HYDROLOGICAL PROCESSES
Hydrol. Process. (2008)
Published online in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/hyp.7136

Comparison of artificial neural network models for hydrologic predictions at multiple gauging stations in an agricultural watershed

E. Mutlu,[1] I. Chaubey,[2]* H. Hexmoor[3] and S. G. Bajwa[1]

[1] *Department of Biological and Agricultural Engineering, University of Arkansas, Fayetteville, AR 72701, USA*
[2] *Departments of Agricultural and Biological Engineering, and Earth and Atmospheric Sciences, Purdue University, West Lafayette, IN 47907, USA*
[3] *Department of Computer Sciences, Southern Illinois University, Carbondale, IL 62901, USA*

Abstract:

This paper reports on an evaluation of the use of artificial neural network (ANN) models to forecast daily flows at multiple gauging stations in Eucha Watershed, an agricultural watershed located in north-west Arkansas and north-east Oklahoma. Two different neural network models, the multilayer perceptron (MLP) and the radial basis neural network (RBFNN), were developed and their abilities to predict stream flow at four gauging stations were compared. Different scenarios using various combinations of data sets such as rainfall and stream flow at various lags were developed and compared for their ability to make flow predictions at four gauging stations. The input vector selection for both models involved quantification of the statistical properties such as cross-, auto- and partial autocorrelation of the data series that best represented the hydrologic response of the watershed. Measured data with 739 patterns of input–output vector were divided into two sets: 492 patterns for training, and the remaining 247 patterns for testing. The best performance based on the RMSE, R^2 and CE was achieved by the MLP model with current and antecedent precipitation and antecedent flow as model inputs. The MLP model testing resulted in R^2 values of 0·86, 0·86, 0·81, and 0·79 at the four gauging stations. Similarly, the testing R^2 values for the RBFNN model were 0·60, 0·57, 0·58, and 0·56 for the four gauging stations. Both models performed satisfactorily for flow predictions at multiple gauging stations, however, the MLP model outperformed the RBFNN model. The training time was in the range 1–2 min for MLP, and 5–10 s for RBFNN on a Pentium IV processor running at 2·8 GHz with 1 MB of RAM. The difference in model training time occurred because of the clustering methods used in the RBFNN model. The RBFNN uses a fuzzy min-max network to perform the clustering to construct the neural network which takes considerably less time than the MLP model. Results show that ANN models are useful tools for forecasting the hydrologic response at multiple points of interest in agricultural watersheds. Copyright © 2008 John Wiley & Sons, Ltd.

KEY WORDS ANN models; rainfall-runoff prediction, agricultural watershed

Received 4 August 2006; Accepted 16 July 2008

INTRODUCTION

Over the last few decades, a plethora of mathematical rainfall–runoff models have been developed to quantify and understand watershed-scale hydrologic processes. Based on the description of the governing processes, these models can be classified as either physics based or system theoretic. Physics based model involve a detailed description of various physical processes controlling the hydrologic behaviour of a system. However, system theoretic models do not consider the physical characteristics of the parameters; they map the data from input to output using transfer functions. Artificial neural network (ANN) models are example of system theoretic models that have gained considerable popularity in recent years in describing rainfall–runoff processes.

ANNs are artificial intelligence-based computational tools that can mimic the biological processes of a human brain. They are considered suitable tools for large search spaces where human expertise is needed. They do not require detailed knowledge of internal functions of a system in order to recognize relationships between inputs and outputs. For various complex nonlinear environmental problems, ANNs have an advantage over distributed parameter models in that the data requirements are usually less, and they are more suited for long-term forecasting. The measured data used for ANN model development are divided into two groups: training and testing. First the ANN model is trained to represent the relationships and processes within the measured training data set. Once the model is adequately trained, it is able to generalize relevant output for the set of input data. This output is subsequently compared with the measured testing data set. The model is considered to behave satisfactorily if its performance during the testing period is similar to that during the training period.

Many researchers have successfully utilized ANNs to simulate rainfall–runoff processes (Hapuarachchi and Zhijia, 2003; Zakermoshfeg *et al.*, 2004; Anctil and Rat, 2005; Cigizoglu, 2005; Kumar *et al.*, 2005). Example applications include real time flood forecasting (Sudheer,

* Correspondence to: I. Chaubey, Departments of Agricultural and Biological Engineering, and Earth and Atmospheric Sciences, Purdue University, West Lafayette, IN 47907, USA. E-mail: ichaubey@purdue.edu

2000; Thirumalaiah and Deo, 2000), climate change impacts on hydrology (Clair and Ehrman, 1996), and choice of predictor variable on simulated flow (Abrahart *et al*., 2001) among others. The two most-commonly used ANN methods for modelling rainfall–runoff processes are the multi layer perceptron (MLP) (Kumar *et al*., 2005) and radial basis function neural network (RBFNN) models. Many studies have reported on comparisons of the two models in simulating rainfall–runoff processes (Zakermoshfegh *et al*., 2004; Kumar *et al*., 2005). Generally both models are found to give satisfactory performance in modelling hydrologic processes and the performance of the model is reported to depend upon many factors, including choice of network type and number of input variables used. These studies have recommended that many more evaluations are needed before a clear choice of network could be established. In addition, analyses of the published results indicate that ANNs should be regarded as an alternative to more traditional rainfall–runoff methods rather than a replacement (Maier and Dandy, 2000).

One of the limitations of the previously published ANN models is that all models were evaluated at a single point of interest, i.e. ability of the ANN model to predict runoff at a single gauging station, such as, watershed outlet. Although previous studies have clearly shown that ANN models can be used effectively to predict rainfall–runoff processes at a single gauging station for various temporal resolutions (Kumar *et al*., 2005), the ability of these models to predict flow at more than one gauging stations is currently not known. While, flow predictions at the watershed outlet are very important, often, there is great interest in predicting flow at intermediate points or gauging stations within a watershed. Many distributed rainfall–runoff and water quality models such as the Soil and Water Assessment Tool model (Arnold *et al*., 1998) have been developed to meet this need. There is a need to develop and evaluate the ability of ANN models in predicting flow at multiple gauging stations.

The objectives of this study were to develop and evaluate the ability of MLP and RBFNN models to predict flow at multiple gauging stations in an agricultural watershed. Various input vectors and their impact on the flow prediction abilities of the two models were evaluated.

METHODOLOGY

Multi-layer perceptron (MLP) model description

The MLP is the most popular neural network architecture in use today (Dawson and Wilby, 2001; Kumar *et al*., 2005). An MLP is a network of simple neurons called perceptrons. The perceptron computes a single output from multiple real-valued inputs by forming a linear combination according to its input weights and then possibly putting the output through some nonlinear activation function. Mathematically this can be represented as

$$y = \varphi\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{1}$$

where w_i denotes the vector of weights, x_i is the vector of inputs ($i = 1, 2..n$), b is the bias, y is the output and φ is the activation function. A signal-flow graph of this operation is shown in Figure 1. The activation function is often chosen to be the logistic sigmoid function defined as

$$1/\left(1 + e^{-x}\right) \tag{2}$$

The MLP is usually trained using the error backpropagation algorithm. This popular algorithm works by iteratively changing a network's interconnecting weights such that the overall error (i.e. between observed values and modelled network outputs) is minimized (Govindaraju and Rao, 2000; Sudheer, 2000).

Radial basis function neural network (RBFNN) model description

The RBFNN developed by Powell (1987) and Broomhead and Lowe (1988) also consists of an input layer, a single hidden layer, and an output layer. Figure 2 shows a typical RBFNN model. The number of input and output nodes is, like the MLP neural network, determined by the nature of the actual input and output variables. The number of hidden nodes, however, is not determined by trial and error; instead, the fuzzy min-max clustering algorithm is used to decide the number of hidden nodes. The output of the RBFNN is calculated according to

$$Y = \sum W\theta\left(||X - C||\right) \tag{3}$$

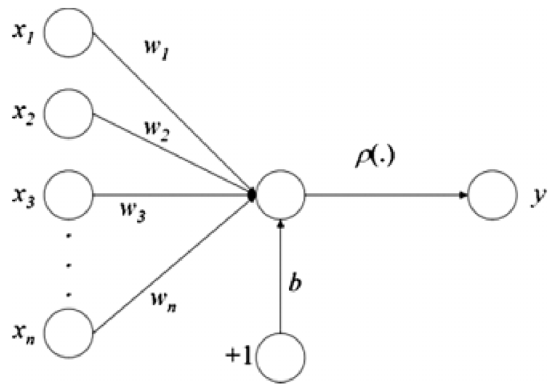

Figure 1. Conceptual schematic of a typical MLP network

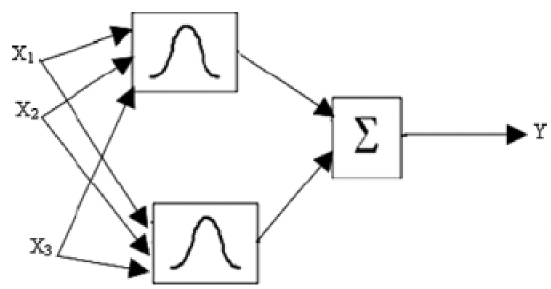

Figure 2. Schematic representation of a RBFNN model with one output

DOI: 10.1002/hyp

where X = input value, Y = output value, $\theta()$ = radial basis function, W = weights connecting the hidden and output nodes; and C represents the centre of each hidden node, which depends on the observed input data. $||X - C||$ is the Euclidean distance between the input and hidden nodes. Each hidden node represents a group of input nodes that have similar information from the input data. The transformation associated with each node of the hidden layer is a Gaussian function (Govindaraju and Rao, 2000; Sudheer, 2000).

Description of the study site

Measured data from Eucha watershed were used to develop and compare the ability of both MLP and RBFNN models to predict stream flow at multiple gauging stations. Eucha watershed is a 1203·5 km^2 drainage basin located with 70% in Delaware County, Oklahoma and the remainder located in Benton County, Arkansas (Figure 3). Elevation in the watershed ranges from approximately 233 to 433 m above mean sea level (Wagner and Woodruf, 1997). The watershed is located in the Ozark Highlands and Central Plains ecoregions of the Ozark Plateau. The land cover of the watershed was derived from Landsat data from 1999 to 2003, which indicated that 50% of the watershed area was forest and 40% pasture.

The USGS operates four stream gauging stations in the watershed; these stations are: 07 191 160 (Spavinaw Creek near Cherokee, AR); 07 191 179 (Spavinaw Creek near Maysville, AR); 7 191 220 (Spavinaw Creek near Sycamore, OK), and 71 912 213 (Spavinaw Creek near Colcord, OK) and are herein referred to as Station 1, Station 2, Station 3, and Station 4, respectively (Figure 3).

The mean temperature ranges from 3·3 °C in January to 28 °C in July in the watershed. Temperatures greater than 38 °C occur on average 15 days per year, temperatures above 32 °C occur on average 71 days per year, and temperatures below freezing occur on average 85 days per year. In spring an average of 38% of rainfall occurs, while 16% of the total rainfall occurs in winter. Total annual precipitation averages approximately 1·143 m. Snowfall ranges from 0·127–0·177 m (Wagner and Woodruf, 1997).

Input set preparation and model architecture

In this study, different models using various combinations of input data sets were developed and compared for their ability to make flow predictions at four gauging stations for 2002–2004. The input variables used in models included various combinations of daily precipitation (inches) and antecedent flow at the gauging stations at which flow predictions were made as well as flow at hydrologically connected upstream gauging stations (Table I). Daily stream flow values and daily rainfall totals for each of the four gauging stations located in the watershed (Figure 3) were obtained from the USGS. Figure 4 shows the approach taken to develop the ANN models for flow predictions at the four gauging stations. The model development started at the most upstream site (Station 1) and proceeded in the downstream direction. Predicted flows from upstream sites were used as an input to the model.

The goal of an ANN is to generalize a relationship of the form

$$Y^m = f(X^n) \tag{4}$$

where X^n is an n-dimensional input vector consisting of variables $x_1, \ldots, x_i, \ldots, x_n$; Y^m is an m-dimensional output vector consisting of the resulting variables of interest $y_1, \ldots, y_i, \ldots, y_m$. In rainfall-runoff process,

Table I. Input scenarios used to develop and test ANN models for flow prediction in Eucha watershed

Scenario number	Input variable
1	Precipitation (P)
2	P, antecedent P
3	P, antecedent P, antecedent Flow (F)

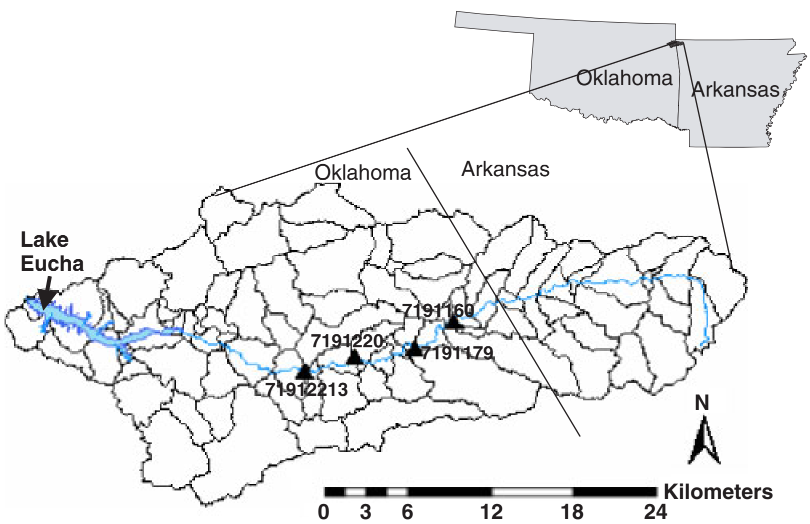

Figure 3. Location of the Eucha watershed with major streams and gauging stations that were used in developing and comparing ANN models

DOI: 10.1002/hyp

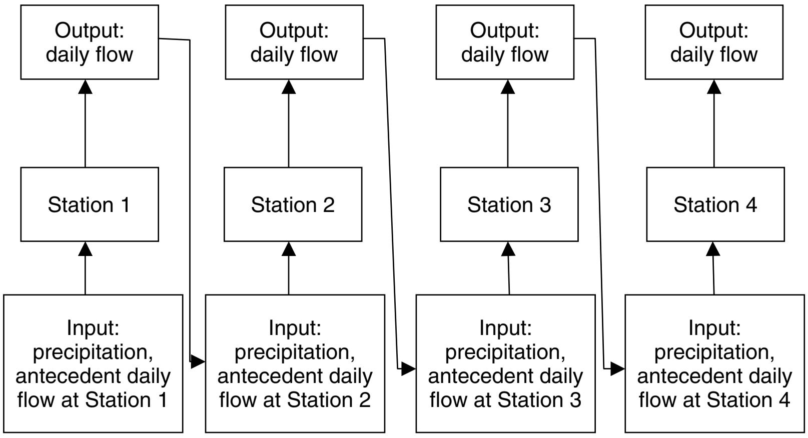

Figure 4. Approach taken to develop ANN models for flow predictions at four gauging stations in the watershed

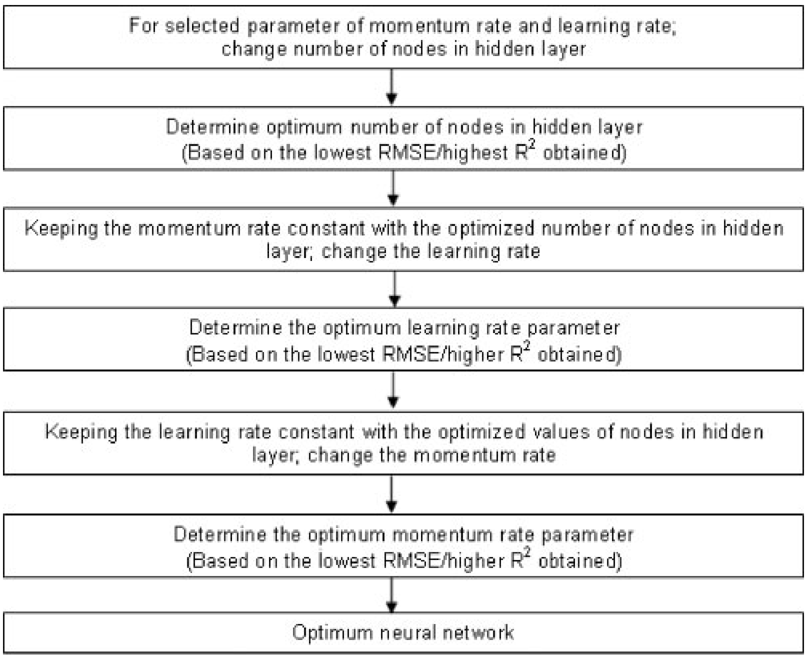

Figure 5. Schematics of the procedure used for determining the optimum neural network schema

values of x_i may be rainfall-runoff values with different lags and the value of y_i is generally the next day's flow. However, the number of antecedent rainfall–runoff values that need to be included in the vector X^n is watershed specific and may not be known *a priori*. Determining the number of rainfall-runoff parameters involves finding the lags of rainfall-runoff that have a significant influence on the predicted flow. These influencing values corresponding to different lags can be very well established through statistical analysis of the data series.

Several statistics such as cross–correlation, autocorrelation and partial autocorrelation were generated to evaluate rainfall and runoff values to be included in the model. The cross-correlation is computed at several different lags and shows the degree of linear relationship between the data values. The autocorrelation describes the correlation between all the pairs of points in the time series with a given separation in time or lag. A partial autocorrelation is the autocorrelation of a series with itself under stationary conditions, while controlling for the effect of intervening lags. A partial autocorrelation reveals the precise autocorrelation of a series with itself without the confounding effects of intervening lagged autocorrelation. In this analysis, many networks were trained with various combinations of rainfall corresponding to different lags (varying from 1 to 10 days) and runoff lags (varying from 1 to 8 days).

After inputs were determined, the MLP and RBFNN models were optimized to obtain the best possible prediction model. Figure 5 shows the procedure used in this study. The number of nodes was changed in the hidden layer to determine the optimum number for the MLP and RBFNN models. The momentum rate was kept constant with the optimized number of hidden layer or prototype layer nodes and the learning rate was varied to determine the optimum learning rate parameter. The learning rate was kept constant with the optimized values of hidden

DOI: 10.1002/hyp

layer or prototype layer nodes and the momentum rate was changed to determine the optimum momentum rate parameter.

The above mentioned steps were followed for MLP and RBFNN models to determine the optimum number of nodes, learning rate and momentum rate. In this study the backpropagation algorithm, delta learning rule and the sigmoid transfer function were used for model development. Learning and momentum rates were changed between 0·1 and 0·9. RMSE, CE and R^2 were used to evaluate the performance of the model for each station. The ANN was trained using the error backpropagation algorithm. This popular algorithm works by iteratively changing a network's interconnecting weights such that the overall error (i.e. between observed values and modelled network outputs) is reduced.

The data were subdivided into two sets: a first set to train the model (training set) and a second set to test (testing set) the model. For the daily time step model, 2002 to 2003 data were used for training and the data for the year 2004 for testing. The total data consisting of 739 patterns of input–output vector were divided into two sets: 492 patterns for training, the remaining 247 patterns for testing. The software package used for ANN simulations in this study was Neuralworks Professional II Plus (Neuralware, 2003).

Model evaluation criteria

Multiple performance criteria have been used by researchers to evaluate the adequacy of ANN models in simulating rainfall–runoff processes. The most commonly employed error measures are the RMSE and coefficient of efficiency (Solomatine and Dulal, 2003). The Nash–Sutcliffe coefficient of model efficiency (CE) (Nash and Sutcliffe, 1970) and the percentage of total error were used by Hapuarachchi and Zhijia (2003). Zakermoshfegh *et al.* (2004) used the sum of square error (SSE) and RMSE. Abrahart *et al.* (2001), and Thirumalaiah and Deo (2000) used optimization algorithms such as cascade correlation, and genetic algorithms. The three numeric error measures used in this study were RMSE, R^2 and CE defined as

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(Q_P - Q_O)^2}{n}} \tag{5}$$

$$\mathrm{R}^2 = \left[\frac{\sum_{i=1}^{n}(Q_O - \overline{Q}_O)(Q_P - \overline{Q}_P)}{\sqrt{\sum(Q_O - \overline{Q}_O)^2 \sum(Q_P - \overline{Q}_P)^2}}\right]^2 \tag{6}$$

$$\mathrm{CE} = 1 - \frac{\sum_{i=1}^{n}(Q_O - Q_P)^2}{\sum(Q_O - \overline{Q}_P)^2} \tag{7}$$

where Q_P are the n predicted values ($I = 1$ to n), Q_O are the n observed values, $\overline{Q}_O$ is the mean of n observed values, and $\overline{Q}_P$ is the mean of n predicted values.

RESULTS AND DISCUSSION

Data analysis for model input preparation

The cross-correlation, autocorrelation, and partial autocorrelation functions were analysed for the four USGS gauging stations. The cross-correlation statistics of the precipitation and flow series are presented in Figure 6 and were found to initially increase with lag for all gauging stations. The greatest values were obtained at a lag 2

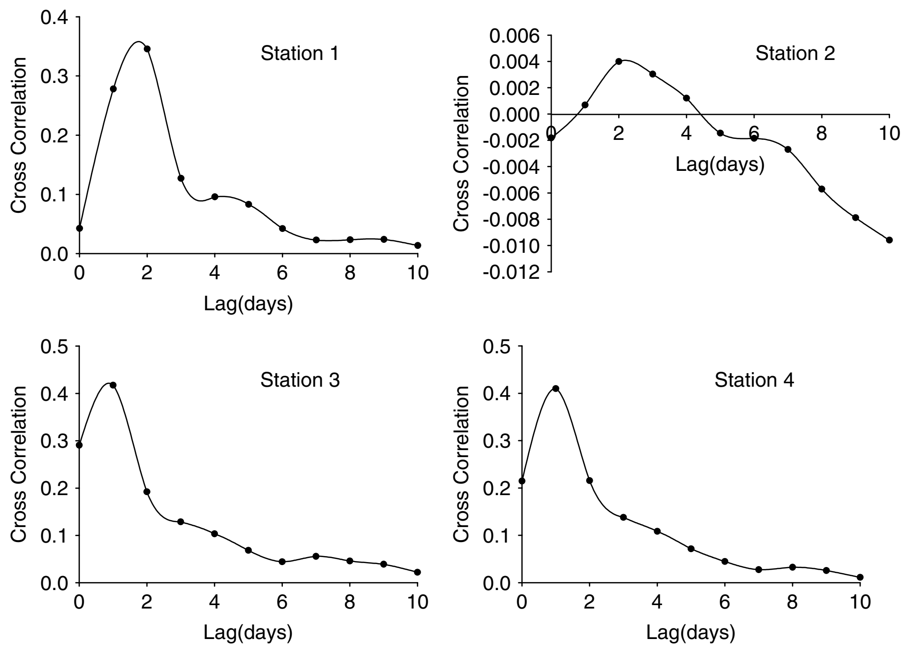

Figure 6. Cross-correlation plots of the daily precipitation-flow series of Eucha watershed for four USGS gauging stations

DOI: 10.1002/hyp

(2 days) for Stations 1 and 2 and lag 1 (1 day) for Stations 3 and 4. Lag time for each station is a function of the runoff time of concentration and depends on watershed geophysical characteristics such as watershed area, slope, shape, flow length, and land use. The lag time for each station was different because of differences in geophysical characteristics of the contributing subwatersheds.

The autocorrelation function and the corresponding 95% confidence intervals from lag 0 to lag 8 (0 to 8 days) were estimated for the flow series (Figure 7). The autocorrelation function showed a significant correlation up to lag 5 for Station 1and 2, lag 6 for Station 3 and lag 7 for Station 4, at the 95% confidence level, and thereafter, fell within the confidence interval. The gradual decaying pattern of the autocorrelation indicates the presence of a dominant autoregressive process. Similarly, the partial autocorrelation function and corresponding 95% confidence limits were estimated for lag 0 to lag 8 (Figure 8). The partial autocorrelation function showed significant correlation up to lag 1 (1 day) for all stations and, thereafter, fell within the confidence band. The rapid decaying pattern of the partial autocorrelation function confirms the dominance of the autoregressive process, relative to the moving-average process. The input variables are determined by autocorrelation function and partial autocorrelation function analyses on flow values, and cross-correlation function analysis between precipitation and flow values. This quantitative analysis of the data series relieves the modeller of a long trial- and-error procedure in identifying the appropriate input vector that best represents the process in the watershed. The above analysis of auto and partial correlation coefficients suggested incorporating flow values of up to 1 day lag in the input vector to the network for all stations.

The coefficient of determination (R^2) values for the MLP model for training ranged from 0·001 to 0·02 for the four stations for input scenario 1 when only current day precipitation was used to make flow predictions. Similarly, the range of R^2 was 0·007 to 0·08 when the current day precipitation and antecedent precipitation derived from the aforementioned analyses were used for input scenario 2 for model training. A very small R^2 values for all four gauging stations for these two scenarios indicated that precipitation alone could not be used to make accurate flow predictions using the MLP model. However, when antecedent flow was also used in the input matrix the range of R^2 values dramatically increased to 0·92–0·95 for all four gauging stations. The best performance based on the RMSE, R^2 and CE was achieved for scenarios 3, with consistently higher R^2 and lower RMSE than other scenarios during training of the MLP model.

It should be noted that RBFNN models were also trained using the input vectors that were identified from the above analysis. Other scenarios were not considered, as they were unable to improve the performance of the MLP and RBFNN models. We have not presented those results here for brevity. To have a true comparison with MLPs, the RBFNN models were developed using scenario 3 data sets that gave the best flow predictions for the MLP model. Comparison of MLP and RBFNN models are discussed in the following sections.

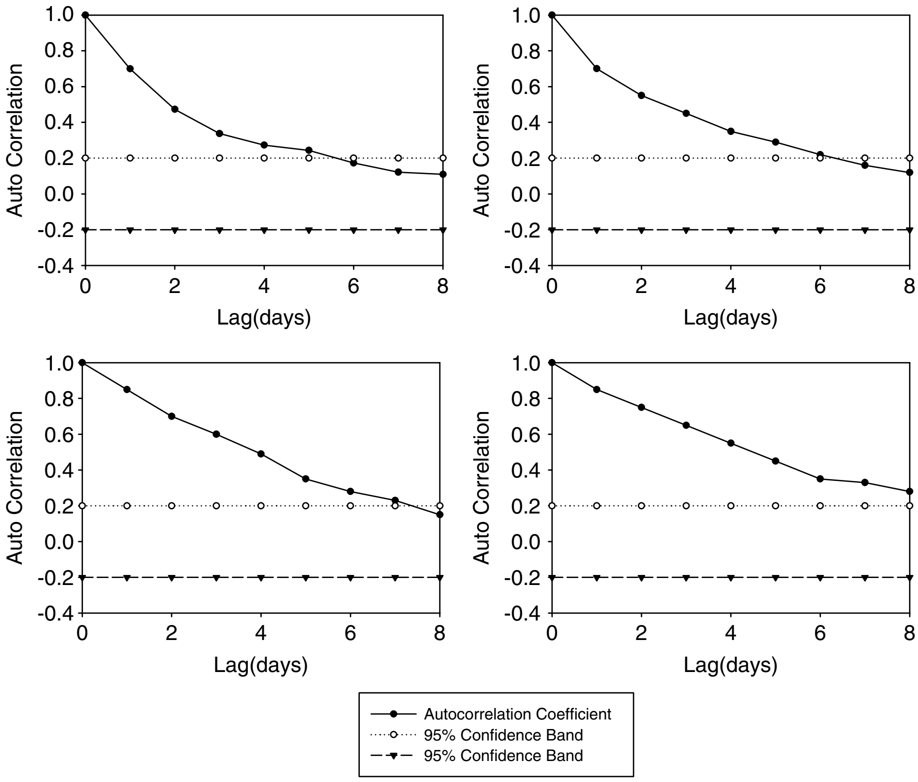

Figure 7. Autocorrelation plot of the flow series at four USGS gauging stations in the Eucha watershed

DOI: 10.1002/hyp

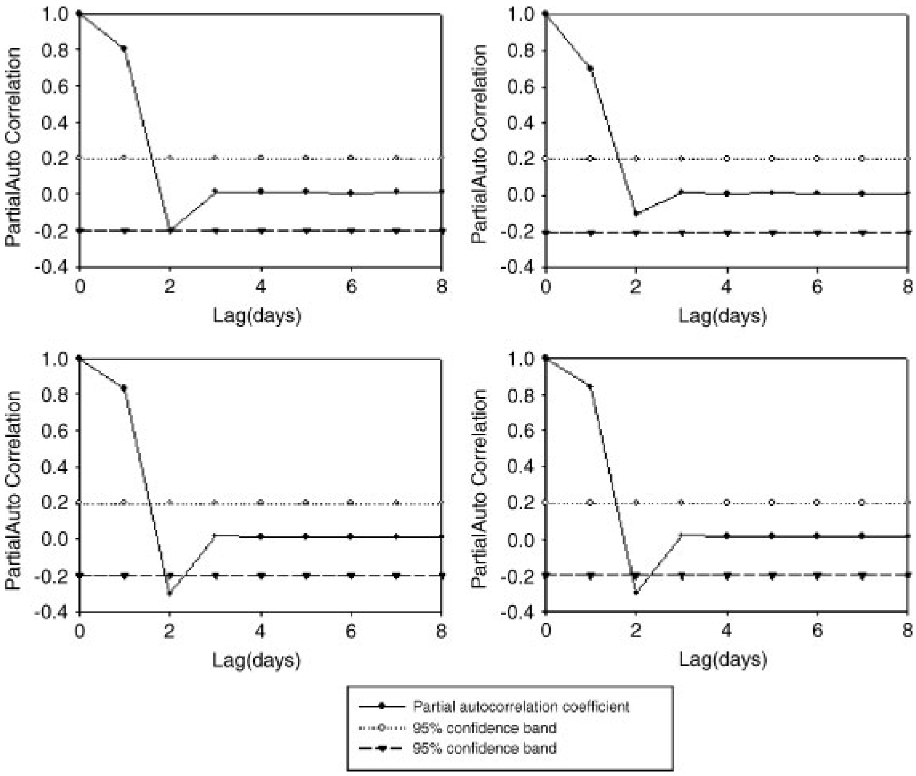

Figure 8. Partial autocorrelation plots of the flow series at four USGS gauging stations in the Eucha watershed

Table II. Performance of MLP and RBFNN models simulating flow at various gauging stations for input scenario 3

	MLP				RBFNN			
Training	Station 1	Station 2	Station 3	Station 4	Station 1	Station 2	Station 3	Station 4
R^2	0·92	0·93	0·95	0·94	0·93	0·81	0·79	0·82
RMSE	0·155	0·217	0·184	0·348	0·15	0·547	0·806	0·975
CE	0·918	0·923	0·952	0·935	0·921	0·806	0·789	0·818
Testing								
R^2	0·86	0·86	0·81	0·79	0·6	0·57	0·58	0·56
RMSE	0·226	0·203	0·4	0·398	0·455	0·573	0·676	0·673
CE	0·789	0·812	0·729	0·664	0·575	0·471	0·544	0·431
Overall								
R^2	0·9	0·91	0·92	0·91	0·85	0·76	0·75	0·79
RMSE	0·128	0·181	0·264	0·533	0·254	1·234	2·326	3·06
CE	0·891	0·905	0·917	0·91	0·846	0·752	0·754	0·785

Comparison of MLP and RBFNN models

The values of R^2, CE and RMSE for scenario 3 at each station are presented in Table II. The R^2 for the MLP model ranged from 0·92 to 0·95 for the training data set and from 0·79 to 0·86 for the testing data set. Similarly, RMSE ranged from 0·155 $m^3\ s^{-1}$ to 0·348 $m^3\ s^{-1}$ for the training data set and from 0·203 $m^3\ s^{-1}$ to 0·4 $m^3\ s^{-1}$ for the testing data set. CE ranged from 0·9178 to 0·9516 for the training data set and from 0·6635 to 0·8121 for the testing data set. It should be noted that although the predicted flows at upstream gauging stations were used as an input to the downstream flow predictions (Figure 3), the greatest R^2 was obtained for gauging Station 3 for the training data set, and gauging Station 2 and 1 for the testing data set, showing the robustness of the model in adequately simulating flow at multiple gauging stations within the watershed. In this study all four gauging stations were hydrologically connected, i.e. flow from Station 1 directly affected flow at Station 2, 3 and 4. When ANN models are used to make flow predictions at multiple gauging stations that are hydrologically connected, attention should be paid to the accuracy of the model predictions at upstream stations. It is possible that a poor performance at upstream stations may lead to error propagation at the downstream stations.

The R^2 values for the RBFNN model ranged from 0·79 to 0·93 for the training data set and from 0·56 to 0·60 for the testing data set with the greatest R^2 obtained at gauging Station 1. The RMSE ranged from 0·150 $m^3\ s^{-1}$ to 0·975 $m^3\ s^{-1}$ for the training data set

DOI: 10.1002/hyp

and from 0·455 $m^3 s^{-1}$ to 0·676 $m^3 s^{-1}$ for the testing data set. The CE for the RBFNN model ranged from 0·431 to 0·921 during training and testing. At Station 1, the RBFNN models outperform the MLPs during training; however, they fail to preserve their performance during testing, implying poor generalization properties for RBFNN models. Unlike MLPs, RBFNN networks do not have any associated connections between input hidden nodes, which give a weighted input to each hidden node before the nonlinear transformation takes place. Thus, in an RBFNN, for any point in the input space the response of the closest basis function plays a major role in the output of the network. Consequently, a trained RBFNN network's output will be accurate only if the input pattern falls close to the centre of the basis function, and hence exhibits poor generalization properties (Moody and Darken, 1989). It can be seen from Table II that the MLP model has higher R^2, CE and lower RMSE than the RBFNN model at Stations 2, 3 and 4 during training and testing.

The results in Table II emphasize the importance of not relying on any single error measure to assess model performance. Figure 9 shows scatter plots of predicted and observed data during the testing period for scenario

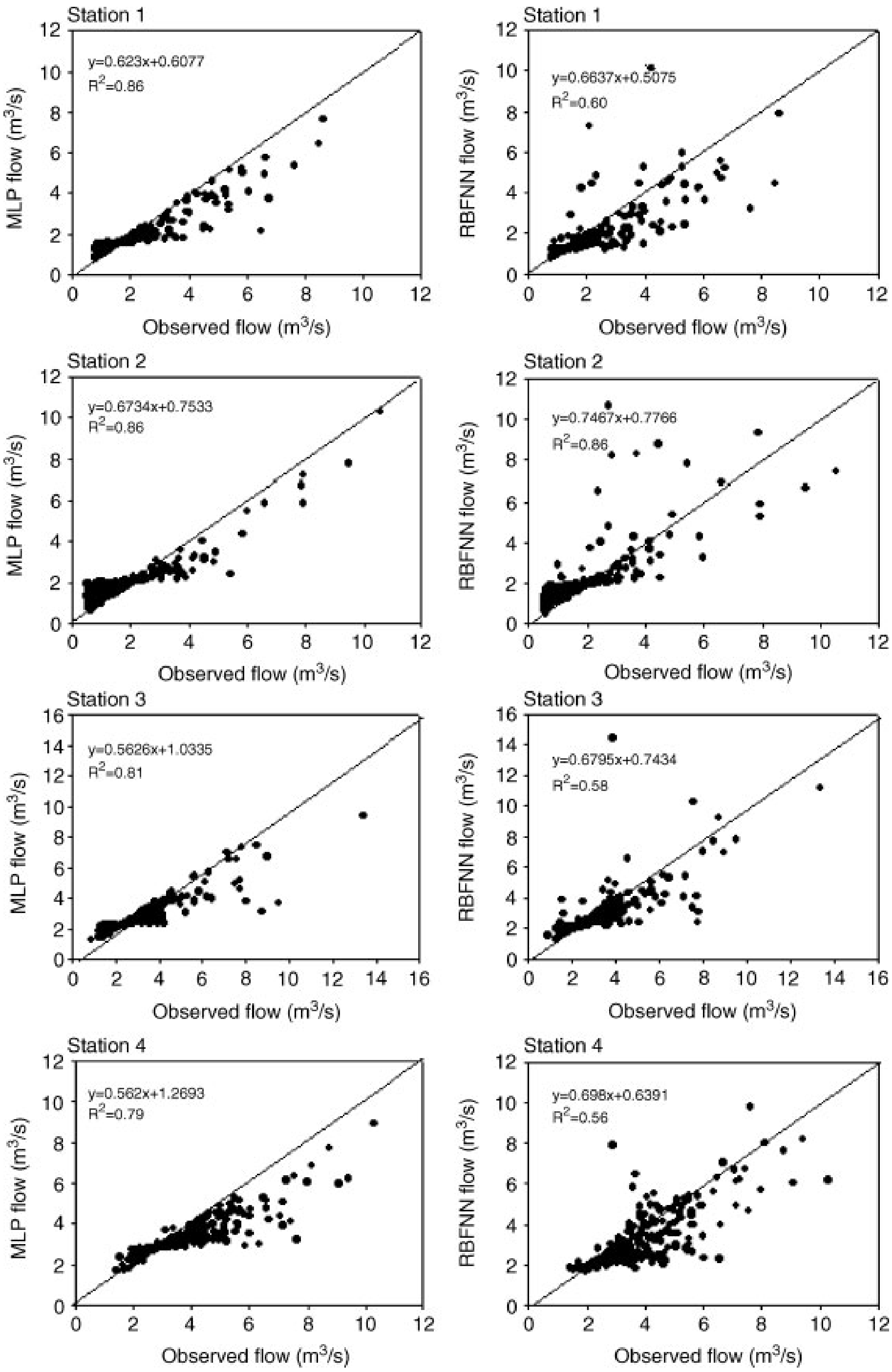

Figure 9. Scatter plots of observed versus MLP and RBFNN model flow during testing for the four stations

DOI: 10.1002/hyp

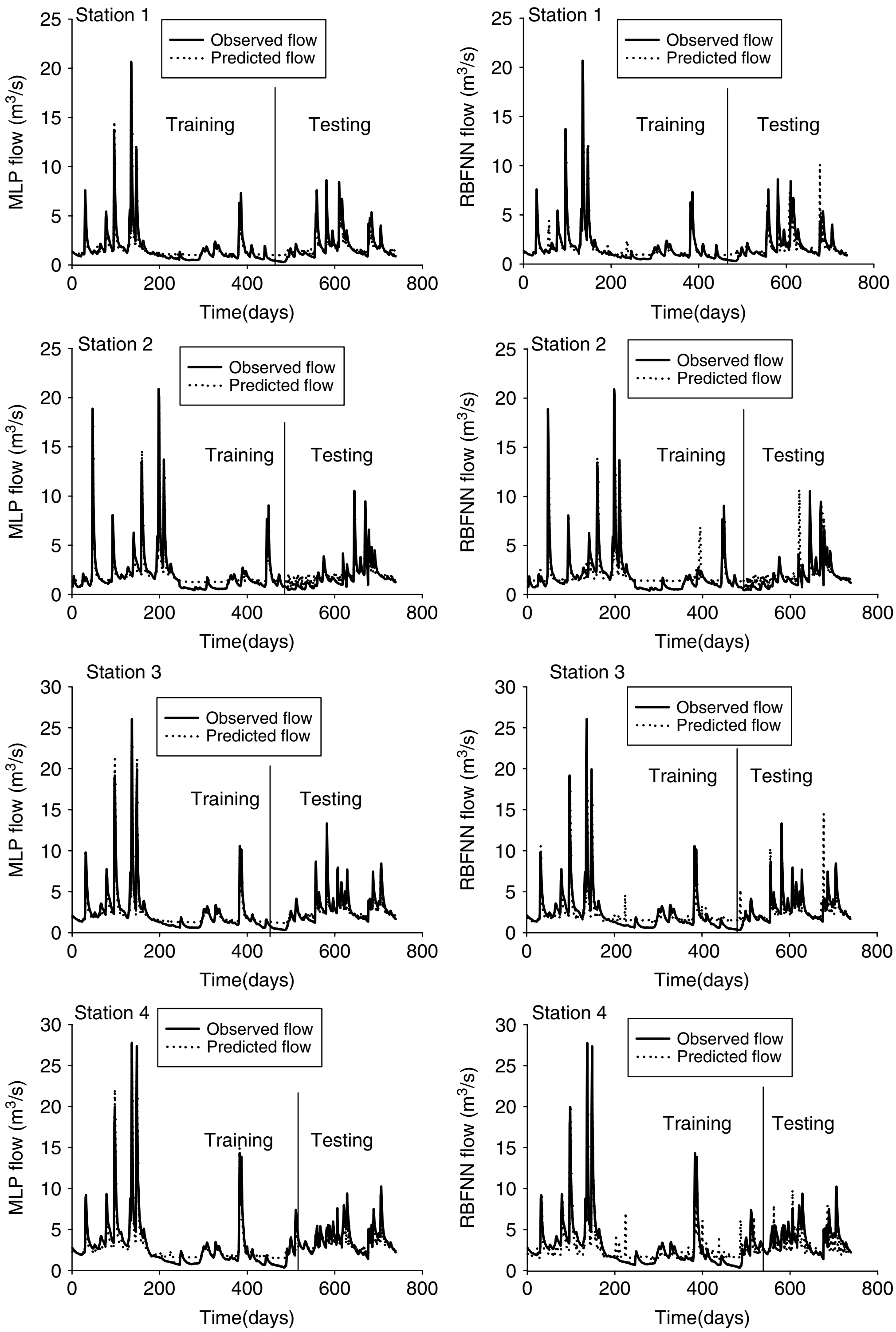

Figure 10. Hydrographs of time versus flow during training and testing for the four stations

3 for all stations for both MLP and RBFNN models. Scatter plots between measured and predicted flow data serve as a useful visual aid to assess a model's accuracy. The closer the scatter points are to the line of the best fit, the better the model. Figure 9 shows that the MLP models had smaller scatter around the best fit line than the RBFNN models for all four gauging stations. Figure 10 shows the corresponding hydrographs for the four stations during training and testing. These plots, when combined with results shown in Table II, indicate that model performance for the full range of flow data evaluated in this study was superior for the MLP model, as indicated by greater R^2 and CE and lower RMSE values compared with those for the RBFNN model. Both models underpredicted high flow events and over predicted low flow events at some of the gauging stations (Figures 9 and 10). The cause of these discrepancies needs to be further evaluated to improve

DOI: 10.1002/hyp

model performance. However, the overall results confirm that the ANN model performed well simulating daily stream flow based on daily precipitation, antecedent precipitation and antecedent flow.

One performance criterion for which the RBFNN model worked better than the MLP was the training time. The observed training time of the MLP was around 1–2 min, whereas the training time of the RBFNN was 5–10 s on a Pentium IV processor running at 2·8 GHz with 1 MB of RAM. The difference occurs because of clustering methods used in RBFNN where a fuzzy min-max method is used to perform the clustering to construct the neural network. These result in faster training of the RBFNN network compared with the MLP model. However, with the availability of fast computers, this should not be a significant concern affecting model selection for flow predictions at multiple gauging stations.

SUMMARY AND CONCLUSIONS

The objectives of this study were to develop and compare ANN models to forecast daily flows at multiple gauging stations in the Eucha watershed in north-west Arkansas and north-east Oklahoma. MLP and RBFNN models were developed and their abilities to predict stream flow at four gauging stations were compared. The input vector selection for both models involved quantification of the statistical properties such as cross-, auto- and partial autocorrelation of the data series that best represented the hydrologic response of the watershed. The MLP model testing resulted in R^2 values of 0·86, 0·86, 0·81 and 0·79 at the four gauging stations. Similarly, the R^2 values for the RBFNN model were 0·60, 0·57, 0·58, and 0·56 for the four gauging stations. The MLP model performed better for forecasting daily flow at multiple gauging stations in the watershed. Based on statistical analysis the MLP outperformed the RBFNN in terms of the RMSE, CE and R^2 during training and testing. However, additional research on watersheds with heterogeneous hydrologic characteristics is needed to establish the model of choice as the RBFNN model is reported to have superior performance in different studies (Fernando and Jayawardena, 1998; Sudheer, 2000) primarily in terms of less time required for model development and training.

The training time was in the range 1–2 minutes for MLP, and 5–10 seconds for RBFNN on a Pentium IV processor running at 2·8 GHz with 1 MB of RAM. The difference in model training time occurred because of the clustering methods used in the RBFNN model. The RBFNN uses a fuzzy min-max network to perform the clustering to construct the neural network which takes considerably less time than the MLP model. The results obtained from this study indicate that ANN models are useful tools for forecasting the hydrologic response at multiple points of interest in agricultural watersheds.

REFERENCES

Abrahart RJ, See L, Kneale PE. 2001. investigating the role of saliency analysis with a neural network rainfall-runoff model. *Computers & Geosciences* **27**: 921–928.

Anctil F, Rat A. 2005. evaluation of neural network stream flow forecasting on 47 watersheds. *Journal of Hydrologic Engineering* **10**: 85–88.

Arnold JG, Srinivasan R, Muttiah RS, Williams JR. 1998. Large-area hydrologic modeling and assessment. Part I. Model development. *Journal American Water Resources Association* **34**: 73–89.

Broomhead DS, Lowe D. 1988. Multivariable functional interpolation and adaptive networks. *Complex System* **2**: 321–355.

Cigizoglu HK. 2005. Application of generalized regression neural networks to intermittent flow forecasting and estimation. *Journal of Hydrologic Engineering* **10**: 336–341.

Clair TA, Ehrman JM. 1996. Variations in discharge and dissolved organic carbon and nitrogen export from terrestrial basins with changes in climate: a neural network approach. *Limnology and Oceanography* **41**: 921–927.

Dawson CW, Wilby RL. 2001. Hydrological modeling using artificial neural networks. *Progress in Physical Geography* **25**: 80–108.

Fernando AK, Jayawardena AW. 1998. Runoff forecasting using RBF networks with OLS algorithm. *Journal of Hydrologic Engineering, ASCE.* **3**: 203–209.

Govindaraju RS, Rao AR. 2000. *Artificial Neural Networks in Hydrology*. Kluwer Academic Publisher: Netherlands.

Hapuarachchi HAP, Zhijia L. 2003. Application of models with different types of modeling methodologies for river flow forecasting. *Weather Radar Information and Distributed Hydrological Modeling* 218–226.

Kumar APS, Sudheer KP, Jain SK, Agarwal PK. 2005. Rainfall-runoff modeling using artificial neural networks: comparison of network types. *Hydrological Processes* **19**: 1277–1291.

Maier HR, Dandy GC. 2000. Neural networks for the prediction and forecasting of water resources variables: a review of modeling issues and applications. *Environmental Modeling and Software* **15**: 101–23.

Moody J, Darken C. 1989. Fast learning in networks of locally tuned processing units. *Neural Computation* **1**: 281–294.

Nash JE, Sutcliffe JV. 1970. River flow forecasting through conceptual models: Part I. A discussion of principles. *Journal of Hydrology* **10**: 282–290.

Neuralware. 2003. *Neuralworks Professional II Plus*. Carnegia, PA http://www.neuralware.com/index.jsp.

Powell MJD. 1987. Radial basis functions for multivariable interpolation: a review. *Proceedings of IMA Conference on Algorithms for Approximation*, Mason JC, Cox MG (eds). Oxford University Press: New York; 143–167.

Solomatine DP, Dulal KN. 2003. Model trees as an alternative to neural networks in rainfall–runoff modeling. *Hydrological Sciences Journal* **48**: 399–411.

Sudheer KP. 2000. *Modeling hydrological processes using neural computing technique*. PhD thesis, Indian Institute of Technology, Delhi, India.

Thirumalaiah K, Deo MC. 2000. Hydrological forecasting using neural networks. *Journal of Hydrologic Engineering* **5**: 180–189.

Wagner K, Woodruf S. 1997. Phase 1 Clean Lakes Project Diagnostic and Feasibility Study of Lake Eucha. Final Report. Executive Summary 1–148.

Zakermoshfegh M, Ghodsian M, Montazer GA. 2004. River flow forecasting using artificial neural networks. *Hydraulics of Dam and River Structures* **90**: 425–430.

DOI: 10.1002/hyp