

# Identification of a Class of Nonlinear Systems by a Continuous-Time Recurrent Neurofuzzy Network

Marcos A. Gonzalez-Olvera and Yu Tang

**Abstract**—In this paper we present a new continuous-time recurrent neurofuzzy network structure for modeling and identification of a class of nonlinear systems, using a training algorithm motivated from previous works in adaptive observers. Using only output measurements and the knowledge of an excitation input signal, the proposed network is trained by generating estimates of an ideal network and jointly identifying its parameters. The objective is to make the network to dynamically behave as the plant. The stability of the network and the convergence of the training algorithm are established based on the Lyapunov stability theory. Two numerical examples and an experimental result are included to demonstrate the effectiveness of the proposed method.

**Keywords:** System Identification, Nonlinear Systems, Neural Networks, Lyapunov stability.

## I. INTRODUCTION

Neurofuzzy networks have been used for modeling and designing control systems for their enhanced learning and function approximation capabilities by combining the advantages of both of them [1], [2], [3]. These structures have been used not only as static input-output functions, but also as dynamic models. These latter are known as *recurrent* or *dynamic* networks, where feedback connections in some or all layers within the network are present. They have been used to model and identify dynamic systems. One of the very first approaches was given by [4], used to learn the dynamic behavior of certain computational systems.

The structure of neural networks is defined according to the problem in which they are used. For example, [2] proposed networks with feedback within the internal layers for identification and control of nonlinear systems, [5] used time-delay networks and [6] proposed fully recurrent single-hidden layer networks, where all layers are considered to be inputs. Networks such as globally static-locally recurrent structures have been proposed by [5], [7], [8], using neurons with linear filters in the synapsis.

As pointed out by [9], these structures do not imply stability in the training algorithms. Common approaches, such as Backpropagation-Trough-Time [10] and Real-time recurrent learning [6], have been used for network training; but these algorithms tend to be slow and lack stability analysis. So, linear approximations such as recursive-least-

squares [11], [12] and Kalman filters [13] have been reported and successfully applied.

Low convergence speed and parameter number explosion (the *curse of dimensionality*) are challenges to be faced during training. Some research works have proposed simpler structures by focusing on the specific task to be analyzed: on control and identification tasks, [14], [15] proposed externally recurrent networks; while [16] studied structures with internal feedback via fuzzy systems and [9] proposed input-output neurofuzzy networks where the consequent part of each rule is itself a neural network function. However, many of these structures do need the direct measurement of the actual states of the system, which may not be available in a practical situation.

State-space representations are preferred in some situations for control design. In this sense, internal states to model the dynamics have been considered. For example, [17], [18] have presented stable training algorithms that require the direct measurement of the states of the plant to be identified. In a different approach, [3] required only the information of the output of the plant, thus generating internal state variables (not related to those of the system), but stable training is not guaranteed.

In a previous work [19], we have focused on a Controllable-Form Recurrent NeuroFuzzy Network (CReNN) in discrete-time, using the delayed outputs as state variables of the network with a Least-Squares-based training. In this paper, we continue this research line by presenting a continuous-time neurofuzzy network that identifies the input/output dynamics of a plant based on a state-space model, and keeps the convergence of the parameters and the ultimate boundness of the identification error when the persistent excitation condition is fulfilled. The novelty of the presented structure relies on both the structure and the training algorithm. The network considers an output feedback into the state equation, and the training is based in the work on adaptive observer design [20], [21], [22]. So, the proposed network can identify a class of nonlinear systems using only output measurements, while the training algorithm estimates the states of an *ideal* network such that minimizes the identification error, jointly with parametric identification.

The rest of the paper is organized in the following sections: in Section II the identification problem is analyzed, and a proposed identifier structure is presented in Section III, where also the boundedness and convergence of the training algorithm are discussed. Examples of the performance of the neural identifier are shown in Section IV by identifying a

This work is partly supported by UNAM-PAPIIT IN120009. The work of Marcos A. González Olvera is supported by CONACyT.

Marcos A. Gonzalez-Olvera and Yu Tang (corresponding author) are with Faculty of Electrical Engineering, National Autonomous University of Mexico, Mexico City, Mexico. [mangel@verona.fi-p.unam.mx](mailto:mangel@verona.fi-p.unam.mx), [tang@servidor.unam.mx](mailto:tang@servidor.unam.mx). Tel (52)-55-56234142

pair of nonlinear systems, including an experiment in a real system with nonlinearities. Finally, conclusions and future work are drawn in Section V.

## II. PROBLEM STATEMENT

Consider the following single-input-single-output system

$$\begin{aligned}\dot{\mathbf{x}}(t) &= f(\mathbf{x}, u), \\ y(t) &= h(\mathbf{x}),\end{aligned}\quad (1)$$

where  $u(t)$  and  $y(t)$  are the input and the output, respectively, and  $\mathbf{x} \in \mathbb{R}^n$  is the state of the system,  $f : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$ ,  $h : \mathbb{R}^n \rightarrow \mathbb{R}$ , are smooth functions. We assume that (1) is observable in the region on interest [21].

We consider the class of nonlinear systems transformable to

$$\begin{aligned}\dot{\mathbf{z}} &= A\mathbf{z} + \bar{f}(y, u), \\ y &= C\mathbf{z},\end{aligned}\quad (2)$$

with  $(A, C)$  observable, and  $\bar{f}(y, u)$  smooth.

In [21], the sufficient and necessary conditions are given in order to transform (1) through a local diffeomorphism  $\mathbf{z} = \Phi(\mathbf{x})$  into the system (2). We assume that these conditions are fulfilled here. The objective is to find a continuous-time recurrent neural network described by

$$\begin{aligned}\dot{\eta} &= A\eta + \varphi(y, u)\theta_\eta, \\ y &= C\eta,\end{aligned}\quad (3)$$

such that, by using only the input signal  $u(t)$  and output measurement  $y(t)$ , can generate a dynamic mapping  $u \rightarrow \hat{y}$ , such that  $\sup_{t \geq 0} |\hat{y}(t) - y(t)|$  is minimized tuning the network parameter  $\theta_\eta$ , as illustrated in Fig. 1.

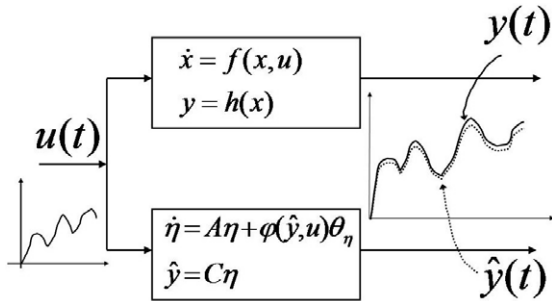


Fig. 1. Identification objective

Notice that the function  $\varphi(y, u)\theta_\eta$  is intended to approximate  $\bar{f}$  in (2). We will approach this problem by borrowing ideas from adaptive observer design [20], [21], [22], where under observability conditions of  $(A, C)$  and a persistent excitation for  $\varphi(y, u)$ , an adaptive observer can be designed such that jointly estimates  $\hat{\eta}$  and  $\theta_\eta$  via a decoupled algorithm [22].

## III. PROPOSED RECURRENT NEUROFUZZY NETWORK AND TRAINING ALGORITHM

### A. Network structure

We propose the following neurofuzzy network, described by the  $r$  fuzzy rules, with  $r = n$

$$\begin{aligned}R_1: & \text{ If } y \text{ is } F_1^y \text{ AND } u \text{ is } F_1^u \text{ then} \\ & \dot{\eta}^1 = A\eta + \begin{bmatrix} \theta_{11} + \theta_{12}u + \theta_{13}\rho(y) \\ 0_{r-1 \times 1} \end{bmatrix}, \\ & \vdots \\ R_i: & \text{ If } y \text{ is } F_i^y \text{ AND } u \text{ is } F_i^u \text{ then} \\ & \dot{\eta}^i = A\eta + \begin{bmatrix} 0_{(i-1) \times 1} \\ \theta_{i1} + \theta_{i2}u + \theta_{i3}\rho(y) \\ 0_{r-i \times 1} \end{bmatrix}, \\ & \vdots \\ R_r: & \text{ If } y \text{ is } F_r^y \text{ AND } u \text{ is } F_r^u \text{ then} \\ & \dot{\eta}^r = A\eta + \begin{bmatrix} 0_{(r-1) \times 1} \\ \theta_{r1} + \theta_{r2}u + \theta_{r3}\rho(y) \end{bmatrix},\end{aligned}\quad (4)$$

where  $A$  is a Hurwitz matrix and  $\rho(y)$  is a monotonic increasing sigmoid function, such that  $\lim_{y \rightarrow \pm\infty} \rho(y) = \pm 1$ ,  $\rho(0) = 0$ . The fuzzy sets for  $u$  and  $y$  are characterized the membership functions  $F_i^y = \exp(-\sigma_{yi}^2(y - \mu_{yi})^2)$ ,  $F_i^u = \exp(-\sigma_{ui}^2(u - \mu_{ui})^2)$ , so the activation value of each rule is  $F_i = F_i^y F_i^u = \exp(-\sigma_{yi}^2(y - \mu_{yi})^2 - \sigma_{ui}^2(u - \mu_{ui})^2)$  [23]. If the defuzzification is defined as  $\dot{\eta} = \sum_{i=1}^r F_i \dot{\eta}^i$ , the regressor can be expressed as

$$\varphi(y, u) = \begin{bmatrix} \text{diag}(F) & \text{diag}(F)u & \text{diag}(F)\rho(y) \end{bmatrix} \in \mathbb{R}^{n \times 3r}, \quad (5)$$

where  $F = [F_1 \ \dots \ F_r] \in \mathbb{R}^r$ . Consequently,  $\theta_\eta \in \mathbb{R}^{3r \times 1}$ .

*Remark 1:* It can be noticed that through the transformation from (1) to (2), the regressor is a function of the plant input and output. The consequent part in the above fuzzy rules is designed in such a way to avoid parameter explosion and to meet the persistent excitation condition (see below). However, other type of fuzzy rules may be considered, as long as they comply with the persistent excitation condition.

### B. Training algorithm

Let the network defined by (3) to work as a parallel model for (2). The problem is then to jointly estimate  $\eta$  and  $\theta_\eta$ . As the pair  $(A, C)$  is observable, a vector  $K \in \mathbb{R}^n$  can be found such that the eigenvalues of  $A - KC$  are in any desired position in the complex plane. Following this, an output injection is considered for the state equation of (3) in the form

$$\dot{\eta} = (A - KC)\eta + \varphi\theta_\eta + Ky.$$

Following a similar treatment as [22], the states of this *ideal* network can be decomposed in

$$\eta = \eta_y + \eta_\varphi, \quad (6)$$

where

$$\dot{\eta}_y = (A - KC)\eta_y + Ky, \quad (7)$$

$$\dot{\eta}_\varphi = (A - KC)\eta_\varphi + \varphi\theta_\eta. \quad (8)$$

Clearly, an estimator for  $\eta_y$  can be easily obtained by

$$\dot{\hat{\eta}}_y = (A - KC)\hat{\eta}_y + Ky, \quad (9)$$

and an estimator for  $\eta_\varphi$  is proposed to have the form

$$\dot{\hat{\eta}}_\varphi = (A - KC)\hat{\eta}_\varphi + \varphi\hat{\theta}_\eta + \Upsilon(t)\hat{\theta}_\eta, \quad (10)$$

where  $\Upsilon$  defines a dynamic transformation  $\hat{\eta}_\varphi = \Upsilon(t)\hat{\theta}_\eta$ . By developing the derivative of  $\hat{\eta}_\varphi$ ,

$$\frac{d\hat{\eta}_\varphi}{dt} = \dot{\Upsilon}\hat{\theta}_\eta + \Upsilon\dot{\hat{\theta}}_\eta, \quad (11)$$

$$= (A - KC)\Upsilon\hat{\theta}_\eta + \varphi\hat{\theta}_\eta + \Upsilon\dot{\hat{\theta}}_\eta. \quad (12)$$

we have that

$$\dot{\Upsilon}\hat{\theta}_\eta = (A - KC)\Upsilon\hat{\theta}_\eta + \varphi\hat{\theta}_\eta, \quad (13)$$

$$(\dot{\Upsilon} - (A - KC)\Upsilon - \varphi)\hat{\theta}_\eta = 0. \quad (14)$$

This last equation gives

$$\dot{\Upsilon} = (A - KC)\Upsilon + \varphi, \quad (15)$$

which describes a filtered matrix of  $\varphi$ .

Since the regressor  $\varphi$  is bounded, so is  $\Upsilon$  because  $A - KC$  is a Hurwitz matrix. The training algorithm is proposed to be

$$\dot{\hat{\theta}}_\eta = \Gamma\Upsilon^T C^T (y - C\hat{\eta}), \quad (16)$$

where  $\Gamma$  is a positive definite constant gain matrix. The stability and convergence of the proposed network and the training algorithm are summarized in the following theorem.

*Theorem 1:* Consider the system (1). Assume that it is observable in the region of interest, and transformable through a diffeomorphism into the equivalent system (2). The following neurofuzzy network

$$\dot{\hat{\eta}} = A\hat{\eta} + \varphi(u, y)\hat{\theta}_\eta + (K + \Upsilon\Gamma\Upsilon^T C^T)(y - C\hat{\eta}), \quad (17)$$

with the training algorithm

$$\dot{\hat{\theta}}_\eta = \Gamma\Upsilon^T C^T (y - C\hat{\eta}), \quad (18)$$

where  $(A, C)$  is observable,  $\Gamma > 0$ , the gain matrix  $K \in \mathfrak{R}^n$  is such that  $A - KC$  is Hurwitz, and

$$\dot{\Upsilon} = (A - KC)\Upsilon + \varphi, \quad (19)$$

the filtered version of  $\varphi$ , guarantees that all signals are bounded, and the state estimation error  $\tilde{\mathbf{z}} = \hat{\eta} - \mathbf{z}$  and parameter error  $\tilde{\theta} = \hat{\theta}_\eta - \theta_\eta$  are ultimately bounded, provided that the regressor matrix is persistently exciting, *i.e.*,

$$\int_t^{t+T} \Upsilon^T(\tau) C^T C \Upsilon(\tau) d\tau \in [\alpha I, \beta I]. \quad (20)$$

for some  $0 < \alpha \leq \beta$ , and  $T > 0$ .

*Proof:* Following the Stone-Weierstrass Theorem, the smooth function  $\bar{f}(y, u)$  can be approximated by  $\varphi(y, u)\theta_\eta$

with arbitrary precision in a compact set contained in the region of interest, and then  $\bar{f}(y, u) - \varphi(y, u)\theta_\eta = \varepsilon(t)$ , where  $\varepsilon(t)$  is the approximation error for the set of optimal parameters  $\theta_\eta$  (depending on the minimizing criteria considered). In this sense, the system (2) can be represented as

$$\dot{\mathbf{z}} = A\mathbf{z} + \varphi(y, u)\theta_\eta + \varepsilon(t), \quad (21)$$

$$y = C\mathbf{z}, \quad (22)$$

Now, considering a network (3) such that  $\eta$  is meant to be an estimate of  $\mathbf{z}$ , and defining the error signals

$$\tilde{\theta} = \theta - \hat{\theta}, \quad \tilde{\mathbf{z}} = \mathbf{z} - \hat{\mathbf{z}}, \quad \xi(t) = \tilde{\mathbf{z}} - \Upsilon\tilde{\theta}, \quad (23)$$

the derivative of the previous equation represents the error signal dynamics, which are

$$\dot{\xi}(t) = (A - KC)\xi - \varepsilon, \quad (24)$$

$$\dot{\tilde{\theta}} = -\Gamma\Upsilon^T C^T C(\xi + \Upsilon\tilde{\theta}). \quad (25)$$

It follows that, while  $\varepsilon$  is bounded,  $\xi$  is bounded, as the matrix  $A - KC$  is designed to be Hurwitz. Moreover,  $\Upsilon$  must be bounded as it is obtained from a stable filtering of  $\varphi$ , that is also bounded.

Now, it can be proven [22] that under a persistent excitation condition (20) the origin of the system

$$\dot{\chi} = -\Gamma\varphi^T \varphi \chi \quad (26)$$

is exponentially stable. It follows that the homogeneous part of (25) is also exponentially stable. Therefore, the equation (25) is exponentially stable with a bounded perturbation, which implies the boundness and the ultimate boundness of  $\tilde{\theta}$ . ■

#### IV. EXAMPLES

In order to validate the proposed recurrent network and illustrate its performance, two simulation examples and an experimental result are presented. In the first one, we identify a nonlinear system that is another recurrent network, which represents an *ideal* plant with no model mismatching. In the second example, a nonlinear system in the form of (2) was considered, where the same set of parameters used in the first example was employed. The third example consists on the identification of the dynamics of a DC motor with associated nonlinearities. In general, a sigmoid function  $\rho(y) = \tanh(y)$  was used, and the antecedent part only depends on  $y$ .

##### A. Example 1: Identification of a recurrent neural network

The system to be identified is a network of the form (3) with three states ( $n = 3$ ). The main objective in this example is to show the performance of the training algorithm under no model mismatch present. The membership functions with  $\sigma_{yi} = 0.2$ ,  $i = 1, 2, 3$ , and  $\mu_{y1} = -1$ ,  $\mu_{y2} = 0$  and  $\mu_{y3} = 1$ . The vector  $\theta_\eta$  was

$$\theta_\eta = [3.495 \ 1.881 \ 0.2252 \ 1.055 \ -2.090 \ 5.088 \dots \\ 0.1772 \ 5.391 \ 0.7922]^T.$$

The vector  $K$  was chosen such that the eigenvalues of  $A - KC$  were  $\{-40, -50, -60\}$ , and  $\Gamma = 10^7 I_{3r \times 3r}$ . The input signal  $u$  was a bounded random signal with zero mean and unitary variance, that changed each 5 seconds. In Fig. 2 it can be seen how the parameters of the proposed recurrent network evolve during training. In Fig. 3 it is shown the performance of the network states running in parallel with the original network, while Fig. 4 depicts the output of both networks. After training, it was obtained that

$$\hat{\theta}_\eta = \begin{bmatrix} 3.604 & 2.038 & 0.2770 & 1.055 & -2.087 & 5.093\dots \\ 0.1756 & 5.389 & 0.7874 \end{bmatrix}^T.$$

Note that the parameter estimates, the state estimate and the output estimate follow closely to their true values; and more importantly, that the input-output dynamics are identified.

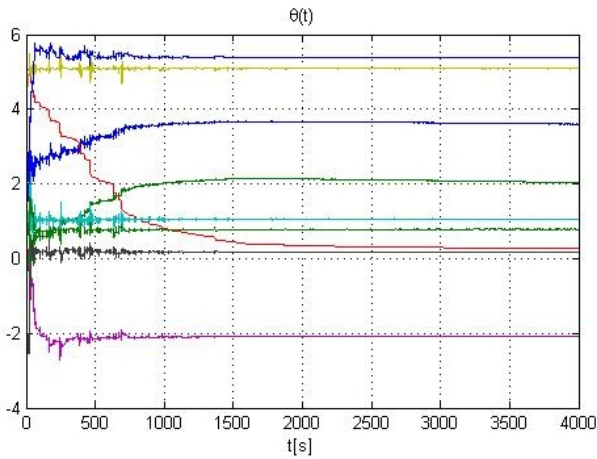


Fig. 2. Parameter evolution for Example I

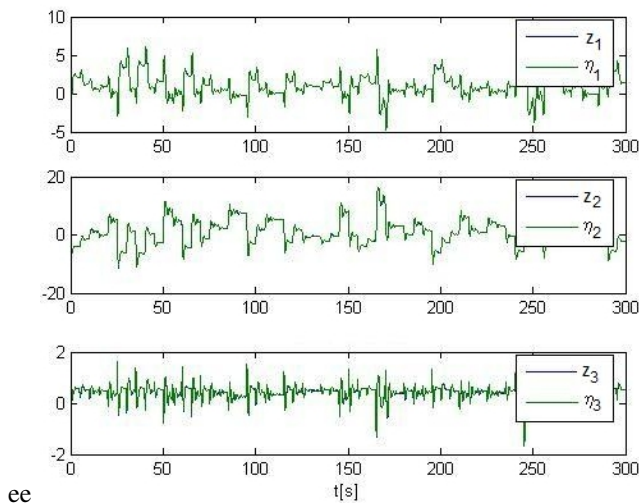


Fig. 3. State evolution for Example I

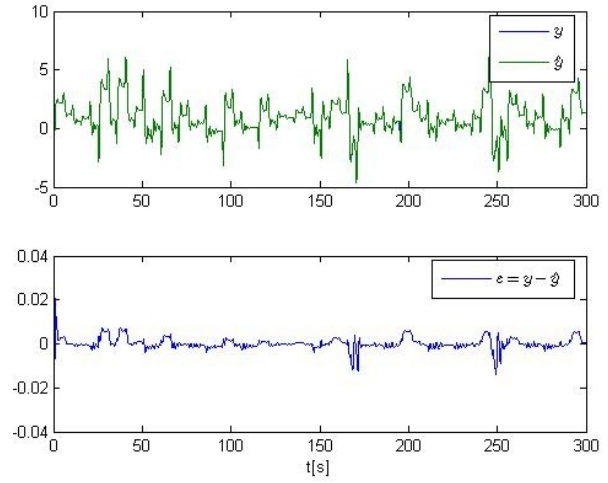


Fig. 4. Output evolution for Example I

### B. Example 2: Identification of a nonlinear system

The nonlinear system is already transformed into the form of (2), given by

$$\begin{aligned} \dot{\mathbf{z}} &= \begin{bmatrix} 0 & 0 & -8 \\ 1 & 0 & -12 \\ 0 & 1 & -5 \end{bmatrix} \mathbf{z} + \begin{bmatrix} 0.1 \\ 0 \\ -0.1 \sin(\frac{1}{2}y) \end{bmatrix}, \\ &+ \begin{bmatrix} 0.1 \sin(\frac{1}{2}y) + 0.2 \\ 0.1 \\ 0 \end{bmatrix} u, \\ y &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \mathbf{z} \end{aligned} \quad (27)$$

The main objective here is to show the performance of the network where theoretical model mismatch is present. The pair  $(A, C)$  is observable, and the network was chosen with  $\dim\{\eta\} = 3 = n$ . For this simulation, we chose the same parameters and input as in the previous example. So, the membership functions have parameters  $\sigma_{yi} = 0.2$ ,  $i = 1, 2, 3$ , and  $\mu_{y1} = -1$ ,  $\mu_{y2} = 0$  and  $\mu_{y3} = 1$ . In particular, after some iterations, these parameters were found to converge. The vector  $K$  was chosen such that the eigenvalues of  $A - KC$  were  $\{-40, -50, -60\}$ , and  $\Gamma = 10^7 I_{3r \times 3r}$ . In Fig. 5 it can be seen how the parameters of the proposed recurrent network evolve during training. In Fig. 6 it is shown the performance of the network running in parallel with the original system comparing their states, and in Fig. 7 is shown the output of both systems.

In the proposed network, the antecedent part of the fuzzy rules are fixed. In some situations, it would be desirable to get an a priori estimate of these parameters, or tune them on-line. If the membership functions are not well chosen, the performance can deteriorate. In this case it can be seen that using the mentioned membership functions, the network can follow closely the dynamics of the system, with a small modeling error. For future work, we plan to adapt the nonlinear parameters in the antecedent part of the rules in order to improve the identification and modeling performance of the network.

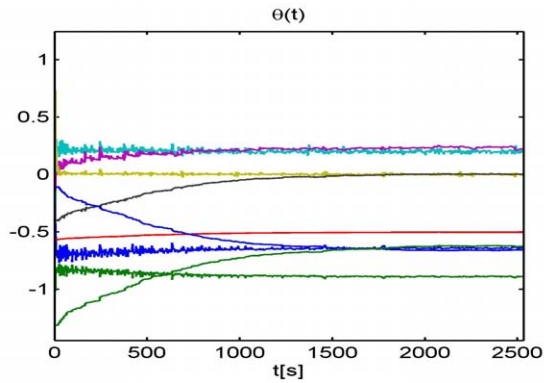


Fig. 5. Parameter evolution for Example II

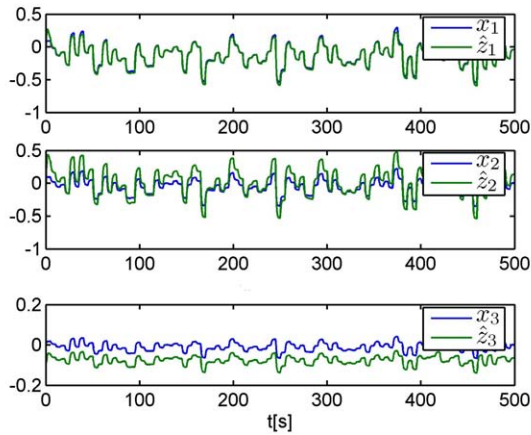


Fig. 6. State evolution for Example II

To see how these parameters affect the performance of the network, several simulations were done with different widths of the membership functions, and it was observed that, for  $\sigma_{y_i}$  between 0.2 and 0.1 the results did not have a strong variation. When  $\sigma_{y_i} = 0.5$  and higher were considered, the performance of the network deteriorated. Fig.9 shows the results with  $\sigma_{y_i} = 0.5$ . As can be noticed in this figure, the performance of the network in producing the dynamic mapping  $u \rightarrow \hat{y}$  is degenerated, though still is acceptable in parameter boundness (shown in Fig.8). This indicates that a careful selection of the space partition of  $(y)$  must be made prior to training. This is also shown in the next example.

### C. Example 3: Identification of a real nonlinear physical system

For this example, experimental data were obtained from a DC motor that has associated nonlinearities, such as Coulomb friction and nonlinear inertia was added.

This motor is controlled by voltage, with a range of  $\pm 5[V]$ , and 400s of input-output signals were used for this experiment, with a sample time of 20ms. For training purposes, the input and output signals were normalized to their maximum values. A sample of these normalized training signals is depicted in Fig. 10. For the network, it was chosen  $\dim\{\eta\} = 3 = n$ ,  $\sigma_{y_i} = 2$ ,  $i = 1, 2, 3$ , and the centers of the

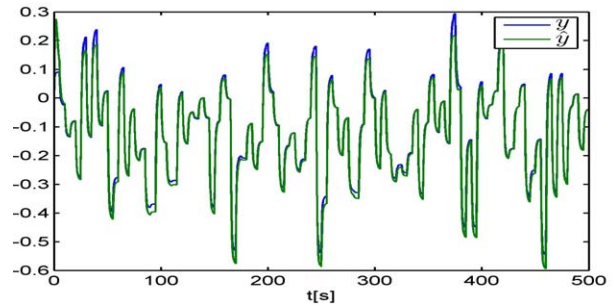


Fig. 7. Output evolution for Example II

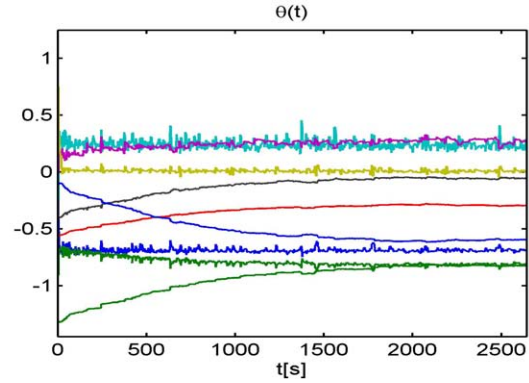


Fig. 8. Parameter evolution for Example II with  $\sigma_i = 0.5$

membership functions were obtained using a Fuzzy C-Means [24] algorithm on output data  $y$ , resulting on  $\mu_{y1} = -0.3180$ ,  $\mu_{y2} = 0.3047$  and  $\mu_{y3} = -0.0129$ . The eigenvalues of  $A$  were chosen such that  $\lambda\{A\} = \{-0.1, -0.2, -0.3\}$ , and vector  $K$  was chosen via a LQR algorithm that the weighting matrices were  $Q_{LQR} = I_{3 \times 3}$  and  $R = 1$ . The training algorithm was run for 7 epochs, and  $\Gamma = 1_{3r \times 3r}$ . In Fig. 11 it can be seen how the parameters of the proposed recurrent network evolve during training and in Fig. 12 is shown the output of both systems.

## V. CONCLUSIONS

In this paper we have presented a new recurrent neuro-fuzzy network for modeling and identification a class of nonlinear systems using only input and output measurements. It gives both the state and parameter estimates of the plant. It was shown that the algorithm results in bounded identification error as well as bounded parametric error, that will depend on the approximation capabilities of the network. Two numerical examples and an two experimental results were presented in order to show the identification effectiveness of this network. It was observed that approximation capabilities vary depending on how the antecedent parameters of the network are selected, so we propose for future work that space-partition schemes (such as Fuzzy C-means or other types of unsupervised techniques) could be used to improve the performance of the network by, either choosing the antecedent parameters prior to training, or modify them into the algorithm. Also, for future work we consider to train



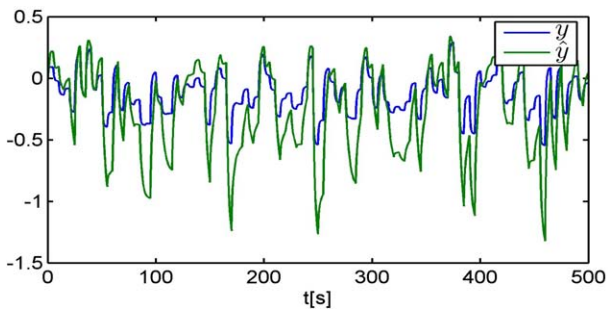


Fig. 9. Output evolution for Example II with  $\sigma_z = 0.5$

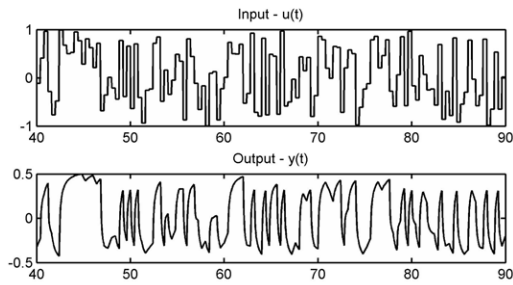


Fig. 10. Training signals for Example III

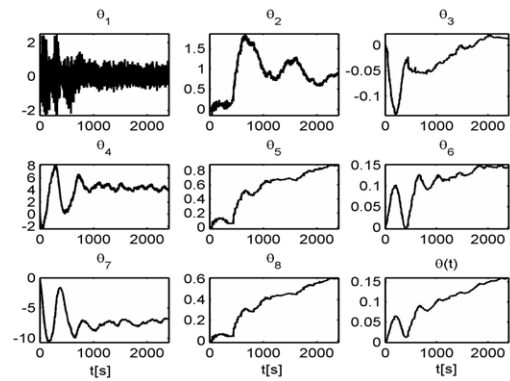


Fig. 11. Parameter evolution for Example III

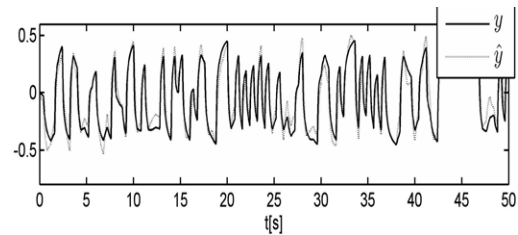


Fig. 12. Output evolution for Example III

also the antecedent parameters to improve the quality of the identification.

#### REFERENCES

- [1] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Math. Con. Signal Syst.*, vol. 2, pp. 303–314, 1989.
- [2] K. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Tras. Neural Networks*, vol. 1, pp. 4–27, March 1990.
- [3] C.-F. Juang, "A tsk-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms," *IEEE Transactions of Fuzzy Systems*, vol. 10, no. 2, pp. 155–170, April 2002.
- [4] J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," *Proceedings of the National Academy of Sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.
- [5] A. Weibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, "Phenomena recognition using time-delay neural networks," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 37, pp. 328–339, March 1989.
- [6] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Computation*, vol. 2, no. 1, pp. 270–280, 1989.
- [7] E. A. Wan, "Temporal back-propagation for fir neural networks," *Proc. of the International Joint Conference on Neural Networks*, pp. 575–580, 1990.
- [8] A. D. Back and A. C. Tsoi, "A time series modeling methodology using fir and iir synapses," *Proc. Workshop on Neural Networks for Statistical and Economic Data*, pp. 187–194, 1990.
- [9] P. A. Mastorocostas and J. B. Theocharis, "A recurrent fuzzy-neural model for dynamic system identification," *IEEE Trans. on Systems, Man and Cybernetics - Part B: Cybernetics*, vol. 32, no. 2, pp. 176–190, April 2002.
- [10] D. Rumelhart, G. Hinton, and R. Williams, *Learning Internal Representations by Error Propagation*. Cambridge: MIT Press, 1986, ch. 8.
- [11] R. Johansson, *System Modeling and Identification*, 1st ed., ser. Prentice Hall Information and System Sciences, T. Kailath, Ed. Prentice Hall, 1993.
- [12] K. M. Passino, *Fuzzy Control*. Menlo, CA: Addison Wesley Longman, 1998.
- [13] S. Haykin, *Kalman Filtering and Neural Networks*. Wiley Chichester, 2001.
- [14] G. C. Mouzouris and J. M. Mendel, "Dynamic non-singleton fuzzy logic systems for nonlinear modeling," *IEEE Trans. Fuzzy Systems*, vol. 5, no. 2, pp. 199–207, May 1997.
- [15] J. S. R. Jang, "Self-learning fuzzy controllers based on temporal back propagation," *IEEE Transactions on Neural Networks*, vol. 3, no. 5, pp. 714–723, 1992.
- [16] V. Gorrini and H. Bersini, "Recurrent fuzzy systems," *Proc. IEEE Int. Conference on Fuzzy Systems*, pp. 193–198, 1994.
- [17] A. Poznyak, T. Poznyak, and I. Chairez, "Dynamic neural observers and their application for identification and purification of water by ozone," *Automation and Remote Control*, vol. 67, no. 6, pp. 887–899, 2006.
- [18] W. Yu and X. Li, "Fuzzy identification using fuzzy neural network," *IEEE Trans. on Fuzzy Systems*, vol. 12, no. 3, June 2004.
- [19] M. Gonzalez Olvera and Y. Tang, "A new recurrent neurofuzzy network for identification of dynamic systems," *Fuzzy Sets and Systems*, vol. 158, no. 10, pp. 1023–1035, 2007.
- [20] G. Bastin and M. Gevers, "Stable adaptive observers for nonlinear time-varying systems," *Automatic Control, IEEE Transactions on*, vol. 33, no. 7, pp. 650–658, 1988.
- [21] R. Marino, "Adaptive observers for single output nonlinear systems," *Automatic Control, IEEE Transactions on*, vol. 35, no. 9, pp. 1054–1058, 1990.
- [22] Q. Zhang, "Adaptive observer for multiple-input-multiple-output (MIMO) lineartime-varying systems," *Automatic Control, IEEE Transactions on*, vol. 47, no. 3, pp. 525–529, 2002.
- [23] M. A. Gonzalez-Olvera and Y. Tang, "Identification and Control of Nonlinear Dynamic Systems via a Constrained Input-Output Neuro-fuzzy Network," *Advances in Control Systems Theory and Applications (Edited by G. Tao and J. Sun)*, 2008.
- [24] J. Bezdek, "Cluster Validity with Fuzzy Sets," *Cybernetics and Systems*, vol. 3, no. 3, pp. 58–73, 1973.