Journal of Applied Sciences

ISSN 1812-5654

Journal of Applied Sciences 7 (22): 3451-3459, 2007
ISSN 1812-5654

Application of Fuzzy Systems and Artificial Neural Networks for Flood Forecasting

R. Tareghian and S.M. Kashefipour
Faculty of Water Sciences Engineering, Shahid Chamran University of Ahwaz, Iran

Abstract: This study presents the development of Artificial Neural Networks (ANNs) and Fuzzy Logic (FL) models for prediction of daily reservoir inflow. Furthermore, a Linear Regression (LR) model was also developed as a traditional method for flood forecasting. To illustrate the applicability and capability of the ANNs and FL models, the Dez reservoir, located in the south-west of Iran, was used as a case study. The results demonstrated that ANNs model can predict the reservoir inflow for 1-day-ahead, especially for training pattern better than the FL and LR models. It was found that the accuracy of ANNs model predictions decreased for flood forecasting more than 1-day ahead (e.g., 2, 3, or 4 days ahead), whereas the results obtained from the FL and LR models showed better correlation with the corresponding measured values in this conditions. One of the main findings of this research was that the fuzzy logic model generally underestimated the flood even for, whereas the other two considered models predicted the flood discharge relatively good. The peak value of the hydrograph, which is very important from the flood hazard viewpoint, was estimated good by the ANNs and LR models for the short period (1-day ahead), with the error being 3, 4.5 and 26% for the ANNs, LR and FL models, respectively. For the long periods (e.g., 3-days ahead) the flood discharge was predicted by the LR and FL models slightly better than the ANNs model.

Key words: Flood forecasting, reservoir inflow, fuzzy logic, artificial neural networks, dez reservoir, flood hazard

INTRODUCTION

Flood forecasting is one of the most important tasks of the reservoirs management systems. The magnitude of economic losses associated with floods highlights the importance of flood management. An efficient flood alarm system may significantly improve public safety and mitigate economical damages caused by inundations. Flood forecasting is undoubtedly a challenging field of operational hydrology and a huge literature has been developed in years (Xiong *et al*., 2001; Gopakumar and James, 2002; Chau *et al*., 2005; Tayfur and Singh, 2006); in particular, the rainfall-runoff relationship has been recognized to be nonlinear. Although conceptual models allow a deep understanding of the hydrological processes, their calibration requires to collect a great amount of information regarding the physical properties of the watershed under study (for example, characteristics of terrain and river networks, rainfall and runoff), which may be expensive and very time consuming. Sophisticated physical models may not be ideal for real-time forecasting due to the huge data requirement and the associated long computation time for model calibration. Since flood warning systems do not aim at providing an explicit knowledge of the rainfall-runoff process and the main concern is making accurate and timely predictions at appropriate locations, a simple black-box model is then preferred for identifying a direct mapping between inputs and outputs (Corani and Guariso, 2004). Furthermore, the inherently nonlinear relationships between input and output variables complicate attempts to forecast stream flow events. There is thus a need for improvement in forecasting techniques.

In recent years, many nonlinear approaches, such as the artificial neural networks, fuzzy logic and genetic algorithm approaches have been used in solving flood forecasting problems. Over the last decades, Artificial Neural Networks (ANNs) have been increasingly used in hydrological forecasting (Maier and Dandy, 2000); furthermore, their computational speed in simulating and forecasting is very welcomed in real time operations. Dawson and Wilby (1998) discussed the application of ANNs to flow forecasting in two flood-prone catchments in England using hourly hydrometric data. Liong *et al*.

Corresponding Author: S.M. Kashefipour, Faculty of Water Sciences Engineering, Shahid Chamran University of Ahwaz, Ahwaz, Iran Tel: 0098 611 3364055 Fax: 0098 611 3365670

(2000) achieved to get a high degree of accuracy with ANNs for river stage forecasting in Bangladesh. Ni and Xue (2003) established an ANN model based on Radial-Basis-Function (RBF) for flood risk ranking at five safety polders in Yangtze River, China. Bhattacharya and Solomatine (2005) built water level-discharge relationship models with an ANN and a M5 model tree on the river Bhagirathi in India. Based on casual meteorological parameters, Ahmad and Simonovic (2005) applied ANNs for predicting the peak flow, timing and shape of runoff hydrograph for the Red River in Manitoba, Canada. Chau (2006) adopted a particle swarm optimization model to train an ANN model to predict water levels in Shing Mun River of Hong Kong with different lead times on the basis of upstream gauging stations. Tareghian and Kashefipour (2006) developed an ANN model to forecast daily runoff for Karoon River in Iran. Dawson *et al*. (2006) used ANNs to predict T-year flood events and the index flood for 850 catchments across the UK. Wu and Chau (2006) employed a genetic algorithm based Artificial Neural Network (ANN-GA) for flood forecasting in a channel reach of the Yangtze River in China. Pang *et al*. (2007) developed a Nonlinear Perturbation Model (NLPM) based on ANN, defined as NLPM-ANN, for the purpose of improving the rainfall-runoff forecasting efficiency and accuracy. Chang *et al*. (2007) presented a systematic investigation of the three common types of ANNs for Multi-Step-Ahead (MSA) flood forecasting for two watersheds in Taiwan.

Since Zadeh (1965) publication regarding an extension of the classical fuzzy set theory, the fuzzy method has been widely used in many fields of applications, such as pattern recognition, data analysis, system control, etc. (Kruse *et al*., 1994; Klir *et al*., 1997; Theodoridis and Koutroumbas, 1999). Hundecha *et al*. (2001) demonstrated that a Fuzzy Logic (FL) approach could be used to simulate actual component hydrologic processes in areas where sufficient data were available to model these processes physically. Ozelkan and Duckstein (2001) proposed a fuzzy conceptual rainfall-runoff framework to deal with parameter uncertainties of conceptual rainfall-runoff models. Cheng *et al*. (2002) combined a fuzzy optimal model with a genetic algorithm to solve multi-objective rainfall-runoff Xinanjiang model calibration in the Shuangpai reservoir. Luchetta and Manetti (2003) used a fuzzy clustering approach to forecast a real time hydrological model in the Padule di Fucecchio basin in middle-north of Italy. Mahabir *et al*. (2003) applied FL to forecast seasonal runoff for the Lodge and Middle Creek basins, Canada. Blazkova and Beven (2004) used FL to estimate flood frequency by continuous simulation of sub-catchments rainfalls and discharges for a dam site in a large catchment in the Czech Republic. Vernieuwe *et al*. (2005) described the catchment's response to rainfall input through fuzzy relationships for Zwalm River in Belgium. Rao and Srinivas (2006) tested a fuzzy clustering for regionalization of watersheds with 245 gauging stations data in Indiana.

The main objective focused in this paper is to model the daily reservoir discharge inflow (e.g., flood events) using the Computation Intelligence Tools (i.e., fuzzy logic and artificial neural networks) with the results obtained from these sophisticated methods being compared with the traditional linear regression model and the corresponding measured values. From where, the suitable methods for predicting the flood peak for the short (e.g., 1-day ahead) and long (e.g., 4-days ahead) periods would be introduced.

Artificial Neural Network (ANN): ANNs have gained popularity in a large array of engineering applications where conventional analytical methods show inferior performance. ANNs have shown a good potential to efficiently model complex input-output relationships where the presence of nonlinearity and inconsistent noisy data adversely affects other approaches (Deka and Chandramouli, 2005).

ANNs have an ability to capture a relationship from given patterns and hence this makes them suitable for employment in the solution of large-scale complex problems, such as pattern recognition, nonlinear modeling, classification, association and control. In applications, a three layer-feed forward type of artificial neural network is commonly considered. In a feed forward ANN, the input quantities (x_i) are fed into the input layer neurons that, in turn, pass them on to the hidden layer neurons (z_i) after multiplication by connection weights (v_{ij}) (Fig. 1). A hidden layer neuron adds up the weighted input received from each input neuron ($x_i v_{ij}$) and associates it with a bias (b_j) (i.e., $net_j = \sum x_i v_{ij} + b_j$). The result ($net_j$) is then passed on through a nonlinear transfer function (activation function) to produce an output i.e., sigmoid function; $f(net_j) = [1/(1+e^{-net_j})]$. The output neurons do the same operation as does a hidden neuron. The back propagation algorithm finds the optimal weights by minimizing a predetermined error function (E) of the following form (ASCE Task Committee, 2000):

$$E = \sum_{P}\sum_{n}\left(y_i - t_i\right)^2 \qquad (1)$$

Where:

y_i = Component of a network output vector
Y; t_i = Component of a target output vector
T; n = No. of output neurons
P = No. of training patterns

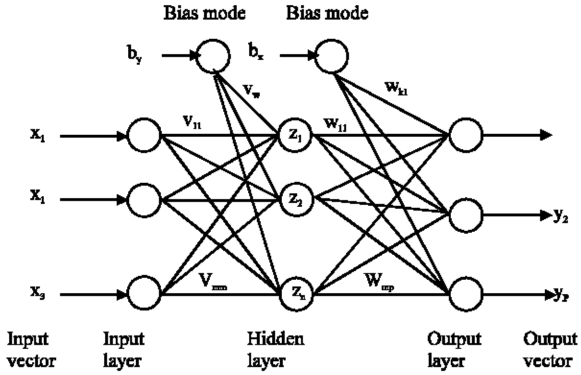

Fig. 1: Schematic representation of ANN architecture

In the back propagation algorithm, the optimal weights would generate an output vector $Y = (y_1, y_2,..., y_n)$ as close as possible to the target values of the output vector $T = (t_1, t_2,..., t_n)$ with a pre-selected accuracy. The back propagation algorithm employs the gradient-descent method, along with the chain rule of differentiation, to modify the network weights as (ASCE Task Committee 2000):

$$v_{ij}^{new} = v_{ij}^{old} - \delta \frac{\partial E}{\partial v_{ij}} \tag{2}$$

Where:

v_{ij} = Weight from ith neuron in the previous layer to the jth neuron in the current layer

δ = Learning rate

The network learns by adjusting the biases and weights that link its neurons.

Before training begins, a network's weights and biases are set equal to small random values. Also, due to the nature of the sigmoid function used in the back propagation algorithm, all external input and output values before passing them into a neural network are normalized. Without standardization and normalization, large values input into an ANN would require extremely small weighting factors to be applied and this could cause a number of problems (Dawson and Wilby, 1998).

Artificial neural network contains three distinctive modes: Training, cross validation and testing. In the training mode, the training datasets consisting of input-output patterns are presented to the network. The weights are found through an iterative process, in which the back propagation learning algorithm is used to find the weights such that the difference between the given outputs and the outputs computed by the network is sufficiently small. While training, it is a usual practice that the training

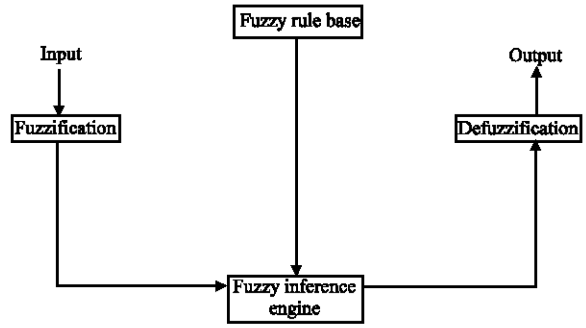

Fig. 2: Schematic representation of fuzzy system

datasets are further subdivided into two sets, training and cross validation sets, according to data availability. During the training patterns, Mean Square Error (MSE) of training and cross validation datasets are monitored together to find the optimal termination point for training. This check avoids overtraining. After training, the network is tested with the testing dataset to determine how accurately the network can simulate the input-output relationship.

Fuzzy Logic (FL): A general fuzzy system has basically four components, fuzzification, fuzzy rule base, fuzzy output engine and defuzzification (Fig. 2). Fuzzification converts each piece of input data to degrees of membership by a look-up in one or more several membership functions. The key idea in fuzzy logic, in fact, is the allowance of partial belongings of any object to different subsets of the universal set instead of belonging to a single set completely.

Partial belonging to a set can be numerically described by a membership function, which takes on values between 0 and 1 inclusive. This intuitive approach is used rather commonly because it is simple and derived from the innate intelligence and understanding of human beings. Fuzzy membership functions may take on many forms, like triangular, trapezoidal, Gaussian and generalized bell membership functions.

The fuzzy rule base contains rules that include all possible fuzzy relations between inputs and outputs. These rules are expressed in the IF-THEN format. In the fuzzy approach, there are no mathematical equations and model parameters. All the uncertainties, nonlinear relationships, or model complications are included in the descriptive fuzzy inference procedure in the form of IF-THEN statements. There are basically two types of rule systems, namely, Mamdani and Sugeno (Jantzen, 1999). Depending upon a problem under consideration, a user can choose the appropriate rule system. According to the Sugeno rule system, the consequent part of the fuzzy rule is expressed as a mathematical function of the input variable and such a system is more appropriate for neuro-fuzzy systems (Sen, 1998; Jantzen, 1999). In the Mamdani rule system, however, the consequent part of the fuzzy rule is also expressed as verbally.

The fuzzy inference engine takes into account all the fuzzy rules in the fuzzy rule base and learns how to transform a set of inputs to the corresponding outputs. To do so, it uses either min or prod activation operators. The activation of a rule is the deduction of the conclusion, possibly reduced by its firing strength. The prod activation (multiplication) scales the membership curves, thus preserving the initial shape, rather than clipping them as the min activation does.

Jantzen (1999) pointed out that both methods, in general, work well. In order to have a good understanding of the methodology, let consider a simple case presented in Fig. 3, where there are two input variables X and Y (Fig. 3a, b) and one output variable Z (Fig. 3c). For this simple system the following fuzzy rules are assumed:

If X is low and Y is low then z is high; if X is high and Y is high then Z is low

As can be seen from Fig. 3a, X = 20 is a part of low and high subsets with the different degrees of membership to be 0.8 and 0.2, respectively. Similarly, Y = 30 is a part of low and high subsets with 0.4 and 0.6 degrees of membership, respectively (Fig. 3b). The fuzzy inference engine will consider the previous rules and by min activation find the fuzzy output set of high from the first rule with 0.4 firing strength (this value would be 0.32 by prod activation) and output set of low from the second rule with 0.2 firing strength (this value would be 0.12 by prod activation) (Fig. 3c). It should be noted that inference produces not a crisp output value but assigns whole fuzzy output subsets to the output variable (Fig. 3c). The next sub-process in inference is the composition sub-process where all of the fuzzy subsets assigned to the output variable are combined together to form a single subset for the output variable. For this purpose, there are basically two composition methods including: Maximization (max) and summation (sum). In max composition, the combined output fuzzy subset is constructed by taking the pointwise maximum over all of the fuzzy subsets assigned to the output variable by the inference rule. In sum composition, the combined output fuzzy subset is constructed by taking the pointwise sum over all of the fuzzy subsets. Consequently, in sum composition it is sometimes possible to obtain truth values greater than one. Note that the sum composition must be followed by the Center of Gravity (COG) defuzzification method (Jantzen, 1999). Figure 4a and b present combined fuzzy output subsets derived by the max and sum compositions for the previous example, respectively.

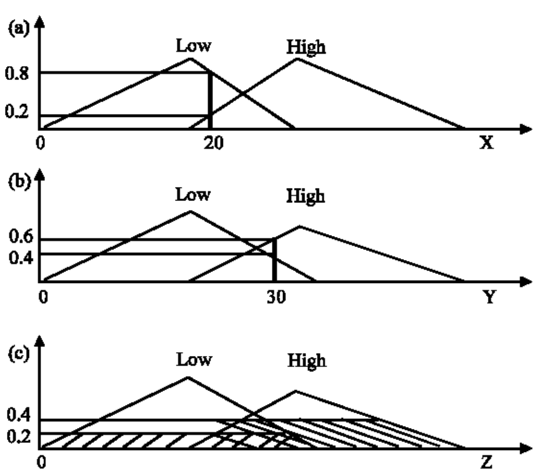

Fig. 3: Schematic representation of fuzzy inferencing: (a) X = 20, (b) Y = 30 and (c) fuzzy output sets for Z

Defuzzification is a process by which a solution set is converted into a single crisp value. The fuzzy logic solution set is in the form of a function, relating the value of the result to the degree of membership. The entire range of possible solutions may be contained in the fuzzy solution set. Defuzzification is also a process to extract an easily comprehensible answer from the set. The Center Of Gravity (COG), Bisector Of Area (BOA), the smallest, median and largest maxima methods are some of commonly used defuzzification methods.

Study area: Data used in this study were taken from the Dez watershed, southwest of Iran. This watershed covers about 21720 km^2, which lies between longitudes 48°10 and 50°21 east and latitudes 31°34 and 34°7 north. At the end of the mountainous part of this watershed the Dez dam is located as one of the main reservoir dams in Iran (Fig. 5).

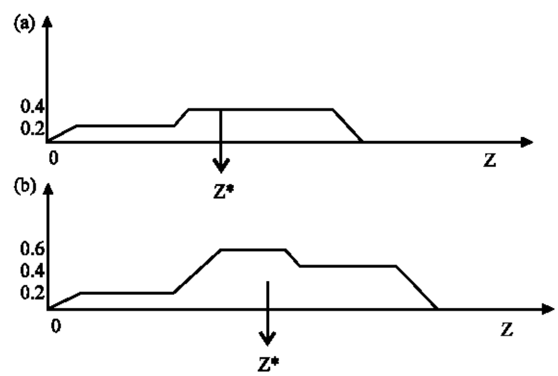

Fig. 4: Schematic representation of (a) max composition and BOA defuzzification and (b) sum composition and COG defuzzification

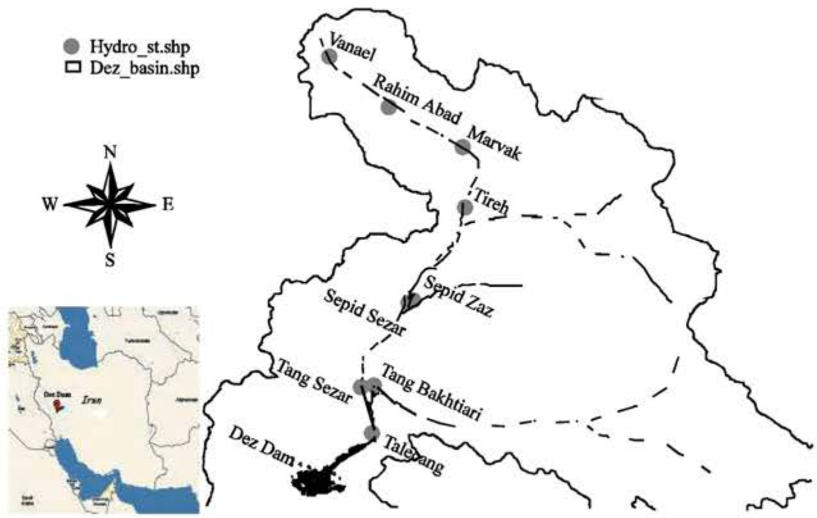

Fig. 5: Dez dam and watershed and selected stations

The main variable parameters affecting the discharge entering to a reservoir dam from a watershed and suitable to use for the statistical models are usually discharges from the tributaries of river, precipitation and evaporation. Precipitation and evaporation data collected from the synoptic stations and discharge data from the hydrometric stations located at the upstream of dam were used to forecast the Dez dam reservoir inflow. For input parameters, four rain gauges, two evaporation gauges and eight discharge gauging stations were selected from a lot of stations, which are distributed in the Dez watershed. The Talezang gauging station data was intended as the output data and the Dez dam reservoir inflow indication (Fig. 5). The original data consists of 6 years (1993-1999), which 1363 input-output pairs were intended for training set and 313 and 314 dataset were used for validation and test set, respectively.

Evaluation criteria for model performance: The performance of the predictions resulting from training, validation and testing is evaluated by the following measures for goodness-of-fit: RMSE (root mean square error) and CC (coefficient of correlation):

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}[(Q_m)_i - (Q_p)_i]}{n}} \quad (3)$$

$$\mathrm{CC} = \frac{\sum_{i=1}^{n}[(Q_m)_i - (\overline{Q_m})_i][(Q_p)_i(\overline{Q_p})_i]}{\sqrt{\sum_{i=1}^{n}[(Q_m)_i - (\overline{Q_m})_i]^2[(Q_p)_i(\overline{Q_p})_i]^2}} \quad (4)$$

$$Q_p = aQ_m \quad (5)$$

Where, subscripts m and p = measured and predicted discharge, respectively; n = total number of data pairs considered; $\overline{Q_m}$ and $\overline{Q_p}$ = mean value of the measured and predicted data, respectively, a = a coefficient which relates the predicted and measured values and for better model performance this coefficient should be close to unity. RMSE furnishes a quantitative indication of the model error in units of the variable, with the characteristic that larger errors receive greater attention than smaller ones. The qualitative evaluation of the model performance is made in terms of the coefficient of correlation between the measured and simulated data.

Another important criterion of the model performance for flood forecasting is to predict the peak flood as accurate as possible. These evaluation criteria were applied to compare the considered models.

Models development and evaluation: The developed ANN and FL models were calibrated and tested for prediction of 1 to 4-days ahead reservoir inflow. Traditionally, linear black box models of ARMAX are frequently used in time series modeling. In discrete time, these models have a form where the model's response at time t depends linearly on the data points at preceding time (t-1, t-2, etc.). In this study, the following model similar in concept to time series model structure is assumed and applied for flood forecasting based on past daily rainfall, evaporation and stream flow data. The following equation is assumed as the main and general function for the dam discharge inflow:

$$Q_{Tal}(t+i) = f(E_1(t), E_2(t), R_1(t), ..., R_4(t), Q_1(t), ..., Q_8(t)) \quad (6)$$

where, Q_{Tal} (t+i) is the discharge in Talezang station (immediate station at the upstream of Dez dam) at time t+i, which i is 1,2,3,4; R(t) is daily rainfall at the synoptic stations 1 to 4 at time t. E(t) is daily evaporation at the synoptic stations 1 and 2 at time t; Q(t) is stream flow data at the hydrometric stations 1 to 8 at time t.

RESULTS AND DISCUSSION

FL model calibration and validation: In order to apply fuzzy logic to predict reservoir inflow, a sensitivity analysis was performed for the fuzzy logic operator AND and for the methods of implication, aggregation and defuzzification. The results of changing a single operator or method while the rest of the model was held constant were compared with the results from the baseline model. Based on this sensitivity analysis, the AND operator product and the implication method minimum were found to perform better. The model was found to be relatively less sensitive to the method of aggregation, in that operators, maximum and summation demonstrated the same result. The model results were most sensitive to the method of defuzzification. The largest of maxima method produced better results than the centroid, bisector and the median of maxima methods. Based on this, a prototype model configuration was developed: using product for the AND operator; minimum for the implication method; maximum for the aggregation method and the largest of maxima method for the defuzzification method. Furthermore, 3, 4, 6 and 9 linguistic terms were applied for each input, which fuzzy logic model with 4 linguistic terms, low, median, high and very high, showed the best sequel. The input-output variables were then fuzzified using different types of fuzzy membership functions. The triangular, trapezoidal, a simple Gaussian, two-sided Gaussian, generalized bell, Sigmoidal, pi curves membership functions were applied in this study and a simple Gaussian membership function showed the best results which can be expressed as:

$$f(x,\sigma,c) = e^{\frac{-(x-c)^2}{2\sigma^2}} \quad (7)$$

c and σ parameters must be determined during the calibration period. In order to define the fuzzy rules, the two well-known fuzzy inference systems, Mamdani and Takagi-Sugeno were also used. The constructed FL model predicted the reservoir inflow at 1 day ahead with 93.2 and 65.2% correlations with the measured data using Mamdani and Takagi-Sugeno fuzzy inferences, respectively, which shows the superiority of Mamdani over the Takagi-Sugeno fuzzy inference. As it can be seen in Table 1, fuzzy logic models can predict the reservoir inflow at 1 to 4-day-ahead with more than 87% correlations for validation period. According to Eq. 5 the FL model generally underestimated the flood discharges.

ANN model training and testing: For training of the ANN by the generalized data rule, dataset comprising 1363 inputs and the corresponding desired output were constructed. Two separate datasets including 313 and 314 datasets were used for model validating and testing. The ASCE Task Committee (2000) reported that ANNs are not very capable at extrapolation. Thus, in the present study, care was taken to have the training data include the highest as well as the lowest values, i.e., the two extreme input patterns.

The Alyuda Neurointeligence software was deployed and ideal network architecture was found through a number of trial and errors. In order to train the network, Quick Back propagation, Conjugate gradient descent, Quasi-Newton, limited memory Quasi-Newton, Levenberg-Marquardt, batch propagation, Incremental Back propagation were applied and the Quick Back propagation demonstrated the best results. Training pattern was completed with a 0.7 learning rate and 15000 iterations. Furthermore, three activation functions-linear, sigmoid and hyperbolic tangent were used and the sigmoid activation function, which can be expressed as the following equation revealed the best results:

$$f(x) = \frac{1}{(1+e^{-x})} \quad (8)$$

As it can be seen in Table 2, training of the ANN model was successfully accomplished with e.g., a CC = 0.97 and RMSE = 2.45 $m^3\ sec^{-1}$ for prediction of 1-

Table 1: FL model results for 1 to 4-day-ahead reservoir inflow

		Calibration		Validation	
Time	a (Eq. 5)	CC	RMSE	CC	RMSE
T+1	0.731	0.932	2.92	0.930	5.93
T+2	0.725	0.879	5.75	0.873	6.56
T+3	0.761	0.841	5.82	0.888	5.52
T+4	0.751	0.812	5.88	0.871	6.54

Table 2: ANN model results for 1 to 4-day-ahead reservoir inflow

		Train		Validation		Test	
Time	a (Eq. 5)	CC	RMSE	CC	RMSE	CC	RMSE
T+1	0.958	0.970	2.45	0.930	4.85	0.939	4.13
T+2	0.902	0.945	3.25	0.863	6.96	0.851	6.27
T+3	0.994	0.881	3.65	0.848	6.72	0.873	5.51
T+4	1.010	0.883	3.65	0.825	6.33	0.743	9.31

Table 3: LR model results for 1 to 4-day-ahead reservoir inflow

		Calibration		Validation	
Time	a (Eq. 5)	CC	RMSE	CC	RMSE
T+1	0.982	0.902	3.65	0.897	4.73
T+2	0.972	0.777	6.25	0.872	6.05
T+3	1.010	0.711	5.20	0.873	5.59
T+4	1.040	0.628	7.82	0.824	4.94

day-ahead reservoir inflow. Table 2 also shows that for the long period prediction (e.g., 3 and 4-days ahead) the accuracy of the model was slightly declined. The values of a coefficient in Table 2 show that the relationship line between the predicted and measured values has almost coincided with the line of (Y = X).

Linear regression model calibrating and validating: Because the basic characteristics of the Dez watershed have remained unaltered in years, there exists a certain correlation between the upstream and downstream conditions. A linear regression model is the simplest and well-developed representation of a casual, time-invariant relationship between an input and output function. Hence a linear regression model was developed as the benchmark for models comparison in flood forecasting. As it can be implied from Table 3 in calibration period, linear regression performed poorly but the result accuracy of validation period is fairly good (e.g., for 1-day-ahead reservoir inflow forecasts, CC = 0.897 and RMSE = 4.73).

Table 1-3 shows that the ANN and regression models generally predict the reservoir inflow for 1-4 day-ahead satisfactorily. Another consequence acquired from this study was that for all 1 to 4-day-ahead forecasts, the ANN model was generally superior in the reservoir inflow (the a values were close to unity, Table 2) in comparison with the FL model (Table 1). The FL model generally underestimates the reservoir inflow. According to the RMSE evaluate function, for the longer period of time forecasting (e.g., 3 and 4-day ahead) the FL model showed the better results in comparison with the corresponding results obtained from the ANN model, in particular for the validating and testing patterns (Table 1, 2). Therefore, for the short period of flood forecasting it seems that the ANN model performed better than the FL model. However, the important point is that if the models would able to predict the peak discharge as accurate as possible. To evaluate how powerful models are in predicting the peak discharge, two observed floods during 4/13/1996 to 4/29/1996 and 4/4/1998 to 4/18/1998 were shown in Fig. 6 and 7 for short and long periods of time prediction respectively. As it can be seen in these figures, ANN and LR models could estimate and predict the flood hydrograph with a fairly good accuracy for 1 and 3-day-ahead. The calculated errors were 3 and 4.5% for 1-day-

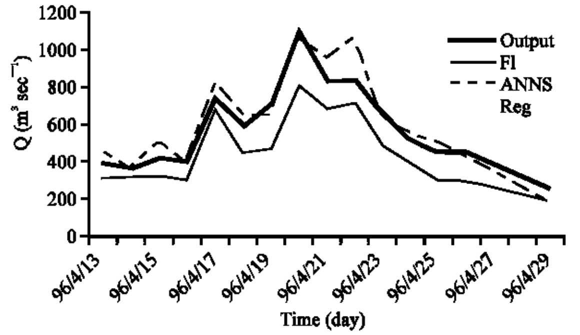

Fig 6: Predicted and measured flood hydrograph for 1-day-ahead forecasts

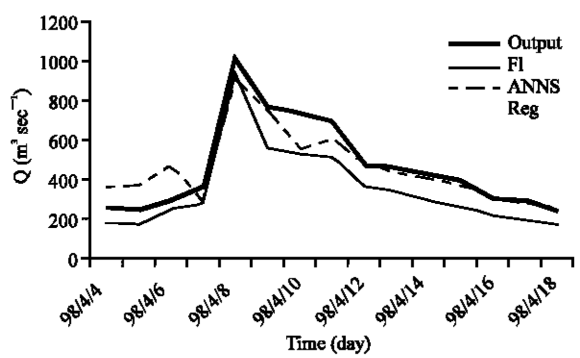

Fig 7: Predicted and measured flood hydrograph for 3-day-ahead forecasts

ahead discharge prediction for ANN and regression models, respectively. The underestimation stream flow prediction of FL model can be also seen in Fig. 6 with the error being about 26%. The error in peak discharge estimation by the regression model was about 0.2% for 3-days-ahead flood predictions. Whereas, the corresponding values for ANN and FL models were 9.3 and 6.0%, respectively. These results show that the FL model was able to predict the peak discharge only for the long period of time flood predictions.

CONCLUSIONS

The main goal of this study was to evaluate the application of fuzzy logic and artificial neural network models for prediction of reservoir inflow in order to control and manage occurring floods. The application results demonstrated that ANNs can predict the reservoir inflow for 1-day-ahead, especially in training period (CC = 0.970, RMSE = 2.45) better than FL (CC = 0.932, RMSE = 2.92) and LR (CC = 0.902, RMSE = 3.65) models. As the prediction time go ahead, FL model showed the better results and ANN model results slightly declined. Another result acquired from this study was that for all 1 to 4-day-ahead forecasts, ANN model generally estimated the flow relatively well, while FL model underestimated the reservoir inflow. According to the errors calculated for peak discharge predictions, it can be finally concluded that the ANN and regression models are more suitable than the FL model for the short term of time flood predictions, e.g., 1-day ahead. The FL model performed slightly better than the ANN model for the long term, e.g., 4-days ahead, flood discharge predictions.

REFERENCES

Ahmad, S. and S.P. Simonovic, 2005. An artificial neural network model for generating hydrograph from hydro-meteorological Parameters. J. Hydrol., 315: 236-251.

ASCE Task Committee, 2000. Artificial neural networks in hydrology. É: Preliminary concepts. J. Hydrol. Eng., 5: 115-123.

Bhattacharya, B. and D.P. Solomatine, 2005. Neural networks and m5 model trees in modeling water level-discharge relationship. Neuro Comput., 63: 381-396.

Blazkova, S. and K. Beven, 2004. Flood frequency estimation by continuous simulation of subcatchment rainfalls and discharges with the aim of improving dam safety assessment in a large basin in the czech republic. J. Hydrol., 292: 153-172.

Chang, F.J., Y.M. Chiang and L.C. Chang, 2007. Multi-step-ahead neural networks for flood forecasting. Hydrol. Sci. J., 52: 114-130.

Chau, K.W., C.L. Wu and Y.S. Li, 2005. Comparison of several flood forecasting models in yangtze river. J. Hydrol. Eng., 10: 485-491.

Chau, K.W., 2006. Particle swarm optimization training algorithm for ANNs in stage prediction of shing Mun river. J. Hydrol., 329: 363-367.

Cheng, C.T., C.P. Ou and K.W. Chau, 2002. Combining a fuzzy optimal model with a genetic algorithm to solve multi-objective rainfall-runoff model calibration. J. Hydrol., 268: 72-86.

Corani, G. and G. Guariso, 2004. Fuzzy modeling of basin state and neural networks for flood forecasting. 2nd International Environmental Modeling and Software Society Conference, Osnabruck.

Dawson, C.W. and R.L. Wilby, 1998. An artificial neural network approach to rainfall-runoff modeling. Hydrol. Sci., 43: 47-66.

Dawson, C.W., R.J. Abrahart, A.Y. Shamseldin and R.L. Wilby, 2006. Flood estimation at ungauged sites using artificial neural networks. J. Hydrol., 319: 391-409.

Deka, P. and V. Chandramouli, 2005. Fuzzy neural network model for hydrlogic flow routing. J. Hydrol. Eng., 10: 302-314.

Gopakumar, R. and E.J. James, 2002. Flood forecasting of achencoil river basin using artificial neural networks. Hydroinformatics 2002: Proceedings of the 5th International Conference on Hydroinformatics, Cardiff, UK.

Hundecha, Y., A. Bardossy and H.W. Theisen, 2001. Development of a fuzzy logic based rainfall-runoff model. Hydrol. Sci., 46: 363-376.

Jantzen, J., 1999. Design of fuzzy controllers. Technical Rep. No. 98-E 864, Dep. Of Automation, Technical University. of Denmark, Denmark.

Klir, G.J., U.H. St Clair and B. Yuan, 1997. Fuzzy Set Theory Foundations and Applications. Prentice Hall: New Jersey, pp: 245.

Kruse, R., J. Gebhardt and F. Klawonn, 1994. Foundations of Fuzzy Systems. Wiley, New York.

Liong, S.Y., W.H. Lim and G.N. Paudyal, 2000. River stage forecasting in bangladesh: Neural network approach. J. Comput. Civ. Eng., 14: 1-8.

Luchetta, A. and S. Manetti, 2003. A real time hydrological forecasting system using a fuzzy clustering approach. Comput. Geosci., 29: 1111-1117.

Mahabir, C., F.E. Hicks and A.R. Fayek, 2003. Application of fuzzy logic to forecast seasonal runoff. Hydrol. Proc., 17: 3749-3762.

Maier, H.R. and G.C. Dandy, 2000. Neural networks for the prediction and forecasting of water resources variables: A review of modeling issues and applications. Environ. Modeling Software, 15: 101-123.

Ni, J.R. and A. Xue 2003. Application of artificial neural network to rapid feedback of potential ecological risk in flood diversion zone. Eng. Applic. Artif. Intell., 16: 105-119.

Ozelkan, E.C. and L. Duckstein, 2001. Fuzzy conceptual rainfall-runoff models. J. Hydrol., 253: 41-68.

Pang, B., S. Guo, L. Xiong and C. Li, 2007. A nonlinear perturbation model based on artificial neural network. J. Hydrol., 333: 504-516.

Rao, A.R. and V.V. Srinivas, 2006. Regionalization of watersheds by fuzzy cluster analysis. J. Hydrol., 318: 57-79.

Sen, Z., 1998. Fuzzy algorithm for estimation of solar irrigation from sunshine duration. Sol. Eng., 63: 39-49.

Tareghian, R. and S.M. Kashefipour, 2006. Karoon river flow forecasting using. Proceedings of Regional Conference on Optimum Utilization of Water Resources in the Karoon and Zayanderood River Basin, Shahrekord, Iran (In Persian).

Tayfur, G. and V.P. Singh, 2006. ANN and fuzzy logic models for simulating event-based rainfall-runoff. J. Hydrol. Eng., 132: 1321-1330.

Theodoridis, S. and K. Koutroumbas, 1999. Pattern Recognition. Academic Press, New York, pp: 482-483.

Vernieuwe, H., O. Georgieva, B. De Baets, V.R.N. Pauwels, N.E.C. Verhoest and F.P. De Troch, 2005. Comparison of data-driven takagi-sugeno models of rainfall-discharge dynamics. J. Hydrol., 302: 172-186.

Wu, C.L. and K.W. Chau, 2006. A flood forecasting neural network model with genetic algorithm. Int. J. Environ. Poll., 28: 261-273.

Xiong, L., A.Y. Shamseldin and K.M. O'Connor, 2001. A nonlinear combination of the forecasts of rainfall-runoff models by the first-order takagi-sugeno fuzzy systems. J. Hydrol., 245: 196-217.

Zadeh, L.A. 1965. Fuzzy sets. Inf. Control, 8: 338-353.