*Proc. Of the 7th Jornadas Hispano-Lusas de Ingenieria Eléctrica*
*Madrid, July 4-6, 2001*

# Application of an Extended Kalman Filter for On-line Identification with Recurrent Neural Networks

**L. Palma**[1]
lbp@mail.fct.unl.pt

**P. Gil**[1,2]
pgil@dei.uc.pt

**J. Henriques**[2]
jh@dei.uc.pt

**A. Dourado**[2]
dourado@dei.uc.pt

**H. Duarte-Ramos**[1]
hdr@mail.fct.unl.pt

**[1] Dep. de Engenharia Electrotécnica**
**Universidade Nova de Lisboa**
**2825-114 Monte de Caparica**
**Portugal**

**[2] CISUC**
**Dep. de Engenharia Informática, UC**
**3030 Coimbra**
**Portugal**

**Abstract:**

Given their good approximation capabilities for reasonable non-linear systems artificial neural networks have attracted an increasing interest in a number of fields such as system identification and filtering. The main goal of this work is to emphasise the potential benefits of non-linear state-space neural networks for real-time identification with an extended Kalman filter. Experimental results from a laboratory heating system confirm the feasibility of this methodology.

**Keywords**: Identification; recurrent networks; on-line learning; real-time systems.

## 1. Introduction

In the last few decades neural networks have regained the research community enthusiasm, concerning their potential for modelling, filtering and control of non-linear systems [1], [2]. At the basis for this interest were the demonstrated capabilities as universal approximation structures exhibited by multi-layer perceptrons (MLP) [3], [4**]**.
Although purely feedforward networks together with the tapped-delay-line method have been applied to capture spatio-temporal information it is well known that a neural network containing a state feedback possesses more computational advantages [5] and are less susceptible to external noise. Additionally, as pointed out by I. Rivals and L. Personnaz [6], state-space models need a lesser number of parameters and they can describe a larger class of dynamic system than input-output models. However, other factors such as the learning method and the quality of the dada set [7], not to mention the initial values for the neural network's weights are crucial to the accuracy of the neural model.
When using neural networks for identification of dynamical systems it is imperative to provide the learning algorithm with fresh data as soon as new information is available so improving the precision of the original model and/or to cope with eventually unmodelled dynamics and structural changes. Thus, whenever possible, it is worthwhile

to reflect in the global training procedure these issues combining a previous batch training for offline identification with an online updating algorithm.
It is a primary intention of this work to get evidences of the applicability of this dual mode training of neural network structures for identification of real world processes.
Within this perspective, an hybrid recurrent neural network under the form of a non-linear state-space description is considered as a non-linear black-box model structure for capturing the non-linear dynamics of a laboratory heating system, being its parameters updated according to the strategy mentioned above. First a batch estimation of the networks weights is carried out by solving a minimisation problem through the Levenberg-Marquardt algorithm. Subsequently, an extended Kalman filter (EKF) is applied for online updating of the network's parameters by using at each discrete-time the input-output sample just measured.

## 2. Recurrent Neural Network for Identification

Let a general deterministic discrete-time non-linear system with $m$ inputs and $p$ outputs to be identified on the basis of input-output sampled data be described in the state-space form by the following equations:

$$\begin{aligned} x(k+1) &= f\left(x(k),u(k),k\right); \quad x(0)=x_0 \\ y(k) &= g\left(x(k),k\right) \end{aligned} \qquad (1)$$

where $f:\Re^n \times \Re^m \times \Re \to \Re^n$ and $g:\Re^n \times \Re \to \Re^p$ are non-linear functions; $u(k)\in \Re^m$, $x(k)\in \Re^n$ and $y(k)\in \Re^p$ are, respectively, the inputs, the states and the outputs at discrete-time $k$.

### 2.1. The Neural Network Topology

Recurrent neural networks are multi-layer networks having feedback loops in a number of neurons so that a particular system's dynamics can be represented and processed by means of a dynamic mapping from the external input to the output. Several recurrent networks architectures such as the Hopfield network [8], the Elman network [9] or the non-linear state-space neural network [10] can be found in literature. In the context of system identification the latter class of neural models are particularly appropriated to describe general non-linear dynamics, given a set of input-output data collected from the plant.
In this paper an affine hybrid neural state-space model comprising linear and non-linear parts is used for capturing the system's dynamics, being the neural network structure depicted in figure 1.

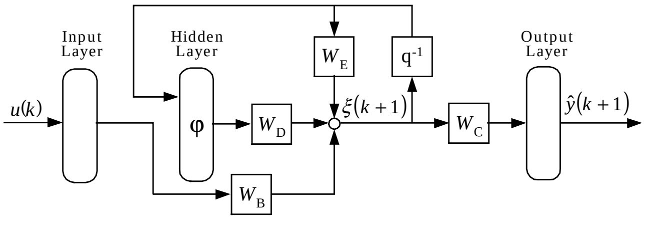



Figure 1. Recurrent neural network.

In this neural network architecture $\xi(k) \in \Re^n$ is the neural state vector, $\hat{y}(k) \in \Re^p$ is the predicted output vector, $W_B, W_C, W_D$ and $W_E$ are weight matrices of appropriate dimensions, which represent the synaptic strengths between neurons, $\varphi$ is a non-linear vector activation function, typically a hyperbolic tangent or sigmoid function and $q^{-1}$ is the backward shift operator.
Considering the neuron activation function for the hidden layer as a hyperbolic tangent function, the neural network dynamics can be described by the following state-space non-linear difference equations:

$$\begin{aligned} \xi(k+1) &= W_D \tanh(\xi(k)) + W_E\, \xi(k) + W_B\, u(k); \quad \xi(0) = \xi_0 \\ \hat{y}(k) &= W_C\, \xi(k) \end{aligned} \tag{2}$$

## 2.1. Learning Methodology

The neural network training comprises two phases: a prior offline learning from which a model is derived and subsequently an online weights updating allowing its improvement. In the first stage a representative set of input-output data is previously collected from the plant and carried out a minimisation of the sum of squared prediction error $J(W) = \sum_{k=1}^{N} \left(y(k) - \hat{y}(k|W)\right)^T \left(y(k) - \hat{y}(k|W)\right)$ by means of the Levenberg-Marquardt algorithm, being the weights updated iteratively according to:

$$\Delta W = -(H + \lambda \mathrm{I})^{-1} \nabla J(W) \tag{3}$$

where $H$ denotes the Hessian of the cost-function $J(W)$, $\nabla J(W)$ is the gradient, $\lambda \in \Re^+$ and $\mathrm{I}$ is an identity matrix of appropriate dimensions.
With respect to the online updating, that is accomplished by using an EKF, as proposed initially by Singhal and Wu [11] and further developed in [12].
In order to apply the filter to the solution of the recursive network learning problem a new state-space representation is constructed as follows [13]:

$$\begin{aligned} W_{k+1} &= W_k + \mu_{k+1} \\ \hat{y}_k &= h(W_k, \mu_k, k) + \upsilon_k \end{aligned} \tag{4}$$

where the process noise $\mu$ is a zero-mean Gaussian white-noise process with positive semi-definite covariance matrix $Q \in \Re^{N_w \times N_w}$, with $N_w$ the weight vector dimension and the measurement noise $\upsilon$ is considered a zero-mean Gaussian white-noise process with positive semi-definite covariance matrix $R \in \Re^{p \times p}$.
The EKF algorithm comprises two groups of equations: time update equations where a one step head prediction is evaluated and measurement update equations providing a correction to the *a priori* estimates taking into account the new information embodied in the most recent measurement.
The specific equations for the weight vector and the covariance matrix of the weight-estimation error predictions are given by:

$$\hat{W}_k^- = \hat{W}_{k-1}$$
$$P_k^- = P_{k-1} + Q \qquad (5)$$

with the "super minus" meaning the time update.
The measurement update equations are defined as:

$$K_k = P_k^- (C_k)^T \left( C_k P_k^- C_k{}^T + R \right)^{-1}$$
$$\hat{W}_k = \hat{W}_k^- + K_k \left( y_k - \hat{y}_k \right) \qquad (6)$$
$$P_k = \left( I - K_k C_k \right) P_k^-$$

with $K_k$ the Kalman gain matrix and $C_k$ given by:

$$C_k = \left. \frac{\partial h(\cdot)}{\partial W} \right|_{W = \hat{W}_k^-} \qquad (7)$$

The essence of the Kalman filter is that it reflects the minimisation of the variance of prediction error and provides in a recursive way a better and better approximation to the solution as new information becomes available.

## 3. Case Study: The Heating System

### 3.1. Process Description

In the laboratory process, depicted in figure 2 air is forced to circulate by a fan blower through a duct, being heated immediately after its inlet. This is a non-linear process with a pure time delay, which happens to depend on the position of the temperature sensor and the air flow rate, depending on the damper position $\Omega$. The system input $u(k)$, is the voltage on the heating device, which consists of a mesh of resistor wires and the output, $y(k)$, is the outlet air temperature.

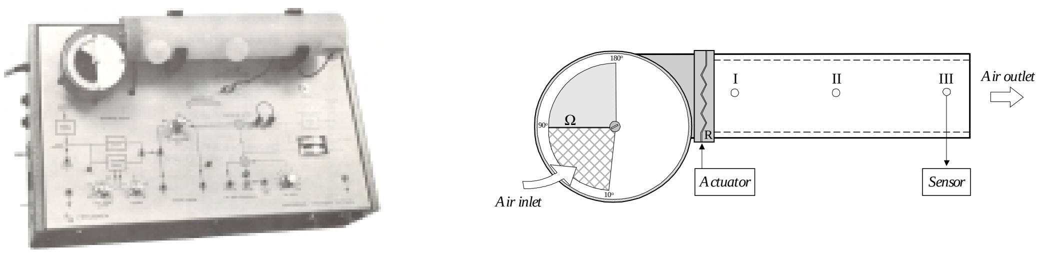
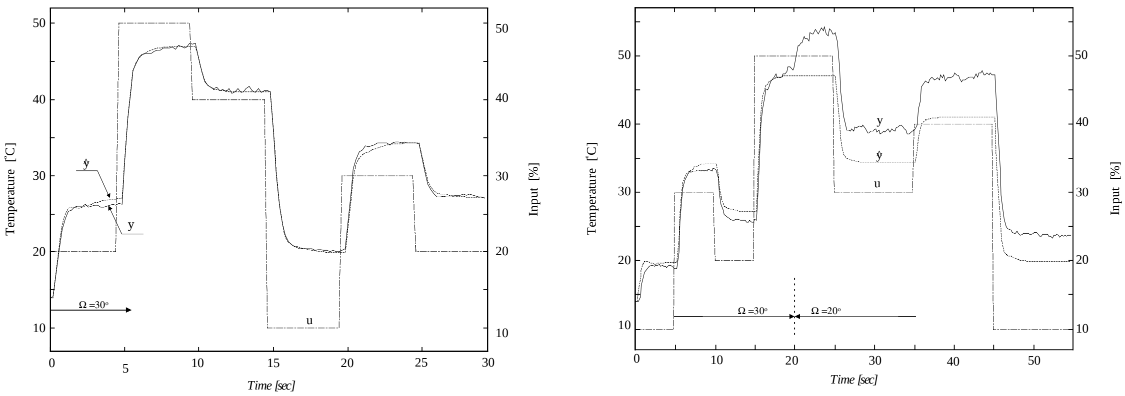



Figure 2. The heating system: picture and schematics.

### 3.2. Experiments

The experiments were designed to assess on the one hand the feasibility of a practical online training by means of an extended Kalman filter and on the other hand to stress the crucial importance of this mechanism when the system's dynamics is changing.

The neural network used for identification purposes has the form of that depicted in figure 1. Both the input layer and the output layer have one neuron whereas the hidden layer contains 3 neurons.
The neural network training algorithms were implemented in Matlab, being used a common PC to run the software and chosen for sampling period 0.2 second.
Concerning the offline identification stage the heating system response to an input sequence consisting in steps of 25 samples was first collected and used subsequently as a training data set. In this experiment the air damper position was set as $\Omega = 30°$ and the sensor element placed in the intermediate location (II).
The neural state-space model parameters were then tuned by applying the Levenberg-Marquardt algorithm to the minimisation of the prediction error. As can be seen from figure 3.a the neural predictor performs quite well in the case of no changes in the system's dynamics. However, when the system undergoes a structural modification resulting in a different dynamics it is evident, as expected, the neural predictor is no longer able of emulating this new behaviour. This evidence was confirmed in a experiment where the position of the air damper changed from $\Omega = 30°$ to $\Omega = 20°$ at the instant 20 seconds and the results plotted in figure 3.b.




a) $\Omega = 30°$ b) $\Omega = 20°$

Figure 3. Neural state-space predictor – batch training.

In order to enable the neural model to reflect changes in the system's dynamics the training procedure must be carried out in real-time as well, that is, as soon as new data from the plant becomes available. For this purposes an EKF has been incorporated in the learning stage. As can clearly be observed from figure 4 the online weights adjust enables the neural predictor to cope with the structural changes affecting the system.

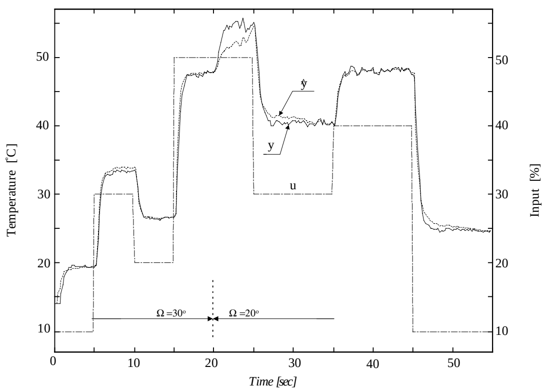



Figure 4. Neural predictor with online weights adaptation.

## 4. Conclusions

In this paper the application of a state-space neural network as a mean for modelling dynamic systems together with an extended Kalman filter to adjust in real-time the weights have been presented. Given the simplicity of this network topology along with the fast training provided by the online learning algorithm it is a viable alternative for real-time system identification, as proved by the experiments conducted on the laboratory plant. In this context, given the adaptive features revealed by the state-space neural network as well as their ability for modelling non-linearities of time-variant plants they bring an added-value to the control field and particularly to those strategies using an explicit model of the plant to be controlled such as the model based predictive control.

## Acknowledgements

This work is partially supported by POSI - Programa Operacional Sociedade de Informação of Portuguese Fundação para a Ciência e Tecnologia and European Union FEDER

## References

[1] K. Narendra and K. Parthasarathy, *Identification and Control of Dynamical Systems Using Neural Networks*, IEEE Trans. On Neural Networks, 1, 1, pp. 4-27, 1990.

[2] K. Hunt, D. Sbarbaro, R. Zbikowski and P. Gawthrop, *Neural Networks for Control Systems – A Survey*, Automatica, 28, 6, pp. 1083-1112, 1992.

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

[4] K. Funashi, *On the approximate realization of continuous mapping by neural networks*, Neural Networks, 2, pp. 183-192, 1989.

[5] L. Jin, P. Nikiforuk and M. Gupta, *Approximation of Discrete-Time State-Space Trajectories Using Dynamic Recurrent Neural Networks*, IEEE Trans. Automatic Control, 40, 7, pp. 1266-1270, 1995.

[6] I. Rivals and L. Personnaz, *Black Box Modelling with State-space Neural Networks*, Neural Adaptive Control Technology, R. Zbikowski and K. Hunt eds., World Scientific, pp. 237-264, 1996.

[7] R. Babuska, J. Sousa and H. Verbruggen, *Predictive Control of Nonlinear Systems based on Fuzzy and Neural Models*, Proc. ECC'99, pp. CA-12-4, 1999.

[8] J. Hopfield, *Neurons with graded response have collective computational properties like those of two-state neurons*, Proc. National Academic of Sciences USA, 81, pp. 3088-3092, 1984.

[9] J. Elman, *Finding Structures in Time*, Cognitive Science, 14, pp. 179-211, 1990.

[10] A. Delgado, C. Kambhampi and K. Warwick, *Dynamic recurrent neural network for system identification and control*, IEE Proc.-Control Theory Appl., 142, 4, pp. 307-314, 1995.

[11] S. Singhal and L. Wu, *Training multilayer perceptrons with the extended Kalman filter*, In advances in Neural Information Processing Systems 1, Morgan Kauffman, pp. 133-140, 1989.

[12] E. Plumer, *Training Neural Networks Using Sequential-Update Forms of The Extended Kalman Filter*, LA-UR-95-422 C.2, Los Alamos National Laboratory, 1995.

[13] E. Wan and A. Nelson, *Dual EKF Methods*, In Kalman Filtering and Neural Networks, ed. Simon Haykin, Wiley Publishing, 2000.