

Multi-layer perceptrons with Levenberg-Marquardt training algorithm for suspended sediment concentration prediction and estimation

ÖZGÜR KIŞI

Civil Engineering Department, Hydraulics Division, Engineering Faculty, Erciyes University, 38039 Kayseri, Turkey

kisi@erciyes.edu.tr

Abstract The prediction and estimation of suspended sediment concentration are investigated by using multi-layer perceptrons (MLP). The fastest MLP training algorithm, that is the Levenberg-Marquardt algorithm, is used for optimization of the network weights for data from two stations on the Tongue River in Montana, USA. The first part of the study deals with prediction and estimation of upstream and downstream station sediment data, separately, and the second part focuses on the estimation of downstream suspended sediment data by using data from both stations. In each case, the MLP test results are compared to those of generalized regression neural networks (GRNN), radial basis function (RBF) and multi-linear regression (MLR) for the best-input combinations. Based on the comparisons, it was found that the MLP generally gives better suspended sediment concentration estimates than the other neural network techniques and the conventional statistical method (MLR). However, for the estimation of maximum sediment peak, the RBF was mostly found to be better than the MLP and the other techniques. The results also indicate that the RBF and GRNN may provide better performance than the MLP in the estimation of the total sediment load.

Key words estimation; generalized regression neural networks; multi-layer perceptrons; multi-linear regression; prediction; radial basis function; suspended sediment concentration

Prévision et estimation de la concentration en matières en suspension avec des perceptrons multi-couches et l'algorithme d'apprentissage de Levenberg-Marquardt

Résumé Des perceptrons multi-couches (PMC) sont testés pour la prévision et l'estimation de la concentration en matières en suspension. L'algorithme d'apprentissage de PMC le plus rapide, l'algorithme de Levenberg-Marquardt, est utilisé pour optimiser les poids du réseau, avec les données de deux stations de la Rivière Tongue dans le Montana, aux États-Unis. La première partie de l'étude traite de la prévision et de l'estimation des données sédimentaires en abordant séparément les stations amont et aval, tandis que la deuxième partie porte sur l'estimation des données de matières en suspension à l'aval à partir des données des deux stations. Dans chaque cas, les résultats de test des PMC sont comparés aux résultats obtenus avec des réseaux de neurones de régression généralisée (RNRG), des réseaux à fonctions de base radiale (RBR) et des régressions multi-linéaires (RML) pour les meilleures combinaisons de données d'entrée. Ces comparaisons montrent que les estimations de la concentration en matières en suspension sont en général meilleures avec le PMC qu'avec les autres techniques de réseau de neurones et qu'avec la méthode statistique conventionnelle (RML). Cependant, le RBR est apparu meilleur que le PMC et que les autres techniques pour l'estimation du pic maximum de transport solide. Les résultats indiquent également que le RBR et le RNRG peuvent présenter de meilleures performances que le PMC pour l'estimation de la charge solide totale.

Mots clefs estimation; réseaux de neurones de régression généralisée; perceptrons multi-couches; régression multi-linéaire; réseaux de neurones; prévision; fonction de base radiale; concentration en matières en suspension

INTRODUCTION

The assessment of the volume of sediment being transported by a river is of particular interest in hydraulic engineering due to its importance in the design and management of water resources projects. The prediction of river sediment load constitutes an important issue in hydraulic and sanitary engineering. In reservoir design, a “dead storage” volume is meant to accommodate the sediment income that will accumulate over the “economic life”. The underestimation of sediment yield results in insufficient reservoir capacity while overestimation leads to over-capacity reservoirs. Appropriate reservoir design and operation justify every effort to determine sediment yield accurately, but in environmental engineering the prediction of river sediment load has an additional significance, especially if the particles also transport pollutants.

A number of attempts have been made to relate the amount of sediment transported by a river to flow conditions such as discharge, velocity and shear stress. However, none of the equations derived have received universal acceptance. Usually, either the weight or the concentration of sediment is related to the discharge. These two forms are often used interchangeably. McBean & Al-Nassri (1988) examined this issue and concluded that the practice of using sediment load *vs* discharge is misleading because the goodness of fit implied by this relationship is spurious. Instead they recommended that the regression link be established.

The physically-based models are based on the simplified partial differential equations of flow and sediment flux as well as on some unrealistic simplifying assumptions for flow and empirical relationships for erosive effects of rainfall and flow. Examples of such models are presented by Wicks & Bathurst (1996), Refsgaard (1997) and others. These highly sophisticated and complex models have components that correspond to physical processes. They are theoretically capable of accounting the spatial variation of catchment properties as well as uneven distribution of precipitation and evapotranspiration. The model complexity should, however, be keyed to utilizable information about the catchment characteristics and density and frequency of the available input data. In particular, because the real spatial distribution of precipitation is not presently measurable for much of the world, process-oriented distributed models offer no practical advantage over lumped models and have many disadvantages (Guldal & Muftuoglu, 2001).

Neural networks (NN) have been successfully applied in a number of diverse fields including water resources. In the hydrological forecasting context, artificial neural networks (ANNs) may offer a promising alternative for rainfall–runoff modelling (Zhu & Fujita, 1994; Shamseldin, 1997; Tokar & Johnson, 1999; Wilby *et al.*, 2003; Solomatine & Dulal, 2003), streamflow prediction (Kang *et al.*, 1993; Clair & Ehrman, 1998; Chang & Chen, 2001; Sivakumar *et al.*, 2002; Cigizoglu, 2003; Chibanga *et al.*, 2003; Kisi, 2004a; Cigizoglu & Kisi, 2004), and reservoir inflow forecasting (Saad *et al.*, 1996; Jain *et al.*, 1999). There are few published works in the field of suspended sediment data prediction using artificial intelligence methods such as neural networks and fuzzy logic approach (Jain, 2001; Tayfur, 2002; Tayfur *et al.*, 2003; Kisi, 2002; Kisi, 2004b). Coulibaly *et al.* (1999) reviewed the ANN-based modelling in hydrology over the last years, and reported that about 90% of the experiments extensively make use of the multi-layer feed-forward neural networks (FNN) trained by the standard backpropagation (BP) algorithm (Rumelhart *et al.*, 1986). Maier & Dandy (2000) reviewed 43 papers dealing with the use of the ANN model for the prediction and forecasting of water resources variables. They reported

that in 41 papers multi-layer perceptron (MLP) neural networks with gradient descent algorithm were used. Since they often yield sub-optimal solutions (Hagan & Menhaj, 1994; El-Bakyr, 2003), a more powerful MLP learning algorithm, that is, the Levenberg-Marquardt algorithm is used for all applications in this study.

The main purpose of this study is to analyse and discuss the performances of three different ANN techniques, namely, the MLP, generalized regression neural networks (GRNN) and the radial basis function neural networks (RBF), in the prediction and estimation of suspended sediment concentration.

This paper is organized as follows: the next section provides an overview description of the ANN techniques; the third section gives a brief description of the data used in the study; and the fourth section provides two different applications to prediction and estimation of suspended sediment concentration. In each application, different numbers of input combinations are tried and the performance of the MLP models is compared to those of the GRNN, RBF and multi-linear regression (MLR) for the best input combinations that give the lowest mean root square errors (MRSE) and mean absolute error (MAE). Finally, the last section provides findings and concluding remarks.

METHODOLOGY

Multi-layer perceptrons (MLP)

An MLP distinguishes itself by the presence of one or more hidden layers, with computation nodes called hidden neurons, whose function is to intervene between the external input and the network output in a useful manner. By adding hidden layer(s), the network is enabled to extract higher-order statistics. In a rather loose sense, the network acquires a global perspective despite its local connectivity due to the extra set of synaptic connections and the extra dimension of NN interconnections. For more information about MLP, see Haykin (1994).

The MLP can have more than one hidden layer; however, theoretical works have shown that a single hidden layer is sufficient for an ANN to approximate any complex nonlinear function (Cybenko, 1989; Hornik *et al.*, 1989). Therefore, in this study, a one-hidden-layer MLP is used. Throughout all MLP simulations, the adaptive learning rates were used to speed up training. The numbers of hidden layer neurons were found using simple trial-and-error method in all applications.

The MLPs were trained using the Levenberg–Marquardt technique as this technique is more powerful than the conventional gradient descent techniques (Hagan & Menhaj, 1994; El-Bakyr, 2003; Cigizoglu & Kisi, 2004).

The Levenberg-Marquardt algorithm

While back-propagation with gradient descent technique is a steepest descent algorithm, the Levenberg-Marquardt algorithm is an approximation to Newton's method (Marquardt, 1963). If a function $V(\underline{x})$ is to be minimized with respect to the parameter vector \underline{x} , then Newton's method would be:

$$\Delta \underline{x} = -[\nabla^2 V(\underline{x})]^{-1} \nabla V(\underline{x}) \quad (1)$$

where $\nabla^2 V(\underline{x})$ is the Hessian matrix and $\nabla V(\underline{x})$ is the gradient. If $V(\underline{x})$ reads:

$$V(\underline{x}) = \sum_{i=1}^N e_i^2(\underline{x}) \quad (2)$$

then it can be shown that:

$$\nabla V(\underline{x}) = J^T(\underline{x}) \underline{e}(\underline{x}) \quad (3)$$

$$\nabla^2 V(\underline{x}) = J^T(\underline{x})J(\underline{x}) + S(\underline{x}) \quad (4)$$

where $J(\underline{x})$ is the Jacobian matrix and

$$S(\underline{x}) = \sum_{i=1}^N e_i \nabla^2 e_i(\underline{x}) \quad (5)$$

For the Gauss-Newton method it is assumed that $S(\underline{x}) \approx 0$, and equation (1) becomes:

$$\Delta \underline{x} = [J^T(\underline{x})J(\underline{x})]^{-1} J^T(\underline{x}) \underline{e}(\underline{x}) \quad (6)$$

The Levenberg-Marquardt modification to the Gauss-Newton method is:

$$\Delta \underline{x} = [J^T(\underline{x})J(\underline{x}) + \mu I]^{-1} J^T(\underline{x}) \underline{e}(\underline{x}) \quad (7)$$

The parameter μ is multiplied by some factor (β) whenever a step would result in an increased $V(\underline{x})$. When a step reduces $V(\underline{x})$, μ is divided by β . When the scalar μ is very large the Levenberg-Marquardt algorithm approximates the steepest descent method. However, when μ is small, it is the same as the Gauss-Newton method. Since the Gauss-Newton method converges faster and more accurately towards an error minimum, the goal is to shift towards the Gauss-Newton method as quickly as possible. The value of μ is decreased after each step unless the change in error is positive; i.e. the error increases. For the neural network-mapping problem, the terms in the Jacobian matrix can be computed by a simple modification to the back-propagation algorithm (Hagan & Menhaj, 1994).

The generalized regression neural networks (GRNN)

A schematic of the GRNN is shown in Fig. 1. The basics of the GRNN can be found in the literature (Specht, 1991; Tsoukalas & Uhrig, 1997). The GRNN consists of four layers: an input layer, a pattern layer, a summation layer, and an output layer. The number of input units in the first layer is equal to the total number of parameters. The first layer is connected to the second, pattern layer, where each unit represents a training pattern and its output is a measure of the distance of the input from the stored patterns. Each pattern layer unit is connected to the two neurons in the summation layer: S-summation neuron and D-summation neuron. The former computes the sum of the weighted outputs of the pattern layer while the latter calculates the unweighted outputs of the pattern neurons. The connection weight between the i th neuron in the pattern layer and the S-summation neuron is y_i ; the target output value corresponding to the i th input pattern. For the D-summation neuron, the connection weight is unity. The output layer merely divides the output of each S-summation neuron by that of each D-summation neuron, yielding the predicted value to an unknown input vector x as:

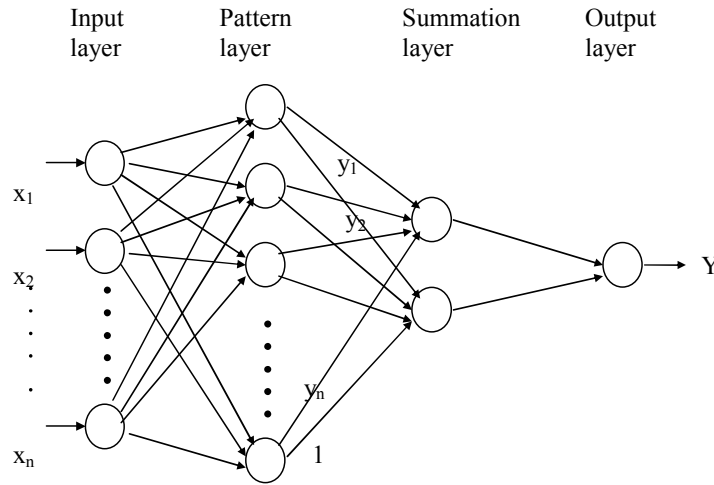


Fig. 1 Schematic diagram of a GRNN.

$$\hat{y}_i(x) = \frac{\sum_{i=1}^n y_i \exp[-D(x, x_i)]}{\sum_{i=1}^n \exp[-D(x, x_i)]} \quad (8)$$

where n indicates the number of training patterns and the Gaussian D function in equation (8) is defined as:

$$D(x, x_i) = \sum_{j=1}^p \left(\frac{x_j - x_{ij}}{\zeta} \right)^2 \quad (9)$$

where p indicates the number of elements of an input vector. The terms x_j and x_{ij} represent the j th element of x and x_i , respectively. The term ζ is generally referred to as the spread factor, whose optimal value is often determined experimentally (Kim *et al.*, 2003). The GRNN method does not require an iterative training procedure as for the MLP method. The GRNN method is used for estimation of continuous variables, as in standard regression techniques. It is related to the radial basis function network and is based on a standard statistical technique called kernel regression (Specht, 1991). The joint probability density function (pdf) of x and y is estimated during a training process in the GRNN. Because the pdf is derived from the training data with no preconceptions about its form, the system is perfectly general. The success of the GRNN method depends heavily on the spread factors (Specht, 1991; Wasserman, 1993). The larger that spread is, the smoother the function approximation. Too large a spread means a lot of neurons will be required to fit a fast changing function. Too small a spread means many neurons will be required to fit a smooth function, and the network may not generalize well.

The radial basis function type neural networks (RBF)

Radial basis function networks were introduced into the neural network literature by Broomhead & Lowe (1988). The RBF network model is motivated by the locally tuned

response observed in biological neurons. Neurons with a locally tuned response characteristic can be found in several parts of the nervous system, for example, cells in the visual cortex sensitive to bars oriented in a certain direction or other visual features within a small region of the visual field (Poggio & Girosi, 1990). These locally tuned neurons show response characteristics bounded to a small range of the input space. The theoretical basis of the RBF approach lies in the field of interpolation of multivariate functions. The objective of interpolating a set of tuples $(x^s, y^s)_{s=1}^N$ with $x^s \in R^d$ is to find a function $F: R^d \rightarrow R$ with $F(x^s) = y^s$ for all $s = 1, \dots, N$. In the RBF approach, the interpolating function F is a linear combination of basis functions:

$$F(x) = \sum_{s=1}^N w_s \phi(\|x - x^s\|) + p(x) \quad (10)$$

where $\|\cdot\|$ denotes Euclidean norm, w_1, \dots, w_N are real numbers, ϕ a real valued function, and $p \in \prod_n^d$ a polynomial of degree at most n (fixed in advance) in d variables. The interpolation problem is to determine the real coefficients w_1, \dots, w_N and the polynomial term $p = \sum_{l=1}^D a_l p_l$ where p_1, \dots, p_D is the standard basis of \prod_n^d and a_1, \dots, a_D are real coefficients. The interpolation conditions are:

$$F(x^s) = y^s, \quad s = 1, \dots, N \quad (11)$$

and

$$\sum_{s=1}^N w_s p_j(x^s) = 0, \quad j = 1, \dots, D \quad (12)$$

The function ϕ is called a radial basis function if the interpolation problem has a unique solution for any choice of data points. In some cases the polynomial term in equation (10) can be omitted, and by combining it with equation (11), one obtains:

$$\phi w = y \quad (13)$$

where $w = (w_1, \dots, w_N)$, $y = (y^1, \dots, y^N)$, and ϕ is a $N \times N$ matrix defined by:

$$\phi = \left(\phi(\|x^k - x^s\|) \right)_{k,s=1,\dots,N} \quad (14)$$

Provided the inverse of ϕ exists, the solution w of the interpolation problem can be explicitly calculated and has the form: $w = \phi^{-1}y$. The most popular and widely used radial basis function is the Gaussian basis function:

$$\phi(\|x - c\|) = e^{-\|x-c\|^2/2\sigma^2} \quad (15)$$

with peak at centre $c \in R^d$ and decreasing as the distance from the centre increases.

The solution of the exact interpolating RBF mapping passes through every data point (x^s, y^s) . In the presence of noise, the exact solution of the interpolation problem is typically a function oscillating between the given data points. An additional problem with the exact interpolation procedure is that the number of basis functions is equal to the number of data points and so calculating the inverse of the $N \times N$ matrix ϕ becomes intractable in practice. The interpretation of the RBF method as an artificial neural network consists of three layers: a layer of input neurons feeding the feature vectors

into the network; a hidden layer of RBF neurons, calculating the outcome of the basis functions; and a layer of output neurons, calculating a linear combination of the basis functions (Taurino *et al.*, 2003). In the present study, the orthogonal least squares algorithm (Chen *et al.*, 1991) is used in the RBF method. The RBF method does not perform the parameter's learning as in the MLP networks, but just performs linear adjustment of the weights for the radial basis. This characteristic of the RBF method gives the advantage of very fast convergence without local minima, since its error function is always convex (Poggio & Girosi, 1990; Lee & Chang, 2003).

CASE STUDY

The time series of daily streamflow and suspended sediment concentration from two stations on the Tongue River in Montana, USA, are used. The upstream station below Brandenburg Bridge near Ashland and the downstream station at Miles City are operated by the US Geological Survey (USGS). The drainage areas at these sites are 10 521 km² for the upstream station and 13 932 km² for the downstream station. The daily time series for these stations were downloaded from the USGS web server (<http://webserver.cr.usgs.gov/sediment>).

After examining the data and noting the periods in which there were gaps in one or more of the two variables, the periods for calibration and validation were chosen arbitrarily. The data of 1 October 1977–30 September 1980 were chosen for calibration, and data for 1 October 1980–30 September 1981 were chosen for validation. It may be noted that the periods from which calibration and validation data were chosen span the same temporal seasons (October–September).

The scatter plots of the downstream and upstream station data are given in Fig. 2. It can be seen that there is a nonlinear and scattered relationship between discharge and sediment data for both stations. Figure 2 indicates the presence of an outlier in the data set (see also Table 1). A suspended sediment concentration value of 6400 mg l⁻¹ is observed for a discharge as low as 20 m³ s⁻¹, while the other concentration values are below 3000 mg l⁻¹. This outlier is also used in the calibration period. This value creates additional difficulties for the models in prediction or estimation. The models calibrated using such an outlier (6400 mg l⁻¹) generally overestimate low sediment values. The daily statistical parameters of the streamflow and sediment data for the stations are

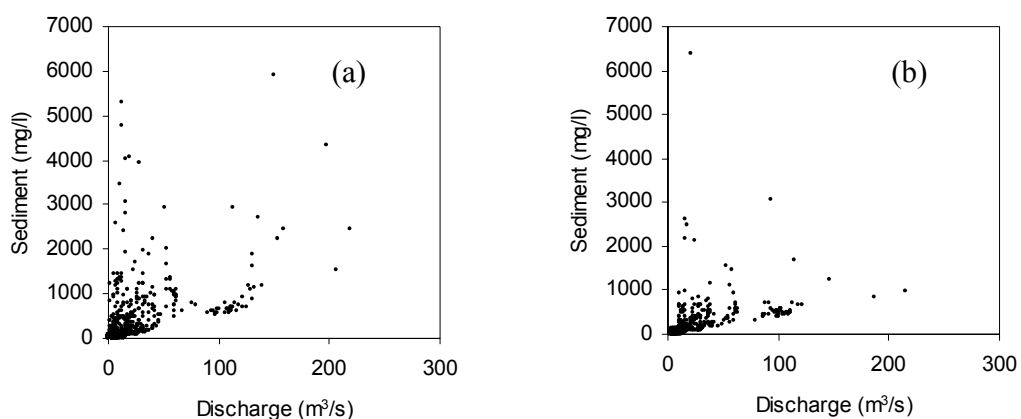


Fig. 2 Scatterplots of the (a) downstream and (b) upstream data.

Table 1 The daily statistical parameters of data set for the stations.

Data set	Station	Basin area (mi ²)	Data type	x_{mean}	S_x	C_v (S_x/x_{mean})	C_{sx}	x_{max}	x_{min}
Training	Downstream (6308500)	5379	Flow (m ³ s ⁻¹)	15.6	24.9	1.60	4.1	218	1.9
			Sediment (mg l ⁻¹)	219	484	2.21	5.4	5900	6
	Upstream (6307830)	4062	Flow (m ³ s ⁻¹)	15.1	20.6	1.36	4.4	215	1.8
			Sediment (mg l ⁻¹)	126	299	2.37	11.4	6400	2
Testing	Downstream (6308500)	5379	Flow (m ³ s ⁻¹)	9.0	12.9	1.43	3.1	62.3	0.1
			Sediment (mg l ⁻¹)	147	283	1.93	3.3	1870	9
	Upstream (6307830)	4062	Flow (m ³ s ⁻¹)	10.3	12.5	1.21	3.1	62.9	1.9
			Sediment (mg l ⁻¹)	87	157	1.80	4.9	1560	5

x_{mean} : mean; S_x : standard deviation; C_v : coefficient of variation; C_{sx} : skewness; x_{max} : maximum; x_{min} : minimum.

given in Table 1. The skewness and coefficient of variation of flow and sediment data of the upstream and downstream stations are high, particularly for the training (calibration) data. In the calibration flow data, x_{min} and x_{max} values fall in the ranges 1.9–218 m³ s⁻¹ for the downstream data. However, the testing flow data set extremes are $x_{\text{min}} = 0.1$ m³ s⁻¹, $x_{\text{max}} = 62.3$ m³ s⁻¹. The value of x_{min} for the calibration flow data is higher than that for the corresponding validation set for the downstream station. This may cause extrapolation difficulties in prediction and estimation of low sediment values.

The auto- and cross-correlation coefficients for the downstream and upstream station data are given in Table 2. It may be seen that the autocorrelation of the sediment data is good for both stations. However, the correlations between the sediment concentrations and discharges are normally not good; in fact they are very poor for the calibration period data of the upstream and downstream stations. The cross-correlations between the upstream and downstream stations are also not high.

Table 2 The auto- and cross-correlations of the data of the downstream and upstream stations.

Autocorrelations:							
<i>Downstream station:</i>		Sd_{t-1}	Sd_{t-2}	Sd_{t-3}	Qd_t	Qd_{t-1}	Qd_{t-2}
Calibration period	Sd_t	0.749	0.511	0.435	0.444	0.416	0.377
Validation period	Sd_t	0.875	0.724	0.644	0.700	0.657	0.600
<i>Upstream station:</i>		Su_{t-1}	Su_{t-2}	Su_{t-3}	Qu_t	Qu_{t-1}	Qu_{t-2}
Calibration period	Su_t	0.707	0.387	0.308	0.565	0.490	0.426
Validation period	Su_t	0.910	0.842	0.778	0.796	0.734	0.669
Cross-correlations between downstream and upstream data:							
	Sd_t	Sd_{t-1}	Sd_{t-2}	Qd_t	Qd_{t-1}	Qd_{t-2}	
Su_t	0.475	0.535	0.513	0.467	0.464	0.445	
Su_{t-1}	0.428	0.436	0.535	0.424	0.467	0.464	
Su_{t-2}	0.366	0.428	0.436	0.369	0.424	0.467	
Qu_t	0.452	0.429	0.396	0.931	0.931	0.906	
Qu_{t-1}	0.471	0.452	0.429	0.920	0.931	0.931	
Qu_{t-2}	0.490	0.471	0.452	0.914	0.920	0.931	

Sd_t and Su_t represent the sediment concentration at time t for the downstream and upstream stations, respectively. Qd_t and Qu_t represent the discharge at time t for the downstream and upstream stations, respectively.

APPLICATION AND RESULTS

A difficult task with the MLP method is choosing the number of hidden nodes. There is no theory yet to tell how many hidden units are needed to approximate any given function. The network geometry is problem dependent. Here, the three-layer MLP with one hidden layer is used and the common trial-and-error method is used to select the number of hidden nodes.

Before applying the MLP method, the input data were normalized to fall in the range [0, 1]. The river flow Q was standardized by the following formula:

$$Q_s = Q/Q_{\max} \quad (16)$$

where Q_s is standardized flow; and Q_{\max} is the maximum of the flow values. The sediment concentration data were also standardized in a similar way. These normalized data were used to train each of the ANN models. After training was over, the weights were saved and used to test (validate) each network performance on test data. The ANN results were transformed back to the original domain, and the mean root square error (*MRSE*) and mean absolute error (*MAE*) were computed using test data for each of the ANN models. The *MRSE* and *MAE* are denoted as:

$$MRSE = \frac{1}{N} \sqrt{\sum_{i=1}^N (Y_{i_{\text{observed}}} - Y_{i_{\text{predicted}}})^2} \quad (17)$$

$$MAE = \frac{1}{N} \sum_{i=1}^N |Y_{i_{\text{observed}}} - Y_{i_{\text{predicted}}}| \quad (18)$$

in which N is the number of data, and Y_i is the sediment concentration.

Next, the performance of the MLP method was tested for two different applications. In the first application, downstream and upstream suspended sediment concentrations were predicted and estimated using either downstream or upstream station data. In the second application, the downstream sediment concentrations were estimated using data from both stations. In each case, a different number of input combinations was tried and the MLP models were compared to GRNN, RBF and MLR models for the best input combinations that give the minimum *MRSE* and *MAE*. For the GRNN applications, different spreads were tried to find the best one that gave the minimum *MRSE* and *MAE* for the given problem. The optimum spreads were found between 0.02 and 0.08. For the RBF applications, different numbers of hidden layer neurons and spread constants were tried. The numbers of hidden layer neurons that gave the minimum *MRSE* and *MAE* were found to be 8 and 9. The spreads between 0.5 and 0.8 that gave the minimum *MRSE* and *MAE* were also found with simple trial-and-error method.

Prediction and estimation of the downstream and upstream sediment concentrations

Several input combinations (Table 3) were tried to predict and estimate suspended sediment concentration values for the downstream station. In all cases, the output layer had only one neuron, that is, the sediment concentration, Sd_i . For each input combination, the number of nodes in the hidden layer that gave the minimum *MRSE* and *MAE* was determined. The node numbers in the hidden layer were found to vary

Table 3 Values of *MRSE*, *MAE* and determination coefficient, R^2 , for each MLP model in the testing period.

Input combinations	Nodes in hidden layer	<i>MRSE</i> (mg l ⁻¹)	<i>MAE</i> (mg l ⁻¹)	R^2
<i>Downstream station:</i>				
(i) Qd_t	3	10.78	97.3	0.470
(ii) Sd_{t-1}	2	7.50	57.9	0.771
(iii) Qd_t and Sd_{t-1}	5	6.94	57.2	0.791
(iv) Sd_{t-1} and Sd_{t-2}	2	7.31	58.4	0.787
(v) Qd_t , Sd_{t-1} , and Sd_{t-2}	5	6.15	50.3	0.830
(vi) Qd_t , Qd_{t-1} , and Sd_{t-1}	5	6.90	52.7	0.800
(vii) Qd_t , Qd_{t-1} , Sd_{t-1} , and Sd_{t-2}	6	6.37	56.2	0.827
<i>Upstream station:</i>				
(i) Qu_t	2	5.52	47.2	0.551
(ii) Su_{t-1}	2	3.81	25.1	0.812
(iii) Qu_t and Su_{t-1}	2	3.83	25.3	0.812
(iv) Su_{t-1} and Su_{t-2}	4	3.71	21.9	0.803
(v) Qu_t , Su_{t-1} , and Su_{t-2}	5	3.53	23.6	0.829
(vi) Qu_t , Qu_{t-1} , and Su_{t-1}	4	3.53	27.4	0.883
(vii) Qu_t , Qu_{t-1} , Su_{t-1} , and Su_{t-2}	5	2.85	21.8	0.888

between 2 and 6. Table 3 gives the *MRSE*, *MAE* and determination coefficients (R^2) for each combination. It can be seen from Table 3 that the MLP whose inputs are the current discharge and the sediment concentration of two previous periods (combination (v)) has the smallest *MRSE* (6.15 mg l⁻¹) and *MAE* (50.3 mg l⁻¹) and the highest R^2 (0.830). The MLP performance for the first input combination (only current discharge) is the worst due to the hysteresis effect between sediment concentration and discharge. That is to say, the sediment concentrations for a given level of streamflow discharge in the rising stage of a streamflow hydrograph are greater than in the falling stage. This confirms that the practice of using sediment load vs discharge is misleading as stated by McBean & Al-Nassri (1988). The GRNN, RBF and MLR methods were also applied to the fifth input combination and the results compared with those of the MLP model, as shown in Table 4. The MLP method provides better performance in sediment estimation compared to the other methods. The MLR method also gave better test results than GRNN and RBF methods.

Table 4 Values of *MRSE*, *MAE* and determination coefficient, R^2 , for each model in the testing period.

Models	<i>MRSE</i> (mg l ⁻¹)	<i>MAE</i> (mg l ⁻¹)	R^2
<i>Downstream station:</i>			
MLP	6.15	50.3	0.830
GRNN	8.70	69.1	0.662
RBF	7.17	57.1	0.793
MLR	7.08	48.6	0.787
<i>Upstream station:</i>			
MLP	2.85	21.8	0.888
GRNN	4.05	36.1	0.771
RBF	3.63	23.2	0.835
MLR	3.30	22.2	0.857

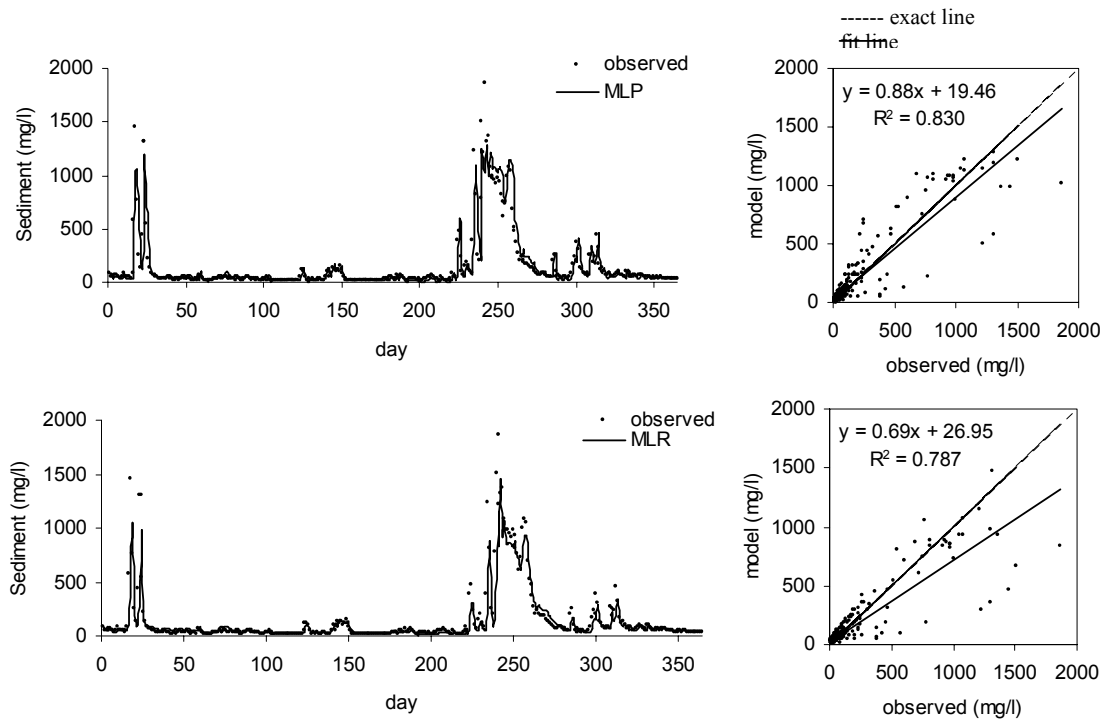


Fig. 3 The observed and estimated suspended sediments of the downstream station in the validation period.

The MLP and MLR estimates for the testing period were compared with the observed suspended sediments in the form of hydrograph and scatterplot (see Fig. 3). The MLP gave R^2 coefficient of 0.830, which was higher than the value of 0.787 obtained using the MLR model. It can be seen from the fit line equations (assume that the equation is $y = a_0x + a_1$) that the a_0 and a_1 coefficients for the MLP are respectively closer to the 1 and 0 than those of the MLR. This confirms the $MRSE$ statistics, which are shown in Table 4. The estimation of total sediment load was also considered for comparison due to its importance in reservoir management. The MLP, GRNN and RBF methods estimated the observed total sediment load of 122 067 t as 130 083, 129 297 and 131 335 t, respectively, with overestimations of 6.6, 5.9 and 7.6%, while the MLR method computed total sediment load as 109 577 t, with an underestimation of 10.2%. The ANN techniques were generally better than the MLR in total sediment load estimation. However, while the RBF method estimated the maximum peak as 1943 mg l^{-1} , compared to the measured 1870 mg l^{-1} , with an overestimation of 3.9%, the MLP, GRNN and MLR methods resulted in 1288, 1208 and 1465 mg l^{-1} , with underestimations of 31, 35 and 22%, respectively. The RBF method gave a much better estimate than the other techniques.

For the upstream station, seven input combinations were also tried (cf. Table 3). Here also the output layer had one neuron, that is Su_t . The $MRSE$, MAE and R^2 values of the MLP models for the testing period are given in Table 3. The number of hidden layer nodes of the MLP that gave the minimum $MRSE$ and MAE were found to vary between 2 and 5. The MLP whose inputs were the current and one previous discharge and the sediment concentration of two previous periods (combination (vii)) gave the smallest $MRSE$ (2.85 mg l^{-1}) and MAE (21.8 mg l^{-1}) and the highest R^2 (0.888). Here

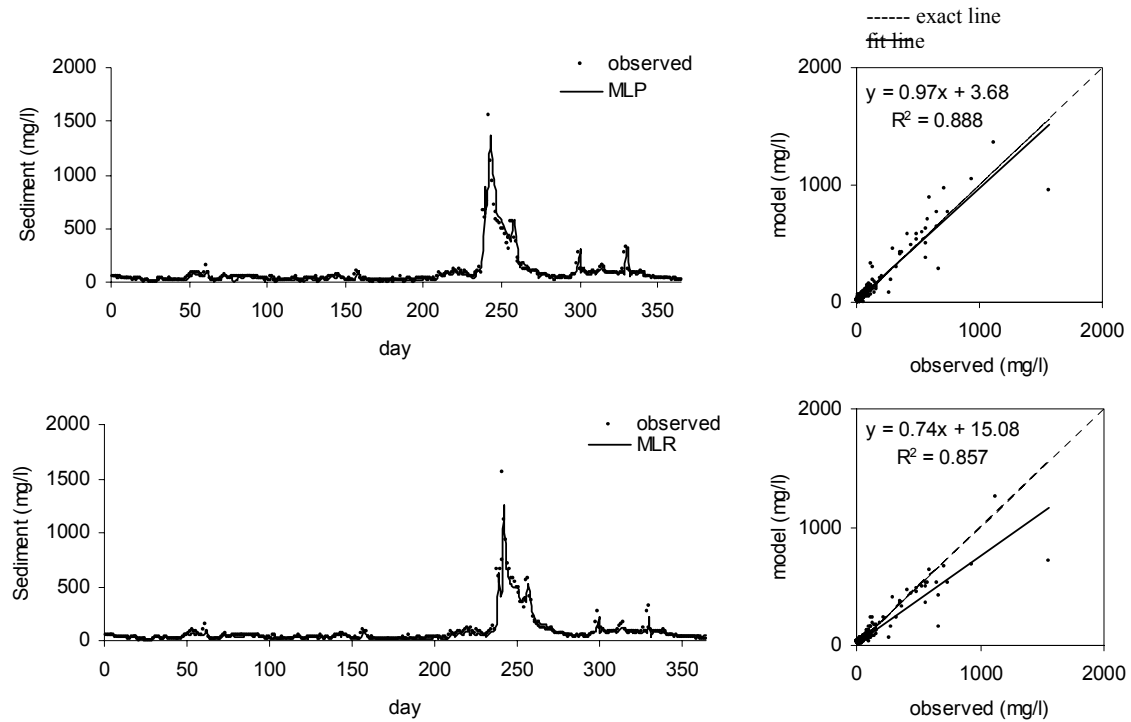


Fig. 4. The observed and estimated suspended sediments of the upstream station in the validation period.

also the first combination gave the worst performance among the input combinations tried. The comparison of MLP, GRNN, RBF and MLR test results for the input combination (vii) is presented in Table 4. Here also the MLP model performs better than the other models. For the upstream station also the MLR model is ranked as the second best. Figure 4 depicts the observed and estimated suspended sediment of the upstream station in the form of hydrograph and scatterplot. It may be seen, particularly from the fit line equations and R^2 values in the scatter diagrams, that the MLP estimates are closer to the corresponding observed values than those of the MLR method. The total sediment load estimates of the MLP, RBF, GRNN and MLR methods are 7.5 and 2.7% higher and 7.9 and 9.2% lower, respectively, than the observed value (73 765 t). Here also the RBF is the best method. The MLP, GRNN and MLR models computed the maximum peak as 1366, 1128 and 1260 mg l^{-1} compared to the observed value of 1560 mg l^{-1} , with underestimations of 12, 27 and 19%, respectively, while the RBF model resulted in 1964 mg l^{-1} , with an overestimation of 26%. The MLP estimate is the closest to the observed one.

Estimation of downstream sediment concentrations using upstream and downstream data

As a last application, the upstream and downstream data were used as inputs to the MLP model to estimate the downstream sediment. The input combinations as in the preceding applications were tried (Table 5). The optimum number of nodes in the hidden layer was found to vary between 2 and 6 (Table 5). The performance criteria of the MLP models are given in Table 5. It can be seen from the table that the MLP

Table 5 Values of *MRSE*, *MAE* and determination coefficient, R^2 , for each MLP model in the testing period—downstream sediment estimation.

Input combinations	Nodes in hidden layer	<i>MRSE</i>	<i>MAE</i>	R^2
(i) Qu_t	3	10.3	97.2	0.515
(ii) Su_t	2	9.50	99.0	0.612
(iii) Qu_t and Su_t	2	9.27	95.5	0.616
(iv) Su_t and Su_{t-1}	4	9.46	104	0.617
(v) Qu_t , Su_t and Su_{t-1}	4	10.0	97.9	0.602
(vi) Qu_t , Qu_{t-1} and Su_t	4	9.06	96.1	0.645
(vii) Qu_t , Qu_{t-1} , Su_t and Su_{t-1}	5	12.2	97.2	0.536
(viii) Qu_t and Qd_t	3	10.9	96.6	0.479
(ix) Qu_t , Su_t and Qd_t	4	8.96	81.2	0.633
(x) Qu_t , Qd_t and Sd_{t-1}	3	6.99	52.4	0.780
(xi) Qu_t , Su_t , Qd_t and Sd_{t-1}	6	7.26	61.3	0.803

Table 6 Values of *MRSE*, *MAE* and determination coefficient, R^2 , for each model in testing period—downstream sediment estimation.

Model	<i>MRSE</i> (mg l^{-1})	<i>MAE</i> (mg l^{-1})	R^2
MLP	6.99	52.4	0.780
GRNN	8.61	56.3	0.662
RBF	7.02	53.5	0.783
MLR	7.67	56.3	0.758

model whose inputs are the current discharge of both stations and the one previous sediment concentration of the downstream station (combination (x)) has the smallest *MRSE* (6.99 mg l^{-1}) and *MAE* (52.4 mg l^{-1}). Here the model performances were not good for the first seven input combinations relative to the other combinations. This shows the difficulties in estimating downstream sediment data using only upstream data. The MLP test results were compared to those of the GRNN, RBF and MLR models for input combination (x) (see Table 6). The MLP and RBF models have better performance criteria than the GRNN and MLR models. The MLP method seems to be slightly better than the RBF model from the *MRSE* and *MAE* viewpoints. The estimates for the testing period were compared with the observed suspended sediment values in the form of hydrograph and scatterplot (Fig. 5). As can be seen from the figure, the MLP and RBF models approximated the observed hydrograph better than the MLR model. The MLP and RBF models gave R^2 coefficients of 0.780 and 0.783, which was higher than the value of 0.758 obtained using the MLR model. In contrast to the *MRSE* and *MAE* criteria viewpoints (Table 6), the RBF model seems to be better than the MLP model with respect to the fit line equation and R^2 coefficient in Fig. 5. The a_0 and a_1 coefficients for the RBF model are respectively closer to the 1 and 0 than those of the MLP model. The total sediment amount estimates of the MLP, GRNN, RBF and MLR models are 1.7, 5 and 7.4% higher and 13.9% lower than the observed value (122 066 t), respectively. The MLP model gave the closest estimate of the total sediment load. The other ANN techniques also gave better results than the MLR model. The MLP, GRNN, RBF and MLR models estimated the maximum peak of 1870 mg l^{-1} as 1490, 1400, 1646 and 1327 mg l^{-1} , with underestimations of 20, 25, 11 and 29%, respectively. Here also the RBF model gave the closest estimate of the

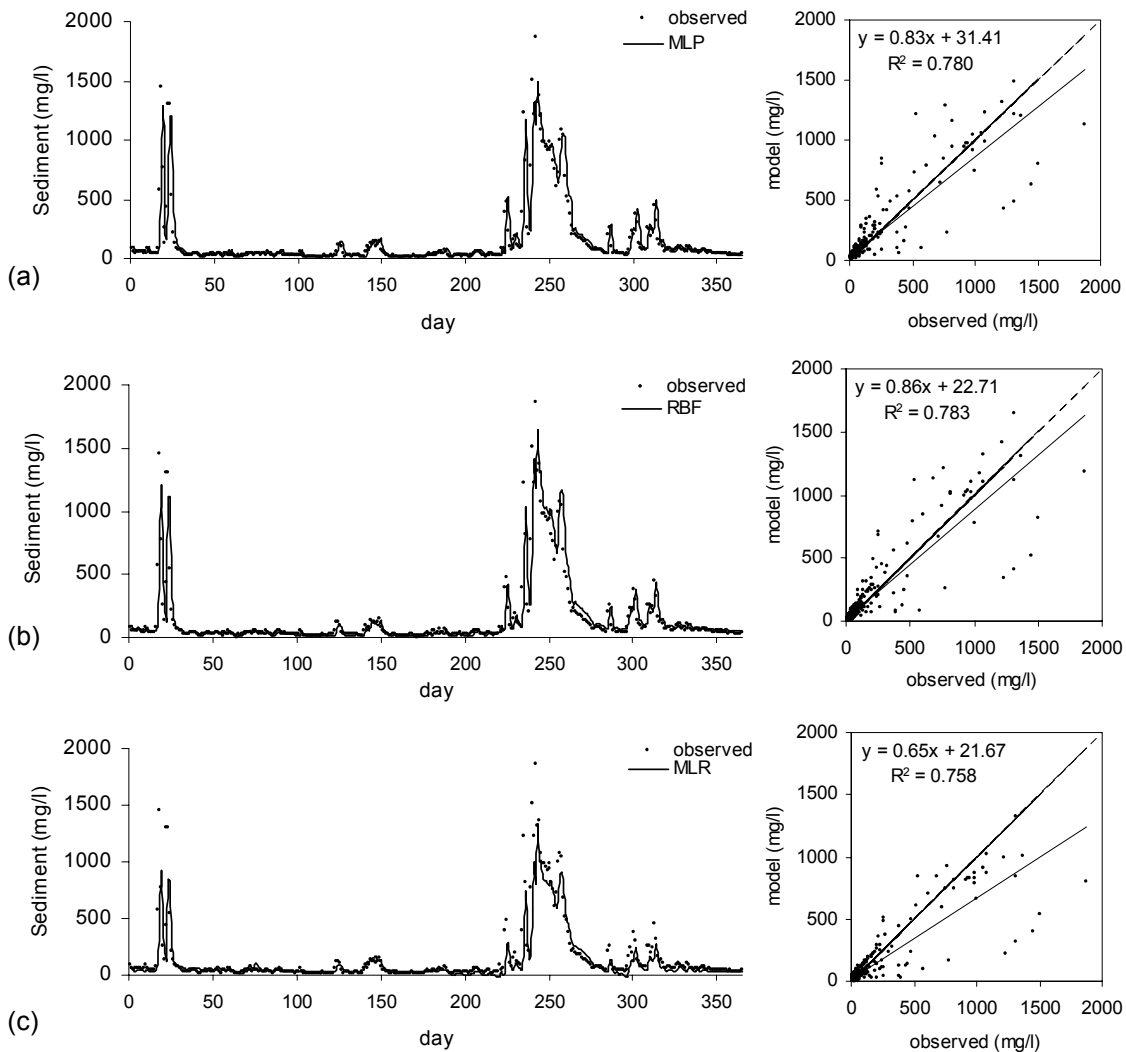


Fig. 5. Plotting of estimation performances for the validation period using (a) MLP and (b) RBF and (c) MLR—downstream sediment estimation.

maximum, as found in the preceding application. The other ANN techniques also gave better maximum estimates than the MLR method.

CONCLUDING REMARKS

The main conclusions of the study can be summarized as follows:

- This study demonstrated that modelling of the daily suspended sediment concentration is possible through the use of the MLP method.
- The MLP method was generally found to perform better than the RBF, GRNN and the conventional statistical method (MLR) in estimation of suspended sediment concentration.
- The GRNN method provided the worst suspended sediment concentration estimates of all the applications.
- The applications indicated that the RBF and GRNN models may yield better sediment load estimates than the MLP model.

- The RBF method was mostly found to be better than the other techniques in approximation of maximum suspended sediment concentration. The MLP method ranked as the second best. The first application indicated that the MLR method may provide better maximum estimates than the MLP and RBF models.
- The difficulties in the estimation of suspended sediment concentration using only current discharge, resulting from the hysteresis effect, were indicated.
- The last application showed that the estimation of sediment data from one station using only streamflow and sediment data from the other station was very difficult.

The study used data from only two areas and further studies using more data from various areas may be required to strengthen these conclusions.

In the present study, the MLP model was compared with the GRNN, RBF and MLR models using only the LM training algorithm. If the other MLP training algorithms (gradient descent, conjugate gradient, etc.) were used, the results from the GRNN, RBF and MLR models may turn out to be better than those from the MLP model. This may be a subject of another study.

REFERENCES

- Broomhead, D. & Lowe, D. (1988) Multivariable functional interpolation and adaptive networks. *Complex Syst.* **2**, 321–355.
- Chang, F.-J. & Chen, Y.-C. (2001) A counterpropagation fuzzy-neural network modeling approach to real time streamflow prediction. *J. Hydrol.* **245**, 153–164.
- Chen, S., Cowan, C. F. N. & Grant, P. M. (1991) Orthogonal least squares learning algorithm for radial basis function networks. *IEEE Trans. Neural Networks* **2**(2), 302–309.
- Chibanga, R., Berlamont, J. & Vandewalle, J. (2003) Modelling and forecasting of hydrological variables using artificial neural networks: the Kafue River sub-basin. *Hydrol. Sci. J.* **48**(3), 363–379.
- Cigizoglu, H. K. (2003) Estimation, forecasting and extrapolation of river flows by artificial neural networks. *Hydrol. Sci. J.* **48**(3), 349–361.
- Cigizoglu, H. K. & Kişi, O. (2004) Flow prediction by three back propagation techniques using k-fold partitioning of neural network training data. *Nordic Hydrol.* **36**(1). (in press)
- Clair, T. A. & Ehrman, J. M. (1998) Using neural networks to assess the influence of changing seasonal climates in modifying discharge, dissolved organic carbon, and nitrogen export in eastern Canadian rivers. *Water Resour. Res.* **34**(3), 447–455.
- Coulibaly, P., Anctil, F. & Bobée, B. (1999) Prévision hydrologique par réseaux de neurones artificiels: état de l'art. *Can. J. Civil Engng* **26**(3), 293–304.
- Cybenco, G. (1989) Approximation by superposition of a sigmoidal function. *Mathematics of control. Signals and Systems* **2**, 303–314.
- El-Bakyr, M. Y. (2003) Feed forward neural networks modeling for K–P interactions. *Chaos, Solutions & Fractals* **18**(5), 995–1000.
- Guldal, V. & Muftuoglu, R. F. (2001) 2D unit sediment graph theory. *J. Hydrol. Engng* **6**(2), 132–140.
- Hagan M. T. & Menhaj M. B. (1994) Training feed forward networks with the Marquardt algorithm. *IEEE Trans. Neural Networks* **6**, 861–867.
- Haykin, S. (1994) *Neural Networks: A Comprehensive Foundation*. IEEE Press, New York, NY, USA.
- Hornik, K., Stinchcombe, M. & White, H. (1989) Multilayer feedforward networks are universal approximators. *Neural Networks* **2**, 359–366.
- Jain, S. K. (2001) Development of integrated sediment rating curves using ANNs. *J. Hydraul. Engng ASCE* **127**(1), 30–37.
- Jain, S. K., Das, D. & Srivastava, D. K. (1999) Application of ANN for reservoir inflow prediction and operation. *J. Water Resour. Plan. Manage. ASCE* **125**(5), 263–271.
- Kang, K. W., Park, C. Y. & Kim, J. H. (1993) Neural network and its application to rainfall–runoff forecasting. *Korean J. Hydrosoci.* **4**, 1–9.
- Kim, B., Kim, S. & Kim, K. (2003) Modelling of plasma etching using a generalized regression neural network. *Vacuum*, **71**, 497–503.
- Kisi, O. (2002) River sediment yield modeling using artificial neural networks. *ASCE's First Virtual World Congress for Civil Engineering*, (<http://www.ceworld.org/ceworld/Presentations/WaterandEnvironment/Kisi.cfm>).
- Kisi, O. (2004a) River flow modeling using artificial neural networks. *J. Hydrol. Engng ASCE* **9**(1), 60–63.
- Kisi, O. (2004b) Daily suspended sediment modeling using fuzzy-differential evolution approach. *Hydrol. Sci. J.* **49**(1), 183–197.
- Lee, G. C. & Chang, S. H. (2003) Radial basis function networks applied to DNBR calculation in digital core protection systems. *Annals of Nuclear Energy* **30**, 1561–1572.

- Maier, H. R. & Dandy, G. C. (2000) Neural networks for prediction and forecasting of water resources variables: review of modelling issues and applications. *Environ. Modelling and Software* **15**, 101–124.
- Marquardt, D. (1963) An algorithm for least squares estimation of non-linear parameters. *J. Soc. Ind. Appl. Math.* 431–441.
- McBean, E. A. & Al-Nassri, S. (1988) Uncertainty in suspended sediment transport curves. *J. Hydrol. Engng ASCE* **114**(1), 63–74.
- Poggio, T. & Girosi, F. (1990) Regularization algorithms for learning that are equivalent to multilayer networks. *Science* **2247**, 978–982.
- Refsgaard, J. C. (1997) Parameterisation, calibration and validation of distributed hydrological models. *J. Hydrol.* **198**, 69–97.
- Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986) Learning internal representation by error propagation. In: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition* (ed. by D. E. Rumelhart & J. L. McClelland), vol. 1, 318–362. MIT Press, Cambridge, Massachusetts, USA.
- Saad, M., Bigras, P., Turgeon, A. & Duquette, R. (1996) Fuzzy learning decomposition for the scheduling of hydroelectric power systems. *Water Resour. Res.* **32**(1), 179–186.
- Shamseldin, A. Y. (1997) Application of a neural network technique to rainfall–runoff modelling. *J. Hydrol.* **199**, 272–294.
- Sivakumar, B., Jayawardena, A. W. & Fernando, T. M. K. G. (2002) River flow forecasting: use of phase space reconstruction and artificial neural networks approaches. *J. Hydrol.* **265**, 225–245.
- Solomatine, D. P. & Dulal, K. N. (2003) Model trees as an alternative to neural networks in rainfall–runoff modelling. *Hydrol. Sci. J.* **48**(3), 399–411.
- Specht, D. F. (1991) A general regression neural network. *IEEE Trans. Neural Networks* **2**(6), 568–576.
- Taurino, A. M., Distante, C., Siciliano, P. & Vasanelli, L. (2003) Quantitative and qualitative analysis of VOCs mixtures by means of a microsensor array and different evaluation methods. *Sensors and Actuators* **93**, 117–125.
- Tayfur, G. (2002) Artificial neural networks for sheet sediment transport. *Hydrol. Sci. J.* **47**(6), 879–892.
- Tayfur, G., Ozdemir, S. & Singh, V. P. (2002) Fuzzy logic algorithm for runoff-induced sediment transport from bare soil surfaces. *Adv. Water Resour.* **26**, 1249–1256.
- Tokar, A. S. & Johnson, P. A. (1999) Rainfall–runoff modeling using artificial neural networks. *J. Hydrol. Engng ASCE* **4**(3), 232–239.
- Tsoukalas, L. H. & Uhrig, R. E. (1997) *Fuzzy and Neural Approaches in Engineering*. John Wiley Inc., New York, USA.
- Wasserman, P. D. (1993) *Advanced Methods in Neural Computing*, 155–161. Van Nostrand Reinhold, New York, USA.
- Wicks, J. M. & Bathurst, J. C. (1996) SHESED: a physically based, distributed erosion and sediment yield component for the SHE hydrological modeling system. *J. Hydrol.* **175**, 213–238.
- Wilby, R. L., Abrahart, R. J. & Dawson, C. W. (2003) Detection of conceptual model rainfall–runoff processes inside an artificial neural network. *Hydrol. Sci. J.* **48**(2), 163–181.
- Zhu, M. L. & Fujita, M. (1994) Comparisons between fuzzy reasoning and neural network methods to forecast runoff discharge. *J. Hydrosci. Hydraul. Engng* **12**(2), 131–141.

Received 17 November 2003; accepted 23 August 2004