Neural Processing Letters **13:** 101–113, 2001.

Using a New Model of Recurrent Neural Network for Control

L. BOQUETE, L. M. BERGASA, R. BAREA, R. GARCÍA and M. MAZO
Electronics Department, Alcalá University, 28801, Spain; e-mail: boquete@depeca.alcala.es

Abstract. This paper shows the results obtained in controlling a mobile robot by means of local recurrent neural networks based on a radial basis function (RBF) type architecture. The model used has a Finite Impulse Response (FIR) filter feeding back each neuron's output to its own input, while using another FIR filter as a synaptic connection. The network parameters (coefficients of both filters) are adjusted by means of the gradient descent technique, thus obtaining the stability conditions of the process. As a practical application the system has been successfully used for controlling a wheelchair, using an architecture made up by a neurocontroller and a neuroidentifier. The role of the latter, connected up in parallel with the wheelchair, is to propagate the control error to the neurocontroller, thus cutting down the control error in each working cycle.

Key words: intelligent control, Lyapunov stability, radial basis function, recurrent neural network.

1. Introduction

Recurrent neural systems have demonstrated their usefulness in non-linear, time-variant systems. When setting up a recurrent neural network the difference between the various architectures resides in how to include the feedback in the network [1]: *externally* as in the networks Tapped Delay Line, the model of Elman [2] or the model of Narendra Parthasarathy (NARX neural network) [3] or *internally*. Within the latter group one option is a totally recurrent network formed by one or several neuron layers totally connected up to each other. But this type of architecture has serious drawbacks, such as great structural complexity and slow and laborious training [4]. This is because the models in question are very general and in principle valid for any type of dynamic system; in the case of specific problems, therefore, involving a certain previous knowledge, it is better to use simpler models. Other possibilities could be to include the feedback within each neuron, using FIR/IIR filters as synaptic connections [5], memory units [6], etc.

Past studies have shown that local recurrence architectures behave better and converge more quickly than totally connected-up networks. They also have certain stability and learning advantages. It is said that these models work well in systems that can be uncoupled into several low-dynamic-order systems, such as chaotic systems with recognizable periodicity and oscillatory models and in speech recognition with format content [7].

This paper presents a neural model based on RBF type architecture with local recurrence and weights formed by FIR filters. Both ideas arose from the works of Ku and Lee [8] and Ciocoiu [5]. The main contribution of this paper being the generalization of the model to systems with any number of inputs and two outputs, plus the practical application of said model.

The use of neural networks as control systems has recently been the subject of several scientific reports. One of these involved Etxebarria [9] using a linear network as the identifier for the adaptive control of discrete linear systems. Zhang et al. [10] used a direct controller in which, instead of calculating the plant Jacobian, the variation of the output over the input is obtained to control a ship. Yuan et al. [11] used an algorithm that is valid for functions of the type $y(k+1) = u(k) + f[y(k), y(k-1)]$. Maeda and Figueiredo [12] used a time-delay type neural network to control a two-link planar arm, using simultaneous perturbation without requiring information about the sensitivity function, while Noriega and Wang [13] also used time-delay type neural networks to implement a control architecture for unknown non-linear systems, minimizing an error function that includes forecasts of the future performance of the plant to be controlled (predictive control).

The main objective of our research was to set up an adaptive neural control system for controlling the movements of a wheelchair. This type of vehicle is characterized by its nonlinear dynamics, which changes with the working conditions: type of floor, state of batteries, weight of the person, etc. Preliminary works with linear controllers showed the control system have excess thrust in the initial moments of operation leading to abrupt movements of the wheelchair [14].

2. Neural Model

2.1. ARCHITECTURE AND TRAINING

The neural network used is a model with an architecture based on radial basis functions (RBF) with FIR filters for feedback and with additional FIR filters as synaptic connections (Figure 1). For a two-input ($x_1(k)$, $x_2(k)$) two-output ($y_{N1}(k)$, $y_{N2}(k)$) system, the model equations are

$$g_i(k) = e^{-\frac{(x_1(k)+x'_{i1}(k)-C_{i1})^2+(x_2(k)+x'_{i2}(k)-C_{i2})^2}{\sigma^2}}, \qquad i = 1, \ldots N. \tag{1}$$

The center of each function is $\boldsymbol{C_i} = (C_{i1}, C_{i2})$ and they are shared out evenly in the input space; σ is a constant modulating the activation zone of each neuron. The filter output acting as feedback for each neuron depends on the previous outputs of said neuron and the filter coefficients:

$$x'_{im} = \sum_{j=1}^{S} a_{imj}.g_i(k-j), \qquad m = 1, 2. \tag{2}$$

FIR filters are used as synaptic connections so that the previous outputs of each

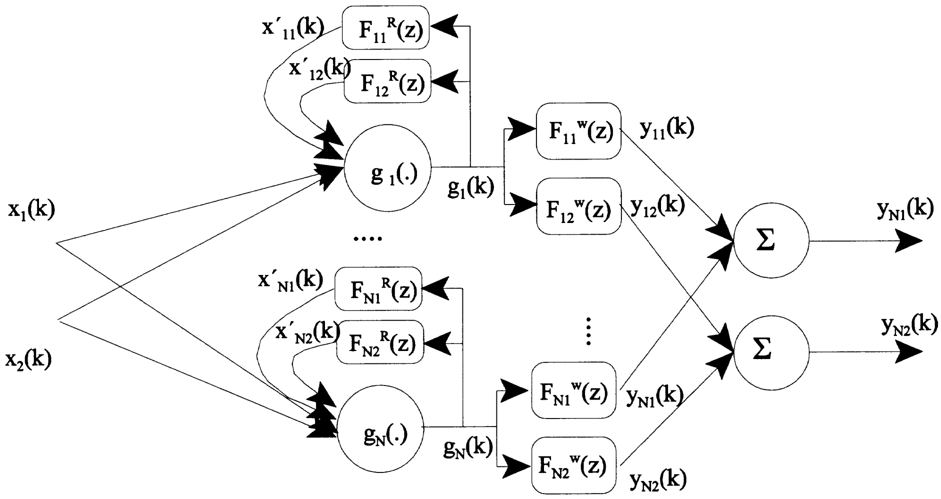

Figure 1. Neural network model.

neuron are taken into account:

$$y_{ip}(k) = \sum_{j=0}^{R-1} w_{ipj}(k).g_i(k-j). \tag{3}$$

Finally the outputs of the neural model are the sum of the filter outputs acting as synaptic connections:

$$y_{Np}(k) = \sum_{i=1}^{N} y_{ip}(k), \qquad p = 1, 2. \tag{4}$$

As can be seen the neural model used has local activation feedback in each neuron plus FIR-filter-based local synapse feedback according to the criterion indicated in [4]. The error function to be minimized is obtained from the following equation:

$$E(k) = \frac{1}{2}(y_{N1}(k) - y_{d1}(k))^2 + \frac{1}{2}(y_{N2}(k) - y_{d2}(k))^2. \tag{5}$$

The objective of the training phase is to vary the synaptic filter coefficients (w_{ipj}) and the feedback filter coefficients (a_{imj}) to minimize (5). To this end the gradient descent technique is used. For the first one:

$$\begin{aligned}\Delta w_{ipj}(k) &= -\alpha.\frac{\partial E(k)}{w_{ipj}(k)} = -\alpha.\frac{\partial E(k)}{\partial y_{Np}(k)}.\frac{\partial y_{Np}(k)}{\partial y_{ip}(k)}.\frac{\partial y_{ip}(k)}{\partial w_{ipj}(k)} \\ &= +\alpha.(y_{dp} - y_{Np}).g_i(k-j).\end{aligned} \tag{6}$$

For feedback filter coefficients:

$$\Delta a_{imj}(k) = -\alpha.\frac{\partial E(k)}{\partial a_{imj}(k)} = -\alpha.\left[\frac{\partial E(k)}{\partial y_{N1}(k)}.\frac{\partial y_{N1}(k)}{\partial g_i(k)}.\frac{\partial E(k)}{\partial y_{N2}(k)}.\frac{\partial y_{N2}(k)}{\partial g_i(k)}\right].\frac{\partial g_i(k)}{\partial a_{imj}(k)}$$
$$= +\alpha.[(y_{d1}(k) - y_{N1}(k)).w_{i10} + (y_{d2}(k) - y_{N2}(k)).w_{i20}] \times$$
$$\times \left[g_i(k-j) + \sum_{t=1}^{S} a_{imt}.\frac{\partial g_i(k-t)}{\partial a_{imj}}\right].\frac{\partial g_i(k)}{\partial x'_{im}(k)}. \tag{7}$$

As can be seen in the above equation, adjustment of the neuron's feedback filter coefficients has to be done by means of a recurrent expression that takes into account adjustments already made in working cycles before the moment k.

2.2. STABILITY

In this section a maximum in the value of the learning factor (α) is found in such a way that it ensures that the training error $E(k)$ decreases or at least does not increase. A vector $\boldsymbol{W}$ containing all the adjustable coefficients of the neural network is considered. It can be demonstrated that in a two-output neural network using the gradient descent technique, a sufficient condition for making sure that $E(k) - E(k-1) \leqslant 0$ is [15]:

$$0 < \alpha < \frac{1}{\parallel \frac{\partial y_{Np}(k)}{\partial \boldsymbol{W}(k)} \parallel^2}. \tag{8}$$

This expression has to be individualized for each neural model considered. In the model of Figure 1:

$$\boldsymbol{W} = [w_{110}, \ldots w_{N2(R-1)}, a_{111}, \ldots a_{N2S}]^T. \tag{9}$$

The total number of elements of $\boldsymbol{W}$ is $2NR + 2NS$ corresponding respectively to the synaptic filters and feedback filters. If all elements of matrix $\boldsymbol{W}$ are limited between $+1$ and -1, and considering the equations of the neural model:

$$\parallel \frac{\partial y_{Np}}{\partial w_{ipj}} \parallel_{\max} = \parallel g_i(k-j) \parallel_{\max} = 1 \tag{10}$$

$$\parallel \frac{\partial y_{Np}}{\partial a_{imj}} \parallel_{\max} = M_d.\frac{1}{1 - M_d.S}. \tag{11}$$

where

$$M_d = \max \parallel \frac{\partial g_i(k)}{\partial x'_{im}(k)} \parallel = \frac{\sqrt{2}}{\sigma}.e^{-\frac{1}{2}} \tag{12}$$

and the following condition also has to be observed:

$$M_d < \frac{1}{S}. \tag{13}$$

Thus the maximum value of the learning factor is:

$$0 < \alpha < \frac{1}{2NR + 2NS\left(\frac{M_d}{1 - S.M_d}\right)^2}. \tag{14}$$

It should be pointed out that this equation is valid only when using the neural network in isolation, for example when identifying a given dynamic process. As will be seen in the following sections, if it is used as a neurocontroller, the relevant adjustments need to be made to allow for the effect of the plant dynamics.

3. Neurocontrol of the Wheelchair

When using an inverse control system, in which the controller is a neural network, the problem is how to propagate the control error to the adjustable coefficients of the neurocontroller in such a way that the latter varies in the right direction, so that the error is reduced. In short, the problem is how to obtain the sensitivity of each plant output with respect to each input. This problem has been solved in different ways: thus, [8, 13, 16], use a neuroidentifier in parallel with the physical system to be controlled which serves as a path for the propagation of the error. This neuroidentifier may be a recurrent neural network or a 'feed-forward' network with inputs of different moments of time. Zhang et al. [10] use only the sign produced in each output when an input varies, as this information suffices for evaluating in which direction each of the neurocontroller's coefficients need to be adjusted. Acosta et al. [17] calculate this sensitivity by finding the relation between the plant input at two consecutive moments of time and the output variation produced by said variation, i.e.:

$$\frac{\partial y(k)}{\partial u(k)} \cong \frac{y(k) - y(k-1)}{u(k-1) - u(k-2)}. \tag{15}$$

Another possibility is that used by [12], who obtains said sensitivity by increasing each one of the neurocontroller's adjustable coefficients and making the corresponding observation of the variations in each one of the outputs of the plant to be controlled, thereby estimating the Jacobian of the plant.

In this paper a neuroidentifier in parallel with the wheelchair is used (Figure 2), its mission being to propagate the control error to the neurocontroller, always providing that the identification error is negligible. Both the neuroidentifier and the neurocontroller are designed on the basis of the neural model explained in point 2 (Neural Model).

3.1. CHARACTERISTICS OF THE WHEELCHAIR

The wheelchair used in the practical tests, is a commercial model which has been equipped with a sensorial system (ultrasound sensors, infrared sensing devices, cameras, etc.) which facilitates its guidance. There are also different user-operated control modes (joystick, vocal commands, air expulsion, eye movements) and various user interfaces. The mechanical structure of the wheelchair consists of a platform (measuring $100 \times 80 \times 58$ cm, and weighing approximately 35 Kg) on two motor wheels and two idle wheels. The motor wheels, with a radius $R_d = 16$ cm and separated by a distance $D = 54$ cm, have independent traction provided by two DC motors.

There is a low-level control loop governing the electronic system of the DC traction motors. This system is implemented with a PID and its mission is to ensure that the turning speed of the right- and left-hand wheels (w_R, w_L) is approximately that indicated on the electronic control cards (w'_R, w'_L):

$$\begin{aligned} w'_R &\simeq w_R \\ w'_L &\simeq w_L \end{aligned} \tag{16}$$

Given that this control loop is not sufficient in itself to ensure reliability in the wheelchair movements [14], another external loop is needed (neural control) to govern adequately the linear speed ($V(k)$) and angular speed ($\Omega(k)$) of the mobile robot. This external control loop acts on the inputs of the PID controller (w'_R, w'_L).

3.2. CONTROL SCHEME

Figure 2 shows the classic model-oriented control structure, which uses a neurocontroller (this generates the control signal ($\boldsymbol{U}(k)$) on the basis of the reference signals ($V_d(k)$, $\Omega_d(k)$), a reference model (with output $V_m(k)$, $\Omega_d(k)$), an adjustment algorithm for minimizing the control error $E^c(k)$ and the identification error $E^i(k)$ and a system for converting the angular speed of each one of the two wheels to the linear and angular speeds of the wheelchair. The variables involved are:

$$\boldsymbol{Y}(k) = \begin{bmatrix} V(k) \\ \Omega(k) \end{bmatrix} \quad \boldsymbol{R}(k) = \begin{bmatrix} V_d(k) \\ \Omega_d(k) \end{bmatrix} \quad \boldsymbol{U}(k) = \begin{bmatrix} w'_R(k) \\ w'_L(k) \end{bmatrix} \quad \boldsymbol{Y}^i_N(k) = \begin{bmatrix} y^i_{N1}(k) \\ y^i_{N2}(k) \end{bmatrix}$$
$$\boldsymbol{Y}_m(k) = \begin{bmatrix} V_m(k) \\ \Omega_m(k) \end{bmatrix}. \tag{17}$$

Where $\boldsymbol{U}(k)$ is the neurocontroller output and also the input of the plant and neuroidentifier. The advantage of this set-up is that no assumption of the wheelchair's dynamic is called for, so the same scheme can be valid for controlling any other two-input, two-output system.

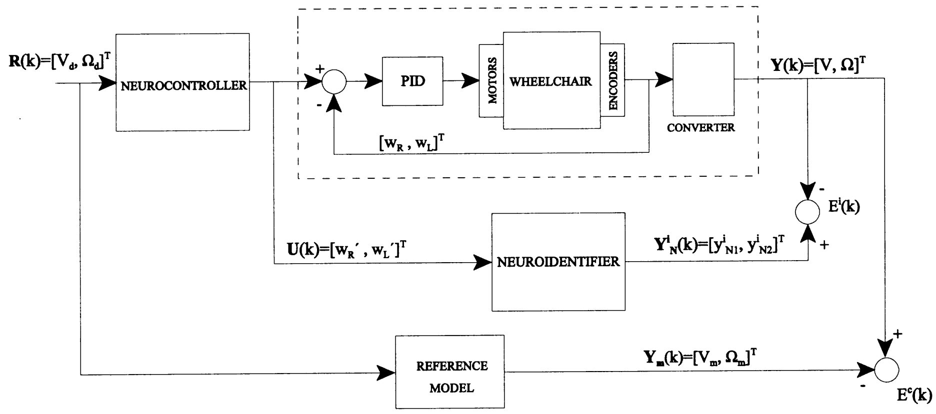

Figure 2. Control system implemented.

In each working cycle (k), two functions must be minimized; the identification error $E^i(k)$ and the control error $E^c(k)$. The former is defined as:

$$E^i(k) = \frac{1}{2}(y^i_{N1}(k) - V(k))^2 + \frac{1}{2}(y^i_{N2}(k) - \Omega(k))^2, \tag{18}$$

the superscript 'i' indicates neuroidentifier coefficients. The neuroidentifier is adjusted in each working cycle by means of (6) and (7), considering the linear speed $(V(k))$ and angular speed $(\Omega(k))$ of the wheelchair to be the desired network outputs.

The control error function to be minimized is the difference between the real output of the plant $(\boldsymbol{Y}(k))$ and the output given by the reference model:

$$E^c(k) = \frac{1}{2}(V(k) - V_m(k))^2 + \frac{1}{2}(\Omega(k) - \Omega_m(k))^2. \tag{19}$$

This error has to be propagated by the dynamic of wheelchair to the neurocontroller, so that the latter's coefficients are adjusted by means of the equations:

$$\begin{aligned} \Delta w^c_{ipj}(k) = -\alpha.\frac{\partial E^c(k)}{\partial w^c_{ipj}(k)} &= -\alpha.[e^c_1(k).J_{11}(k) + e^c_2(k).J_{12}(k)].\frac{\partial w'_r(k)}{\partial w^c_{ipj}(k)} \\ &- \alpha.[e^c_1(k).J_{21}(k) + e^c_2(k).J_{22}(k)].\frac{\partial w'_L(k)}{\partial w^c_{ipj}(k)} \end{aligned} \tag{20}$$

$$\begin{aligned} \Delta a^c_{imj}(k) = -\alpha.\frac{\partial E^c(k)}{\partial a^c_{imj}(k)} &= -\alpha.[e^c_1(k).J_{11}(k) + e^c_2(k).J_{12}(k)].\frac{\partial w'_r(k)}{\partial a^c_{imj}(k)} \\ &- \alpha.[e^c_1(k).J_{21}(k) + e^c_2(k).J_{22}(k)].\frac{\partial w'_L(k)}{\partial w^c_{imj}(k)} \end{aligned} \tag{21}$$

where

$$e_1^c(k) = (V(k) - V_m(k)); \qquad e_2^c(k) = (\Omega(k) - \Omega_m(k)) \tag{22}$$

and:

$$\begin{bmatrix} J_{11}(k) & J_{12}(k) \\ J_{21}(k) & J_{22}(k) \end{bmatrix} = \begin{bmatrix} \frac{\partial V}{\partial w'_R}(k) & \frac{\partial \Omega}{\partial w'_R}(k) \\ \frac{\partial V}{\partial w'_L}(k) & \frac{\partial \Omega}{\partial w'_L}(k) \end{bmatrix}. \tag{23}$$

When the identification error is negligible ($E^i(k) \approx 0$), each of the elements of (23) can be obtained from the neuroidentifier model, according to the following equation:

$$J_{mp}(k) = -2. \sum_{i=1}^{N^i} \frac{x_m^i + x'^i_{im} - C^i_{im}}{\sigma^{i^2}} .w^i_{ip0}.g^i_i(k). \tag{24}$$

3.3. STABILITY

Applying the result indicated by Equation (8) and considering the neurocontroller + neuroidentifier unit to be a single neural network, so in each learning cycle the coefficients of the neurocontroller and the neuroidentifier are adjusted, then, for stability purposes, consideration has to be given to the following vector $\boldsymbol{W}$

$$\boldsymbol{W} = [w^c_{110}, \ldots w^c_{N^c 2(R^c-1)}, a^c_{111}, \ldots a_{N^c 2S^c}, w^i_{110}, \ldots w^i_{N^i 2(R^i-1)}, a^i_{111}, \ldots a^i_{N^i 2S^i}]^T. \tag{25}$$

The number of elements of vector $\boldsymbol{W}$ is $2N^cR^c + 2N^cS^c + 2N^iR^i + 2N^iS^i$, corresponding to the neurocontroller and neuroidentifier respectively.

From the Equations (10) and (11):

$$\| \frac{\partial y^i_{Np}}{\partial w^i_{ipj}} \|_{\max} = \| g^i_i(k-j) \|_{\max} = 1 \tag{26}$$

$$\| \frac{\partial y^i_{Np}}{\partial a^i_{imj}} \|_{\max} = M^i_d . \frac{1}{1 - M^i_d.S^i}. \tag{27}$$

To obtain the maximum variation of the neurocontroller coefficients, consideration has to be given to the effect of the neuroidentifier, given by Equation (23):

$$\| \frac{\partial y^i_{N1}}{\partial w^c_{ipj}} \|_{\max} = \| J_{p1}(k) \|_{\max} = M^i_d.N^i \tag{28}$$

$$\| \frac{\partial y^i_{N1}}{\partial a^c_{imj}} \|_{\max} = M^c_d . \frac{1}{1 - M^c_d.S^c} .2.M^i_d.N^i. \tag{29}$$

The following conditions need to be met:

$$M_d^c < \frac{1}{S^c}; \quad \| w_{ipj}^c \| < 1; \quad M_d^i < \frac{1}{S^i}; \quad \| w_{ipj}^i \| < 1. \tag{30}$$

In short, the maximum value of the learning factor to be used in the control scheme of Figure 2 is:

$$0 < \alpha < \frac{1}{2N^iR^i + 2N^i.S^i.\left(\frac{M_d^i}{1-M_d^i.S^i}\right)^2 + 2N^cR^c(M_d^i.N^i)^2 + 2N^c.S^c.\left(\frac{M_d^c}{1-M_d^i.S^c}.2M_d^iN^i\right)^2}. \tag{31}$$

4. Practical Trials

Three examples are shown of the wheelchair's behavior with the neural control system: movement in a straight line at constant speed (Figure 3), movement in a straight line with speed in triangular form (Figure 4) and an example (Figure 5) in which the wheelchair describes a circular trajectory with a radius of 1 m. ($V = 37.7$ m/s, $\Omega = 0.38$ rd/s). In all the examples the coefficients of both the neurocontroller and the neuroidentifier start from random values at the beginning

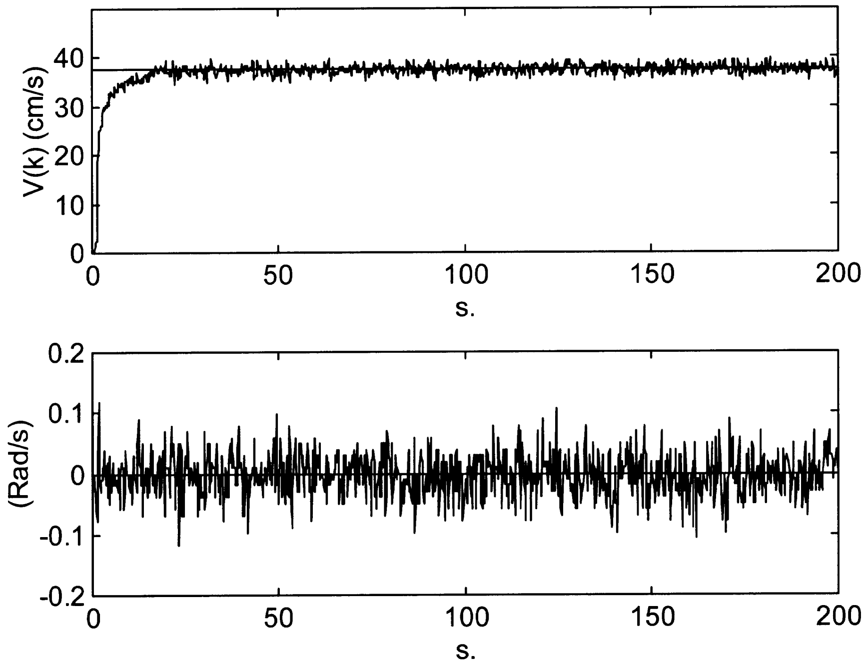

Figure 3. Advancing in a straight line.

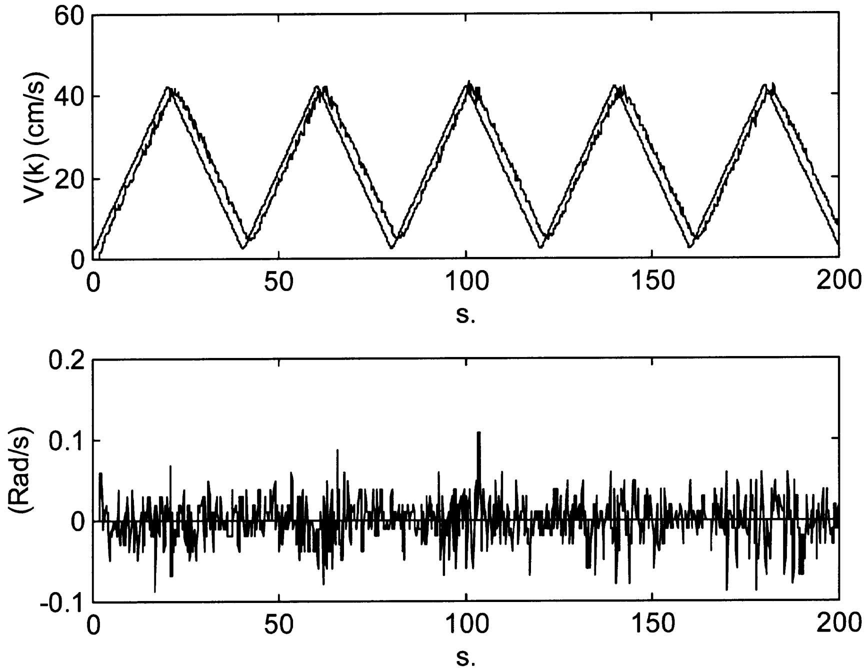

Figure 4. Linear speed in triangular form.

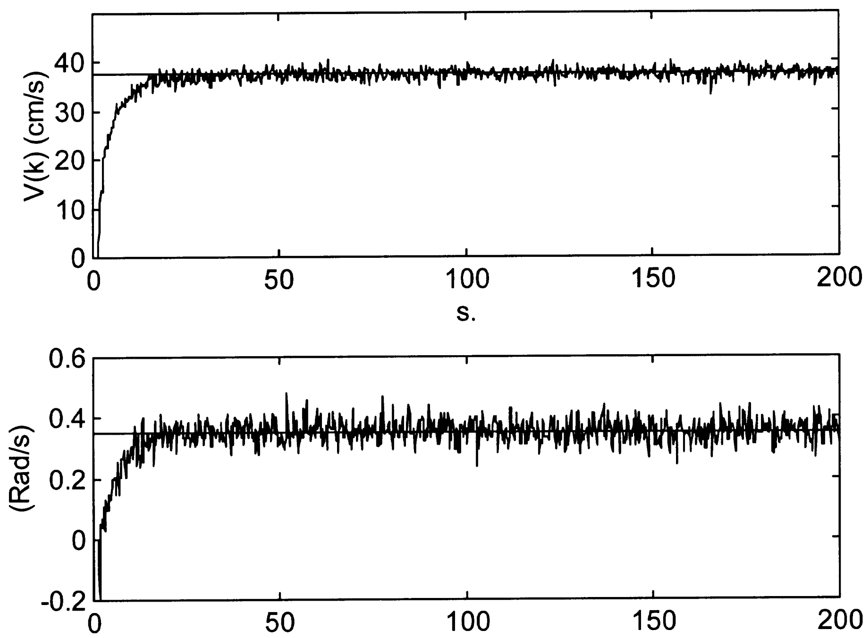

Figure 5. Tracing a circumference.

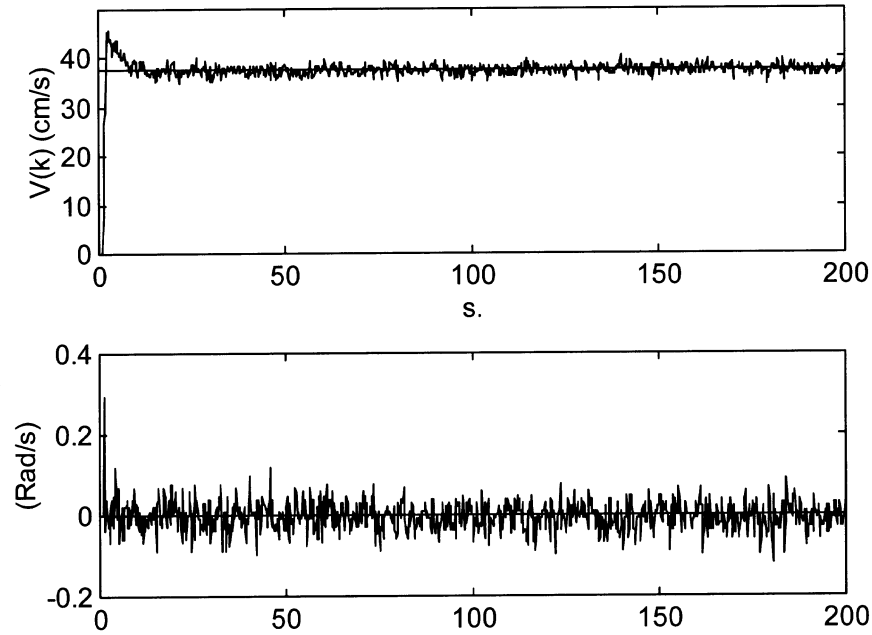

Figure 6. Response with overshooting.

Table I. Parameters of neuroidentifier and neurocontroller.

Neurocontroller	$N^c = 9$	$R^c = 1$	$S^c = 2$	$\sigma^c = 2.2$
Neuroidentifier	$N^i = 9$	$R^i = 2$	$S^i = 2$	$\sigma^i = 2.2$

of the sampling. The outcome therefore shows the time needed by the algorithm to adapt to the working conditions. The following reference model has been used [3]:

$$\begin{aligned} V_m(k) &= \beta . V_m(k-1) + V_d(k) \\ \Omega_m(k) &= \beta . \Omega_m(k-1) + \Omega_d(k) \\ |\beta| &= 0.7 < 1. \end{aligned} \tag{32}$$

The configuration of the architecture used corresponds to the data shown in Table I and the learning factor was $\alpha = 1/5800$.

All the experiments were carried out with a person weighing about 50 kg sitting in the wheelchair. The graphs show that the various speeds evolve smoothly and without overshooting from initial to final values, although quicker responses can be obtained in the transitory part of the bends by increasing the value of the learning factor (α) or the number of neurons (N^c) or the number of adjustable parameters of the neurocontroller (S^c, R^c), all this at the cost of system stability, as indicated by Equation (31). For example Figure 6 shows an example in which the configuration indicated in Table I has been used, but with $N^c = 16$, the above-mentioned effect being quite noticeable.

5. Conclusions

This paper shows a practical case of controlling the movements of a real system (a wheelchair) by means of a neural system using a new recurrent neural network architecture based on a RBF model. Equations have been obtained for the adjustment of the coefficients (FIR filters) and stability conditions for 2-output models. The system was then tested in practice for guiding the movements of a wheelchair, whereby several trials conducted under real working conditions have proven that it works correctly.

Acknowledgements

This work has been carried out thanks to the grants received from CICYT (Interministerial Science and Technology Committee-Spain) for project TER96-1957-C03-01 and for project E013/99 received from Alcalá University.

References

1. Campolucci, P., Uncini, A., Piazza, F. and Rao, B. D.: On-line learning algorithms for locally recurrent neural networks, *IEEE Transactions on Neural Networks* **10**(2) (1999), 253–271.
2. Elman, J. L.: Finding structure in time, *Cognitive Science* **14** (1990), 179–211.
3. Narendra, K. S. and Parthasarathy, K.: Identification and control of dynamical systems using neural networks, *IEEE Transactions on Neural Networks* **1**(1) (1990), 4–27.
4. Tsoi, A. C. and Black, A. D.: Locally recurrent globally feedforward networks: a critical review of architectures, *IEEE Transactions on Neural Networks* **5**(2) (1994), 229–239.
5. Ciocoiu, I. B.: Radial base function networks with FIR/IIR synapses, *Neural Processing Letters* **3** (1996), 17–22.
6. Sastry, P. S., Santharam, G. and Unnikrishnan, K. P.: Memory neuron networks for identification and control of dynamical systems, *IEEE Transactions on Neural Networks* **5**(2) (1994), 30–319.
7. Sivakumar, S. C., Robertson, W. and Phillips, W. J.: On-line stabilitation of block-diagonal recurrent neural networks, *IEEE Transactions on Neural Networks* **10**(1) (1999), 167–175.
8. Ku, C. C. and Lee, K. Y.: Diagonal recurrent neural networks for dynamic systems control, *IEEE Transactions on Neural Networks* **6**(1) (1995), 144–156.
9. Etxebarria, V.: Adaptive control of discrete systems using neural networks, *IEE Proc. Control Theory Appl.* **141**(4) (1994), 209–215.
10. Zhang, Y., Sen, P. and Hearn, G. E.: An on-line trained adaptive neural controller, *IEEE Control Systems* **15**(5) (1995), 67–75.
11. Yuan, M., Poo, A. N. and Hong, G. S.: Direct neural control system: nonlinear extension of adaptive control, *IEE Proc. Control Theory Appl.* **142**(6) (1995), 661–667.
12. Maeda, Y. and Figueiredo, R. J. P.: Learning rules for neuro-controller via simultaneous perturbation, *IEEE Transactions on Neural Networks* **8**(5) (1997), 1119–1130.
13. Noriega, J. R. and Wang, H.: A direct adaptive neural-network control for unknown nonlinear systems and its application, *IEEE Transactions on Neural Networks* **9**(1) (1998), 27–34.

14. Boquete, L., García, R., Barea, R. and Mazo, M.: Neural control of the movements of a wheelchair, *J. Intelligent Robotic Systems* **25** (1999), 213–226.
15. Boquete, L., Barea, R., García, R., Mazo, M. and Espinosa, F.: Identification and control of a wheelchair using recurrent neural networks, *Engineering Applications of Artificial Intelligence* **12**(4) (1999), 443–452.
16. Hunt, K. J. and Sbarbaro, D.: Neural networks for non linear internal model control, *IEE Proceedings-d* **138**(5) (1991), 431–438.
17. Acosta, L., Méndez, J. A., Torres, S., Moreno, L. and Marichal, G. N.: On the design and implementation of a neuromorphic selft-tuning controller, *Neural Processing Letters* **9**(3) (1999), 229–242.