Computing and Informatics, Vol. 26, 2007, 649–661

USE OF AUTOREGRESSIVE PREDICTOR IN ECHO STATE NEURAL NETWORKS

Štefan Babinec

Department of Mathematics, Faculty of Chemical and Food Technology
Slovak University of Technology, 812 37 Bratislava, Slovakia
e-mail: stefan.babinec@stuba.sk

Manuscript received 15 May 2007; revised 31 May 2007
Communicated by Vladimír Kvasnička

Abstract. "Echo State" neural networks (ESN), which are a special case of recurrent neural networks, are studied with the goal to achieve their greater predictive ability by the correction of their output signal. In this paper standard ESN was supplemented by a new correcting neural network which has served as an autoregressive predictor. The main task of this special neural network was output signal correction and therefore also a decrease of the prediction error. The goal of this paper was to compare the results achieved by this new approach with those achieved by original one-step learning algorithm. This approach was tested in laser fluctuations and air temperature prediction. Its prediction error decreased substantially in comparison to the standard approach.

Keywords: Echo State neural networks, recurrent neural networks, prediction, autoregressive predictor

Mathematics Subject Classification 2000: 68T05

1 INTRODUCTION

From the point of information transfer during processing, neural networks (NN) can be divided into two types: feed-forward neural networks and recurrent neural networks [1]. Unlike the feed forward NN, recurrent NN contain at least one cyclical path, where the same input information repeatedly influences the activity of the

neurons in a cyclical path. The advantage of such networks is their close correspondence to biological NN, but there are many theoretical and practical difficulties connected with their adaptation and implementation. The common problem of all such networks is the lack of an effective supervised training algorithm. The problem was partially overcome with ESN [2]. On one hand their application bypasses a problem of efficient training, but on the other hand, by imposing an echo-state property we restrict the ESN recurrent dynamics to contractions, making it less general (unlike fully trained recurrent neural networks, ESN cannot learn, e.g. the context-free grammar). A very fast algorithm is used in these networks consisting of a calculation of one pseudo-inverse matrix, which is a standard numerical task. But the advantage of "one step" learning turns into a disadvantage when we try to improve the predictive abilities of the network. The pseudo-inverse matrix approach does not offer any straightforward solution in this case.

The trouble with this approach is that it offers nearly absolutely perfect learning of the training set for the given recurrent neural network, but the predictive ability of such a network is not very good. In this paper we shall concentrate on the possibility to supplement standard ESN by a new correcting NN which will serve as an autoregressive predictor. The task of this special NN is output signal correction. We can find similar approach for different type of neural networks in [7] and analysis of learning process for noisy time series in [8]. Connection between "liquid state" computing, related to echo states was mentioned previously in [4, 6]. The predictive abilities of optimized ESN were tested on laser-fluctuations and air temperature data. When we apply this new approach, we loose slightly the advantage of fast computation of the "one-step" learning typical for ESN, but we gain flexibility and better quality of prediction.

2 ECHO STATE NEURAL NETWORK

Echo State neural networks are atypical in architecture and training of recurrent NN. This new approach leads to a fast, simple and constructive supervised learning algorithm for the recurrent NN. The basic idea of ESN is the application of a huge "reservoir", as a source of dynamic behavior of a neural network, from which neural activities are combined into the required output.

The activity of hidden layer neurons in an RNN is further denoted as $\mathbf{x}(n) = (x_1(n), x_2(n), \ldots, x_N(n))$, where $x_i(n)$ is the output of the ith hidden neuron in time n, and N is the number of hidden neurons. Under certain conditions, each $x_i(n)$ is a function of the networks previous inputs $\mathbf{u}(n), \mathbf{u}(n-1), \ldots$, previously processed by the network. The input vector is denoted as $\mathbf{u}(n) = (u_1(n), u_2(n), \ldots, u_K(n))$, where $u_i(n)$ is the input of the ith input neuron at time n and K is the number of input neurons. Therefore, there exists such a function, E, so that:

$$\mathbf{x}(n) = E(\mathbf{u}(n), \mathbf{u}(n-1), \ldots). \tag{1}$$

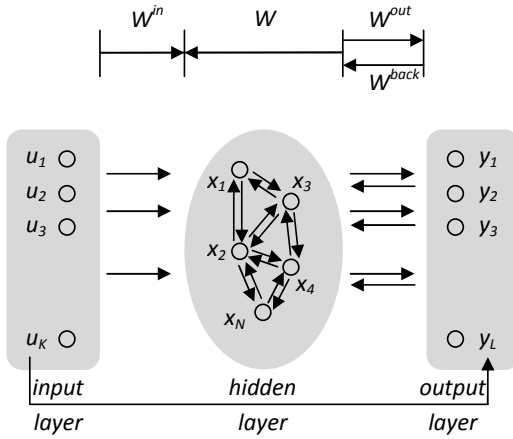

Fig. 1. The typical architecture of Echo State neural networks

Metaphorically speaking, the state of the neural network $\mathbf{x}(n)$ can be considered as an "echo", or in other words, a reflection of its previous inputs.

2.1 Description of the Neural Network

Neural network consists of K input, N hidden and L output neurons. The state of the neurons in the input layer at time n is characterized by the vector

$$\mathbf{u}(n) = (u_1(n), u_2(n), \ldots, u_K(n)),$$

in the output layer by the vector

$$\mathbf{y}(n) = (y_1(n), y_2(n), \ldots, y_L(n)),$$

and in the hidden layer by the vector

$$\mathbf{x}(n) = (x_1(n), x_2(n), \ldots, x_N(n)).$$

The values of all the synaptic weights will be stored in matrices. An input weight matrix will be created: $\mathbf{W}^{in} = (w_{ij}^{in})$ of size $N \times K$, a weight matrix between hidden neurons: $\mathbf{W} = (w_{ij})$ of size $N \times N$, a matrix of output weights: $\mathbf{W}^{out} = (w_{ij}^{out})$ size of $L \times (K + N + L)$, and a matrix of weights from the output back to the reservoir: $\mathbf{W}^{back} = (w_{ij}^{back})$ size of $N \times L$. It is notable that, in this type of network, both direct input-output weights, as well as the weights between output neurons are allowed.

The structure and topology of ESN can be adjusted according to their current task. It is not necessary, for example, to use sigmoid output neurons, back weights from the output layer to the reservoir may or may not exist, and even the input

neurons may not be used. In this application, sigmoid output neurons were not used (linear activation function was used), however back weights from the output layer to the reservoir were used. No loops were used for output neurons. We can find detailed description of the learning algorithm in [2, 3]. We will just introduce computation of activities of the internal and output neurons. The states of hidden neurons in "dynamical reservoir" are calculated from the formula

$$\mathbf{x}(n+1) = f(\mathbf{W}^{in}\mathbf{u}(n) + \mathbf{W}\mathbf{x}(n) + \mathbf{W}^{back}\mathbf{d}(n)), \tag{2}$$

where f is the activation function of hidden neurons. The states of output neurons are calculated from the formula

$$\mathbf{y}(n+1) = f^{out}(\mathbf{W}^{out}(\mathbf{u}(n+1), \mathbf{x}(n+1), \mathbf{y}(n)), \tag{3}$$

where f^{out} is the activation function of output neurons.

3 TESTING DATA

Most publications about prediction strive to achieve the best prediction, which is then compared with results of other prediction systems on selected data. This paper is different in this aim; its goal was to compare results by original "one-step" learning algorithm, with our new approach. We have used two data sets.

The first testing set was composed of a time sequence of 1000 samples of laser fluctuations data, and the quality of prediction was measured by an error of prediction in the next 100 steps.

The second testing set was composed of a time sequence of 1096 samples of average air temperature in Slovakia in years 1999–2002. The quality of prediction was measured by an error of prediction in the next 31 steps – that means January of the year 2002.

A mean absolute percentage error (MAPE)[1] was used to measure the quality of prediction on these testing sets, where test values P_i^{real} and predicted values P_i^{calc} are used, and N is the number of couples of values (the length of the predicted time series):

$$MAPE = \frac{\sum_{i=1}^{N} \left| \frac{P_i^{real} - P_i^{calc}}{P_i^{real}} \right|}{N} \times 100. \tag{4}$$

A mean squared error (MSE) and standard deviation (SD) were also used for the final evaluation of results:

$$MSE = \frac{\sum_{i=1}^{N} (P_i^{real} - P_i^{calc})^2}{N}. \tag{5}$$

[1] The MAPE error was used in order to allow a comparison with results of prediction on the predicted data achieved by other methods [10, 11], which also used MAPE error.

$$SD = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(P_i^{calc} - y')^2}, \tag{6}$$

where y' is the mean (average) of predicted values.

4 AUTOREGRESSIVE PREDICTOR

The main task connected with a design of a predictor based on NN is to find such a configuration of synaptic weights, which represents functional dependency between ideal output of the NN and input signal. So we want a prediction system with good approximation properties on testing sets ([5]).

The process of adaptation of the chosen topology for the training set may not be optimal and the performed prediction may not be complete. But we can obtain an error signal from such an inaccurate prediction and this signal can be used for model correction. We can improve the prediction ability of the whole system with such modification.

One of the possibilities how to solve this problem is to create a specific model (Figure 2). With this model, we are trying to extrapolate error in the next step based upon knowledge of the error in the past steps. We basically assume that the series of errors produced in time series prediction is a predictable time series by itself.

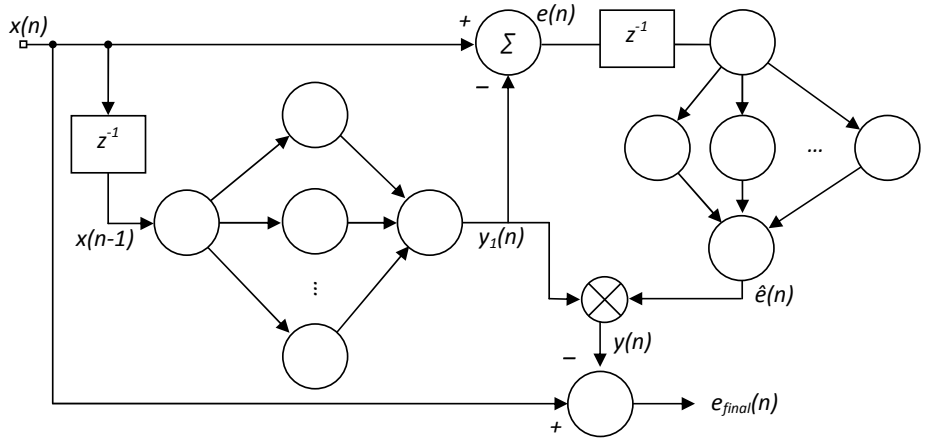

Fig. 2. Autoregressive error model

The symbol z^{-1} in Figure 2 means the time-delay element. The ability of error prediction is also based on the level of noise, which is undoubtedly also a part of testing sets. In a case when the error signal still carries some information, but the noise part is dominant, it is very difficult to predict such a kind of error.

As we can see from Figure 2, the next step is the processing of error $e(n-1)$, which is then used as the input of the next neural network. This correcting NN

serves as an error predictor, which is estimating the error $\widehat{e}(n)$. This estimated error is then added to output $y_1(n)$ in the next time step. The prediction error $e_{final}(n)$ of the final system should be lower than the error of uncorrected system:

$$E[e_{final}] < E[e], \tag{7}$$

where E is statistical average operator. Based on the facts mentioned above, we can divide this method into two phases:

- In the first phase, the maximum of the information is extracted and coded into the synaptic weights of the first NN, which is then used as a time series predictor for the next time steps.
- In the second phase, a secondary neural network is trained to predict the error of the first neural network, provided that the error is still predictable.

This procedure is usable in NN which are using iterative learning algorithms. As the ESN are using one-step learning algorithm, this procedure must be slightly modified. This varied procedure is described in details in the next section.

5 EXPERIMENTS

Experiments were divided into three parts. The task of the first part was to find parameters of ESN, which would be optimal for the quality of prediction on the laser fluctuations and air temperature testing sets. The autoregressive predictor was not used in this part. The achieved results should also serve for comparison with results achieved in the third part.

In the second part, ESN was used for the prediction of time series values of the past (of the training set). Parameters of the main NN and initial synaptic weights of dynamic reservoir were chosen in accordance with the results of experiments in the first part. Following the differences between original and predicted values, the so called "error time series" was created. This error time series was used in this part as a training set for the correcting NN, which then served as output signal corrector of the main NN. The task of this second part was also to find parameters of correcting neural network, which would be optimal for the quality of prediction of the training "error time series" set.

The third part of experiments was focused on the evaluation of prediction results, where the above mentioned correcting NN was used as an autoregressive predictor. Parameters of this correcting NN and initial synaptic weights of dynamic reservoir were chosen in accordance with the results of experiments in the second part.

5.1 The First Part of Experiments

As mentioned in Section 5, the first part involves two tasks: Finding best parameters of the main ESN and obtaining prediction results. These results will be used later for comparison with those achieved in the third part.

The prediction task is based on an implicit model of the dynamic of the predicted process, which is combined from the ESN dynamic reservoir. During the "training" phase, the ESN feeds ideal output to the dynamic reservoir through its output neuron. During the prediction, no input is needed; the output neuron feeds its predictions back into the reservoir. The past values, which would be fed into the input neuron in a typical one-step ahead prediction, are unnecessary in the ESN prediction phase. The ESN in the presented problem therefore does not need any input neuron.

So the NN consists only of dynamic reservoir and one output neuron, which has also the input function in the final consequence. The weight matrix between hidden neurons ($\mathbf{W}$) should be sparse, to encourage rich variety of dynamics in dynamical reservoir. For that reason, only 2 % of all connections in dynamical reservoir were created. The network's hidden units are standard sigmoid units, with a transfer function $f = \tanh$ (hyperbolic tangent) and the synaptic weights were initialized from the interval $[-1, 1]$ with uniform distribution (the same holds for the weight matrix $\mathbf{W}^{back}$).

A considerable number of experiments was carried out, the representative results of which are presented in Table 1 for laser data and in Table 2 for air temperature data.

Index	*Size of DR*	*Alpha*	*Average MAPE*	*Best MAPE*
1	200	0.7	38.27 %	34.23 %
2	250	0.7	36.29 %	31.34 %
3	250	0.8	34.24 %	29.42 %
4	300	0.7	35.94 %	32.86 %

Table 1. Results of representative experiments in the first part: laser time series

Index	*Size of DR*	*Alpha*	*Average MAPE*	*Best MAPE*
1	200	0.8	31.72 %	26.67 %
2	250	0.7	29.27 %	25.16 %
3	250	0.8	26.52 %	23.28 %
4	300	0.8	35.51 %	27.54 %

Table 2. Results of representative experiments in the first part: air temperature time series

In these tables, *DR* represents the dynamic reservoir; *Alpha* is the spectral radius of the weight matrix W, which is influencing the ability of the neural network to exhibit echo states. These *DR* and *Alpha* values were chosen in accordance with the proposal used by Jaeger (see [2]). Experiments were carried out in the following way. For each value of *DR* and the parameter *Alpha*, the values of synaptic weights in *DR* were randomly generated 1000 times. This number was estimated to be large enough for statistical evaluation of prediction error on a testing set, and for each

initialization of weights the error for the testing set was calculated. Further, an average error of all 1000 trials is presented in the columns *Average MAPE* (Tables 1 and 2). Also, the smallest achieved error was recorded in the *Best MAPE* in the same tables.

A clear correlation between Best and Average value columns is apparent from Tables 1 and 2. When a better *Average MAPE* was achieved, there is also a better *Best MAPE*. The best results for both prediction sets were achieved with a *DR* consisting of 250 neurons and for the parameter *Alpha* 0.8. We can see graphical representation of the best error for laser testing set in Figure 3 and for air temperature set in Figure 4.

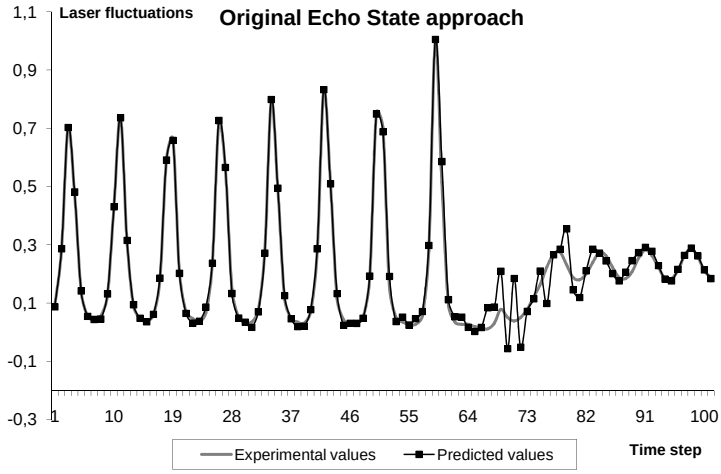

Fig. 3. Testing data: 100 records of laser fluctuations and 100 values predicted by the original Echo State neural network. MAPE 29.42 %, graph corresponds to experiment No. 3 from Table 1

5.2 The Second Part of the Experiments

In this part of experiments we have optimal parameters of the main ESN, in regard to used laser fluctuations and air temperature sets. These NN are already trained and ready to use in this second part.

The training of ESN is based on linear regression and one-step learning algorithm. Therefore, we have no possibility to catch the error signal during this learning process. So the main NN was used for self prediction of the whole laser fluctuations and air temperature sets and the time series obtained in that way reflects the quality of adaptation on this training set.

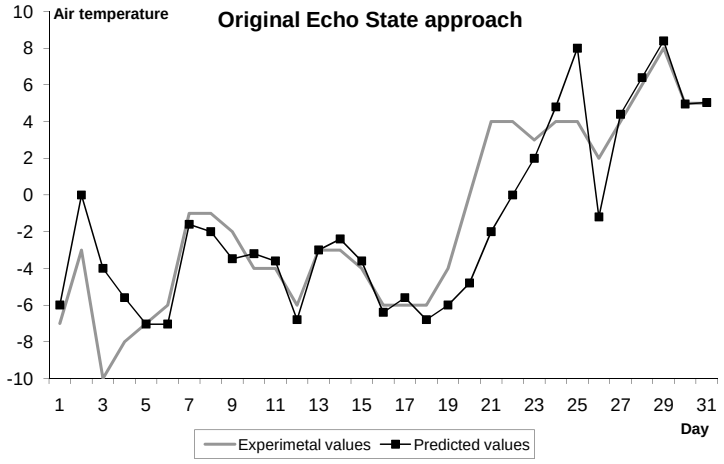

Fig. 4. Testing data: 31 records of air temperature and 31 values predicted by the original Echo State neural network. MAPE 23.28 %, graph corresponds to experiment No. 3 from Table 2

Then we have obtained the new "error time series" as the difference between real and predicted values. This error time series has served in this part as the training set for the correcting NN.

Subsequently, we have tried to find optimal parameters of these correcting NN, in regard to quality of prediction on the newly created error time series.

A considerable number of experiments was carried out, the representative results are in the following Table 3 for laser and in the Table 4 for air temperature time series.

Index	*Size of DR*	*Alpha*	*Average MAPE*	*Best MAPE*
1	100	0.7	24.29 %	21.56 %
2	150	0.7	20.34 %	18.46 %
3	200	0.8	21.90 %	19.31 %

Table 3. Results of representative experiments in the second part: laser error time series

Experiments in this part and description of attributes from Tables 3 and 4 are identical to experiments and description of attributes from the first part. For that reason these descriptions will not be repeated again.

Index	*Size of DR*	*Alpha*	*Average MAPE*	*Best MAPE*
1	100	0.7	19.24 %	18.55 %
2	150	0.7	23.22 %	20.22 %
3	200	0.7	18.74 %	16.45 %

Table 4. Results of representative experiments in the second part: air temperature error time series

5.3 The Third Part of the Experiments

We are now ready to move on to the last part of experiments. In this part we have already prepared and trained the main NN and also the correcting NN. The experiments proceeded in the following way. First we have predicted the values of the testing set (the main neural network). Subsequently we have predicted the error values (correcting neural network), which represent in each step the estimated error of the main NN. The final time series was formed by superposition of these two predicted time series.

We can see the results of experiments in Table 5, with graphical representation for laser testing set in Figure 5 and for air temperature testing set in Figure 6.

Testing data	*Original Approach* MAPE / MSE / SD	*Use of autoregressive predictor* MAPE / MSE / SD
Laser fluctuations	29.42 % / $1.2e{-}3$ / 0.221	22.36 % / $0.8e{-}3$ / 0.220
Air temperature	23.28 % / $1.7e{-}2$ / 0.264	20.13 % / $1.2e{-}2$ / 0.260

Table 5. Results of experiments in the third part

It is clear from the results shown in Table 5 and figures that this approach can increase the quality of prediction of ESN. The laser data are a good sample of accurate laboratory measurements (we can consider them noiseless). For that reason, the improvement of achieved results with this new approach is much better in the case of easer time series than for air temperature time series.

6 CONCLUSIONS

Echo State neural networks are relatively new in the domain of NN. Their advantage is a closer connection with biological models inherent to recurrent NN and in their usage of the reservoir of dynamic behavior without adjusting the weights within the hidden layer. ESN have a substantial advantage over other types of recurrent networks in their "one-step" learning ability, even though this approach may not be very biologically plausible. As a disadvantage, they can be considered to have a relatively low ability to generalize. However, the existence of this disadvantage is not theoretically founded, it is based only on extensive experimental observations. In

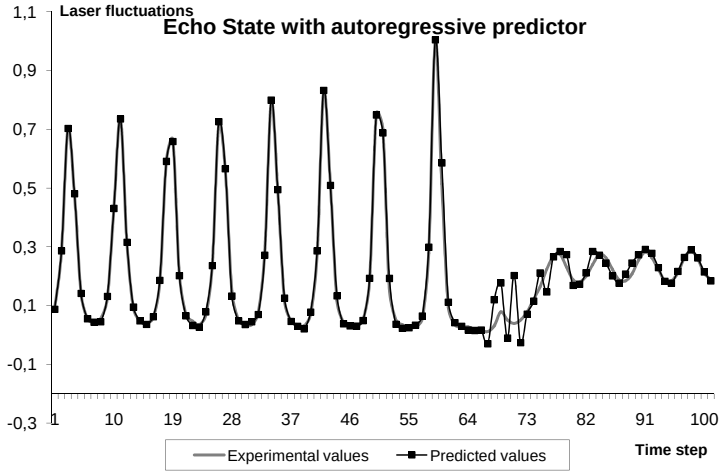

Fig. 5. Testing data: 100 records of laser fluctuations and 100 values predicted by using autoregressive predictor in Echo State neural network. MAPE 22.36 %

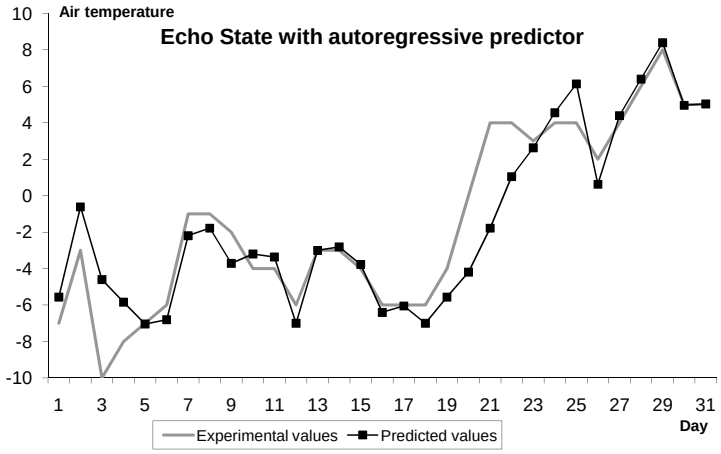

Fig. 6. Testing data: 31 records of air temperature and 31 values predicted by using autoregressive predictor in Echo State neural network. MAPE 20.13 %

some cases, ESN was shown to have better generalization capacity than supervised Elman net (see e.g. [9]).

We have tried to improve predictive performance of ESN in this work, where the standard ESN was supplemented with a new correcting NN, which has served as an autoregressive predictor. The main task of this special NN was the output signal correction.

We have chosen laser fluctuations and air temperature time series as a testing data. Our aim was to find out if this combination of neural networks is able to increase predictive quality of ESN. It is clear from the results shown in the paper that this aim has been accomplished. Supplementation of standard Echo State neural network with error correction network can increase the quality of the prediction.

Acknowledgement

This work was supported by Slovak Scientific Grant Agency, Grant VEGA 1/4053/07 and by Slovak Research Development Agency under the contract No. APVT-20-002504.

REFERENCES

[1] Haykin, S.: Neural Networks – A Comprehensive Foundation. Macmillian Publishing, 1994.

[2] Jaeger, H.: The Echo State Approach to Analysing and Training Recurrent Neural Networks. German National Research Center for Information Technology, GMD report 148, 2001.

[3] Jaeger, H.: Short Term Memory in Echo State Networks. German National Research Center for Information Technology, GMD report 152, 2002.

[4] Natschlager, T.— Maass, W.—Markram, H.: The "Liquid Computer": A Novel Strategy for Real-Time Computing on Time Series. Special Issue on Foundations of Information Processing of TELEMATIK, Vol. 8, 2002, No. 1, pp. 39–43.

[5] Kropuch, M.: Using of Neural Networks in Prediction Systems. Diploma Thesis. Technical university in Košice, 2001 (in Slovak).

[6] Goldenholz, D.: Liquid Computing: A Real Effect. Technical report, Boston University Department of Biomedical Engineering, 2002.

[7] Wan, E. A.: Finite Impulse Response Neural Networks for Autoregressive Time Series Prediction. In A. Weigend and N. Gershenfeld, editors, Predicting the Future and Understanding the Past, Volume XVII. Addison-Wesley, Reading, MA, 1993.

[8] Nakayama, K.—Khalaf A. A. M.: A Hybrid Nonlinear Predictor: Analysis of Learning Process and Predictability for Noisy Time Series. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol. E82-A, 1999, No. 8, pp. 1420–1427.

[9] Frank, S. L.: Learn More By Training Less: Systematicity in Sentence Processing by Recurrent Networks. Connection Science, Vol. 18, 2006, pp. 287–302.

[10] Babinec, Š.—Pospíchal, J.: Optimization in Echo State Neural Networks by Metropolis Algorithm. In R. Matousek, P. Osmera (eds.): Proceedings of the 10th International Conference on Soft Computing, Mendel 2004. VUT Brno Publishing, pp. 155–160, 2004.

[11] Babinec, Š.—Pospíchal, J.: Merging Echo State and Feedforward Neural Networks for Time Series Forecasting. In: S. Kollias et al. (eds): Proceedings of the 16th International Conference ICANN 2006. Springer-Verlag Berlin Heidelberg, pp. 367–375, 2006.

Štefan Babinec received his M. Sc. degree in artificial intelligence in June 2003, from Technical University of Košice, Faculty of Electrical Engineering and Informatics, Slovakia. From October 2003 to September 2006, he was a postgraduate student at Slovak University of Technology, Faculty of Chemical and Food Technology, Department of Mathematics, Bratislava, Slovakia. Now he is working at the same department as assistant professor. He has co-authored 15 works in journals and conferences related to artificial neural networks and evolutionary algorithms.