Neural Networks Application to Neretva Basin Hydro-meteorological Data

Miloš Daković, *Member, IEEE*, Tijana Ružić, Tanja Rogač, Miloš Brajović, *Student Member, IEEE*, Budimir Lutovac, *Member, IEEE*

***Abstract*—Neural networks application to the analysis and prediction of the hydro-meteorological data is presented. The neural networks are trained and tested with water-level and water-flow data measured at three stations in the Neretva river basin. Estimation of the water-level based on water-flow and vice versa is presented. These data are highly (byt nonlineary) correlated. The proposed approach can be used to reconstruct missed measurements caused, for example, by measurement equipment failure. In this way an accurate and complete set of measurements can be obtained. Estimation of downstream measurements based on upstream data is also analysed. It is shown that highly accurate estimations can be obtained when there is no tributaries between measurement stations.**

***Keywords*—Neural networks, Hydro-meteorological data, Data prediction, Signal processing**

I. Introduction

The acquisition and analysis of the hydro-meteorological data is a very important topic today [1]–[6]. These data are crucial for hydro-electric power plants operations, and for avoiding emergency situations.

In this paper we will focus on hydro-meteorological data analysis for Neretva basin measured at three measurement stations: Mostar, Baćevići and Žitomislići. Although specific basin and measurement stations are considered, the proposed methods are not limited to this scenario and could be applied to any other river basin.

The water-flow and water-level are the main hydro-meteorological parameters measured on several stations in the Neretva basin. The measurements are provided by the Agency for watershed of Adriatic sea – Mostar. Time interval between successive measurements varies from 15 min to 1 hour. Some of the measurements are unavailable due to recording problems, measurement devices failure, and other unclassified problems. Very often we have water-level data but do not have water-flow data and vice versa. Here, we propose a method for data restoration based on neural networks. It is obvious that water-level and water-flow data are mutually dependent but the dependence is hard to be expressed in an analytic way. Here we identify a possible application of neural networks in order to find the dependence between water-flow and water-level data at the considered measurement station.

In Section II we will briefly describe the data used in the rest of the paper. The neural network model is introduced in Section III. The obtained results are presented in Section IV.

This research is supported by the project "New ICT Compressive Sensing Based Trends Applied to: Multimedia, Biomedicine and Communications (CS-ICT)" (Montenegro Ministry of Science, Grant No. 01-1002).

Miloš Daković, Miloš Brajović and Budimir Lutovac are with University of Montenegro, Faculty of Electrical Engineering, Cetinjski put bb, 81000 Podgorica, Montenegro (e-mails: {milos,milosb,budo}@ac.me)

Tijana Ružić is with Commercial District Court in Trebinje, Kralja Petra I Oslobodioca 48 89101 Trebinje, Republika Srpska – Bosna and Herzegovina (e-mail: ruzictijana@yahoo.com)

Tanja Rogač is with Public institution "Vode Srpske", Department of Water Management river basin Trebišnjica, Stepe Stepanovića bb, 89101 Trebinje, Republika Srpska – Bosna and Herzegovina (e-mail: trogac@voders.org)

Fig. 1. Locations of the measurement stations (Source: Google Earth)

TABLE I
MEASUREMENT STATIONS, MEASURED VARIABLES, NUMBER OF MEASUREMENTS, MINIMAL AND MAXIMAL VALUE, AND SAMPLING INTERVAL

Station	Variable	N_m	Min	Max	Sampl. int.
Mostar	level (cm)	33656	230	794	15(30) min
	flow (m^3/s)	33626	45	684	15(30) min
Baćevići	level (cm)	17568	165	469	30 min
	flow (m^3/s)	17568	58	721	30 min
Žitomislići	level (cm)	4194	76	211	30 min
	flow (m^3/s)	1252	103	334	30 min

II. Data Description

The analyzed data are collected on three hydro-meteorological stations located at Mostar, Baćevići and Žitomislići, Fig. 1. Water flow and water level are measured and brief data description is presented in Table I. Within the analyzed data not all measurements are available. Some measurements are skipped due to the station or measurement failure. Sampling interval is 30 min, while at Mostar station significant part of the data are measured every 15 minutes. In this paper we work with raw (unprocessed) data.

III. Neural Network

Herein, the neural networks are used for several purposes. The used networks are feed-forward with one hidden layer [7]. We assume that neurons in the hidden layer have bias, and that the output neuron is unbiased. The bias is modeled as an additional input to the neural network with constant input value equal to one. Network function is weighted sum of all inputs

$$u = \sum_{k=0}^{M} w_k x_k \tag{1}$$

while activation function is unipolar sigmoid defined by

$$f_a(u) = \frac{1}{1+e^{-u}}. \tag{2}$$

It is assumed that neurons in the input layer transfer the input value to the output lines without any transformation (weighting or activation function). The network topology of neural networks is presented in Fig. 2 and Fig. 3.

The error backpropagation algorithm is used for network training. For each epoch and for each input-output pair, the network output and error are calculated, and neurons weights are updated according to

$$\begin{aligned}
\mathbf{h} &= f_a(\mathbf{W}\mathbf{x}), \\
Y &= f_a(\mathbf{u}\mathbf{h}) \\
\delta_o &= Y(1-Y)(y-Y) \\
\delta_h &= \mathbf{h} \odot (\mathbf{1}-\mathbf{h}) \odot \mathbf{u}^T \delta_o \\
\mathbf{u}_{new} &= \mathbf{u}_{old} + \mu \delta_o \mathbf{h}^T \\
\mathbf{W}_{new} &= \mathbf{W}_{old} + \mu \delta_h \mathbf{x}^T
\end{aligned}$$

where $\mathbf{h}$ is output vector for neurons in the hidden layer, $\mathbf{x} = [1, x(n)]^T$ is input vector, $\mathbf{W}$ is weight matrix of the neurons in the hidden layer, Y is network output, $\mathbf{u}$ is weighting row-vector of the output neuron and y is desired network output. Element-by-element product is denoted with $\odot$.

The error, mean squared error (MSE), and root mean squared error (RMSE) are used for evaluation of the trained network. The error is defined as difference between desired and actual output for i-th data pair, $e_i = y_i - Y_i$ for $i = 1, 2, \ldots, N_m$ where N_m is total number of available input–output pairs. The MSE (in dB) and RMSE are calculated as

$$MSE = 10 \log_{10} \left(\frac{1}{N_m} \sum_{i=1}^{N_m} e_i^2 \right),$$

$$RMSE = \sqrt{\frac{1}{N_m} \sum_{i=1}^{N_m} e_i^2}.$$

It is very important to properly select the number of neurons in the hidden layer. A large number of neurons could lead to a slow training procedure, while the output error can be too high if the number of neurons is too low.

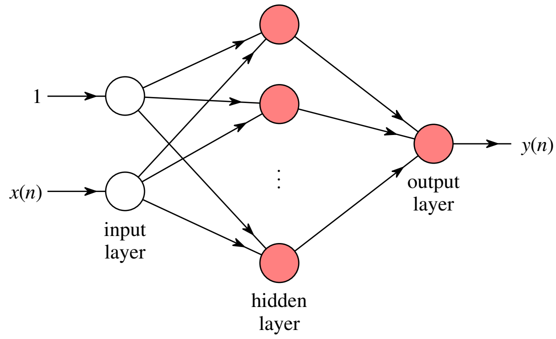

Fig. 2. Simple network topology for algebraic input-output functional dependence.

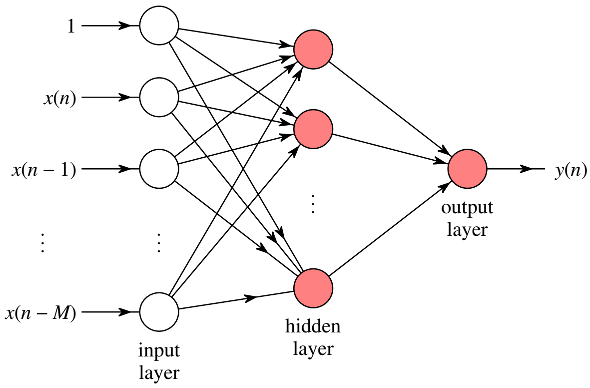

Fig. 3. Autoregressive network topology.

IV. Results

The first type of neural network is designed to provide water-flow data if water-level is available and vice versa. Topology of the used network is presented in Fig. 2. Neurons in the hidden layer are biased and bias is implemented as an additional input with constant value. The output neuron is of the same type as neurons in the hidden layer without bias input. The networks are trained for each measurements station by using part of the data (1000 input-output pairs) for training and all available data for verification. The number of neurons in the hidden layer is varied from 1 to 6. The results are presented in Fig. 4.

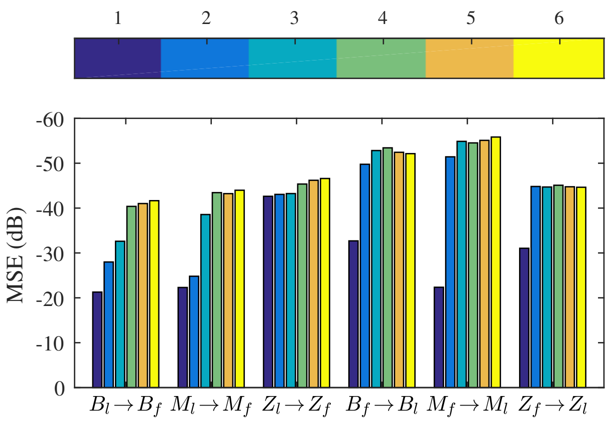

Fig. 4. MSE (in dB) for various number of neurons in the hidden layer and for various input-output combinations. The number of neurons varies form 1 to 6 and it is presented by color.

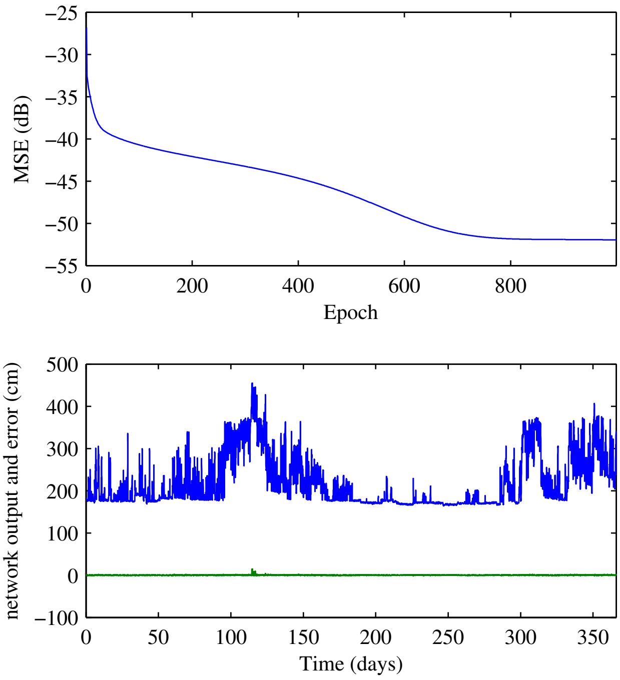

Fig. 5. Learning curve (upper subplot), network output and error (lower subplot). The neural network is trained with water flow as input and water level as output for data measured at station Baćevići. The training dataset was consisted of 1000 randomly positioned measurements out of total 17568 available measurements.

Input and output variables are given below the MSE bars where B stand for Baćevići, M for Mostar, and Z for Žitomislići station. Index l indicates water-level data while index f denotes water-flow data. From Fig. 4 we can conclude that four neurons in the hidden layer are sufficient in all considered cases, while in some cases even two neurons are enough.

The neural networks are trained with 1000 epochs and learning rate $\mu = 20$. The input and output data are scaled with factor 1000 in order to fit data into range [0,1] according to minimum and maximum values presented in Table I. If any of the data are unavailable at the considered time instant, the corresponding input-output pair is removed from the training and verification sets.

The training results for $B_f \rightarrow B_l$ network with 3 neurons in the hidden layer and 1000 epochs are presented in Fig. 5 as well. The learning curve (MSE versus epoch) is presented in the upper subplot. The network output and error is presented in the lower subplot. It is obvious that trained network achieves low error, and that MSE is almost constant for epochs from 800 to 1000. The similar results are obtained in all considered cases.

The presented networks can be used to interpolate missing data pairs, i.e. when either water-flow or water-level is unavailable at the considered time instant it can be replaced with output of the corresponding network. Note that in the considered setup for measurement station Žitomislići there are only 1252 water-flow measurements and 4194 water-level measurements, so by using neural network trained to give water-flow based on water-level input we can obtain reliable data for any further analysis.

Next we will train neural networks for future value

TABLE II
RMSE ERROR FOR WATER LEVEL DOWNSTREAM PREDICTION FOR VARIOUS NUMBER OF USED PAST VALUES M

M	$M_l \rightarrow B_l$	$M_l \rightarrow Z_l$	$B_l \rightarrow Z_l$
0	22.4	31.4	22.9
1	13.7	26.6	16.5
2	9.1	23.3	13.5
3	8.7	19.5	11.6
4	8.6	15.7	11.0
5	8.6	12.9	10.7

TABLE III
RMSE ERROR FOR WATER FLOW DOWNSTREAM PREDICTION FOR VARIOUS NUMBER OF USED PAST VALUES M

M	$M_f \rightarrow B_f$	$M_f \rightarrow Z_f$	$B_f \rightarrow Z_f$
0	13.3	20.0	15.9
1	8.5	18.4	13.9
2	5.8	17.3	13.6
3	5.4	16.2	12.9
4	5.7	15.7	12.8
5	5.9	15.8	13.1

prediction of the water-level and water-flow data. In order to obtain more a reliable prediction, current and M past values of the input data are used as the neural network input, Fig 3.

When the desired network output is future value of the input measurements for a single measurement variable, the obtained errors after network training were only few dB below the simplest predictor $x(n+1) = x(n)$. Here we vary the number of neurons form 1 to 30, the number of past signal values form 1 to 48 (whole day with 30min sampling interval) and the learning rate from 2 to 30. This results are not promising and lead us to the conclusion that neural networks future value prediction, based on a single input (with current and past values), can be replaced with a simple and efficient predictor without high loose in the prediction error.

Next we will utilize neural network in order to estimate values of water-level and water-flow for downstream stations based on values measured on upstream stations. The neural network topology is presented in Fig. 3. The number of neurons in the hidden layer is 5. The networks are trained on a subset of 1000 data and tested on all available data.

Neural networks are tested with various number of inputs (current value and M past values). The results for water level are presented in Table II and the results for water flow prediction are given in Table III. In both cases root mean squared error is calculated for $M = 0, 1, \ldots, 5$.

In this case the results are much more accurate. The prediction results for water level data by using current and $M = 5$ past values of water level measured at the upstream station are presented in Fig. 6. The output value and estimation error are presented. Top subplot is for water level at Baćevići station, predicted from the Mostar data. Middle subplot is water level at Žitomislići station estimated with Mostar input data, and bottom subplot is water level at Žitomislići station predicted from Baćevići data. The error is smallest in the first considered case. Increased error for Žitomislići station can be explained with several tributaries between Baćevići and Žitomislići.

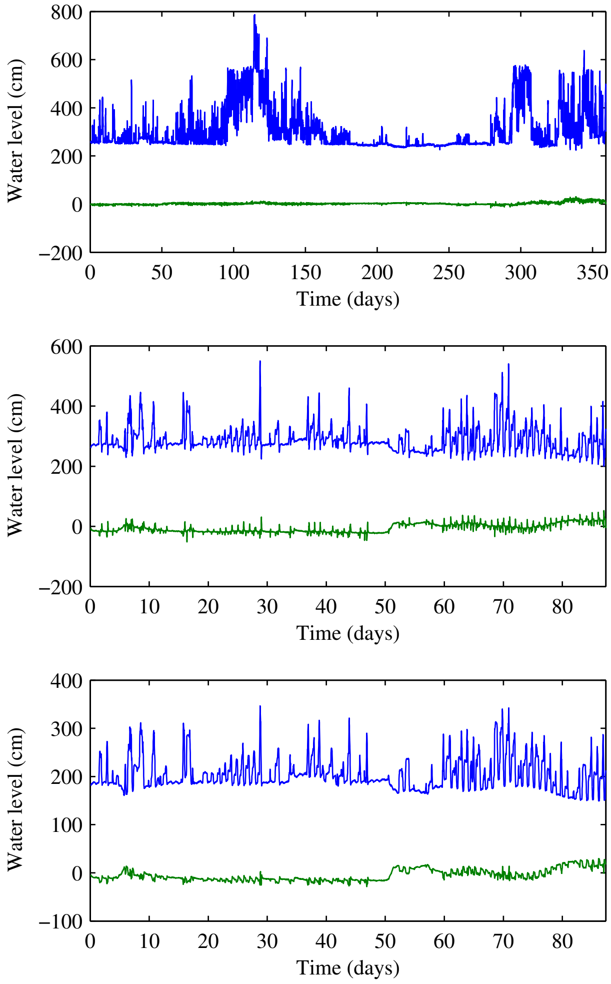

Fig. 6. Water level data obtained by using upstream station data as input to the neural network. Top subplot: Mostar→Baćevići, middle subplot Mostar→Žitomislići, and bottom subplot Baćevići→Žitomislići. Network output (blue) and error (green) are presented.

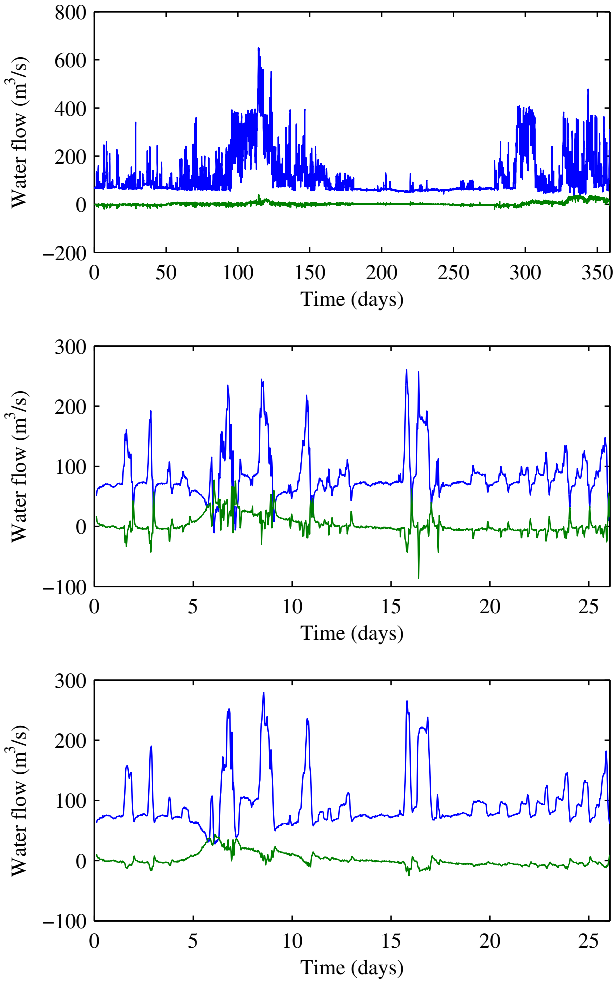

Fig. 7. Water flow data obtained by using upstream station data as input to the neural network. Top subplot: Mostar→Baćevići, middle subplot Mostar→Žitomislići, and bottom subplot Baćevići→Žitomislići. Network output (blue) and error (green) are presented.

Downstream water flow prediction is presented in Fig. 7. As in the previous case prediction at Baćevići station based on Mostar data is highly accurate while prediction at Žitomislići have moderate accuracy due to tributaries.

V. Conclusion

We considered three measurement stations where water-level and water-flow data are available, and we performed network training with varying number of neurons in the hidden layer for each station. It is shown that with appropriately trained neural networks the missing measuremens can be restored with high accuracy. We conclude that four neurons in the hidden layer is enough for all considered stations, although for some of them even smaller number of neurons could be sufficient.

Next, we analyzed possibility of downstream measurements prediction based on the upstream measurements. High prediction accuracy is obtained in the case when there is no tributaries between measurement stations.

VI. Acknowledgments

The authors are thankful to the Agency for watershed of Adriatic sea Mostar — "Agencija za vodno područje Jadranskog mora Mostar" for providing raw data measurements.

References

[1] M. Valipour, M. E. Banihabib, and S. M. R. Behbahani. "Comparison of the ARMA, ARIMA, and the autoregressive artificial neural network models in forecasting the monthly inflow of Dez dam reservoir," *Journal of hydrology* 476 (2013): 433-441.

[2] C. C. Young, W. C. Liu, and W. L. Hsieh, "Predicting the water level fluctuation in an Alpine lake using physically based, artificial neural network, and time series forecasting models," *Mathematical Problems in Engineering*, 2015

[3] M. Y. A. Khan, F. Hasan, S. Panwar, and G. J. Chakrapani. "Neural network model for discharge and water level prediction for Ramganga river catchment of Ganga Basin, India," *Hydrological Sciences Journal*, vol. 61, no. 11, 2015

[4] D. Pokrajac, N. Reljin, M. Reiter, S. Stotts, "Signal processing of St. Jones River, Delaware water level data," *ETRAN 2007*, Herceg Novi, 2007

[5] T. Rogač, T. Ružić, M. Daković, "Primjena vremensko- frekvencijskih transformacija na hidrometeoroloske podatke," *Infoteh*, Jahorina 2016.

[6] T. Ružić, T. Rogač, M. Brajović, and M. Daković, "Digitalna obrada hidrometeorolokih podataka sa sliva rijeka Neretve i Trebinjice," *ETRAN 2016*, Zlatibor, Jun 2016.

[7] LJ. Stanković, *Digital Signal Processing with Selected Topics*, CreateSpace Independent Publishing Platform, An Amazon.com Company, November , 2015