CODEN STJSAO ISSN 0562-1887
ZX470/1569 UDK 62(05)=862=20=30

# Nonlinear Structural Behaviour Identification using Digital Recurrent Neural Networks

***Vesna RANKOVIĆ*** [1], ***Nenad GRUJOVIĆ*** [1], ***Dejan DIVAC*** [2], ***Nikola MILIVOJEVIĆ*** [2] **and** ***Radovan SLAVKOVIĆ*** [1]

1) Fakultet inženjerskih nauka, Univerzitet u Kragujevcu (Faculty of Engineering, University of Kragujevac), Sestre Janjić 6, 34000 Kragujevac, **Republic of Serbia**
2) Institut za vodoprivredu "Jaroslav Černi" (Institute for the Development of Water Resources "Jaroslav Černi"), Jaroslava Černog 80, 11000 Beograd, **Republic of Serbia**

vesnar@kg.ac.rs

Original scientific paper

Dynamical systems contain nonlinear relations which are difficult to model with conventional techniques. Hence, efficient nonlinear models are needed for system analysis, optimization, simulation and diagnosis of nonlinear systems. In recent years, computational-intelligence techniques such as neural networks, fuzzy logic and combined neuro-fuzzy systems algorithms have become very effective tools in the field of structural identification. The problem of the identification consists of choosing an identification model and adjusting the parameters in an way that the response of the model approximates the response of the real system to the same input. This paper investigates the identification of a nonlinear system by Digital Recurrent Neural Network (DRNN). A dynamic backpropagation algorithm is employed to adapt weights and biases of the DRNN. Mathematical model based on experimental data is developed. Results of simulations show that the application of the DRN for the identification of complex nonlinear structural behaviour gives satisfactory results.

**Keywords**

*Identification*
*Nonlinear system*
*Structural behaviour*
*Digital recurrent neural network*
*Radial displacement*

**Ključne riječi**

*Identifikacija*
*Nelinearan sustav*
*Strukturno ponašanje*
*Digitalna povratna neuronska mreža*
*Radijalno pomicanje*

**Primljeno (Received):** 2011-10-10
**Prihvaćeno (Accepted):** 2011-12-21

**Identifikacija nelinearnog strukturnog ponašanja pomoću digitalne povratne neuronske mreže**

Izvornoznanstveni članak

Dinamički sustavi sadrže nelinearne veze koje se teško modeliraju konvencionalnim tehnikama. Nelinearni modeli su neophodni za analizu sustava, optimizaciju, simulaciju i dijagnostiku nelinearnih sustava. Prethodnih godina, tehnike računalne inteligencije kao što su neuralne mreže, fuzzy logika i kombinirani neuro-fuzzy sustavi postaju efikasni alati u identifikaciji nelinearnih objekata. Problem identifikacije se sastoji od izbora identifikacijskog modela i prilagođavanja parametara tako da odziv modela aproksimira odziv realnog sustava za isti ulaz.Ovaj rad proučava identifikaciju nelinearnih sustava pomoću digitalne povratne neuronske mreže. Dinamički algoritam s propagacijom pogreške unazad se primjenjuje za adaptaciju težina i pragova osjetljivosti DRNN. Matematički model se razvija na bazi eksperimentalnih podataka. Rezultati simulacija pokazuju da primjena DRN u identifikaciji kompleksnog nelinearnog strukturnog ponašanja daje zadovoljavajuće rezultate.

## 1. Introduction

Nonlinear system identification and prediction is a complex task. On the other hand, all processes in nature are nonlinear. In many processes, nonlinearities are not prominent, so their behaviour can be described by simple linear models. In the linear systems theory there exist a large number of methods that can be applied for obtaining the linear model of processes. In contrast, nonlinear models must be selected when strong nonlinearities are present.

Neural network modelling and identification from experimental data are effective tools for approximation of uncertain nonlinear dynamic systems. Neural networks can be classified into two major categories: feedforward and recurrent. Most publications in nonlinear system identification use feedforward networks, for example multilayer perceptrons [1]. Furthermore, feedforward neural networks have been applied successfully to a variety of classification problems [2] and for the development of the time prediction models [3]. The main drawback of these neural networks is that the weight updating does not utilize any information on the local data structure and the function approximation is sensitive to the training data [4]. The feedforward neural networks trained with a standard back-propagation algorithm can be used for the identification of nonlinear dynamic systems [5].

**Symbols/Oznake**

| | | | |
|---|---|---|---|
| $y_m(k)$ | - output of the model<br>- izlaz modela | $\eta$ | - update rate<br>- brzina učenja |
| $u(k)$ | - input of the model<br>- ulaz modela | $n_e$ | - maximum lag of the error<br>- maksimalno kašnjenje pogreške |
| $k$ | - time instant<br>- vremenski korak | $n_H$ | - number of hidden nodes<br>- broj neurona u nevidljivom sloju |
| $n_u$ | - maximum lag of the input<br>- maksimalno kašnjenje ulaza | | |
| $n_y$ | - maximum lag of the output<br>- maksimalno kašnjenje izlaza | | **Greek letters/Grčka slova** |
| $f_m$ | - unknown nonlinear function<br>- nepoznata nelinearna funkcija | $\boldsymbol{\varphi}(k)$ | - regression vector<br>- regresijski vektor |
| $e(k)$ | - prediction error<br>- pogreška predviđanja | $\boldsymbol{\theta}$ | - parameter vector<br>- vektor parametara |

Neural network-based metamodels have been applied for determining the dynamic characteristic parameters of structures from field measurement data [6]. Conventional back-propagation is used to train the neural network. However, the conventional back-propagation algorithm has the problems of local minima and slow rate of convergence.

An improvement to the back-propagation algorithm based on the use of an independent, adaptive learning rate parameter for each weight with adaptable nonlinear function is presented in [7]. Adaptive time delay neural network structures have been proposed with satisfactory modelling accuracy [8]. The dynamic neural networks have been applied to functional approximation and nonlinear system identification [9-11]. Predictions of the structural behaviour by fuzzy neural network methodology [12] and dynamic fuzzy wavelet neural network [13] have been considered. The NARMAX-(nonlinear autoregressive moving average with exogenous inputs) approach is used for mapping the input–output relationship.

System identification plays a key role in active control of civil infrastructures. Because of their wide applicability, system identification methods have been studied in civil engineering for various purposes [14]. The main contribution and originality of this paper is to develop a DRNN identification model for time-varying behaviour prediction of civil engineering structures. Dynamic backpropagation algorithm is used to adapt weights and biases. The paper considered the application of DRNN to predict the radial displacement of an arch dam. Dam deformation modelling is very important for its safety monitoring. Radial displacement in any point of the dam body is a nonlinear function of hydrostatic pressure, temperature and other unexpected unknown causes. In dam engineering, soft computing models have been developed for the prediction of dam displacements [15-18]. In this paper, recurrent neural network approach is used for nonlinear system identification. Recurrent networks have been shown more efficient than feedforward neural networks in terms of the number of neurons required to model a dynamic system [19,20]. Models with recurrent networks are shown to have the capability of capturing various system nonlinearities, [21,22].

## 2. Identification of nonlinear dynamic systems

Different methods have been developed in the literature for nonlinear system identification that use a parameterized model. The parameters are updated to minimize an output identification error.

$$y_m(k) = f_m\left(\boldsymbol{\varphi}(k), \boldsymbol{\theta}\right) \tag{1}$$

Since $f_m$ can have a variety of forms, the identification of nonlinear systems becomes a much more complex task than for linear systems, where the key difficulty is to determine the system order [23].

Depending on the choice of the regressors in $\boldsymbol{\varphi}(k)$, different models can be derived:

***NFIR*** (Nonlinear Finite Impulse Response) model:

$$\boldsymbol{\varphi}(k) = \left(u(k-1), u(k-2), ..., u(k-n_u)\right)$$

***NARX*** (Nonlinear AutoRegressive with eXogenous inputs) model:

$$\boldsymbol{\varphi}(k) = \left(u(k-1), u(k-2), ..., u(k-n_u),\right.$$
$$\left. y(k-1), y(k-2), ..., y(k-n_y)\right)$$

***NARMAX*** (Nonlinear AutoRegressive Moving Average with eXogenous inputs) model:

$$\boldsymbol{\varphi}(k) = \left(u(k-1), u(k-2), ..., u(k-n_u),\right.$$
$$y(k-1), y(k-2), ..., y(k-n_y),$$
$$\left. e(k-1), e(k-2), ..., e(k-n_e)\right)$$

***NOE*** (Nonlinear Output Error) model:

$$\boldsymbol{\varphi}(k) = \left(u(k-1), u(k-2), ..., u(k-n_u),\right.$$
$$\left. y_m(k-1), y_m(k-2), ..., y_m(k-n_y)\right)$$

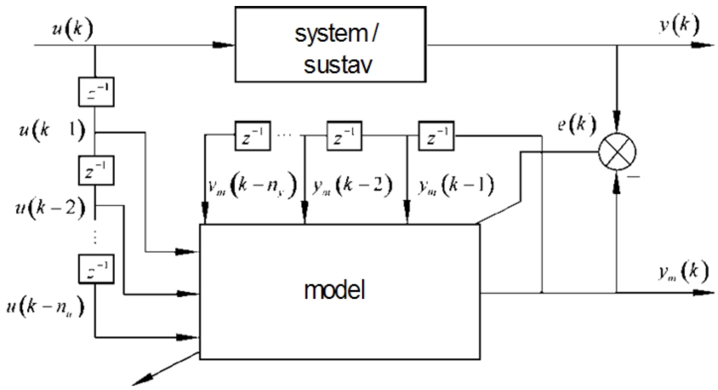


**Figure 1.** The general block schema of the NOE model

**Slika 1.** Općenita blok shema NOE modela

***NBJ*** (Nonlinear Box-Jenkins) model: uses all four regressor types.
In this paper, NOE model (Figure 1) is used for representation of nonlinear systems.

## 3. DRN neural network for nonlinear system identification

Figure 2 depicts a typical configuration of a DRN. The output of the network is feedback to its input. This is a realization of the NOE model. The output of the network is a function not only of the weights, biases and network input, but also of the outputs of the network at previous points in time. Dynamic backpropagation algorithm can be used to adapt weights and biases [24].
DRN network is composed of a nonlinear hidden layer and a linear output layer. The inputs $u(k-1)$, $u(k-2)$ ,..., $u(k-n_u)$ are multiplied by weights $\omega_{u_{ij}}$, while outputs $y_m(k-1)$, $y_m(k-2),\ldots,y_m(k-n_y)$ are multiplied by weights $\omega_{y_{ij}}$ and summed at each hidden node. Then the summed signal at a node activates a nonlinear function. The hidden neurons activation function is the hyperbolic tangent sigmoid function. In Figure 2, $\omega_i$ represents the weight that connects the node $i$ in the hidden layer and the output node; $b_i$ represents the biased weight for $i$-th hidden neuron and $b$ is a biased weight for the output neuron.
The output of the network is:

$$y_m(k)=\sum_{i=1}^{n_H}\omega_i \nu_i + b \tag{2}$$

where:

$$\nu_i=\frac{e^{n_i}-e^{-n_i}}{e^{n_i}+e^{-n_i}} \tag{3}$$

$$n_i=\sum_{j=1}^{n_u}u(k-j)\omega_{u_{ij}}+\sum_{j=1}^{n_y}y_m(k-j)\omega_{y_{mij}}+b_i \tag{3}$$

The difference between the output of the system $y(k)$ and the output of the network $y_m(k)$ is called the prediction error:

$$e(k)=y(k)-y_m(k) \tag{4}$$

This error is used to adjust the weights and biases in the network via the minimization of the following function:

$$\varepsilon=\frac{1}{2}\left[y(k)-y_m(k)\right]^2 \tag{5}$$

Using the gradient descent, the weight and bias updating rules can be described as:

$$\omega_{u_{ij}}(k+1)=\omega_{u_{ij}}(k)-\eta\frac{\partial\varepsilon}{\partial\omega_{u_{ij}}} \tag{6}$$

$$\omega_{y_{mij}}(k+1)=\omega_{y_{mij}}(k)-\eta\frac{\partial\varepsilon}{\partial\omega_{y_{mij}}} \tag{7}$$

$$b_i(k+1)=b_i(k)-\eta\frac{\partial\varepsilon}{\partial b_i} \tag{8}$$

$$b(k+1)=b(k)-\eta\frac{\partial\varepsilon}{\partial b} \tag{9}$$

where:

$$\frac{\partial\varepsilon}{\partial\omega_{u_{ij}}}=\frac{\partial^e\varepsilon}{\partial y_m}\frac{\partial y_m}{\partial\omega_{u_{ij}}};\quad \frac{\partial\varepsilon}{\partial\omega_{y_{ij}}}=\frac{\partial^e\varepsilon}{\partial y_m}\frac{\partial y_m}{\partial\omega_{y_{ij}}};$$

$$\frac{\partial\varepsilon}{\partial b_i}=\frac{\partial^e\varepsilon}{\partial y_m}\frac{\partial y_m}{\partial b_i};\quad \frac{\partial\varepsilon}{\partial b}=\frac{\partial^e\varepsilon}{\partial y_m}\frac{\partial y_m}{\partial b}$$

where the superscript $e$ indicates an explicit derivative, not accounting for indirect effects through time.

The terms $\frac{\partial y_m}{\partial\omega_{u_{ij}}}$, $\frac{\partial y_m}{\partial\omega_{y_{ij}}}$, $\frac{\partial y_m}{\partial b_i}$ and $\frac{\partial y_m}{\partial b}$ must be propagated forward through time [24].

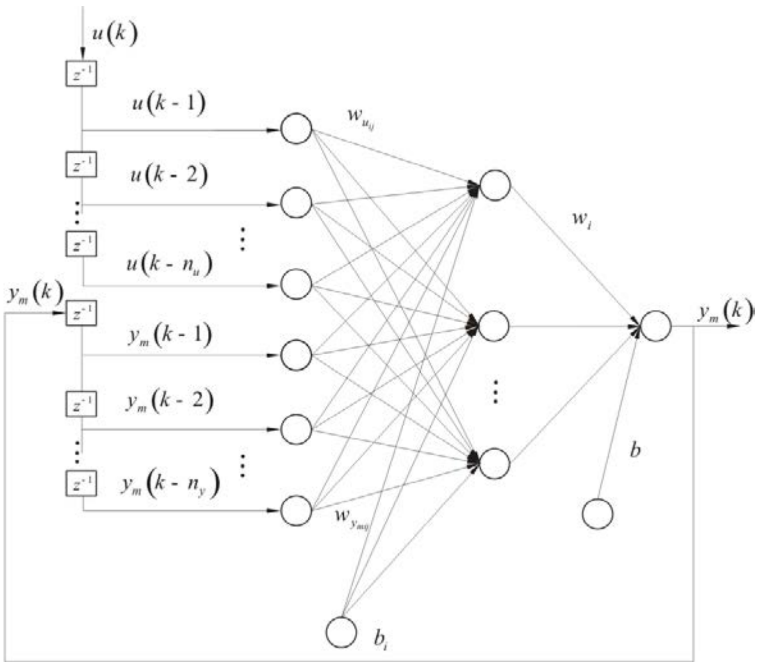


**Figure 2.** Digital Recurrent Neural Network

**Slika 2.** Digitalna povratna neuronska mreža

## 4. Simulation results

The main aim of this study is to construct an efficient DRNN model to predict the radial displacement at the crest of an arch dam. One arch dam after 30 years of operation, is shown in Figure 3. It is a double curvature arch dam, 66 m high, with 221.4 m long crest. The minimum, normal and maximum operating levels are 254, 282 and 283 m above sea level (asl), respectively. The total capacity of reservoir is $52.7 \cdot 10^6 \times \text{m}^3$. The displacement of point V1 at block 8 is predicted with the proposed method. A data set includes 783 data samples. The available set of data was divided into two subsets as training and test sets. Five-day measurements from January 2000 to December 2008 are used to train, and data from January 2009 to December 2010 are used to test the DRNN model.

The behaviour of nonlinear dynamic system with two inputs and one output is considered. The model input vector is defined by:

$$\boldsymbol{\varphi}(k) = \left(u_1(k-1), u_2(k-1), y_m(k-1)\right)$$

where $u_1$ is water level, $u_2 = \dfrac{2\pi j}{365}$ is the season varying between 0 and $2\pi$, $j$ represents the number of days since January 1st.

MATLAB Neural Network Toolbox has been applied for the implementation of the digital recurrent network network. Different DRNN models were constructed and tested in order to determine the optimum number of neurons in the hidden layer. The two-layer network with a tan-sigmoid transfer function at the hidden layer and a linear transfer function at the output layer was used. Optimal network size was selected as the one which obtained in maximum correlation coefficient for the training and test sets, Table 1.

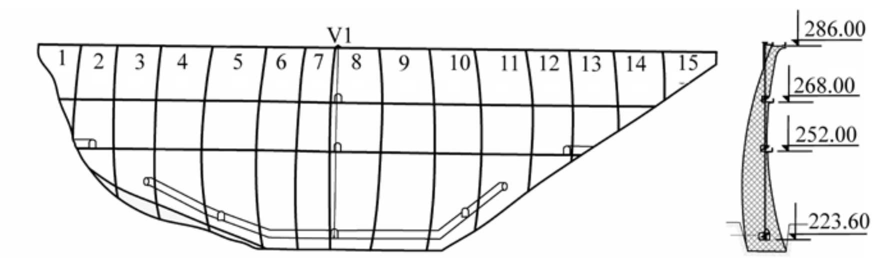


**Figure 3.** Upstream face of dam and cross-section through block 8

**Slika 3.** Uzvodno lice brane i poprečni presjek bloka 8

**Table 1.** Correlation coefficient for the training and test sets

**Tablica 1**.Korelacijski koeficijenti za skupove podataka za učenje i testiranje

| DRNN-structure / DRNN struktura | 3-20-1 | 3-24-1 | 3-27-1 | 3-30-1 |
|---|---|---|---|---|
| Training / Učenje | 0.889 | 0.936 | 0.974 | 0.966 |
| Test / Testiranje | 0.877 | 0.943 | 0.972 | 0.965 |

Based on Table 1, it was concluded that the optimal number of hidden neurons is 27. The total number of the parameters of DRN network is 136. In the learning process, the weights (108) as well as the biases (28) of the neural network were adaptively adjusted. Parameters of the DRN are given in Table 2. The inputs $u_1(k-1)$, $u_2(k-1)$ are multiplied by weights $\omega_{u_{i1(1)}}$ and $\omega_{u_{i1(2)}}$, respectively.

Figure 4 presents the measured and DRNN computed values in training and test sets.

## 5. Conclusion

Dynamical systems contain nonlinear relations which are difficult to model with conventional techniques. In this paper, a DRNN has been successfully applied to unknown nonlinear system identification and modelling. A real-data set was used to demonstrate the effectiveness of the proposed approach. Comparing the modelled values by DRNN with the experimental data indicates that the soft computing model provides accurate results. Designing a neural network model, the main problem is how to determine an optimal architecture of the network and how to achieve an optimal fine tuning of its parameters. The number of DRNN inputs is determined by the number of time lags. The determination of the values of time lags is an open issue. Large time lags result in better prediction of the DRNN.

**Table 2.** Parameters of the DRN network

**Tablica 2.** Parametri DRN mreže

| $i$ | $\omega_{u_{i1(1)}}$ | $\omega_{u_{i1(2)}}$ | $\omega_{y_{mi1}}$ | $\omega_i$ | $b_i$ |
|---|---|---|---|---|---|
| 1 | 0.4278 | 1.0918 | 1.5638 | −0.385 | 3.1645 |
| 2 | 0.1029 | −2.0117 | −0.8312 | −0.5376 | −1.056 |
| 3 | 0.0104 | −1.1102 | 0.4498 | 0.8453 | 0.4556 |
| 4 | 0.1729 | −0.9834 | −0.837 | −1.6245 | −2.674 |
| 5 | −1.7569 | 0.2158 | −1.379 | −0.5386 | 1.4897 |
| 6 | 0.2006 | −1.0001 | 0.7629 | 2.2895 | 2.4756 |
| 7 | −0.2252 | −0.4567 | 1.2987 | −2.5739 | 1.7865 |
| 8 | 1.5782 | 1.2298 | −0.9927 | 1.1086 | −3.4392 |
| 9 | −0.0011 | 0.0012 | −1.1557 | −0.0731 | −1.3629 |
| 10 | 0.0303 | −0.2001 | 0.6623 | −0.9328 | 2.5528 |
| 11 | −0.9741 | 0.9978 | −1.2984 | 2.3289 | 0.3695 |
| 12 | 0.0005 | −0.3789 | 0.5309 | −1.4473 | −1.7547 |
| 13 | −0.2301 | −1.0602 | −0.1287 | −0.2459 | 0.2428 |
| 14 | 0.8002 | −0.0015 | 0.9291 | 0.7843 | −0.761 |
| 15 | 1.3829 | −0.5009 | 2.5289 | −1.2904 | −2.2331 |
| 16 | 0.5389 | 0.0091 | −0.3897 | 3.1068 | −1.1278 |
| 17 | −0.2946 | −1.4916 | 1.1982 | −0.0762 | 1.8804 |
| 18 | 0.2943 | 0.8856 | −0.8856 | 0.9178 | 0.4379 |
| 19 | −0.5204 | 0.0203 | −0.2901 | −0.9924 | −3.6644 |
| 20 | 0.8463 | 0.3825 | 0.6781 | −0.7827 | 1.8575 |
| 21 | −0.6634 | −0.1768 | 1.2754 | −1.0045 | −1.4672 |
| 22 | −0.4639 | 0.3579 | 0.9783 | 0.3901 | −1.4741 |
| 23 | 0.1156 | −1.6381 | −0.3495 | 2.2678 | 0.4829 |
| 24 | 0.2474 | −0.3589 | −0.2983 | −1.1338 | 0.6616 |
| 25 | 0.9178 | −0.2493 | −0.9506 | −0.7071 | −3.3939 |
| 26 | −0.0017 | 0.2537 | 1.4887 | −2.2448 | 1.6728 |
| 27 | 0.4567 | −1.3587 | −0.9508 | 0.3491 | −1.3785 |

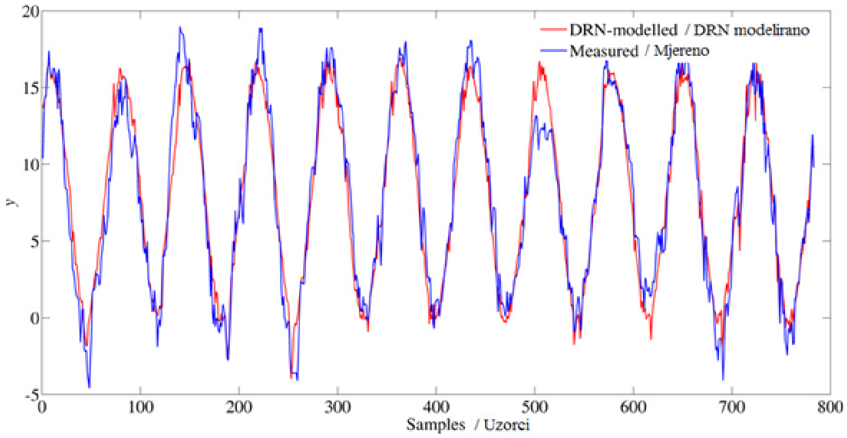


**Figure 4.** The measured and modelled values in training and test sets

**Slika 4.** Mjerene i modelirane vrijednosti za učenje i testiranje

However, large $n_u$ and $n_y$ also result in large number of parameters (weights and biases) that need to be adapted. In the considered dam study, satisfactory results were obtained for $n_{u_1} = n_{u_2} = n_y = 1$.

The main limitation of the methodology presented here is that it does not directly consider mechanical properties and possible damage. Additional analysis in the form of statical and dynamical tests, computational mechanical modelling and inverse analysis are required.

## Acknowledgements

The part of this research is supported by Ministry of Science in Serbia, Grants III41007 and TR37013.

## REFERENCES

[1] YU, W.: *Nonlinear system identification using discrete-time recurrent neural networks with stable learning algorithms*, Information Sciences, 158 (2004) 1, 131-147.

[2] ŠIMUNOVIĆ, K.; ŠIMUNOVIĆ, G.; ŠARIĆ, T.: *Application of artificial neural networks to multiple criteria inventory classification,* Strojarstvo, 51(2009) 4, 313-321.

[3] ŠIMUNOVIĆ, G.; BALIČ, J.; ŠARIĆ, T.; ŠIMUNOVIĆ, K.; LUJIĆ R.; SVALINA, I.: *Comparison of the technological time prediction models,* Strojarstvo, 52 (2010) 2, 137-145.

[4] JIN, L.; GUPTA, M.M.: *Stable dynamic backpropagation learning in recurrent neural networks*, IEEE Transactions on Neural Networks, 10 (1999) 6, 1321–1334.

[5] NARENDRA, K.S.; PARTHASARTHY, K.: *Identification and control of dynamical systems using neural networks*, IEEE Transactions on Neural Networks, 1 (1990) 1, 4-27.

[6] CHEN, C.H.: *Structural identification from field measurement data using a neural network*, Smart Materials and Structures, 14 (2005) 3, S104-S115.

[7] GUPTA, P.; SINHA, N.K.: *An improved approach for nonlinear system identification using neural networks*, Journal of the Franklin Institute, 336 (1999) 4, 721-734.

[8] YAZDIZADEH, A.; KHORASANI, K.: *Adaptive time delay neural network structures for nonlinear system identification*, Neurocomputing, 47 (2002) 1-4, 207-240.

[9] HAN, X.; XIE, W. F.; FU, Z.; LUO, W.: *Nonlinear systems identification using dynamic multi-time scale neural networks*, Neurocomputing, 74 (2011) 17, 3428-3439.

[10] YU, W.; LI, X.: *Some new results on system identification with dynamic neural networks*, IEEE Transactions on Neural Networks, 12 (2001) 2, 412-417.

[11] XIE, W.F.; ZHU, Y.Q.; ZHAO, Z.Y.; WONG, Y.K.: *Nonlinear system identification using optimized dynamic neural network*, Neurocomputing, 72 (2009) 13-15, 3277-3287.

[12] GAO, Y.; E.R, M.J.: *NARMAX time series model prediction: feedforward and recurrent fuzzy neural network approaches*, Fuzzy Sets and Systems, 150 (2005) 2, 331-350.

[13] ADELI, H.; ASCE, F.; JIANG, X.: *Dynamic fuzzy wavelet neural network model for structural system identification*, Journal of Structural Engineering, 132 (2006) 1, 102-111.

[14] TANG, H.; XUE, S.; FAN, C.: *Differential evolution strategy for structural system identification*, Computers and Structures, 86 (2008) 21-22, 2004-2012.

[15] MATA, J.: *Interpretation of concrete dam behaviour with artificial neural network and multiple linear regression models*, Engineering Structures, 33 (2011) 3, 903-910.

[16] CAO, M.; QIAO, P.; REN, Q.: *Improved hybrid wavelet neural network methodology for time-varying behavior prediction of engineering structures*, Neural Computing & Applications, 18 (2009) 7, 821-832.

[17] SU, H.; WU, Z.; WEN, Z.: *Identification model for dam behavior based on wavelet network*, Computer-Aided Civil and Infrastructure Engineering, 22 (2007) 6, 438-448.

[18] DEMIRKAYA, S.: *Deformation analysis of an arch dam using ANFIS*, Second International Workshop on Application of Artificial Intelligence and Innovations in Engineering Geodesy - Proceedings of papers, A. Reiterer, U. Egly, M. Heinert, B. Riedel, (eds.), Braunschweig, Germany, 21-31, 2010.

[19] AL SEYAB, R.K.; CAO, Y.: *Nonlinear system identification for predictive control using continuous time recurrent neural networks and automatic differentiation*, Journal of Process Control, 18 (2008) 6, 568-581.

[20] HUSH, D.R.; HORNE, B.G.: *Progress in supervised neural networks*, IEEE Signal Processing Magazine, 10 (1993) 1, 8-39.

[21] FUNAHASHI, K. L.; NAKAMURA, Y.: *Approximation of dynamical systems by continuous time recurrent neural networks*, Neural Networks, 6 (1993) 6, 183-192.

[22] JIN, L.; NIKIFORUK, P.; GUPTA, M.: *Approximation of discrete-time state-space trajectories using dynamic recurrent neural networks*, IEEE Transactions on Automatic Control, 40 (1995) 7, 1266-1270.

[23] MENDES E.; BILLINGS, S.: *An alternative solution to the model structure selection problem,* IEEE Transactions on Systems, Man, and Cybernetics—PART A: Systems and Humans, 31 (2001) 6, 597-608.[24] HAGAN, M.; JESUS, O.D.; SCHULTZ, R.: *Training Recurrent Networks for Filtering and Control,* Chapter 11 of Recurrent Neural Networks: Design and Applications, L.R. Medsker and L.C. Jain, Eds., CRC Press, (1999), pp 325-354.