Prediction of flood abnormalities for improved public safety using a modified adaptive neuro-fuzzy inference system

M. Aqil*,, I. Kita**, A. Yano** and S. Nishiyama*****

*The United Graduate School of Agricultural Sciences, Tottori University, Tottori 680-8573, Japan
(E-mail: *akilshimane@gmail.com*)

**Faculty of Life and Environmental Science, Shimane University, Shimane, Matsue 690-8504, Japan

***Faculty of Agriculture, Yamaguchi University, Yamaguchi 753-8515, Japan

Water Science & Technology **Vol 54 No 11–12 pp 11–19**

Abstract It is widely accepted that an efficient flood alarm system may significantly improve public safety and mitigate economical damages caused by inundations. In this paper, a modified adaptive neuro-fuzzy system is proposed to modify the traditional neuro-fuzzy model. This new method employs a rule-correction based algorithm to replace the error back propagation algorithm that is employed by the traditional neuro-fuzzy method in backward pass calculation. The final value obtained during the backward pass calculation using the rule-correction algorithm is then considered as a mapping function of the learning mechanism of the modified neuro-fuzzy system. Effectiveness of the proposed identification technique is demonstrated through a simulation study on the flood series of the Citarum River in Indonesia. The first four-year data (1987 to 1990) was used for model training/calibration, while the other remaining data (1991 to 2002) was used for testing the model. The number of antecedent flows that should be included in the input variables was determined by two statistical methods, i.e. autocorrelation and partial autocorrelation between the variables. Performance accuracy of the model was evaluated in terms of two statistical indices, i.e. mean average percentage error and root mean square error. The algorithm was developed in a decision support system environment in order to enable users to process the data. The decision support system is found to be useful due to its interactive nature, flexibility in approach, and evolving graphical features, and can be adopted for any similar situation to predict the streamflow. The main data processing includes gauging station selection, input generation, lead-time selection/generation, and length of prediction. This program enables users to process the flood data, to train/test the model using various input options, and to visualize results. The program code consists of a set of files, which can be modified as well to match other purposes. This program may also serve as a tool for real-time flood monitoring and process control. The results indicate that the modified neuro-fuzzy model applied to the flood prediction seems to have reached encouraging results for the river basin under examination. The comparison of the modified neuro-fuzzy predictions with the observed data was satisfactory, where the error resulted from the testing period was varied between 2.632% and 5.560%. Thus, this program may also serve as a tool for real-time flood monitoring and process control.

Keywords Flood; graphical user interface; modified neuro-fuzzy; prediction

Introduction

Flood forecasting is undoubtedly a challenging field of operational hydrology, and a huge literature has been developed over recent years. Modelling of flood dynamics is performed not only to provide a warning system as a technical way to reduce flood risks but also to assist in managing reservoir operations, particularly during drought periods. In the past, prediction of river flood was performed mainly using conceptual and deterministic models (Bazartseren *et al.*, 2003). Conceptual and deterministic models are designed to simulate the physical mechanisms that determine the hydrological cycle, and involve water transference physical laws, and parameters associated with the characteristics of the catchment area (Sorooshian and Gupta, 1995). Such models may require sophisticated

doi: 10.2166/wst.2006.839

mathematical tools, a significant amount of calibration data, and some degree of expertise and experience with the model (Duan *et al.*, 1992). By considering the complexity of the phenomena involved, there is a strong need to explore alternative solutions through modelling direct relationship between the input and output data without having a complete physical understanding of the system. While data-driven models do not provide any physics of the hydrologic processes, they are, in particular, very useful for modelling rainfall-runoff relationships where the main concern is to predict accurate flows at specific watershed locations.

The adaptive neuro-fuzzy inference system has been adopted in water resource forecasting studies as a powerful alternative modelling tool in recent decades. This method offers more advantages that include the ability to handle large amounts of noisy data from dynamic and nonlinear systems, improvement of model performance, faster model development with less calculation times, and improved opportunities to provide estimates of prediction confidence through comprehensive bootstrapping operations. Successful applications of the neuro-fuzzy model in water resources forecasting have been widely reported. Bazartseren *et al.* (2003) used neuro-fuzzy and neural network models for short-term flow prediction. Valenca and Ludermir (2000) developed a fuzzy neural network models for inflow forecasting in electric power plants. Nayak *et al.* (2004) suggested that the performance of the adaptive neuro-fuzzy inference system is improved significantly if the input data are transformed into the normal domain prior to model building. Vernieuwe *et al.* (2004) demonstrated the use of Takagi–Sugeno models for predicting discharge from rainfall time series by comparing grid partitioning, subtractive clustering, and Gustafson–Kessel clustering identification methods for constructing models. Dixon (2004) examined the sensitivity of neuro-fuzzy models used to predict groundwater vulnerability in a spatial context by integrating GIS and neuro-fuzzy techniques. Chang and Chen (2001) developed a counter-propagation fuzzy-neural network (CFNN), which is the integration of a neural network and fuzzy arithmetic for real-time streamflow forecasting. Finally, Ponnambalam *et al.* (2003) used a combination of fuzzy inference systems, artificial neural networks, and genetic algorithms to minimize variance, which are of benefit in reservoir operations.

In this paper, a Mo-ANFIS model is proposed to modify the traditional ANFIS model. The aim is to modify the error correction rule of the error back propagation (EBP method). The model is developed as a decision support system and applied to a case study of streamflow forecasting of the Citarum River basin in Indonesia.

The study catchment and model development

The proposed modified ANFIS model is applied to forecast the daily streamflow from the Citarum River basin, West Java Province in Indonesia (Figure 1). The total drainage area of the Citarum River basin is 11 000 km^2, with the total length of the channel being about 270 km. The average annual rainfall in the river basin ranges between 2 000–5 000 mm/yr and the temperature ranges from 18° up to 24 °C. Three large, multipurpose reservoirs from upstream to downstream (Saguling, Cirata, and Jatiluhur) in the basin regulate the water flow and are the main source of water supply to Jakarta City, through the West Tarum Canal. The Citarum River can carry about 12.95 billion m^3 annually, consisting of 6 billion m^3 from the Citarum River and 6.95 billion m^3 from its tributaries. The Jatiluhur reservoir is owned and operated by Perum Jasa Tirta II (PJT II), a public corporation formed in 1967. The working area of PJT II covers 74 rivers and their tributaries, which become one hydrology unit in northern West Java.

The streamflow data collected by the PJT II authority was used for model investigation. The data contains information for a period of 16 years (1987 to 2002). The entire

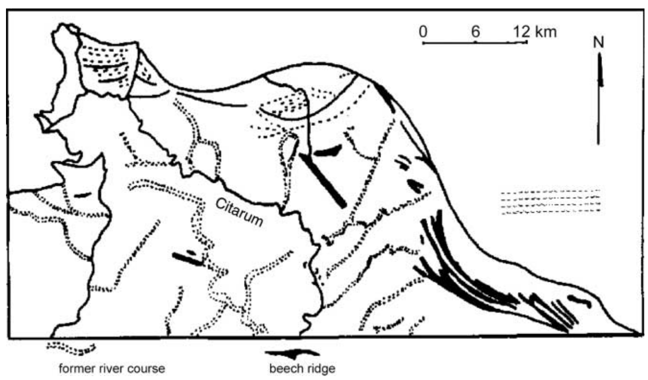

Figure 1 Map of the river basin under consideration

database consisted of 5840 daily values of streamflow data pairs. The input vector is arranged as follows:

$$[x(t-3), x(t-2), x(t-1), x(t); x(t+1)] \quad (1)$$

where $x(t)$ indicates the streamflow event at year t.

In this study, we used the data for the first four years (1987 to 1990) for model training/calibration, while the other remaining data (1991 to 2002) was used for testing the model.

Architecture of modified ANFIS

Modified ANFIS (Mo-ANFIS) is a modification adopted from the traditional ANFIS. It is a five-layer feed forward network. The architecture of the Mo-ANFIS learning algorithm that was implemented in the streamflow forecasting system is described in Figure 2.

The following procedure describes the learning process of the Mo-ANFIS.

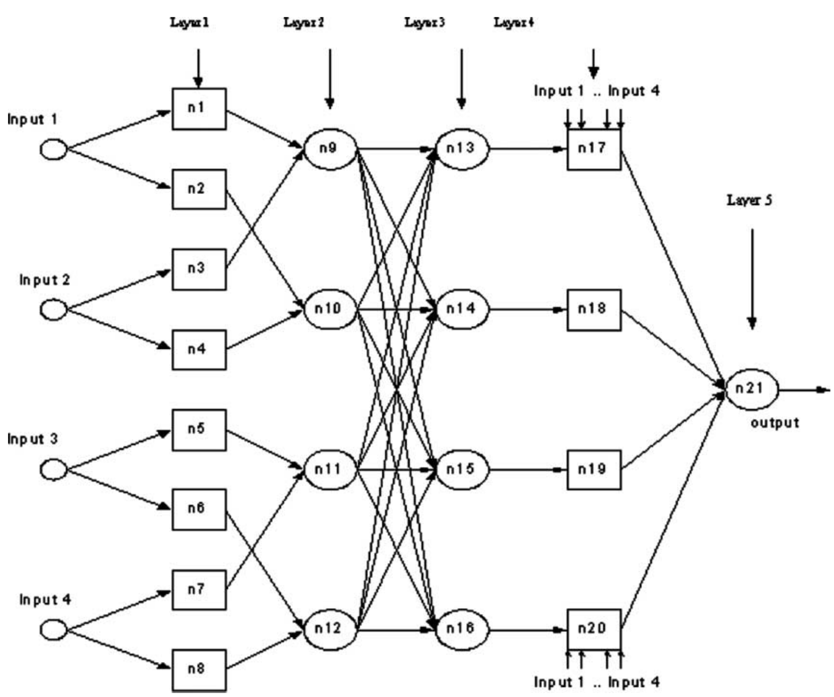

Figure 2 Architecture of the Mo-ANFIS learning algorithm

Forward pass

Layer 1: The membership function used in the Mo-ANFIS is a bell membership function.

$$\mu A(x) = \frac{1}{1 + \left|\frac{x - c_i}{a_i}\right|^{2b}} \tag{2}$$

The parameters of the membership function $\{a_i, b_i, c_i\}$ are predetermined. Every output from this node is labelled by *a,* so the input is given by *n1a* to *n8a*. The symbol *a* is used to differentiate from the new symbol *b* after the correction.

Layer 2: Fuzzy logic AND is used as the node function in this layer. The outputs of this layer are:

$$\begin{aligned} n_{9a} &= \min(n_{1a}, n_{3a}) \\ n_{10a} &= \min(n_{2a}, n_{4a}) \\ n_{11a} &= \min(n_{5a}, n_{7a}) \\ n_{12a} &= \min(n_{6a}, n_{8a}) \end{aligned} \tag{3}$$

Layer 3: The input signals of this layer are normalized. Hence

$$\begin{aligned} n_{13a} &= n_{9a}/n_{tot-a} \\ n_{14a} &= n_{10a}/n_{tot-a} \\ n_{15a} &= n_{11a}/n_{tot-a} \\ n_{16a} &= n_{12a}/n_{tot-a} \end{aligned} \tag{4}$$

where $n_{tot-a} = n_{9a} + n_{10a} + n_{11a} + n_{12a}$

Layer 4: From the incoming signals, we obtain matrix A as follows

$$\begin{aligned} A_1 &= [(n_{13a}.in_1)(n_{13a}.in_2)] \\ A_2 &= [(n_{14a}.in_1)(n_{14a}.in_2)] \\ A_3 &= [(n_{15a}.in_3)(n_{15a}.in_4)] \\ A_4 &= [(n_{16a}.in_3)(n_{16a}.in_4)] \\ A &= [A_1\, A_2\, A_3\, A_4] \end{aligned} \tag{5}$$

By using the LSE method, we obtain the consequent parameters of θ as follows

$$\theta = [A^T A]^{-1} A^T N \tag{6}$$

where N is the desired output of the model. We obtain the parameter

$$\begin{aligned} &\theta = [\theta(1)\ \theta(2)\ \theta(3)\ \theta(4)\ \theta(5)\ \theta(6)\ \theta(7)\ \theta(8)]^T \text{ and} \\ &f_1 = \theta(1).in_1 + \theta(2).in_2 \\ &f_2 = \theta(3).in_1 + \theta(4).in_2 \\ &f_3 = \theta(5).in_3 + \theta(6).in_4 \\ &f_4 = \theta(7).in_3 + \theta(8).in_4 \end{aligned} \tag{7}$$

Then the output of the node n_{17a} n to n_{20a} can be obtained

$$\begin{aligned} n_{17a} &= n_{13a}.f_1 \\ n_{18a} &= n_{14a}.f_2 \\ n_{19a} &= n_{15a}.f_3 \\ n_{20a} &= n_{16a}.f_4 \end{aligned} \tag{8}$$

Layer 5: The output of this layer is the summation of all incoming signals

$$n_{21a} = n_{17a} + n_{18a} + n_{19a} + n_{20a} \tag{9}$$

Backward pass

In this stage, the error between the output of the network and the desired output d_k should be defined. Then the sum of squared error is given by

$$Ep = \sum_{k=1}^{N(\ell)} (d_k^p - x_{L,k}^p)^2 \tag{10}$$

where $E_p = \epsilon_{21}.X_L$ in this layer is n_{21a}, and $d_k = N_1$.

Hence

$$\varepsilon_{21} = -2(N_1 - n_{21a}) \tag{11}$$

next, d_{21} is defined as

$$d_{21} = -\varepsilon_{21}/2 \tag{12}$$

The output of the node n_{21} becomes

$$n_{21b} = n_{21a} + d_{21} \tag{13}$$

In fact

$$n_{21b} = n_{17b} + n_{18b} + n_{19b} + n_{20b} \tag{14}$$

All the mechanisms in the forward pass in layer 1 through to layer 5 are used in the same manner for the backward pass to produce a corrected value (labelled b after added by corrected factors, d_i, $i = 9\ldots21$).

Further, by extending the correction rule of the backward pass of Mo-ANFIS learning algorithm for 4 inputs, allows to further obtain the corrected value of each node. Finally, after the corrected value in layer 1 had been obtained, then all inputs were mapped to this value by interpolation. The mapping function then becomes the mapping function of the learning mechanism of Mo-ANFIS.

Streamflow simulation

A decision support system is developed involving a modified ANFIS decision-making method and applied to the streamflow forecasting in the Citarum River basin, Indonesia. The Mo-ANFIS decision support system is found to be useful due to its interactive nature, flexibility in approach, and evolving graphical features, and can be adopted for any similar situation to predict the streamflow. Figure 3 presents the sample screen of the main module of Mo-ANFIS.

Performance of the Mo-ANFIS methodology was compared with the actual data using two performance criteria as shown in Table 1.

The training period was considered for the first four years and the last of the data was used for training the model. The input layer consisted of four nodes representing the

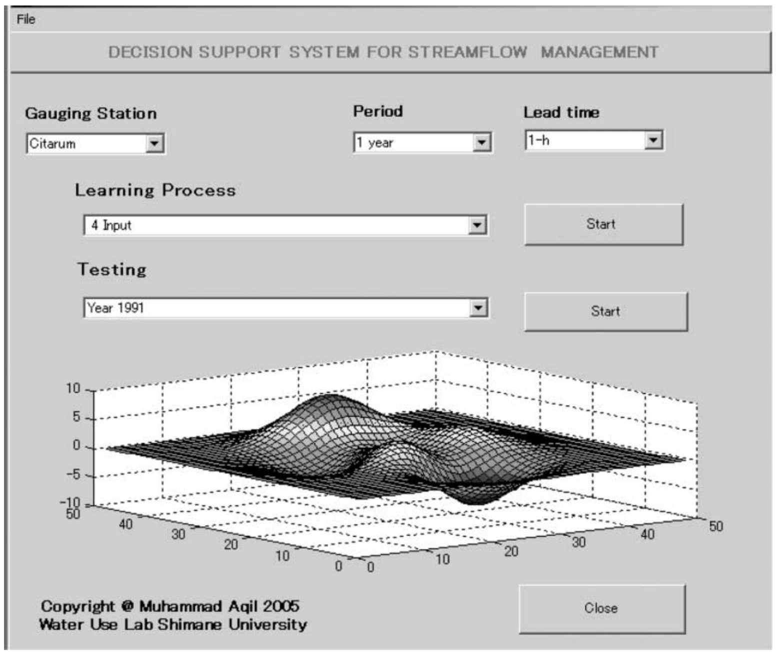

Figure 3 Sample screen of the main module of the Mo-ANFIS model

daily streamflow values at year t, $t-1$, $t-2$ and $t-3$, and the output layer consisted of a single node representing the flow value at $t+1$. As already mentioned, the initial model parameters were formed using a grid partitioning method, and two bell membership functions were used for training the model. Using these membership functions, each company in the sector under analysis is assigned a membership function based upon its position in the range. Training was performed until a maximum of 1 000 iterations was obtained. During the network training, prediction models were generated from historical data, which was provided and was combined with another parameter to deliver the results. In order to find the optimum membership parameters for the input models, the Mo-ANFIS algorithm is employed. The number of rules defined in this model is a product of the number of membership functions in each input. Therefore, the model contains 16 $(2 \times 2 \times 2 \times 2)$ rules. The membership functions of the river stage after training are shown in Figure 4. Comparing the final membership functions with those before training, we see how the final membership functions are trying to catch the local features of the training set.

The comparison between the predicted and actual flow values at training phase is shown in Figure 5. From Figure 5, it can be seen that the comparison of the Mo-ANFIS predictions with the observed data was satisfactory, and Mo-ANFIS can catch the low and peak flows with a good generalization. During the training stage, the MAPE and RMSE resulted were 2.632 and 9.663%, respectively. The final values of model parameters obtained after training were then used as the optimal parameter combination for testing the networks. Testing of the networks was done using the daily data during year

Table 1 Statistical parameter used to test the model

Statistical parameter	Expression
Root mean square error (RMSE)	$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(Q_i^o - Q_i^p)^2}{n}}$
Mean absolute percentage error (MAPE)	$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left\lvert\frac{Q_i^p - Q_i^o}{Q_i^o}\right\rvert \times 100$

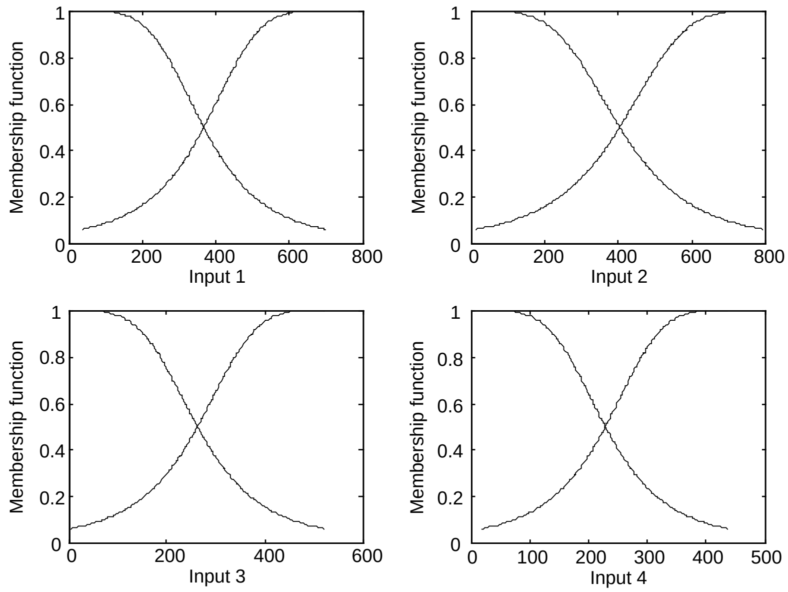

Figure 4 The final membership functions for input parameters

1991 to 2002. A sample screen of the testing module during the testing period of 1992 is shown in Figure 6.

Table 2 shows the distribution of the forecasting errors (MAPE and RMSE) throughout the year for the Citarum River basin. The forecast results regarding the MAPE and RMSE values during the validation phase were approximately 2.632–5.650% and 6.957–11.826%, respectively. Thus, the results indicate that the neuro-fuzzy model is able to identify the events for which it was designed, although the extent to which this model can generalize its ability to forecast events was not included in the training process.

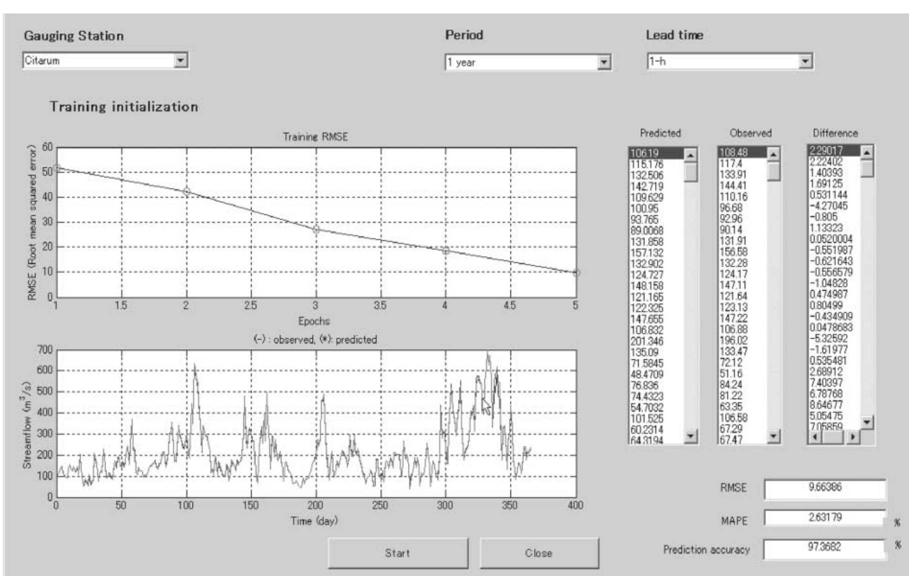

Figure 5 Sample screen of the training module (1987–1990)

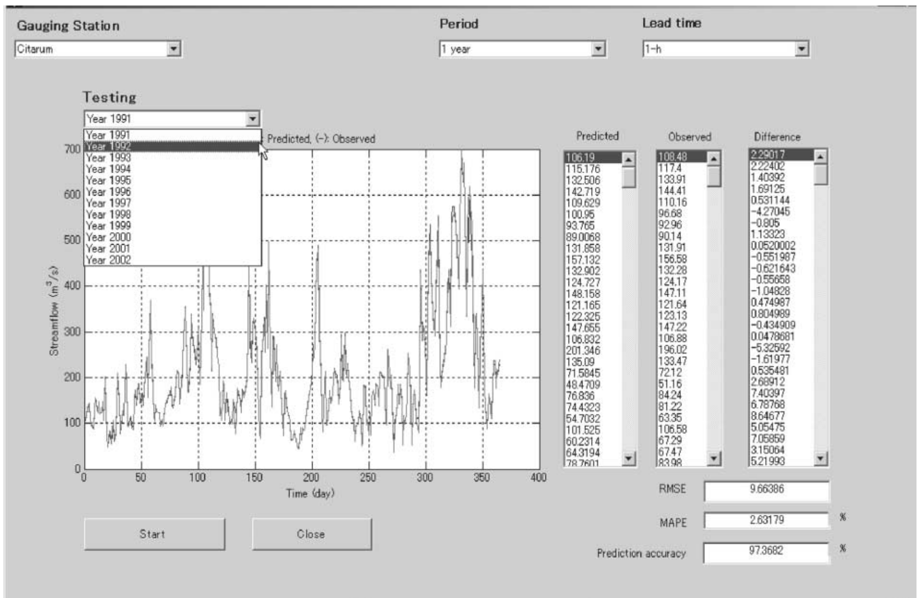

Figure 6 Sample screen of the training module (1987–1990)

Table 2 Forecasting performance of Mo-ANFIS model during the testing period

Number	Testing period	Performance	
		MAPE (%)	RMSE
1	1991	2.632	9.664
2	1992	5.151	11.826
3	1993	5.560	8.578
4	1995	3.514	11.826
5	1996	2.632	8.579
6	1997	5.151	9.664
7	2000	5.151	11.826
8	2001	5.560	8.579
9	2002	3.514	6.957

Conclusion

A Mo-ANFIS model is proposed to modify the traditional ANFIS model. The aim is to modify the error correction rule of the EBP method. Effectiveness of the proposed identification technique is demonstrated through a simulation study on the flood series of the Citarum River in Indonesia. The first four-year data (1987 to 1990) was used for model training/calibration, while the other remaining data (1991 to 2002) was used for testing the model. The number of antecedent flows that should be included in the input variables was determined by two statistical methods, i.e. autocorrelation and partial autocorrelation between the variables. Performance accuracy of the model was evaluated in terms of two statistical indices, i.e. mean average percentage error and root mean square error. In order to enable users to process the data, a modified neuro-fuzzy decision support system is developed. The Mo-ANFIS decision support system is found to be useful due to its interactive nature, flexibility in approach, and evolving graphical features, and can be adopted for any similar situation to predict the streamflow. The main data processing includes gauging station selection, input generation, lead-time selection/generation, and length of prediction. This program enables users to process the flood data, to train/test the model using various input options, and to visualize results. The program code consists of a set of files, which can be modified as well to match other purposes. This program may also serve as a tool for real-time flood monitoring and process control. The results indicate that the modified neuro-fuzzy model applied to the flood prediction seems to have reached encouraging results for the river basin under examination. The comparison of the modified neuro-fuzzy predictions with the observed data was satisfactory, where the error resulted from the testing period was varied between 2.632 and 5.560%. Thus, this program may also serve as a tool for real-time flood monitoring and process control.

Acknowledgements

The authors would like to extend grateful thanks to Perusahaan Umum Jasa Tirta II (PJT II), West Java, Indonesia for cooperation and assistance during the data collection exercise.

References

Bazartseren, B., Hildebrandt, G. and Holz, K.P. (2003). Short-term water level prediction using neural networks and neuro-fuzzy approach. *Neurocomputing*, **55**, 439–450.

Chang, F.J. and Chen, Y.C. (2001). A counterpropagation fuzzy-neural network modeling approach to real time streamflow prediction. *J. Hydrol.*, **245**, 153–164.

Dixon, B. (2004). Applicability of neuro-fuzzy techniques in predicting ground-water vulnerability: a GIS-based sensitivity analysis. *J. Hydrol.*, **309**, 17–38.

Duan, Q., Sorooshian, S. and Gupta, V.K. (1992). Effective and efficient global optimization for conceptual rainfall runoff models. *Water Resour. Res.*, **28**, 1015–1031.

Nayak, P.C., Sudheer, K.P., Rangan, D.M. and Ramasastri, K.S. (2004). A neuro-fuzzy computing technique for modeling hydrological time series. *J. Hydrol.*, **291**, 52–66.

Ponnambalam, K., Karray, F. and Mousavi, S.J. (2003). Minimizing variance of reservoir systems operations benefits using soft computing tools. *Fuzzy Sets and Syst.*, **139**, 451–461.

Sorooshian, S. and Gupta, V.K. (1995). Model calibration. In *Computer Models of Watershed Hydrology*, Singh, V.P. (ed.), Water Resources Publications, Colorado, pp. 23–68.

Valenca, M. and Ludermir, T. (2000). Monthly streamflow forecasting using an neural fuzzy network model. *Proceedings of the Sixth Brazilian Symposium on Neural Networks*, **6**, 117–119.

Vernieuwe, H., Georgieva, O., Baets, B.D., Pauwels, V.R.N., Verhoest, N.E.C. and Troch, P.D. (2004). Comparison of data-driven Takagi–Sugeno models of rainfall-discharge dynamics. *J. Hydrol.*, **291**(1-4), 173–186.