SMCia/05
2005 IEEE Mid-Summer Workshop on Soft Computing in Industrial Applications
Helsinki University of Technology, Espoo, Finland, June 28-30, 2005

Order Estimation in Affine State-Space Neural NetworkS

P. Gil[1, 2], J. Henriques[1], A. Dourado[1] and H. Duarte-Ramos[2]
[1] CISUC - Informatics Engineering Department
Pólo II, Universidade de Coimbra, 3030-290 Coimbra – Portugal
Phone: +351 239790000, Fax: +351 239701266, e.mail: pgil@dei.uc.pt
[2] Electrical Engineering Department
The New University of Lisbon, 2829-516 Caparica – Portugal
Phone: +351 212948545, Fax: +351 212948532

Abstract - The problem of order evaluation for an affine state–space neural network or equivalently the estimation of the number of neurons to be inserted in the hidden layer in a recurrent neural network is here addressed. The proposed method is based on a singular value decomposition applied to an oblique subspace projection given as the projection of the row space of future outputs into the past inputs–outputs row space, along the future inputs row space.

I. INTRODUCTION

The increasing necessity for modelling high complex dynamic systems has been a major driving force towards the development of robust black-box methodologies to deal with nonlinearities and uncertainty. One of the research directions that have been pursued in the context of system identification focus on the synthesis of nonlinear model structures and their approximation capabilities.

Among nonlinear black–box models (see e.g. [1] and references therein) artificial neural networks (or simply neural networks) have attracted considerable attention as an invaluable nonlinear structure choice mainly due to their inherent well known approximation capabilities [2]–[5]. As proven, a three–layered feedforward neural network (MPL) incorporating in the hidden layer a sufficient number of processing units (neurons) with sigmoidal activation function can theoretically approximate to any level of accuracy a given continuous nonlinear mapping.

Although MPL are inherently static structures, they have extensively been applied in modelling and control of nonlinear dynamic systems (see e.g. [7]–[10]). The way these structures are able to approximate spatio–temporal sequences is artificially carried out by providing to the network input layer a sequence of past inputs and outputs. The current neural network output is then given by

$$y(k) = \Phi\big(u(k-n_u),\cdots,u(k-1),u(k), \\ ,y(k-n_y),\cdots,y(k-1)\big) \qquad (1)$$

where $u(k) \in \mathbb{R}^m$ and $y(k) \in \mathbb{R}^n$ represent the input vector and the output of the system at time k and n_u and n_y the lag window orders and Φ a real vector valued function.

One well known drawback of this approach is that it can only encode a finite number of previous inputs and outputs [11]. On the other hand, the so called tapped delay line representation suffers from high sensitiveness to the lag windows not to mention its susceptibility to noisy input and output signals.

An alternative to MPL structures involves the incorporation of feedback connections within the hidden layer. This recurrent neural network structure, which conceptually represents a broader class of nonlinear dynamic systems, not only is less vulnerable to noise, since the input vector provided to the network does not include previous outputs, but also the whole system's history can intrinsically be embedded into the model dynamics. Recurrent neural networks topologies provide universal identification models in the restricted sense that they can approximate uniformly any MIMO nonlinear dynamic system over finite-time intervals for every continuous and bounded input signal [12]–[16]. Applications of recurrent neural networks topologies can be found elsewhere, see e.g. [17]–[21].

An important issue concerning the generalization capabilities of a given neural preditor is the size of the underlying network or, by other words, the number of neurons in the hidden layer. As pointed out by Lawrence and Giles [22] neural networks are intrinsically prone to overfitting as a converging result of an excessive number of weights and the unconstrained minimisation of the empirical loss function [23].

One method known to control the smoothness of the fit is to add a regularization term to the loss function being minimised (see e.g. [24] and references therein). Other techniques, which are devoted to selecting the number of hidden-layer neurons, include the network pruning and constructive methods [25], statistical approaches such as the Network Information Criteria (NIC) [26] and methods based on the application of Vapnik–Chevornenkis dimension [27].

This paper presents a conceptually new approach to the problem of order estimation for state–space neural networks. The proposed method is based on subspace projections and on the formal specificity of the nonlinear model structure. The involved subspaces are accordingly generated using input/output data collected form the system to be modelled and the order extracted by means of a singular value decomposition (SVD) applied to a non–orthogonal space projection.

II. AFFINE NEURAL NETWORK ARCHITECTURE

The general class of discrete-time dynamic neural networks considered in this work comprises three layers, as depicted in Fig.1. As this neural network topology is here regarded as a nonlinear model structure in the context of system identification, the input and output layers must incorporate as much neurons as the number of the inputs and outputs of the system to be modelled. In what the hidden layer is concerned, this particular architecture consists of neurons presenting two types of activation functions, namely, sigmoidal activation functions and linear activation functions. The number of neurons to be incorporated within the hidden layer should be accordingly selected in order to provide good generalization capabilities for the neural predictor. In fact, a neural network consisting of a deficient number of neurons in the hidden layer may not be feasible to appropriately emulate the encapsulated nonlinear dynamics, while a neural network larger than required might suffer from poor generalization performance or overfitting.

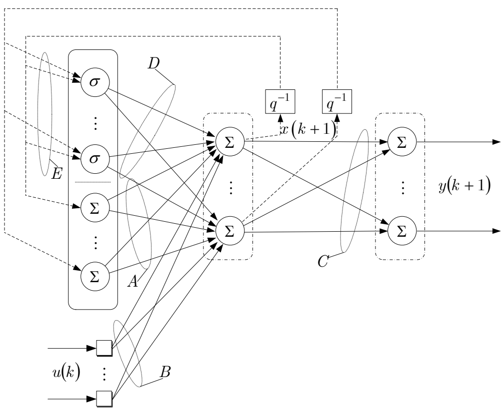

Fig.1. Affine state–space neural network.

The neural network presents in a topological sense hybrid features, which are provided by the two kinds of activation functions (sigmoidal and linear) included in the hidden layer. The incorporation of both neurons is known to improve the neural predictor's modelling performance in case of mild nonlinear dynamics.

Mathematically, the discrete-time nonlinear model can be expressed in state–space form as

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + D\sigma\left(Ex(k)\right) \\ y(k) &= Cx(k) \end{aligned} \tag{2}$$

where $x \in \mathbb{R}^n$ denotes the neural state-space vector, $y \in \mathbb{R}^p$ the neural output and $u \in \mathbb{R}^m$ the input vector. A, B, C, D and E are real-valued matrices of appropriate dimensions. The nonlinear activation function $\sigma(\cdot)$ is here a continuous and differentiable sigmoidal function, upper and lower bounded satisfying the following conditions:

$\lim_{t\to\pm\infty} \sigma(t) = \pm 1$; $\sigma(t) = 0 \Leftarrow t = 0$; $\sigma'(t) > 0$,
$\lim_{t\to\pm\infty} \sigma'(t) = 0$ and $\max\left(\sigma'(t)\right) \le 1$, at $t = 0$.

In [28] the proposed affine state-space neural network structure is used within a model-based predictive control framework and applied to a solar power plant while in [29] some stability results using the Lyapunov theory and the contraction mapping theorem are presented.

III. ORDER ESTIMATION

Consider the affine state-space model described by (2) and assume that the nonlinear term associated to the sigmoidal activation function accomplishes a spatio-temporal compensation to the linear part prediction. Thus, by removing the nonlinear term contribution one comes up with a discrete-time invariant linear system for which it is possible by means of subspace techniques to obtain a vector basis for a given state–space realization. On the other hand, the estimated order can be regarded as a bound on the number of hidden neurons.

A finite dimensional discrete–time linear system can be represented in state–space form as

$$\begin{aligned} x(k+1) &= A\,x(k) + B\,u(k) + \eta(k) \\ y(k) &= C\,x(k) + D\,u(k) + \vartheta(k) \end{aligned} \tag{3}$$

where $x \in \mathbb{R}^n$, $y \in \mathbb{R}^p$, $u \in \mathbb{R}^m$, $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$ and $C \in \mathbb{R}^{p\times m}$. $\eta \in \mathbb{R}^n$ and $\vartheta \in \mathbb{R}^p$ are unobserved, Gaussian distributed, zero mean, white noise vector sequences. Assume also that (3) satisfies the orthogonality property, that is

$$E\left[\begin{pmatrix} x(k) \\ u(k) \end{pmatrix} \begin{pmatrix} \eta^{\mathrm{T}}(k) & \vartheta^{\mathrm{T}}(k) \end{pmatrix}\right] = 0 \qquad (4)$$

In subspace approaches (see e.g. [30]) the central paradigm involves a row space projection of block Hankel matrices generated using the available input-output data. The future input block Hankel matrix $U_{\mathrm{f}} = U_{i|2i\text{-}1}$ is defined as

$$U_{\mathrm{f}} \triangleq \begin{bmatrix} u(i) & u(i+1) & \cdots & u(i+j-1) \\ u(i+1) & u(i+2) & \cdots & u(i+j) \\ \vdots & \vdots & \cdots & \vdots \\ u(2i-1) & u(2i) & \cdots & u(2i+j-2) \end{bmatrix} \qquad (5)$$

while the future output block Hankel $Y_{\mathrm{f}} = Y_{i|2i\text{-}1}$ matrix is given by

$$Y_{\mathrm{f}} \triangleq \begin{bmatrix} y(i) & y(i+1) & \cdots & y(i+j-1) \\ y(i+1) & y(i+2) & \cdots & y(i+j) \\ \vdots & \vdots & \cdots & \vdots \\ y(2i-1) & y(2i) & \cdots & y(2i+j-2) \end{bmatrix} \qquad (6)$$

The number of block rows i should be larger than the "expected" maximum order of the system under identification $(n < i)$ and $j \to \infty$.

Definition 1 (Oblique projection): The oblique projection of the row space of $\mathrm{A} \in \mathbb{R}^{p \times j}$ along the row space of $\mathrm{B} \in \mathbb{R}^{q \times j}$ on the row space of $\Gamma \in \mathbb{R}^{q \times j}$ is given by

$$\mathrm{A} /_{\mathrm{B}} \Gamma = \mathrm{A}\begin{bmatrix} \Gamma^{\mathrm{T}} & \mathrm{B}^{\mathrm{T}} \end{bmatrix} \left[\begin{pmatrix} \Gamma\Gamma^{\mathrm{T}} & \Gamma\mathrm{B}^{\mathrm{T}} \\ \mathrm{B}\Gamma^{\mathrm{T}} & \mathrm{B}\mathrm{B}^{\mathrm{T}} \end{pmatrix}^{\dagger}\right]_{1:r} \Gamma \qquad (7)$$

Theorem 1 [30]: Assume that:

a. The deterministic input is uncorrelated with the process noise and measurement noise;

b. The process noise and the measurement noise are not identically zero;

c. The exogenous input is persistently exciting of order $2i$;

d. The data set is large ($j \to \infty$).

Then

a. The weighted projection Π_i can be defined as the oblique projection of the row space of Y_{f} on the past input/output row space given as

$M_{\mathrm{p}} \triangleq \begin{pmatrix} U_{0|i-1} \\ \hline Y_{0|i-1} \end{pmatrix}$, along the row space of U_{f} ;

$$\Pi_i = Y_{\mathrm{f}} /_{U_{\mathrm{f}}} M_{\mathrm{p}} \qquad (8)$$

b. The order of the system is given by the number of singular values of Π_i different from zero ($n = rank(\Pi_i)$).

After the projection is obtained the singular values can easily be retrieved using a SVD approach,

$$\Pi_i = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^{\mathrm{T}} \\ V_2^{\mathrm{T}} \end{pmatrix} \qquad (9)$$

When applying the above method to finite data-sets embedding nonlinear dynamics the result is a full rank matrix S, irrespective to the number of row blocks i. In this case, the vector space dimension would be $n = p \cdot i$ and so a complexity reduction should be considered in order to get a lower vector basis. The way the problem of complexity reduction is here dealt with involves a comparative magnitude analysis of singular values. This is accomplished by looking at the ratio of each singular value to the largest singular value ($\sigma_{\max}$). Consider the singular value σ_j, $j = 1, \cdots, p \cdot i$ such that the ratio η_j to the largest one is given as

$$\eta_j = \frac{\sigma_j}{\sigma_{\max}} \qquad (10)$$

and define ℓ as the number of significant decimal digits in the data. All singular values for which $\eta_j < 10^{-\ell}$ are associated to the noise subspace and should be accordingly discarded. In case all the computed ratios are above that aforementioned threshold it means that the linear simplification cannot be applied in modelling the dynamics embedded in the data (extremely severe nonlinearities). In these situations other heuristics should be considered, such as cross-validation or constructive techniques.

IV. CASE STUDY

To illustrate the proposed approach for order estimation in the model structure represented here by a state-space neural network it is presented the identification of a distributed solar collector field (DSC). This solar power plant (Fig.2) is located at Plataforma Solar de Almería in the desert of Tabernas, South of Spain.

Fig.2. The distributed solar collector field.

The DSC field (ACUREX) consists of 480 parabolic trough collectors arranged in 20 rows aligned on a West-East axis and forming 10 independent loops. Each solar collector has a linear parabolic-shaped reflector that focuses the sun's beam radiation on a linear absorber tube located at the focus of the parabola. The loops are 172 m long, with an active section of 142 m, while the reflective area of the mirrors is around 264.4 m^2.

The heat transfer fluid used to transport the thermal energy is the Santotherm 55, which is a synthetic oil with a maximum film temperature of 318°C and an auto–ignition temperature of 357°C. The thermal oil is heated as it circulates through the absorber tube before entering the top of the storage tank (Fig.3), while the colder inlet oil is extracted form the bottom of the tank. A three way valve located at the field outlet enables the oil recycling (by-passing the storage tank) until the outlet temperature is high enough to be sent to the storage tank. The thermal energy stored up in the tank can be subsequently used to produce electrical energy in a conventional steam turbine/generator or in the solar desalination plant operation.

In order to follow the yearly variation of the sun's declination, the solar collector field is provided with a sun tracking system, which causes the solar collector to revolve around an axis parallel to the receiver.

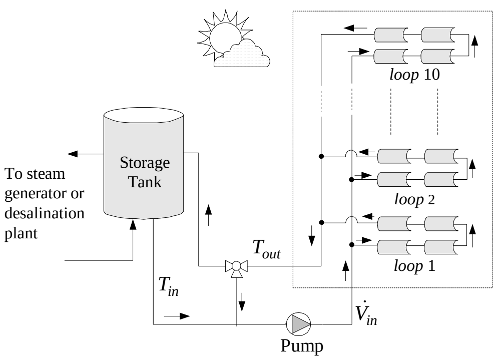

Fig. 3. Distributed solar collector field schematics.

The DSC's dynamics is approximated by using the affine state-space neural network structure proposed in this work. The data-set used in the training process was collected from the ACUREX field on July 23, 2002 (Fig.4).

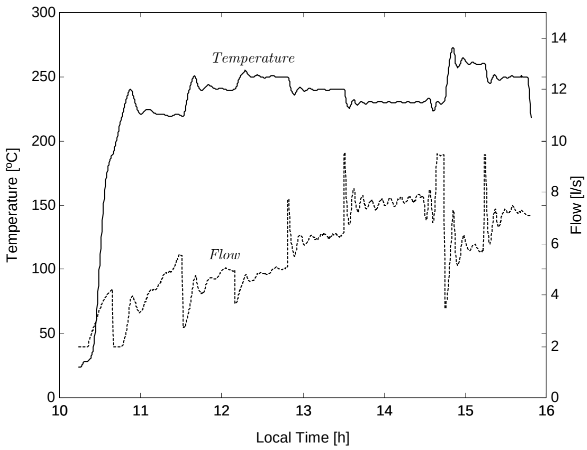

Fig.4. Training set.

By applying the proposed heuristics for order estimation and setting the number of block rows as 15 the following singular value vector σ was obtained.

$$\sigma = \begin{pmatrix} 1279.9 \\ 27.79 \\ 8.5948 \\ 1.0603 \\ 0.6615 \\ 0.4998 \\ 0.3634 \\ 0.3539 \\ 0.3335 \\ 0.3026 \\ 0.2701 \\ 0.2568 \\ 0.1156 \\ 0.0987 \\ 0.0042 \end{pmatrix} \qquad \eta = \begin{pmatrix} 1.0000 \\ 0.0217 \\ 0.0067 \\ 0.0008 \\ 0.0005 \\ 0.0004 \\ 0.0003 \\ 0.0003 \\ 0.0003 \\ 0.0002 \\ 0.0002 \\ 0.0002 \\ 0.0001 \\ 0.0001 \\ 0.0000 \end{pmatrix}$$

Assuming $\ell = 3$, it follows that just the first three entries of the vector η are actually above the prescribed threshold 10^{-3}, and according to the proposed approach the order of the affine state-space model should be selected as 3.

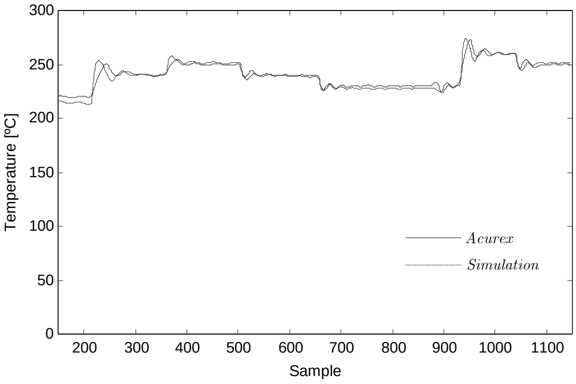

Fig.5. Real plant vs. simulation.

In order to find a parameterization for the neural predictor the neural network was training using the Levenberg–Marquardt algorithm. Figure 5 shows the simulated solar collector field output temperature and the outlet oil temperature measured on the plant. As can be observed the state–space neural predictor with a complexity determined by the proposed approach is able of capturing the system dynamics embedded in the collected data.

V. CONCLUSIONS

A new approach for complexity control in affine recurrent neural networks based on subspace techniques and applied to mild nonlinear systems was presented in this paper. The estimated neural network order can be regarded as a bound on the number of hidden neurons.

ACKNOWLEDGEMENTS

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

REFERENCES

[1] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P. Glorennec, H. Hjalmarsson and A. Juditsky, "Nonlinear Black-Box Modeling in System Identification: A unified Overview," Automatica, vol. 31, no. 12, pp. 1691-1724, 1995

[2] G. Cybenko, "Approximation by Superpositions of a Sygmoidal Function," Mathematics of Control, Signal and Systems, vol. 2, pp. 303-310, 1989.

[3] K. Funahashi, "On the approximate realization of continuous mappings by neural networks". *Neural Networks*, vol. 2, pp. 183-192, 1989.

[4] K. Hornik, M. Stinchcombe and H. White, "Multilayer Feedforward Networks are Universal Approximators," *Neural Networks*, vol.2, pp. 359-366, 1989.

[5] M. Leshno, V. Lin, A. Pinkus and S. Schocken, "Multilayer Feedforward Networks with a Non-Polynomial Activation Function Can Approximate any Function," Neural Networks, vol. 6, pp. 861-867, 1993.

[6] D. Pham and L. Xing, "Neural Networks for identification, Prediction and Control," Springler-Verlag London Limited, 1995.

[7] S. Chen, S. Billings and P. Grand, "Non-linear System Identification Using Neural Networks," Int. Journal of Control, vol. 51, pp. 1191-1214, 1990.

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

[9] H. Siegelmann, B. Horne and L. Giles, "Computational Capabilities of Recurrent NARX Neural Networks," IEEE Trans. On Systems, Man and Cybernetics-Part B, vol. 27, no. 2, pp. 208-2151997.

[10] B. Lennox, G. Montague, A. Frith, C. Gent and V. Bevan, "Industrial Application of Neural Networks – An Investigation," Journal Process Control, 11, pp. 497-507, 2001.

[11] G. Puskorius and L. Feldkamp, "Neurocontrol of nonlinear Dynamical Systems with Kalman Filter Trained Recurrent Networks," IEEE Trans. On Neural Networks, vol. 5, no. 2, 1994.

[12] E. Sontag, "Recurrent Neural Netwoks: Some Systems-Theoretic Aspects," in M. Kamy, K. Warwick and V. Kurkova, eds., Dealing with Complexity: A Neural Network Approach, Springler-Verlag, 1997.

[13] L. Jin, M. Gupta and P. Nikiforuk, "Approximation Capability of Continuous-Time Dynamic Recurrent Neural Networks," Neural, Parallel & Scientific Computations, 9, pp. 239-266, 2001.

[14] L. Jin, M. Gupta and P. Nikiforuk, "Approximation of Discrete-Time State-Space Trajectories Using Dynamic Recurrent Neural Networks," IEEE Trans. On Automatic Control, vol. 4, no. 7, pp. 1266-1270, 1995

[15] L. Li, "Approximation Theory and Recurrent Networks," Proc. of IJCNN, vol.2, pp. 266-271, 1992.

[16] M. Garzon and F. Botelho, "Dynamical Approximation by Recurrent Neural Networks," Neurocomputing, 29, 25-46, 1999.

[17] J. Zamarreño, P. Vega, L. Garcia and M. Francisco, "State-space Neural Network for Modelling, Prediction and Control," Control Engineering Practice, 8, pp. 1063-1075, 2000.

[18] W. Maass and E. Sontag, " Neural Networks as Nonlinear Filters," Neural Computation, 12, pp. 1743-1772, 2000

[19] C. Gan and K. Danai, "Mode-Based Recurrent Neural Network for Modeling Nonlinear Systems," IEEE Trans. On Systems, Man and Cybernetics, vol. 30, no. 2, pp. 344-351, 2000

[20] Y. Zhang and J. Wang, "Obstacle Avoidance for Kinematically Redundant Manipulators Using a Dual Neural Network," IEEE Trans. On Systems, Man and Cybernetics, vol. 34, no. 1, pp. 752-759, 2004

[21] A. Malhi and Robert Gao, "Recurrent Neural Networks for Long-Term Prediction in Machine Condition Monitoring," IMTC'04, Como, Italy, 2004

[22] S. Lawrence and C. Giles, "Overfitting and Neural Networks: Conjugate Gradient and Backpropagation," Proc. of IEEE-IJCNN, pp. 114-119, Como, Italy, 2000.

[23] A. Barron, "Statistical Properties of Artificial Neural Networks," Proc. of the 28th Conference on Decision and Control, pp. 280-285, 1989.

[24] L. Wu and J. Moody, "A Smoothing Regularizer for Feeforward and Recurrent Neural Networks," Neural Computation, 8 (3), pp. 463-491, 1996.

[25] M Costa, A. Braga and B. Menezes, "Constructive and Pruning Methods for Neural Network Design," Proc. of the IEEE-SBRN'02, pp. 49-54, 2002.

[26] N. Murata, S. Yoshizawa and S. Amari, "Network Information Criteria–Determining the Number of Hidden Units for an Artificial Neural Network Model," IEEE Trans. On Neural Networks, vol. 5, no. 6, pp. 865-872, 1994.

[27] V. Cherkassky, X. Shao, F. Mulier and V. Vapnik, "Model Complexity Control for Regression Using VC Generalization Bounds," IEEE Trans. On Neural Networks, vol. 10, no. 5, pp. 1075-1089, 1999.

[28] P. Gil, J. Henriques, P. Carvalho, H. Duarte-Ramos and A. Dourado, "Adaptive Model-Based Predictive Control with Steady State Offset Compensation for a Distributed Solar Collector Field," IEEE Int. Conf. Neural Networks & Signal Processing, pp. 283-288, 2003.

[29] P.Gil, J. Henriques, A. Dourado and H. Duarte-Ramos, "Stability Analysis for a Class of Affine State-Space Neural Networks," 12th Mediterranean Conference on Control and Automation, Kusadasi, Turkey, 2004.

[30] B. De Moor, P. Overschee and W. Favoreel, "Algorithms for Subspace State Space Identification – An Overview," In Biswa Datta eds., Birkhauser Book Series on Applied and Computation Control, Signals and Circuits, pp. 247-311, 1999.