LEARNING AND OUTPUT REGULATION WITH RECURRENT NEURONAL NETWORKS

J. Henriques†, B. Castillo‡, A. Titli‡, P. Gil†, A. Dourado†

jh@dei.uc.pt, bcastill@laas.fr, titli@laas.fr, pgil@dei.uc.pt, dourado@dei.uc.pt

†*CISUC - Informatics Engineering Department, University of Coimbra*
Pólo II, Pinhal de Marrocos - 3030 Coimbra - Portugal

‡*LAAS – CNRS - Laboratorie d'Analyse et d'Architecture des Systèmes*
Avenue du Colonel Roche, 7 - 31077 Toulouse Cedex 4 – France

Abstract: The problem of on-line learning using recurrent neural networks, using within a model based non-linear control scheme, is addressed in this paper. The control output regulation theory is introduced, leading to an indirect adaptive control structure and matching both the classical non-linear control and neural methods. The main goal is to investigate the combination of the well-known learning abilities of neural networks with the stability properties of the output regulator. The practical potentials of the proposed scheme are demonstrated by some experimental results on a laboratory bench process.

Keywords: On-line learning; recurrent networks; output regulation; discrete time systems; adaptive non-linear control.

1. INTRODUCTION

It is recognized that neural networks offer a number of potential benefits for application in the field of control engineering, particularly in modelling non-linear systems. The main advantages are their inherent ability to approximate non-linear mappings (Jin et al, 1995) and their learning capabilities through examples, performed either off-line or on-line. Among neural networks, recurrent neural networks are particularly suitable for modelling purposes and control applications. In fact, recurrent neural networks have a compact structure and they are in a standard non-linear state space form.

Advanced control methods, that explicitly considers the intrinsic non-linearities of the process, have recently been investigated (Isidori, 1995), (Khalil, 1992) and (Nijmeijer and Schaft, 1990). The output regulator is a non-linear model based control algorithm that guarantees the convergence of the tracking error to a small neighbourhood of zero, while assuring that the closed loop system is stable. If the system is linear and time invariant and their parameters are completely known, a linear regulator is derived (Francis, 1977). However, for non-linear systems the problem becomes more complicated (Isidori and Byrnes, 1990). In this case the solution of the output regulator problem leads to a set of non-linear difference equations for which is very difficult, or even impossible, derive a closed solution (Castillo et al, 1993). Here, to overcome the problem, an iterative procedure is presented. The proposed algorithm, based on the recurrent neural network structure, ensures the convergence of the regulator equations solution.

The non-linear control theory, such as feedback linearization or output regulation, offers a straightforward method for solving non-linear regulation problems.

One drawback on non-linear control theory is the assumption of a perfect model, leading to a heavily dependency on the accuracy of the process model. To cope with the modelling errors of the off-line estimated neural parameters, and changing dynamics, an adaptive strategy was employed by providing on-line learning of the network parameters. In this sense, the neural model can adaptively capture the system uncertainties and the regulator law adjusts the control action in order to guarantee zero offsets.

It is intended with this approach to investigate the combination of the identification capabilities of neural networks and the stability properties of the output regulation controller. Employing a recurrent neural network, the unknown system under control can be replaced by the network model, trained by an appropriate on-line learning algorithm, transforming the original problem of control into a non-linear control problem suitable to be designed with the output regulation theory.

This paper is organised as follows. In section 2 the recurrent neural architecture and both off-line and on-line learning laws are presented. In section 3 a brief review of output regulation theory is presented, being the problem stated and described the iterative procedure that solves the output regulator equations. In section 4 the proposed adaptive control structure that combines the recurrent neural and the output regulator is illustrated. Section 5 presents some experimental results with a non-linear laboratory process, and, in section 6, some conclusions are stated.

Controlo'2000: 4th Portuguese Conference on Automatic Control
ISBN 972-98603-0-0

2. MODELLING WITH RNN

Due to their well know ability to approximate uncertain non-linear mappings neural networks become a potential model solution to many adaptive non-linear control problems. As has been observed from recent developments, under a suitable choice of network architecture, learning methodologies, and appropriate data set training, neural networks can be conveniently trained to be included in the control system context (Kulawski and Brdys, 2000).

2.1. Recurrent Neural Network Architecture

Several discrete time recurrent neural network architectures have been proposed. A good review has been reported in (Tsoi and Back, 1997) and (Kremer, 1998). One possible classification for these architectures can be based on the accessibility of their states (Tsoi and Back, 1997). Depending on whether or not the states of the network are observable they can be divided into two groups.

In one group, for which the states are accessible, Time Delay Neural Network (TDNN)(Narendra and Parthasarathy, 1991), is a well-known example. The TDNN provides a simple form of dynamics that gives the network the ability of modelling non-linear dynamic systems. A variation on the TDNN is the Gamma network, where the TDNN is replaced by a set of cascaded filters (Vries and Principe, 1992). This class of networks can be interpreted as a non-linear auto-regressive with exogenous inputs (NARX) model.

The other group includes networks that have hidden neurons not directly accessible. They can be divided into two classes: globally recurrent networks (Williams and Zipser, 1989), and locally recurrent networks (Frasconi et al, 1992). One network that fit into this class is the well know Elman's simple recurrent network (Elman, 1990). Since their behaviour can be describes by a non-linear state space model they are also known as state space neural models.

In this work, a neural network considering hidden neurons that are not accessible is adopted. The architecture proposed is depicted in Fig. 1.

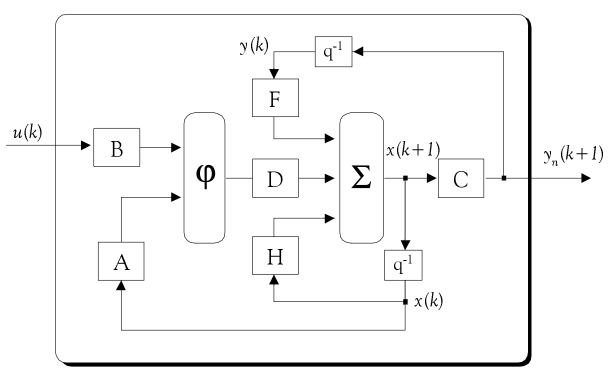

Fig. 1. Block diagram of the recurrent neural network.

The following reasons justify this choice: state space models describes a broader general class of dynamical systems than input-output models; they have a lower order and require a smaller number of past inputs (i.e. a smaller number of regressors), and therefore a smaller number of parameters; finally, they present a compact structure and they are in a standard form, which makes them an ideal candidate for the application in model based control schemes, which are the main interest of this paper.

The behaviour of the proposed structure can be described by the following discrete time equations:

$$x(k+1) = D\varphi\{Ax(k) + Bu(k)\} + Hx(k) + Fy(k) \qquad (1a)$$

$$y_n(k) = Cx(k) \qquad (1b)$$

The vector $x \in \Re^n$ is the state, $u \in \Re^m$ is the input and $y_n \in \Re^p$ is the output. $A \in \Re^{n,n}$, $B \in \Re^{n,m}$, $C \in \Re^{p,n}$, $D \in \Re^{n,n}$ and $H \in \Re^{nxn}$ are the interconnection matrices. This architecture can be seen as a modification of the original discrete time recurrent neural network, proposed by Hopfield (Hopfield and Tank, 1985). The neural activation function $\varphi(.)$ is the hyperbolic tangent function.

The network's weights are adjusted in order to minimise the sum of the squared errors. Once identification is achieved, two modes of operation are possible: series-parallel, where the outputs of the process are used as inputs to the model, and parallel mode, where the output of the network is feed back into the network. While training is greatly simplified with the first approach, the neural model can generate only one step ahead prediction. In the parallel mode more than a one-step ahead prediction is possible, however the training procedure is generally very slow and one cannot expect the identification model to be perfect. In the proposed structure both modes are tacking in account. First, training is performed with the past outputs of the actual system $y(k)$ (series-parallel mode), allowing a fast training. Next, past plant's outputs $y(k)$ are replaced by past outputs of the neural network $y_n(k)$ (parallel mode). Following this procedure a non-linear state space model in the form (2) is derived.

$$x(k+1) = D\varphi\{Ax(k) + Bu(k)\} + \{H + FC\}x(k) \qquad (2a)$$

$$y_n(k) = Cx(k) \qquad (2b)$$

2.2. Learning Algorithms

If learning is performed off-line the neural network is supposed to be accurate enough to model the real process. Yet, when controlling the process, the performance may become very poor in case of process variations, namely, in its structure. In other hand, the implementation of on-line learning algorithms can be computationally very demanding, so the reduction of computational complexity is particularly appealing. The focus of this section is to describe a feasible learning algorithm for on-line post-training of recurrent neural networks.

Off-line learning: The simplest method is the steepest descent algorithm. More advanced methods are quasi-Newton and Levenberg-Marquardt, which are Newton like methods that try to build curvature information of

Controlo'2000: 4th Portuguese Conference on Automatic Control
ISBN 972-98603-0-0

the Hessian, based on gradient information only. For large-scale problems, conjugate gradient algorithms are preferred, because in this algorithm there is no need to store matrices. As has been appointed by Hagan (Hagan and Menhaj, 1994) the Levenberg-Marquardt method is more efficient than other techniques when the network contains no more than a few hundred of parameters. Due to their effectiveness this algorithm was used for off-line training.

On-line learning: Several training algorithms have been proposed to adjust the values of the network parameters in recurrent networks. The well known are the real time recurrent algorithm (Williams and Zipser, 1989), the dynamic backpropagation , and the backpropagation trough time (Werbos, 1990). The algorithm proposed here is a modification of the one proposed by (Chow and Fang, 1998). It is assumed that in equation (2) the matrices C, F and H are fixed and determined off-line. Therefore, only matrices A and B are going to be determined on-line. Let the matrix $W(k)\in\Re^{n,n+p}$ be given as follows:

$$W(k)=\left[A(k)\ B(k)\right] \tag{3}$$

For each time step k an iterative procedure is established. The identification error $e_n(k)$ at each iteration i, is defined by $e_n^i(k)=y(k)-y_n^i(k)$. From equation (2) it is possible to obtain,

$$e_n^i(k) \ = \ y(k) \ -C\varphi\left\{W^i(k)z(k-1)\right\}+C\left\{H+FC\right\}x(k-1) \tag{4}$$

where $z(k)\in\Re^{n+p}$ is given by (5).

$$z(k)=\left[x(k)\,u(k)\right]^T \tag{5}$$

$W^i(k)$ has to be determined from the previous iteration $W^{i-1}(k)$ by (6).

$$W^i(k)=W^{i-1}(k)+\Delta W^i(k) \tag{6}$$

The problem consists in determining the increment $\Delta W^i(k)$, such that, between two consecutive iterations the identification error $e_n^i(k)$ verifies (7).

$$e_n^i(k)=\alpha\, e_n^{i-1}(k) \tag{7}$$

where α is a stable matrix. If this condition holds, the error will converge to zero. For the hyperbolic activation function (a non-decreasing monotonic function) it is possible to write (8).

$$\varphi(x+\Delta x)-\varphi(x)\approx\frac{\partial\varphi(x)}{\partial x}\Delta x \tag{8}$$

Using this approximation equation (4) can be rewritten as follows:

$$e_n^i(k) \ -e_n^{i-1}(k)\le \ -C\Delta W^i(k)z(k-1) \tag{9}$$

Choosing

$$\Delta W^i(k)=\left\{C^T\left[CC^T\right]^{-1}\right\}M\,e_n^{i-1}(k)\left\{\left[z(k-1)^T z(k-1)\right]^{-1}z(k-1)^T\right\} \tag{10}$$

equations (9) becomes

$$e_n^i(k) \ \le (I-M)\,e_n^{i-1}(k) \tag{11}$$

In order to assure that the error converges, M must be chosen such that the eigenvalues of $(I-M)=\alpha$ are placed in the open unit circle of the complex plane. It is assume that the scalar $(CC^T)^{-1}\neq 0$ and in case of the scalar $(z(k-1)^T z(k-1))^{-1}=0$ the weights are not updated. The proposed learning algorithm can be seen as a gradient type algorithm with a varying learning rate.

3. OUTPUT REGULATION THEORY

The output regulation technique is especially appealing from the point view of the non-linear control system design. The main goal is to derive a control law such that the closed loop system is stable and, simultaneously, the tracking output error converges to zero. A complete solution for linear systems was given by Francis (Francis, 1977). For non-linear discrete time systems, Castillo (Castillo et al, 1993) using the zero output constrained algorithm (Monaco and Normand-Cyrot, 1987), shows that the solution for the problem is reduced to the solution of transcendental non-linear equations, which represents the discrete time counterpart of the differential and transcendental equations, found for the continuous time systems by Isidori (Isidori and Byrnes, 1990).

3.1. Solution of the Output regulation Problem

Consider a multivariable system, with m inputs and p outputs described by state equations of the form (12).

$$x(k+1) \ = \ f\left\{x(k),\,u(k),w(k)\right\} \tag{12a}$$

$$y(k) \ = \ h\left\{x(k)\right\} \tag{12b}$$

where $f:\Re^{+2}\to\Re$ and $h:\Re^n\to\Re^p$ are non-linear functions; $u(k)\in\Re^m$, $y(k)\in\Re^p$ and $x(k)\in\Re^n$ are, respectively, the input vector, the output vector and the internal state vector, at a discrete time k. The vector $w(k)\in\Re^p$ defines a reference signal, generated by a so-called *exosystem* (13).

$$w(k+1) \ = \ s\left\{w(k)\right\} \tag{13a}$$

$$y_d(k)=r(w) \tag{13b}$$

The problem of output regulation consists to find a state feedback control law such that the tracking error $e_c(k)$ (14), goes to zero and the whole system is asymptotically stable.

$$e_c(k) \ =h(x)-r(w) \tag{14}$$

Theorem (Castillo): *The state feedback discrete time regulator problem is locally solvable if there exits two* C^h $(h\ge 2)$ *mappings* $x=\pi(w)$ *and* $u=c(w)$ *satisfying* (15).

$$\pi(s(w)) \ = \ f\left\{\pi(w),\,c(w),\ w\right\} \tag{15a}$$

$$0 \ = \ h(\pi(w))-r(w) \tag{15b}$$

It is easy to show that the particular control law given by (16) satisfies (15).

$$u(k)=c(w)+K\left[x-\pi(w)\right] \tag{16}$$

Controlo’2000: 4th Portuguese Conference on Automatic Control
ISBN 972-98603-0-0

$K \in \Re^{m,n}$ is a matrix that places the eigenvalues of the first order approximation of the non-linear state space model in desired locations.

3.2. Solving the Regulator Equations

As given by equation (15) the solution of the output regulator problem is reduced to a set of non-linear difference equations, known as regulator equations. However, except in a very few cases, it is very difficult to derive an analytical solution for mappings $x = \pi(w)$ and $u = c(w)$. One possibility to overcome the difficult originated by the non-linearities is to solve approximately the regulator equations. Castillo (Castillo et al, 1993) has presented and derived conditions for the existence of an approximate solution for the discrete time case, based on a polynomial expansion. Based on a Taylor series expansion, Huang and Rugh (Huang and Rugh, 1992) have proposed an approximation method for the continuous case. Recently, the same authors (Huang and Rugh, 1999) have presented an alternative approximation using a type of recurrent neural network, analogous to a cellular network.

In this paper, taking into account the recurrent neural structure an iterative algorithm that ensures convergence of the regulator equation solution is followed. The regulator equations and the recurrent linear network actually complement each other. In fact, regulator equations can be described by the same architecture as the original recurrent neural network used for modelling tasks. The proposed algorithm is similar to the on-line learning algorithm, leading to a pole placement design and ensuring convergence if the eigenvalues of the error equation are chosen to be stable.

The application of the zero output constrained dynamics algorithm (Monaco and Normand-Cyrot, 1987) leads to the following set of $n+p$ equations with $n+p$ unknowns (17).

$$C_1\,\pi_1 + C_2\,\pi_2 = w(k) \tag{17a}$$

$$\begin{aligned} &C_1\,\varphi\{A_{11}\pi_1 + A_{12}\pi_2 + B_1\gamma(w)\} + (C_1H_{11} + C_2H_{12})\pi_1 \\ &+ C_2\,\varphi\{A_{21}\pi_1 + A_{22}\pi_2 + B_2\gamma(w)\} + (C_2H_{21} + C_2H_{22})\pi_2 \\ &= w(k+1) \end{aligned} \tag{17b}$$

$$\varphi\{A_{21}\pi_1 + A_{22}\pi_2 + B_2\gamma(w)\} + H_{21}\pi_1 + H_{22}\pi_2 = \pi_2 \tag{17c}$$

where $c(w) \in \Re^p$ and $\pi(w) = [\pi_1(w)\ \pi_2(w)\ ]^T \in \Re^n$ are defined in (16). $\pi_1(w) \in \Re$ (neural network relative degree value), $\pi_2(w) \in \Re^{n-1}$ and A_{ij}, B_{ij}, C_{ij} and H_{ij} are matrices of appropriate dimensions.

The problem under investigation can be stated in similar way as the on-line learning algorithm (section 2.2). It is intended to determine the mappings $\Gamma(w) = [c(w)\ \pi(w)\ ]^T \in \Re^{n+1}$ such that the error between two consecutive iterations i and $i-1$, derived from (17), verify (18).

$$E^i = \Lambda\, E^{i-1} \tag{18}$$

The matrix $\Lambda \in \Re^{n+1,n+1}$ has stable eigenvalues (a diagonal matrix is a particularly convenient choice).

4. CONTROL STRUCTURE

The recurrent neural model replace the unknown model, transforming the original problem of control into a non-linear control problem, suitable to be handled by the output regulation theory. On the other hand, if the training is performed off-line, non-linear neural models are not exact in the presence of disturbances, parameter variations and uncertainties. Thus, to overcome this drawback an adaptive strategy is applied by providing on-line learning for the network parameters (Fig. 2).

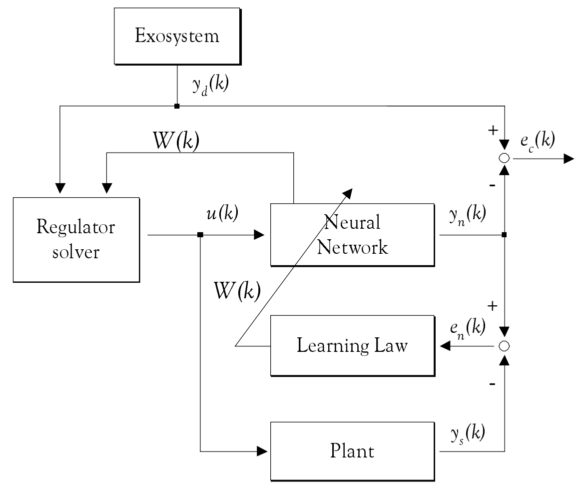

Fig. 2 - Block diagram of the proposed control structure.

In this approach, a learning law updates on-line the neural weights $W(k)$, based on the identification error, $e_n(k) = y_n(k) - y_s(k)$. The regulator design for the neural network ensures the asymptotic convergence of the neural tracking error, $e_c(k) = y_n(k) - y_d(k)$. Thus, if the parameters of the neural model are properly adapted in the presence of parametric variations or uncertainties in the dynamics of the system, the recurrent neural network will be able to capture the process dynamics and, on the other hand, the control will be able to follow the set point trajectory. In fact, the system tracking error, $e(k) = y_s(k) - y_d(k)$, can be written as

$$e(k) = (y_s(k) - y_n(k)) + (y_n(k) - y_d(k)) \tag{19}$$

Since the regulator assures that

$$\lim_{k\to\infty}\{y_n(k) - y_d(k)\} = 0 \tag{20}$$

then, the overall error will converge if the identification error also converges, that is:

$$\lim_{k\to\infty}\{y_s(k) - y_n(k)\} = 0 \tag{21}$$

Following the on-line learning approach and under an appropriate selection of α the identification error converges to zero.

5. EXPERIMENTAL RESULTS

In order to illustrate the applicability of the proposed control scheme, the temperature control in a laboratory heating system is considered. The system dynamics is

Controlo'2000: 4th Portuguese Conference on Automatic Control
ISBN 972-98603-0-0

assumed to be totally unknown and that the control law is only based on input-output measurements.

5.1. Laboratory process

In the laboratory process, illustrated schematically in Fig. 3, air is forced to circulate by a fan blower through a duct and heated at its inlet. This is a non-linear process with a pure time delay, which depends on the position of the temperature sensor element and on the air flow rate, which relies on the damper position Ω, see Fig. 3. The system input $u(k)$, is the voltage on the heating device, which consists of a mesh of resistor wires, and the output, $y_s(k)$, is the outlet air temperature.

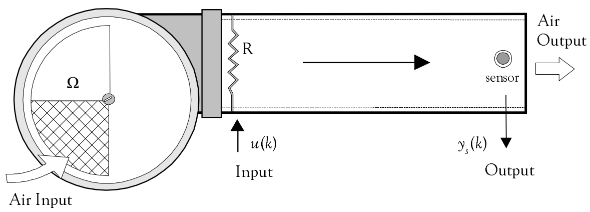

Fig. 3. Schematic diagram of the Laboratory Process.

5.2. Experiments

For modelling purposes a recurrent neural network with five neurones in both hidden unit, $n = 5$, was considered. For the iterative output regulation algorithm the value for $(1-M) = \alpha = 0.8$ was chosen. The recurrent network was previous trained with a fixed sequence of 400 data points (80 seconds), corresponding to a sampling time of 0.2 seconds (Fig. 4) and considering a series-parallel mode.

As can be observed from Fig. 4 the network is satisfactory able of describing the system's dynamics.

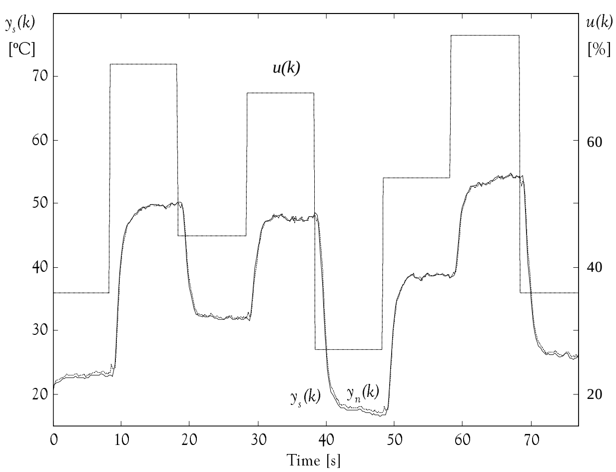

Fig. 4 – Identification performance - off-line learning.

If the recurrent neural network is indeed able to capture the non-linearities, it provides a suitable non-linear predictor to be used within the output regulation algorithm.

The performance of the proposed control scheme was tested for different set points, $y_d(k)$, and results for a square wave presented in Fig. 5. As can be observed, due to the model uncertainties, there are some deviations from the desired output.

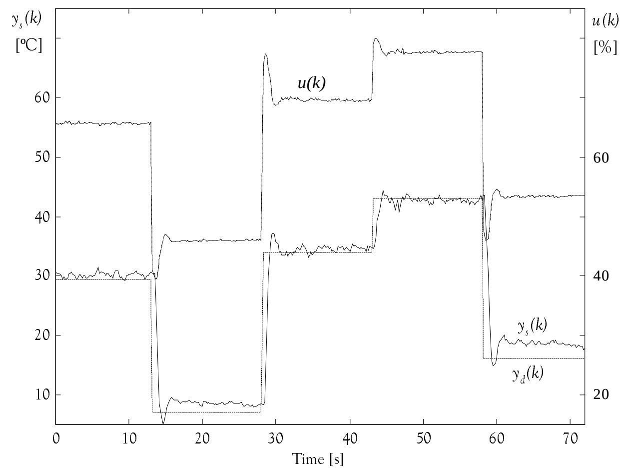

Fig. 5. Controller performance – without learning.

Fig. 6 shows the results when on-line learning is also considered. In this case, as can be observed, the proposed adaptive control approach gives quite satisfactory results. Clearly, it performs considerably well in tracking set points.

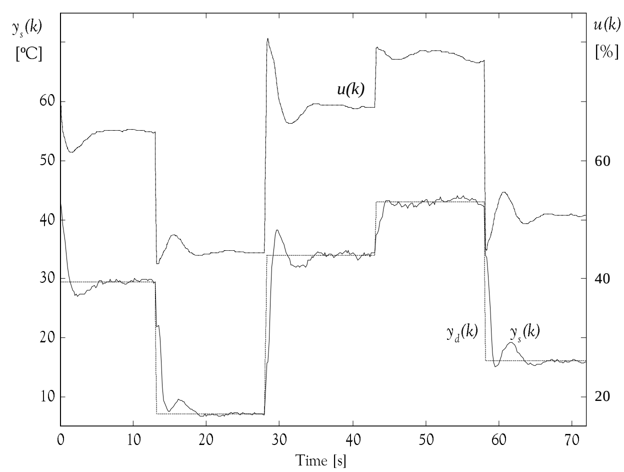

Fig. 6. Controller performance – with on-line learning.

The next experiment addressed the dynamics variation of the laboratory process. Variations were considered in the air flow rate, changing in this way the overall plant delay: $\Delta\Omega = +25\%$ at instant 13 seconds and $\Delta\Omega = -25\%$, at instant 33 seconds and considering a constant reference. Fig. 7 presents the experimental results. As can be observed, due to the adaptive characteristic of the proposed method, the control can accommodate system variations.

Controlo’2000: 4th Portuguese Conference on Automatic Control
ISBN 972-98603-0-0

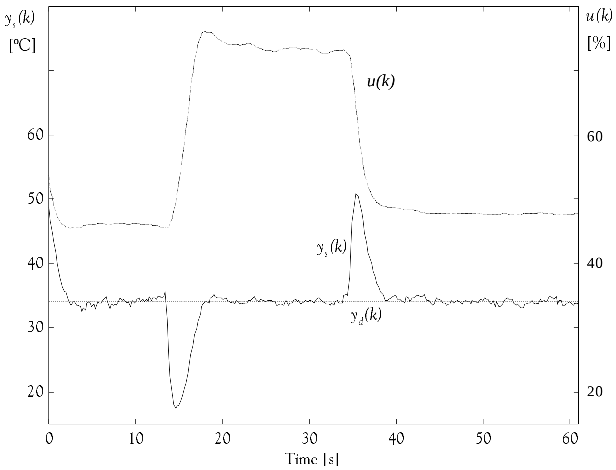

Fig. 7. Controller performance - dynamics variation.

6. CONCLUSIONS

Due to their intrinsic approximate properties, a recurrent neural network for modelling purposes was presented. States are assumed inaccessible and identification is performed using input output information. Its particular architecture, which can be seen as a non-linear state space model, along with the fast training provided by the proposed learning algorithm, makes this approach a feasible one for non-linear model based control techniques, such as the output regulation. For dealing with the output regulator design an approximated iterative procedure, ensuring convergence of the solution was derived. The output regulator guarantees the neural tracking error convergence to zero.

To cope with the inaccuracy of the off-line estimated neural parameters and, possible, changing dynamics, an adaptive strategy was employed by considering on-line learning. In this sense, the neural model can adaptively learn the system uncertainties and, thus, the regulator law adjusts the control action in order to guarantee error convergence.

The effectiveness of this technique has been demonstrated by experimental results from a laboratory process. They show that the control structure can not only adaptively learn the system dynamics but also guarantee asymptotic error convergence.

Acknowledgements

This work was partially supported by the Portuguese Ministry of Science and Technology (MCT), under the program PRAXIS XXI/EEI/11455/1998. Part of this work was done when the first author was visiting LAAS, Toulouse and the second was a CNRS visiting professor at LAAS. We would like to express here our gratitude to LAAS and CNRS.

REFERENCES

Castillo, B., S. Di Gennaro and D. Normand-Cyrot. Non-linear regulation for a class of discrete time systems. *Systems & Cont. Letters*, 20, 57-65, 1993. 1 (Castillo

Chow, T. and Y. Fang. A recurrent neural network based real time learning control strategy applying to non-linear systems with unknown dynamics. *IEEE Trans. Ind. Electronics*, 45, 1, 151-161, 1998. 2 (Chow an

Elman J. Finding structure in time, *Cognitive Science*, 14, 179-221, 1990. 3 (Elman,

Francis, B.The linear multivariable regulator problem. *SIAM J. Control Optim*ization, 15, 484-505, 1977. 4 (Francis,

Frasconi, P., M. Gori and G. Soda. Local feedback multilayered networks. *Neural Computation*, 4, 1, 120-130, 1992. 5 (Frascon

Hagan, M. and B. Menhaj. Training feedforward networks with the Marquardt algorithm. *IEEE Trans. on Neural Networks*, 5, 6, 989-99, 1994. 6 (Hagan a

Hopfield, J. and D. Tank. Neural computation of decisions in optim. problems, *Biological Cybernetics*, 52, 141-152, 1985. 7 (Hopfield

Huang, J, and W. Rugh. An approximation method for the non-linear servomechanism problem. *IEEE Trans. Autom. Control*, 37, 9, 1395-1398, 1992. 8 (Huang a

Huang, J, and W. Rugh. Stabilization on zero-error manifolds and the non-linear servomechanism problem. *IEEE Trans. Autom. Control*, 37, 7, 1009-1003, 1999. 9 (Huang a

Isidori, A. and C. Byrnes. Output regulation for non-linear systems. *IEEE Trans. Automat. Control*, 26, 131-140, 1990. 10 (Isidori

Isidori, A. Non-linear Control Systems. (third edition) *Springer-Verlag*, London, 1995. 11 (Isidori,

Jin, L., P. Nikiforuk and M. Gupta. Approximation of discrete time state space trajectories using dynamic recurrent networks. *IEEE Trans. Automatic Control*, 40, 7, 1266-1270, 1995. 12 (Jin et a

Khalil, H. Non-linear Systems. *Macmilllan Pub. Co.*, New York, 1992. 13 (Khalil,

Kremer, S. Spatio-temporal Connectionist Networks. A Taxonomy and review. *Computer based learning Unit*, University of Leeds, 1998. 14 (Kreme

Kulawski, G. and M. Brdys. Stable adaptive control with recurrent networks. *Automatica*, 36, 5-22, 2000. 15 (Kulaws

Monaco, A. and D. Normand-Cyrot. Minimum phase non-linear discrete time systems and feedback stabilization. Proc. 23th *CDC*, Los Angeles, 979-986, 1987. 16 (Monac

Narendra K. and Y. Parthasarathy, (1991). Gradient methods for the optimization of dynamical systems containing neural networks. *IEEE Trans. on Neural Networks*, 2, 2, 252-262. 17 (Narend

Nijmeijer, H. and V. Schaft. Non-linear Dynamical Control Systems. *Springer-Verlag*, New York, 1990. 18 (Nijmei

Tsoi, A. and A. Back. Discrete time recurrent neural network architectures. *Neurocomputing*, 15, 183-223, 1997. 19 (Tsoi an

Vries, B. and J. Principe. The gamma mode: a new neural net model for temporal processing, *Neural Networks*, 5, 565-576, 1992. 20 (Vries a

Werbos, P. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10), 1550-1560, 1990. 21 (Werbos

Williams, R. and D. Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1, 3, 270-280, 1989. 22 (William