The convergence of the orientation θ to zero can be determined analising:

$$\theta = \theta_\varepsilon = \arctan(\frac{ay}{x}) = \arctan(abx^{a-1}) \tag{23}$$

during the time evolution of x 16, with control 21 and $a > 1$.

Considering the region around the origin and the approximation $\tan(\alpha) \approx \alpha$, to small values of α, is possible to conclude that θ convergers exponentialy to zero.

4 Dynamics Neural Controller

As seen in chapter 2, the control of mobile robots with differential drive can be seen as a cascade of a control loop to the kinematics followed by a control loop to the dynamics of the robot.

In this section is adopted an strategy based on the application of artificial neural networks to control the mobile robot dynamics [18].

A brief introduction to artificial neural networks is presented in the next section and then the control strategy used, as well the proof of stability and some practical results.

4.1 Artificial Neural Networks

An artificial neural networl (figure 4) consists in a set of processing units that communicate by sending signals to the other units throw weighted connections [31]. Each of the basic elements of a neural network (also called neurons) has input signal originated by the other elements or signals externals to the net.
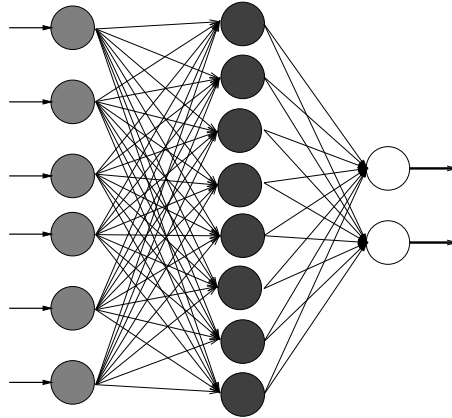

Figure 4: Structure of the neural network adopted.

The neural network used in this has 6 neurons in the input layer, 8 neurons in the hidden layer and, finally, 2 neurons in the output layer. Thus, the output signal of the neural net is given by the following equation:

$$y(x) = \mathbf{W}^T \sigma(\mathbf{V}^T x) \tag{24}$$

7

where $x \in \mathcal{R}^{6 \times 1}$ is the input vector of the neural network, $\mathbf{V} \in \mathcal{R}^{6 \times 8}$ is the weight matrix between the input layer and the intermediate layer and $\mathbf{W} \in \mathcal{R}^{8 \times 2}$ is the weight matrix between the intermediate layer and the output layer. The function $\sigma(\cdot)$ is the so called activation function and in this work we decided to use the following sigmoidal function:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{25}$$

The application of neural networks in the control of the dynamics of a mobile robot is intuited from the intrinsec characteristic of neural networks to map unknown nonlinear functions [32]. This way, a nonlinear function generates sampled pairs (x_1, y_1), (x_2, y_2), ..., (x_n, y_n). The sampled data modify the parameters of the neural estimator, approximating the neural system output to the output of the unknown nonlinear function [33]. The accuracy of this approximation is proportional to the amount of sampled data.

Based on the feature above and considering $\Gamma(x)$ a continuous function $\mathcal{R}^n \mapsto \mathcal{R}^m$ is possible to show that, making x restricted to a compact subset $U_n \in \mathcal{R}^n$, for a given number N of neurons in the hidden layer, there is a configuration of the neural network such that:

$$\Gamma(x) = \mathbf{W}^T \sigma(\mathbf{V}^T x) + \epsilon \tag{26}$$

where ϵ is the neural network approximation error. Fixing the maximum error allowed in the approximation process to ϵ_N, it is possible to find $\epsilon < \epsilon_N$. The most important to the control perspective is the fact that, once the value ϵ_N is specified, there is one configuration of the neural network such that the maximum desired error in the approximation is obtained. Thus, an estimative of $\Gamma(x)$ is given by:

$$\hat{\Gamma}(x) = \hat{\mathbf{W}}^T \sigma(\hat{\mathbf{V}}^T x) + \epsilon \tag{27}$$

where $\hat{\mathbf{V}}$ and $\hat{\mathbf{W}}$ are estimatives of the ideal weight matrices.

It is important to notice the way weights of the neural network are updated. The most used method to update the weights is the off-line technique, where the neural network is trained with a set of values originated from the function to be approximate. This technique requires two phases: one to teach the neural network and another to execute the function approximation. The use of such technique in a control application may not be possible, due to the time required.

In this work we use the on-line training technique, which has only one phase, because the weights are adjusted during the execution phase. The weights are adjusted according to the following equations, which will be used later to satisfy the theory of Lyapunov:

$$\begin{aligned} \Delta \hat{\mathbf{W}} &= \mathbf{F}\sigma(\hat{\mathbf{V}}^{\mathbf{T}}x)e_c^T - \mathbf{F}\sigma'(\hat{\mathbf{V}}^T x)\hat{\mathbf{V}}^T x e_c^T \\ &\quad -k\mathbf{F}\|e_c\|\hat{\mathbf{W}} \end{aligned} \tag{28}$$

$$\begin{aligned} \Delta \hat{\mathbf{V}} &= \mathbf{G}x(\sigma'(\hat{\mathbf{V}}^T x)\hat{\mathbf{W}}e_c)^T \\ &\quad -k\mathbf{G}\|e_c\|\hat{\mathbf{V}} \end{aligned} \tag{29}$$

where the design parameters $\mathbf{F}$ and $\mathbf{G}$ are positive definite matrices and $k > 0$.

5   Overall Control Structure of the Mobile Robot

In this section we present, besides the overall structure of the controller, the proof of stability of the neural network and, as a consequence, the stability of the controller.

Once the dynamic model of the mobile robot is defined, we present in figure 5 the blocks diagram of the control scheme proposed. As already mentioned, we have a block related to the kinematics control and
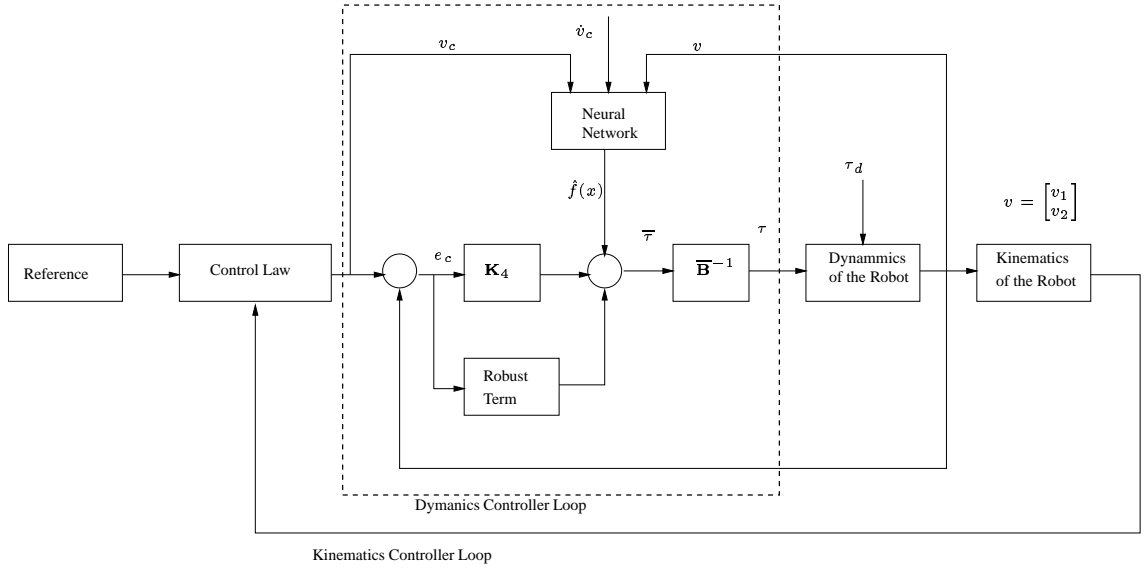
Figure 5: Block diagram of the controller.

another block related to the dynamics control, where the neural network is involved. One can observe that the error in tracking the reference velocity is given by the difference between the value obtained from the kinematic control loop and the value measured from the robot:

$$e_c = v_c - v \tag{30}$$

Taking the first time-derivative of equation 30 and substituting it in equation 6, the dynamics of the robot can be described by the velocity tracking error, as follow:

$$\overline{\mathbf{M}}\dot{e}_c = -\overline{\mathbf{V}}e_c - \overline{\tau} + f(x) + \overline{\tau}_d \tag{31}$$

where the nonlinear function $f(x)$ of the robot is given by:

$$f(x) = \overline{\mathbf{M}}\dot{v}_c + \overline{\mathbf{V}}v_c + \overline{\mathbf{F}} \tag{32}$$

After defining the nonlinear function for the dynamics of the robot, it is important to notice that the parameter involved in this model (mass, inertial moments, friction coefficients, etc) are not known with enough accuracy, once they are of difficult measurement. Due to this lack of precision we make use of the approximation of multivariable nonlinear functios property from the neural networks to map equation 32.

A simple inspection on this equation allows to explicitly define the input variables to the neural network. The input vector of the neural network is defined as follows:

$$x = \begin{bmatrix} v^T & v_c^T & \dot{v}_c^T \end{bmatrix}^T \tag{33}$$

Continuing, we obtain a control law expressed by:

$$\overline{\tau} = \hat{f} + \mathbf{K}_4 e_c - \gamma \tag{34}$$

where $\mathbf{K}_4$ is a positive definite matrix of gains and $\hat{f}(x)$ is an estimative of function $f(x)$ of the robot, which is performed by the neural network. The γ signal is to guarantee the control law robustness to unstructured unmodeled disturbances [18].

9

Applying this control in equation 31, the closed loop system can be described as:

$$\overline{\mathbf{M}}\dot{e}_c = -(\mathbf{K_4} + \overline{\mathbf{V}}_{\mathbf{m}})e_c + \tilde{f}(x) + \overline{\tau}_d + \gamma \tag{35}$$

with $\tilde{f} = f - \hat{f}$.

For a better understanding of the stability proof, which is described bellow, we need to make some definitions:

Definition 5.1 *The solution of a nonlinear system with the state $x(t) \in \mathcal{R}^n$ is uniformly locally stable if there is a compact set $U_x \subset \mathcal{R}^n$, for all $x(t_0) = x_0 \in U_x$ exists $\delta > 0$ and a number $T(\delta, x_0)$ that $\|x(t)\| < \delta$, for all $t \geq t_0 + T$.*

Definition 5.2 *Let $\mathbf{A} = [a_{ij}], \mathbf{A} \in \mathcal{R}^{m \times n}$ be the Frobenius norm, defined by:*

$$\|\mathbf{A}\|_F^2 = tr\{\mathbf{A}^T\mathbf{A}\} = \sum_{i,j} a_{ij}^2 \tag{36}$$

Considering $\mathbf{B} = [b_{ij}], \mathbf{B} \in \mathcal{R}^{m \times n}$, the inner product associated is $\langle \mathbf{A}.\mathbf{B} \rangle_F = tr\{\mathbf{A}^T\mathbf{B}\}$. The Frobenius norm cannot be defined as a norm of the induced matrix to any vectorial norm, but it is compatible with the 2-norm, such that $\|\mathbf{A}x\|_2 \leq \|\mathbf{A}\|_F \|x\|_2$, with $\mathbf{A} \in \mathcal{R}^{m \times n}$ e $x \in \mathcal{R}^n$.

Definition 5.3 *To facilitate the notation, we define a matrix with all weigths of the neural network as $\mathbf{Z} \equiv diag\{\mathbf{W}, \mathbf{V}\}$.*

Definition 5.4 *The errors from the estimations the weigths are given by $\tilde{\mathbf{V}} = \mathbf{V} - \hat{\mathbf{V}}$, $\tilde{\mathbf{W}} = \mathbf{W} - \hat{\mathbf{W}}$ and $\tilde{\mathbf{Z}} = \mathbf{Z} - \hat{\mathbf{Z}}$.*

Definition 5.5 *Let the error of the intermediate layer, for a given x, be defined as:*

$$\tilde{\sigma} = \sigma - \hat{\sigma} = \sigma(\mathbf{V}^T x) - \sigma(\hat{\mathbf{V}}^T x) \tag{37}$$

The expansion in Taylor Series of $\sigma(s)$ for a given x can be described as follows:

$$\sigma(\mathbf{V}^T x) = \sigma(\hat{\mathbf{V}}^T x) + \sigma'(\hat{\mathbf{V}}^T x) + O(\tilde{\mathbf{V}}^T x) \tag{38}$$

with

$$\sigma'(\hat{z}) \equiv \left. \frac{\partial \sigma(z)}{\partial z} \right|_{z=\hat{z}}$$

being the Jacobian matrix and $O(\hat{\mathbf{V}}^T x)$ the higher order terms in the Taylor series. Defining $\hat{\sigma}' \triangleq \sigma'(\hat{\mathbf{V}}^T x)$ it is possible to write:

$$\tilde{\sigma} = \sigma'(\hat{\mathbf{V}}^T x)\tilde{\mathbf{V}}^T x + O(\tilde{\mathbf{V}}^T x) \tag{39}$$

One can observe in definiton 5.5 (equation 37) that $\tilde{\sigma}$ is nonlinear with respect to $\tilde{\mathbf{V}}$ and in equation 39 it is linear with respect to $\tilde{\mathbf{V}}$. The determination of the weight adjustment rule is possible due to this linearization.

Based on practical applications, the following considerations were made [18].

Consideration 5.1 *In a given compact subset of $\mathcal{R}^n$, the ideal weights of the neural network are bounded by a known positive number, which means that $\|\mathbf{V}\| \le V_M$, $\|\mathbf{W}\| \le W_M$ or $\|\mathbf{Z}\| \le Z_M$, with V_M, W_M or Z_M known.*

Consideration 5.2 *The reference trajectory is bounded, i. e., $\|q_r\| \le q_M$ with q_M a known constant scalar . The disturbances are also bounded: $\vec{\tau}_d \le d_m$.*

Consideration 5.3 *Let the robustness term be given by:*

$$\gamma(t) = -K_z(|\|\hat{\mathbf{Z}}\|_F + Z_M)e_c - e_c \tag{40}$$

with $K_z > C_2$.

Lemma 5.1 *For each instant t, $x(t)$ in 33 is bounded by:*

$$\|x\| \le q_M + c_0\|e_c(t_0)\| + c_2\|e_c(t)\| \le c_1 + c_2\|e_c(t)\| \tag{41}$$

where the variables c_i are positive and feasible.

Lemma 5.2 *The disturbance considered in the model is bounded by:*

$$\|\delta(t)\| \le C_0 + C_1\|\tilde{\mathbf{Z}}\|_F + C_2\|\tilde{\mathbf{Z}}\|_F\|e_c\| \tag{42}$$

The proofs of the lemma 5.1 and 5.2 will not be demonstrate in this paper and can be found in [34].

5.1 Stability of the Neural Controller by Lyapunov Theory

This section covers the mathematical proof of the stability of the proposed controller for the dynamics of mobile robots, which is based on neural networks technique.

Let V_0 be a function candidate to be the Lyapunov function, given by:

$$V_0 = V_1 + V_2 \tag{43}$$

where V_1 corresponds to the function from kinematics control loop and we assume that

$$V_1 > 0 \qquad \text{e} \qquad \dot{V}_1 < 0 \qquad\qquad \forall t > 0 \tag{44}$$

The function V_2 refers to the neural network controller, given by:

$$V_2 = \frac{1}{2}\left[e_c^T\overline{\mathbf{M}}e_c\right] + tr\{\tilde{\mathbf{W}}^T\mathbf{F}^{-1}\tilde{\mathbf{W}}\} + tr\{\tilde{\mathbf{V}}^T\mathbf{G}^{-1}\tilde{\mathbf{V}}\} \tag{45}$$

The first time derivative of V_2 is expressed by:

$$\dot{V}_2 = e_c^T\overline{\mathbf{M}}\dot{e}_c + e_c^T\dot{\overline{\mathbf{M}}}e_c + tr\{\tilde{\mathbf{W}}^T\mathbf{F}^{-1}\dot{\tilde{\mathbf{W}}}^T\} + tr\{\tilde{\mathbf{V}}^T\mathbf{G}^{-1}\dot{\tilde{\mathbf{V}}}^T\} \tag{46}$$

11

Substituting equation 35 in 46 and considering the approximation of the function $f(x)$ by the neural network we obtain:

$$
\begin{aligned}
\dot{V}_2 &= e_c^T \overline{\mathbf{M}} \dot{e}_c + e_c^T \big[-(\mathbf{K}_4 + \overline{\mathbf{V}}) e_c + \mathbf{W}^T \sigma(\mathbf{V} x) - \hat{\mathbf{W}}^T \sigma(\hat{\mathbf{V}} x) + \epsilon + \tau_d + \gamma \big] + \\
&\quad + \frac{1}{2} e_c^T \dot{\overline{\mathbf{M}}} e_c + tr\{\tilde{\mathbf{W}}^T \mathbf{F}^{-1} \dot{\tilde{\mathbf{W}}}^T\} + tr\{\tilde{\mathbf{V}}^T \mathbf{G}^{-1} \dot{\tilde{\mathbf{V}}}^T\}
\end{aligned}
\tag{47}
$$

$$
\begin{aligned}
\dot{V}_2 &= -e_c^T \mathbf{K}_4 e_c + \frac{1}{2}(\dot{\overline{\mathbf{M}}} - 2\overline{\mathbf{V}}) e_c + e_c^T \big[\mathbf{W}^T \sigma - \hat{\mathbf{W}}^T \hat{\sigma} + \mathbf{W}^T \hat{\sigma} - \mathbf{W}^T \hat{\sigma} + \epsilon + \tau_d + \gamma \big] + \\
&\quad + tr\{\tilde{\mathbf{W}}^T \mathbf{F}^{-1} \dot{\tilde{\mathbf{W}}}^T\} + tr\{\tilde{\mathbf{V}}^T \mathbf{G}^{-1} \dot{\tilde{\mathbf{V}}}^T\}
\end{aligned}
\tag{48}
$$

Applying the skew-simetry property and after some algebrical manipulations, it is possible to write:

$$
\begin{aligned}
\dot{V}_2 &= -e_c^T \mathbf{K}_4 e_c + e_c^T \big[\hat{\mathbf{W}}^T \tilde{\sigma} + \tilde{\mathbf{W}}^T \tilde{\sigma} + \tilde{\mathbf{W}}^T \hat{\sigma} + \epsilon + \tau_d + \gamma \big] + \\
&\quad + tr\{\tilde{\mathbf{W}}^T \mathbf{F}^{-1} \dot{\tilde{\mathbf{W}}}^T\} + tr\{\tilde{\mathbf{V}}^T \mathbf{G}^{-1} \dot{\tilde{\mathbf{V}}}^T\}
\end{aligned}
\tag{49}
$$

Replacing 39 in equation 49 we have:

$$
\begin{aligned}
\dot{V}_2 &= e_c^T \big\{ \hat{\mathbf{W}}^T [\hat{\sigma}' \tilde{\mathbf{V}} x + O(\tilde{\mathbf{V}} x)] + \tilde{\mathbf{W}}^T [\hat{\sigma}' \tilde{\mathbf{V}} x + O(\tilde{\mathbf{V}} x)] + \tilde{\mathbf{W}}^T \hat{\sigma} + \epsilon + \tau_d + \gamma \big\} + \\
&\quad - e_c^T \mathbf{K}_4 e_c + tr\{\tilde{\mathbf{W}}^T \mathbf{F}^{-1} \dot{\tilde{\mathbf{W}}}^T\} + tr\{\tilde{\mathbf{V}}^T \mathbf{G}^{-1} \dot{\tilde{\mathbf{V}}}^T\}
\end{aligned}
\tag{50}
$$

Using the lemma 5.1 and 5.2 we can rewrite equation 50 as:

$$
\begin{aligned}
\dot{V}_2 &= -e_c^T \mathbf{K}_4 e_c + e_c^T (\delta + \gamma) + e_c^T \big[\tilde{\mathbf{W}}^T \hat{\sigma}' \tilde{\mathbf{V}} x + \hat{\mathbf{W}}^T \hat{\sigma}' \tilde{\mathbf{V}} x + \tilde{\mathbf{W}}^T \hat{\sigma} \big] \\
&\quad + tr\{\tilde{\mathbf{W}}^T \mathbf{F}^{-1} \dot{\tilde{\mathbf{W}}}^T\} + tr\{\tilde{\mathbf{V}}^T \mathbf{G}^{-1} \dot{\tilde{\mathbf{V}}}^T\}
\end{aligned}
\tag{51}
$$

Let the disturbance term δ be given by:

$$
\delta(t) = \tilde{\mathbf{W}}^T \tilde{\sigma}' \mathbf{V}^T x + \mathbf{W}^T O() + \epsilon + \overline{\tau}_d
\tag{52}
$$

and we have equation 50 as follows:

$$
\begin{aligned}
\dot{V}_2 &= -e_c^T \mathbf{K}_4 e_c + e_c^T (\delta + \gamma) + tr\{\tilde{\mathbf{W}}^T (\mathbf{F}^{-1} \dot{\tilde{\mathbf{W}}} + \hat{\sigma} e_c^T - \hat{\sigma}' \hat{\mathbf{V}}^T x e_c^T)\} \\
&\quad + tr\{\tilde{\mathbf{V}}^T (\mathbf{G}^{-1} \dot{\tilde{\mathbf{V}}} + x e_c^T \hat{\mathbf{W}}^T \hat{\sigma}')\}
\end{aligned}
\tag{53}
$$

Based on definiton 5.4, we have that $\dot{\tilde{\mathbf{W}}} = -\dot{\hat{\mathbf{W}}}$ and $\dot{\tilde{\mathbf{V}}} = -\dot{\hat{\mathbf{V}}}$. Thus the equation above is rewritten below:

$$
\begin{aligned}
\dot{V}_2 &= -e_c^T \mathbf{K}_4 e_c + e_c^T (\delta + \gamma) + k\|e_c\| \big[tr\{\tilde{\mathbf{W}}^T (\mathbf{W} - \tilde{\mathbf{W}})\} + tr\{\tilde{\mathbf{V}}^T (\mathbf{V} - \tilde{\mathbf{V}})\} \big] \\
\dot{V}_2 &= -e_c^T \mathbf{K}_4 e_c + e_c^T (\delta + \gamma) + k\|e_c\| \big[tr\{\tilde{\mathbf{Z}}^T (\mathbf{Z} - \tilde{\mathbf{Z}})\} \big]
\end{aligned}
\tag{54}
$$

Considering:

$$
\begin{aligned}
tr\{\tilde{\mathbf{Z}}^T (\mathbf{Z} - \tilde{\mathbf{Z}})\} \big] &= \langle \tilde{\mathbf{Z}}, \mathbf{Z} \rangle_F - \|\tilde{\mathbf{Z}}\|_F^2 \\
&\leq \|\tilde{\mathbf{Z}}\|_F \|\mathbf{Z}\|_F - \|\tilde{\mathbf{Z}}\|_F^2
\end{aligned}
\tag{55}
$$

equation 54 becomes:

$$
\dot{V}_2 \leq -e_c^T \mathbf{K}_4 e_c + e_c^T (\delta + \gamma) + k\|e_c\| \big(\|\tilde{\mathbf{Z}}\|_F \|\mathbf{Z}\|_F - \|\tilde{\mathbf{Z}}\|_F^2 \big)
\tag{56}
$$

12

Now, using consideration 5.3, we have:

$$\dot{V}_2 \;\leq\; \begin{aligned}[t] &-e_c^T \mathbf{K}_4 e_c + \|e_c\|\|\delta\| - e_c^T e_c + \\ &-k\|e_c\|\|\tilde{\mathbf{Z}}\|(\|\tilde{\mathbf{Z}}\| - Z_M) - K_z(\|\tilde{\mathbf{Z}}\| + Z_M)\|e_c\|^2 \end{aligned} \tag{57}$$

and with lemma 5.2:

$$\dot{V}_2 \;\leq\; -\|e_c\|\Big[\mathbf{K}_4\|e_c\| + k\big(\|\tilde{\mathbf{Z}}\|^2 - \|\tilde{\mathbf{Z}}\|Z_M\big) - C_0 - C_1\|\tilde{\mathbf{Z}}\|\Big] - e_c^T e_c \tag{58}$$

Let the auxiliar constant C_3 be given by:

$$C_3 \triangleq \frac{1}{2}\Big(Z_M + \frac{C_1}{k}\Big) \tag{59}$$

Manipulating the terms in the brackets in equation 58, substituting 59 and square completing we obtain:

$$\dot{V}_2 \leq -\|e_c\|\Big[\mathbf{K}_4\|e_c\| + k\big(\|\tilde{\mathbf{Z}}\| - C_3\big)^2 - kC_3^2 - C_0\Big] - e_c^T e_c \tag{60}$$

For the function $\dot{V}_2$ be negative, we must guarantee that

$$\|e_c\| > \frac{k_3 C_3^2 + C_0}{\mathbf{K}_4} \tag{61}$$

or

$$\|\tilde{\mathbf{Z}}\| > C_3 + \sqrt{C_3^2 + \frac{C_0}{k}} \tag{62}$$
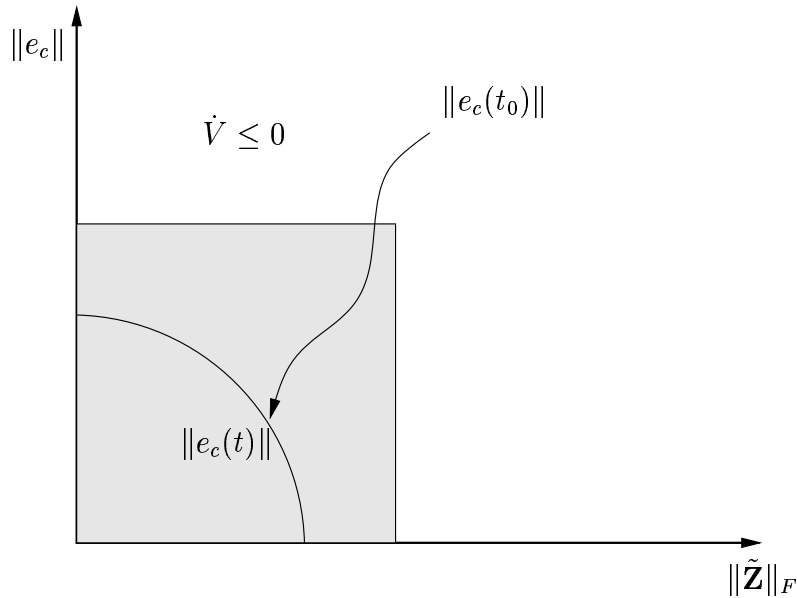
Figure 6: Region of stability to the neural network based controller.

Bearing in mind the Lyapunov theory and LaSalle theorem we show that $\|e_c\|$ and $\|\tilde{\mathbf{Z}}\|$ are uniformly locally stable (according to the figure 6).

It is possible to observe that inside the region delimited by equations 61 and 62 the function $\dot{V}_2$ isn't negative semi-definite. However, $\mathbf{K}_4$ can be such a suitable value for the tracking velocity error be as small as desired.

6   Simulation Results

The results presented in this section were obtained by simulation and in section 7 we present the experimental results.

The gains to the sliding mode kinematic controller are $a = 2.0$, $\xi = 20.0$ and $V_0 = 0.5(m/s)$. The initial posture of the robot is $x_0 = 2.0(m)$, $y_0 = 3.0(m)$ and $\theta_0 = 0.0(rd)$ and the reference posture is $x = 2.0(m)$, $y = 0.0(m)$ and $\theta = 0.0(rd)$.

Figure 7 shows the trajectory of x coordinate during the manouver and figure 8 shows the y coordinate trajectory. The orientation of the robot is in figure 9.
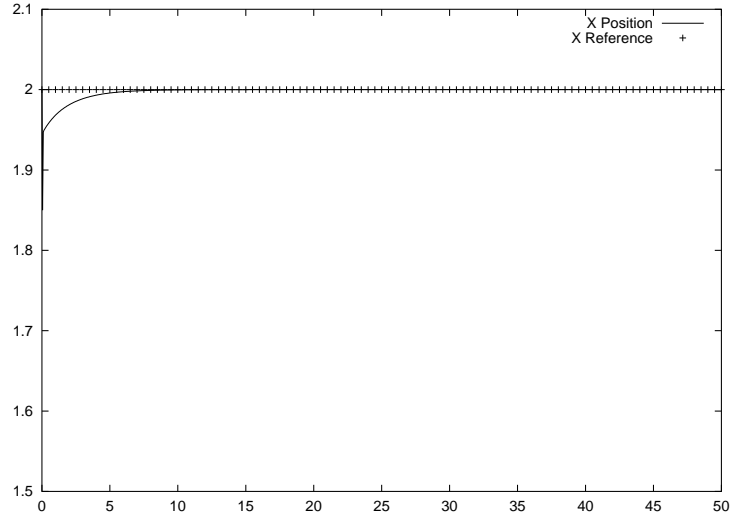
Figure 7: Position x of the robot (solid line) and its reference position (dotted line.
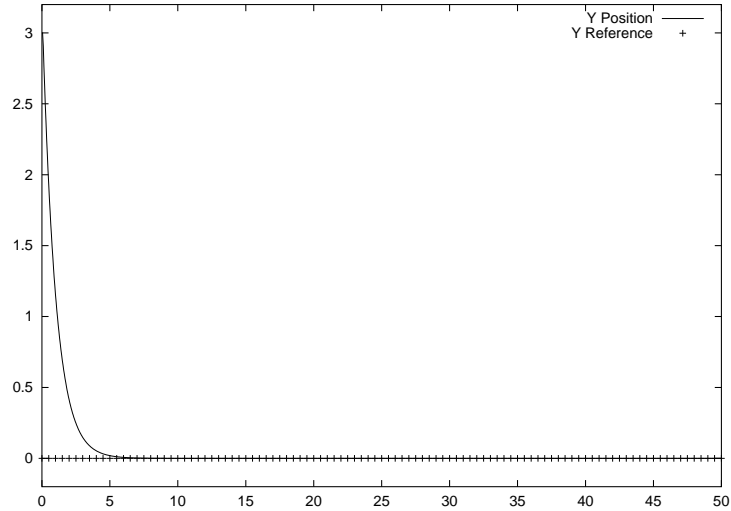
Figure 8: Positon y of the robot (solid line) and its reference y position (dotted line).
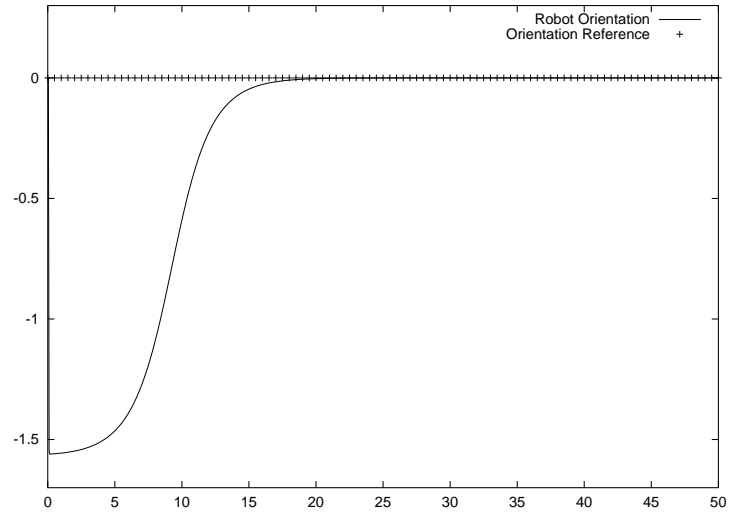
Figure 9: Robot orientation (solid line) and reference orientation (dotted line).

The trajectory described by the robot on the cartesian plan is shown in figure 10 and the velocities imposed to the robot can be analized on figure 11.
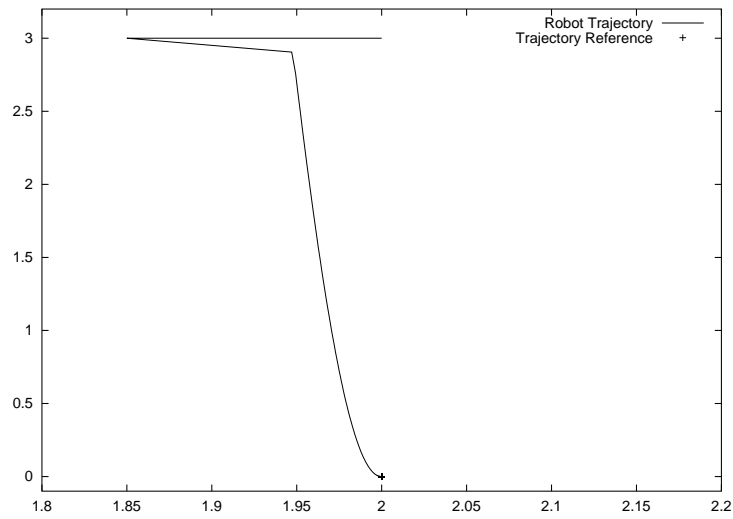

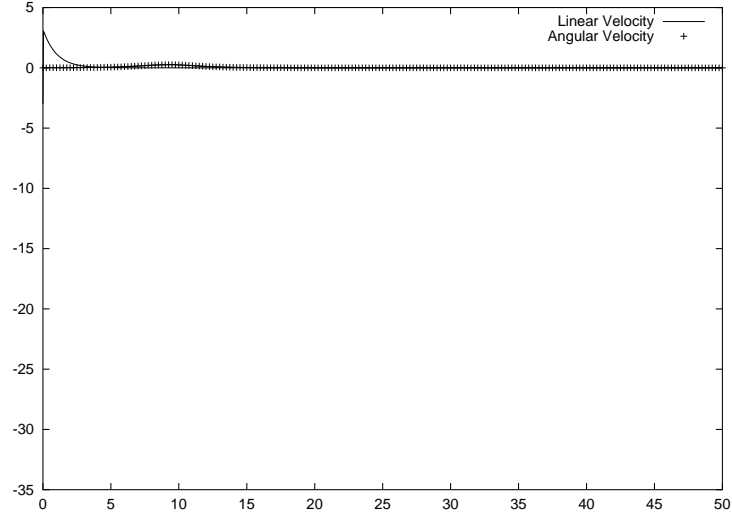Figure 10: Final trajectory of the robot to the reference posture.

Figure 11: Linear (solid line) and angular (dotted line) velocities of the robot.

7 Experimental Results

Finally we present the experimental results regarding the application of the sliding mode technique to the kinematic control loop and neural networks to the dynamic control loop. The objective is to track the robot to the origin of the system.

The initial position of the robot is $x_0 = 1.0(m)$, $y_0 = 1.0(m)$ and $\theta_0 = 0.0(rd)$. The gains of the kinematic controller are $a = 2.0$, $\xi = 10.0$ and $v_0 = 0.5(m/s)$. To the dynamic controller we have $K_4 = 50\,\mathbf{I}$, $k_z = 0.001$, $k = 0.01$, $F = 3.0$ and $G = 4.0$, where $\mathbf{I}$ is the identity matrix with appropriate dimensions.

In figure 12 we have the trajectory described by x coordinate of the robot during the movement to the origin of coordinate system. Next, in figure 13, is described the y coordinate traejctory. The orientation of the robot during the approximation to the origin can be visualized in figure 14.
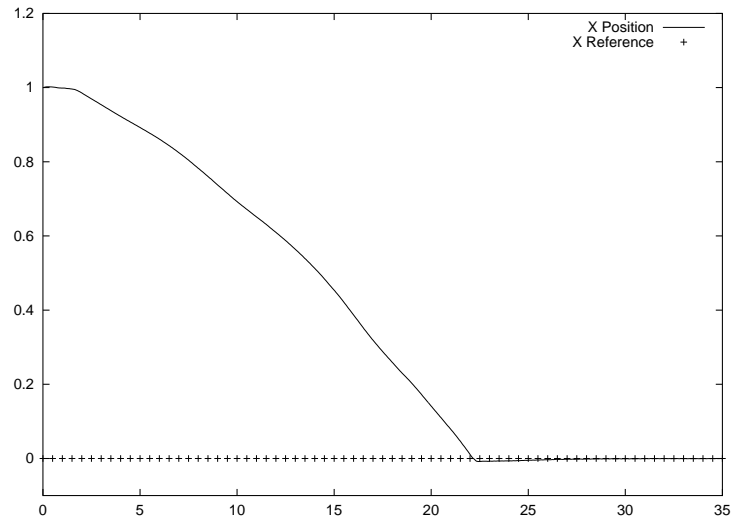

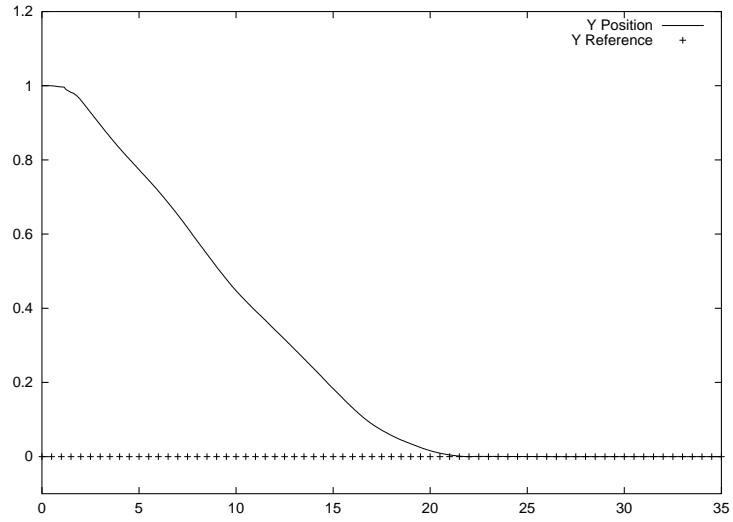Figure 12: Position x of the robot during the convergence to the origin.

16

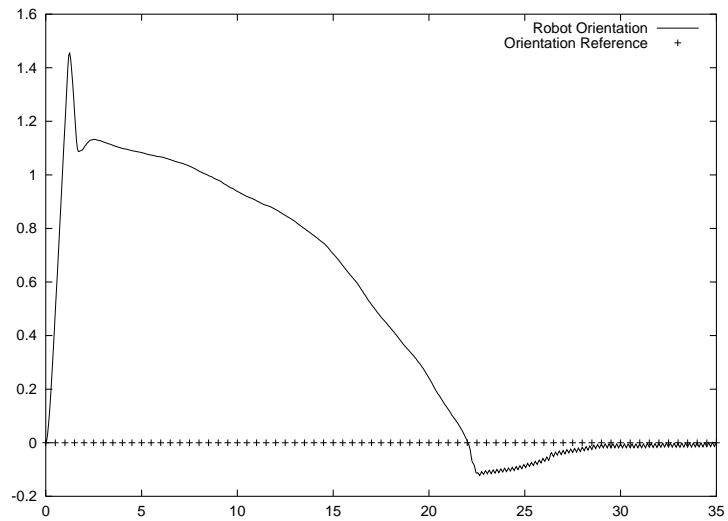Figure 13: Position y of the robot during the convergence to the origin.

Figure 14: Orientation of the robot.

The plannar trajectory executed by the robot to achieve the origin $(0, 0, 0)$ is in figure 15. The torques provided by the right (solid line) and left (dotted line) motors are depicted in figure 16.
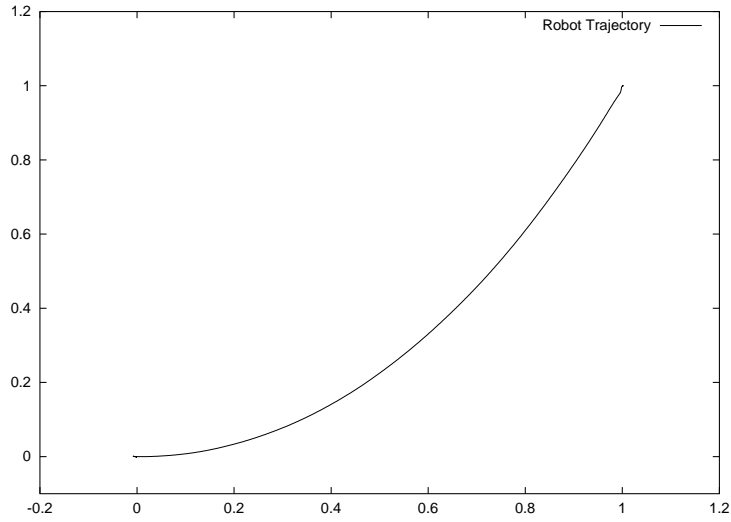
17

Figure 15: Final trajectory of the robot to achieve the origin of the coordinate system.
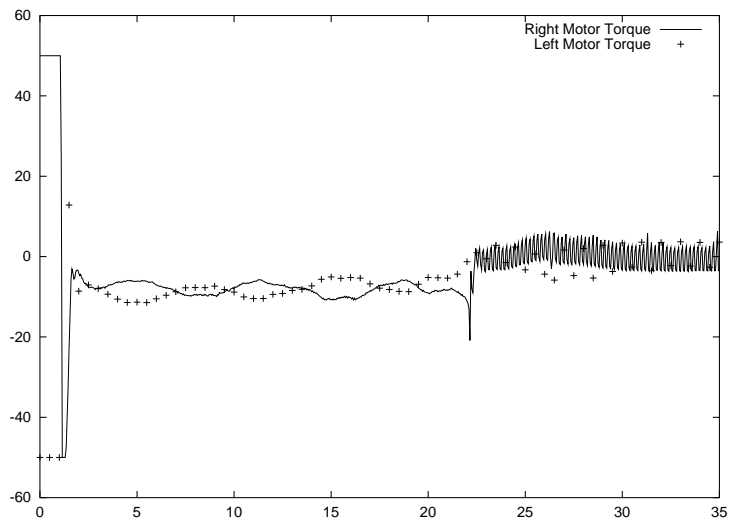

Figure 16: Torques provided by the right (solid line) and left (dotted line) motors.

References

[1] H. A. Yanco, A. Hazel, A. Peacock, S. Smith, and H. Wintermute. Initial report on wheelesley: a robotic wheelchair system. In *Proceedings of the Workshop on Developing AI Applications for the Disabled - International Joint Conference on Artificial Intelligence*, August 1995.

[2] H. A. Yanco. Integrating robotic research: A survey of robotic wheelchair development. In *AAAI Spring Symposium on Integrating Robotic Research*, March 1998.

[3] C. M. Angle and R. A. Brooks. Small planetary rovers. In *IEEE International Workshop on Intelligent Robots and Systems*, pages 383–388, July 1990.

[4] L. Hsu and R. R. Costa. Adaptive control with sliding modes: Theory and control. In *11º Congresso Brasileiro de Automtica*, pages 39–60, Setembro 1996. Mini-curso com durao de 6 horas.

[5] Y. Kanayama, Y. Kimura, F. Miyazaki, and T. Noguchi. A stable tracking control method for an autonomous mobile robot. In *Proceedings of the 1990 IEEE International Conference on Robotics and Automation*, pages 384–389, May 1990.

[6] G. Campion, G. Bastin, and B. D'Andra-Novel. Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, 12(1):47–62, February 1996.

[7] J.-M. Godhavn and O. Egeland. A lyapunov approach to exponential stabilization of nonholonomic systems in power form. *IEEE Transactions on Automatic Control*, 42(7):1028–1032, July 1997.

[8] A. C. Victorino and E. G. O. N. Paulo R. G. Kurka. Controle de trajetria e estabilizao de rob mvel no-holonmico com realimentao variante no tempo. In *Proceedings of XII Brazilian Automatic Control Conference*, volume II, pages 593–598, September 1998.

[9] S. Bentalba, A. E. Haijaji, and A. Rachid. Fuzzy sliding mode control of mobile robot. In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 4264–4265, December 1998.

[10] A. Astolfi. Discontinuous control of nonholonomic systems. Technical report, ETH, Disponível eletronicamente em ftp://ftp.aut.ee.ethz.ch/pub/reports/postscript/AUT95-09.ps, 1995.

[11] J.-B. Pomet. Explicit design of time-varying stabilizing control laws for a class of controllable systems without drift. *Systems & Control Letters*, 18:147–158, 1992.

[12] P. Morin and C. Samson. Control of nonlinear chained system: From the routh-hurwitz stability criteria to time-varying exponential stabilizers. *IEEE Transactions on Automatic Control*, 45(1):141–146, January 2000.

[13] J. E. Normey-Rico, J. G. Ortega, and I. Alcal. Control predictivo para seguimiento de caminos en un robot del tipo "synchro-drive". In *Proceedings of XII Brazilian Automatic Control Conference*, volume II, pages 611–616, September 1998.

[14] J. E. Normey-Rico, J. Gómez-Ortega, and E. F. Camacho. A smith-predictor-based generalised predictive controller for mobile robot path-tracking. *Pergamon - Control Engineering Practice - IFAC*, 7:729–740, 1999.

[15] Y. Yamamoto and X. Yun. Coordinating locomotion and manipulation of a mobile platform. *IEEE Transactions on Automatic Control*, 39(6):1326–1332, June 1994.

[16] Z.-P. Jiang and H. Nijmeijer. A recursive technique for tracking control of nonholonomic systems in chained form. *IEEE Transactions on Automatic Control*, 44(2):265–279, February 1999.

[17] W. F. Lages and E. M. Hemerly. Controle de robôs móveis utilizando transformação descontínua e linearização adaptativa. In *Anais do XIII Congresso Brasileiro de Automática*, pages 2251–2256, Setembro 2000.

[18] R. Fierro and F. L. Lewis. Control of nonholonomic mobile robot using neural networks. *IEEE Transactions on Neural Networks*, 9(4):589–601, July 1998.

[19] V. M. de Oliveira, E. R. de Pieri, and W. F. Lages. Feedforward control of a mobile robot using a neural network. In *Proceedings of the IEEE SMC*, October 2000.

[20] D. K. Chwa, J. H. Seo, P. Kim, and J. Y. Choi. Sliding mode tracking control of nonholonomic wheeled mobile robots. In *Proceedings of the American Control Conference*, pages 3991–3996, May 2002.

[21] W. F. Lages. *Controle e Estimao de Posio e Orientao de Robs Mveis*. PhD thesis, Instituto Tecnolgico de Aeronutica, 1998.

[22] Y. Yamamoto and X. Yun. Effect of the dynamic interaction on coordinated control of mobile manipulators. *IEEE Transactions on Robotics and Automation*, 12(5):816–824, October 1996.

[23] R. A. DeCarlo, S. H. Zak, and G. P. Matthews. Variable structure control of nonlinear multivariable systems: A tutorial. In *Proceedings of the IEEE*, volume 76, pages 212–232, March 1988.

[24] V. I. Utkin. Sliding mode control in dynamic systems. In *Proceedings of the 32nd Conference on Decision and Control*, pages 2446–2451, December 1993.

[25] J. Y. Hung, W. Gao, and J. C. Hung. Variable structure control: A survey. *IEEE Transactions on Industrial Electronics*, 40(1):2–22, February 1993.

[26] A. Bloch and S. Drakunov. Stabilization of a nonholonomic system via sliding modes. In *Proceedings of the 33rd Conference on Decision and Control*, pages 2961–2963, December 1994.

[27] W. Gao and J. C. Hung. Variable structure control of nonlinear systems: A new approach. *IEEE Transactions on Industrial Electronics*, 40(1):45–55, February 1993.

[28] R. A. DeCarlo, S. H. Žak, and S. V. Drakunov. *The Control Handbook*, chapter Variable Structure, Sliding-Mode Controller Design, pages 941–951. CRC Press, 1996.

[29] J. Guldner and V. I. Utkin. Stabilization of nonholonomic mobile robots using lyapunov functions for navigation and sliding modes control. In *Proceedings of the 33rd Conference on Decision and Control*, pages 2967–2972, December 1994.

[30] R. W. Brockett. *New directions in Applied Mathematics*, chapter Control Theory and Singular Riemannian geometry. Springer-Verlag, 1982.

[31] B. J. A. Kröse and P. P. van der Smagt. *An Introduction to Neural Networks*. Faculty of Mathematica & Computer Science - University of Amsterdam, Kruislaan 403, NL-1098 SJ Amsterdam / The Netherlands, fifth edition, January 1993.

[32] R. Żbikowski and A. Dzieliński. *Neural Network Engineering in Dynamic Control Systems*, chapter Neural Approximation: A Control Perspective. Advances in Industrial Control. Springer-Verlag, 1995.

[33] B. Kosko. *Neural Networks for Signal Processing*. Prentice-Hall, 1992.

[34] F. L. Lewis, A. Yesildirek, and K. Liu. Multilayer neural-net robot controller with guaranteed tracking performance. *IEEE Transactions on Neural Networks*, 7:1–12, 1996.