Rainfall-runoff modelling using adaptive neuro-fuzzy systems

D. K. Gautam and K. P. Holz

ABSTRACT

Two important applications of rainfall-runoff models are forecasting and simulation. At present, rainfall-runoff models based on artificial intelligence methods are built basically for short-term forecasting purposes and these models are not very effective for simulation purposes. This study explores the applicability and effectiveness of adaptive neuro-fuzzy-system-based rainfall-runoff models for both forecasting and simulation. For this purpose, an adaptive neuro-fuzzy system with autoregressive exogenous input (ARX) structure is proposed and an application is presented for the modelling of rainfall-runoff processes in the Sieve basin in Italy.

Key words | rainfall-runoff modelling, forecasting, simulation, neuro-fuzzy system, artificial intelligence, subtractive clustering

D. K. Gautam
Department of Hydrology and Meteorology,
P.O. Box 406,
Kathmandu,
Nepal
and
Brandenburg University of Technology at Cottbus,
P.O. Box 101344,
D-03013 Cottbus,
Germany
E-mail: *dilip@bauinf.tu-cottbus.de*

K. P. Holz
Brandenburg University of Technology at Cottbus,
P.O. Box 101344,
D-03013 Cottbus,
Germany
E-mail: *holz@bauinf.tu-cottbus.de*

INTRODUCTION

Flood forecasting and warning constitutes an important task of operational flood management in a river basin. Forecasting of future river flow conditions and early warning of anticipated catastrophic flood events issued to the appropriate authorities and the public greatly helps to prevent and mitigate its effects on the economical, social, environmental and cultural heritage of the area and saves lives and properties. Reliable forecasts of river flows are needed to ensure the maximum benefits from operating the water resources systems. For example, forecasts of reservoir inflows are often required for deciding upon operating rules to effectively operate a system of reservoirs.

The main objective of forecasting is to provide the most accurate predictions of future unknown floods and to give a warning with adequate lead time, and to do so during the entire duration of the flood at specified intervals of time. The model must be run during the flood. Such real-time applications require rapid computation methods to give enough lead time.

Rainfall-runoff models, in combination with on-line data acquisition systems, are used as standard tools for such flood forecasting purposes. At present, lumped conceptual rainfall-runoff models in combination with hydrodynamic river routing models represent the state-of-the-art. However, the real-time operations of these sophisticated models and the initial-state updating usually require powerful computers and highly specialized well-trained personnel for flood forecasting. Due to the decrease in the cost of computer core memory, powerful computers may be available nowadays, but even then the forecasting centres may not want or be able to employ technicians fully capable of dealing with very sophisticated models. Additionally, due to the uncertain nature of the timing of floods such technicians may not be available at the time of a catastrophic flood. Therefore, the prediction tools must be as simple as possible. Consequently, a simplified model should be conceived for use in the forecasting itself (see e.g. Cunge *et al.* 1980, p. 350).

Modern artificial intelligence methods such as fuzzy systems and artificial neural networks can be used for such real-time forecasting problems (see for example French *et al.* 1992; Minns & Hall 1996; Khondker *et al.* 1998; Luk *et al.* 1998; See & Openshaw 1998). These methods provide fast, good-enough and low-cost solutions. Another advantage of these methods is that they can handle dynamic, non-linear and noisy data, especially when the underlying physical relations are very complex and not fully understood.

The forecasting application is a 'one-step-ahead prediction' of river flows where past runoff measurements are usually assimilated into the model for updating the state of the system. It is a relatively easy task because there is significant information accumulated in the runoff history. Usually, there is a high correlation between subsequent measurements. For instance, the naive prediction model $Q(t)=Q(t-1)$ may sometimes give adequate forecasts, especially when the system is sampled rapidly compared to its dynamics.

Another application of rainfall-runoff models is for the simulation of rainfall-runoff processes. In simulation, the state of the system is modelled based on the system forcing. For rainfall-runoff modelling, it means that the runoff is modelled as a function of rainfall history, evapotranspiration and the initial conditions. Such 'simulation' models of rainfall-runoff processes are much more important than the 'one-step-ahead prediction' model and can be employed for forecasting, control, trend assessment, record extension and a better understanding of the dynamics of the system, provided that there are no significant changes in land use or other such factors in the catchment area. The goal of simulation is to employ the fitted model to generate a set of stochastically equivalent sequences of observations which could possibly occur in the future. Simulation is now a widely accepted technique to aid in both the design and operation of water resources systems.

Fuzzy rules can be extracted from observed data to describe the evolution of runoff time series based on the previous states and on other observations such as rainfall and evapotranspiration. This type of model of rainfall-runoff processes can be used for simulation and forecasting the future unknown observations. Another advantage of this type of modelling is that imprecise data can also be taken into account, which is an extremely difficult task in traditional rainfall-runoff models. The performance of the fuzzy system can be improved by using an adaptive network-based structure which updates the membership function parameters by applying a combination of the least-squares method and the backpropagation gradient descent method.

This paper presents the application of a Sugeno-type adaptive neuro-fuzzy-system-based rainfall-runoff model for the simulation and forecasting of floods. The proposed approach is explained in more detail in Section 2. Sections 3 and 4 describe the case study and present the results from an application to data from the Sieve basin in Italy. Some conclusions are presented in the final section.

AUTOREGRESSIVE EXOGENEOUS INPUT FUZZY INFERENCE SYSTEM

AutoRegressive eXogeneous input Fuzzy Inference System (ARXFIS) is a Sugeno-type (Sugeno 1985) fuzzy inference system (FIS) obtained by implementing subtractive clustering on the data (Chiu 1994) and by using an adaptive network-based fuzzy inference system (ANFIS) to update the membership function parameters (Jang 1993). Since the regressor vector is similar to that of ARX models, the model is called ARXFIS. ARXFIS provides a fuzzy inference system as a generic model structure for the modelling of nonlinear dynamic water resources and environmental systems. Figure 1 shows the block diagram of system identification procedure using ARXFIS. TDL in Figure 1 denotes a tapped delay line.

AutoRegressive eXogeneous input Fuzzy Inference System (ARXFIS) can be obtained as follows:

$$y(t)=g(\varphi(t),\theta)+e(t) \tag{1}$$

Regressor vector:

$$\varphi(t)=[y(t\text{–}1)\ \ldots\ y(t-n_a)u(t-n_k)\ \ldots\ u(t-n_k-n_b+1)]^T \tag{2}$$

Predictor:

$$\hat{y}(t \mid \theta)=\hat{y}(t \mid t-1,\theta)=g(\varphi(t),\theta) \tag{3}$$

where $\varphi(t)$ is a vector containing the regressors, θ is a vector containing the parameters of the rule's antecedent

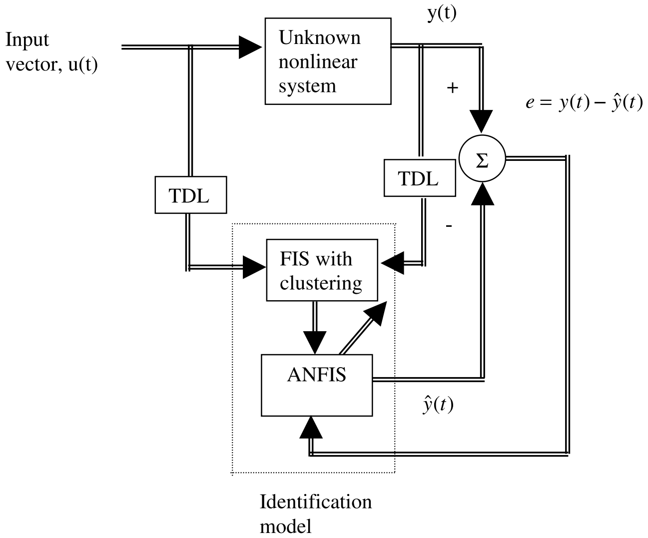


Figure 1 | System identification with adaptive fuzzy systems.

Table 1 | Statistical characteristics of observed runoffs for training and test set

Data set	Mean runoff (m^3/s)	Std. dev.	Maximum runoff (m^3/s)	Minimum runoff (m^3/s)
Training set	53.62	51.30	390.86	12.27
Test set	56.16	85.04	752.65	10.74

and the consequent, $e(t)$ is the prediction error given by $e(t) = y(t) - \hat{y}(t \mid \theta)$ to be minimized and g is the nonlinear function of its arguments realized by the fuzzy-rule-based system. Thus, the input vector consists of a data window made up as follows:

- Present and past values of the input, namely $u(t-n_k)$, $u(t-n_k-1)$, . . ., $u(t-n_k-n_b+1)$, which represent *exogeneous* inputs originated from outside the model.
- Delayed values of outputs, namely $y(t-1)$, . . ., $y(t-n_a)$, which represent *autoregressive* inputs on which the model output $y(t)$ is *regressed*.

Since the output at any instant is a nonlinear function of the past values of both the input and output, the ARXFIS is general enough for most practical applications. An initial fuzzy inference system (FIS) is obtained by implementing subtractive clustering on the data as proposed by Chiu (1994). The identification task then consists of adjusting the parameters of the rule's premise and consequently to optimise a performance function based on the error between the true outputs and the model outputs, which can be done using ANFIS as proposed by Jang (1993). In this way, a dynamic fuzzy inference system can be obtained from observed data alone. The Fuzzy Logic Toolbox Version 2 for MATLAB (1998) can be customised for this purpose.

STUDY AREA AND DATA

The Sieve basin was chosen for this study. The basin has an area of 822 km^2. It is a sub-basin of the Arno basin which is located in Tuscany region, Italy. Three months of data sampled with a one hour interval spanning the time period of 1 December 1959 to 28 February 1960 were available that represent a variety of hydrological conditions and phenomena. The data consist of hourly runoff (m^3/s), rainfall (mm) and potential evapotranspiration (mm). The runoff data were calculated using a rating curve and hourly water levels at Florence. Mean areal rainfall data were calculated based on the Theissen polygon method using 11 rainfall stations.

To evaluate the performance of ARXFIS, a split sample experiment was performed by splitting the total data set into training and test subsets. The first 45 d (1080 h) of data were chosen for training the model and the remaining 45 d data were used for testing the model's performance. The statistical characteristics of the output vector (runoff) for each data set are given in Table 1.

The statistical characteristics of the test set suggest that it contains extreme maximum and minimum values of runoff with a high degree of variability (standard deviation). The model was fitted to the training data set and this model was then employed for simulation and one-step-ahead prediction.

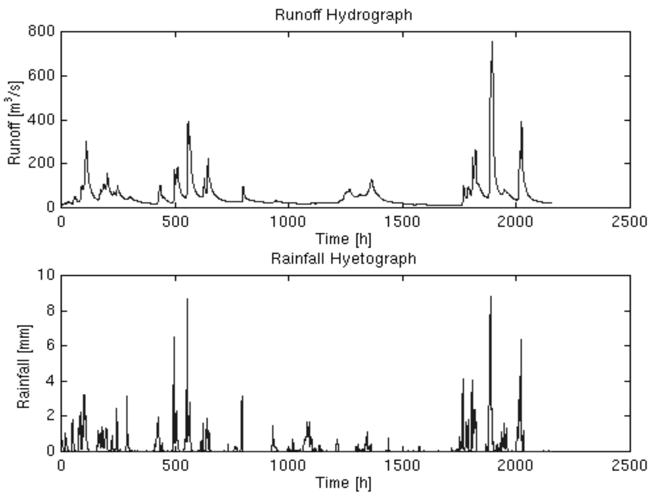


Figure 2 | Runoff hydrograph and rainfall hyetograph.

APPLICATION OF ARXFIS

For rainfall-runoff processes, the stimulus is obviously the rainfall and the response is runoff. Hence, the input vector consists of present and past values of rainfall and evapotranspiration and several delayed values of runoff and the ouput vector consists of current runoff (Q_t). The model then can be written as follows:

$$Q(t) = g(\varphi(t),\theta) + e(t) \tag{4}$$

Regressor vector:

$$\varphi(t) = [Q(t-1) \ldots Q(t-na), R(t-nk_1) \ldots R(t-nk_1-nb_1+1), E(t-nk_2) \ldots E(t-nk_2-nb_2+1)]^T \tag{5}$$

Predictor:

$$\hat{Q}(t \mid \theta) = \hat{Q}(t \mid t-1,\theta) = g(\varphi(t),\theta) \tag{6}$$

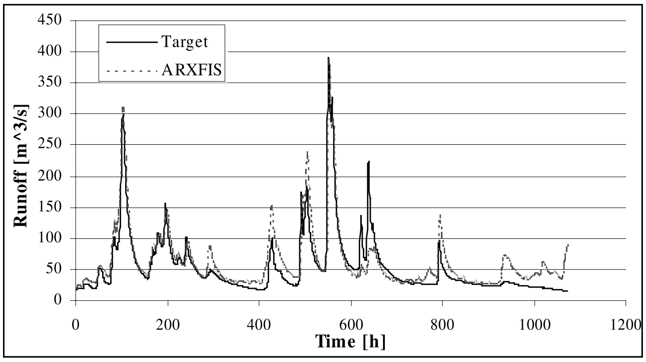
where Q is runoff, R is rainfall, E is evapotranspiration and nk_1 and nk_2 are the pure delay-time (dead-time) in the system.

Several M-files were written in MATLAB and the Fuzzy Logic Toolbox version 2 for MATLAB (1998) has also been customised for the experiment. The maximum cross-correlation between rainfall and runoff was found at lag 6. Based on this preliminary information and by trial and error, the order of the regressor vector components were found to be $na = 3$, $nb_1 = 6$, $nb_2 = 1$, $nk_1 = nk_2 = 1$, to give the best performance. An initial ARXFIS was generated for ANFIS training by first implementing subtractive clustering on the data to determine the number of rules and antecedent membership functions and then extracting a set of rules that models the data behaviour. The cluster centre's range of influence (r_a) plays an important role in determining the number of clusters and hence rules. A larger value of r_a will give fewer clusters and rules, resulting in a coarser model that may not be able to model the processes properly. A smaller value of r_a will give more clusters and rules, which may *overfit* the model. Hence, there exists an optimal value of r_a. The optimal value of r_a was found by trial and error to be 0.46, which gave the best performance on both training and test sets. It means that each cluster centre will have a spherical neighbourhood of influence with 0.46 times the width of the data space. The cluster centre represents the centre of the Gaussian curve and r_a gives the value of spread ($\sigma = 0.3536 r_a \times$ (width of the data space)). In this way, by applying a clustering method we can find the parameters of the Gaussian membership functions for the rule's antecedent. In this application, only two Gaussian membership functions for each input vector and two sets of rules were generated, which is the minimum number of rules required for ANFIS training. Linear least-squares estimation (LSE) was used to determine each rule's consequent equations.

Then ANFIS was used to fine tune the parameters of membership functions of the antecedents and the consequent with an initial learning rate of 0.01. Training was done until either the training error goal (0.0) was achieved or the designated number of training epochs (10) was reached. The error on the test set was also calculated and monitored during each training epoch and the ANFIS returned the model which had the minimum error on the test set. This prevented overfitting of the model.

The Sugeno-type fuzzy-rule-based model has rules of the following form:

If (Past Flow is A and Rainfall is B and Evapotranspiration is C) then (Runoff $= p*$Past Flow + q*Rainfall + $r*$Evapotranspiration + s)



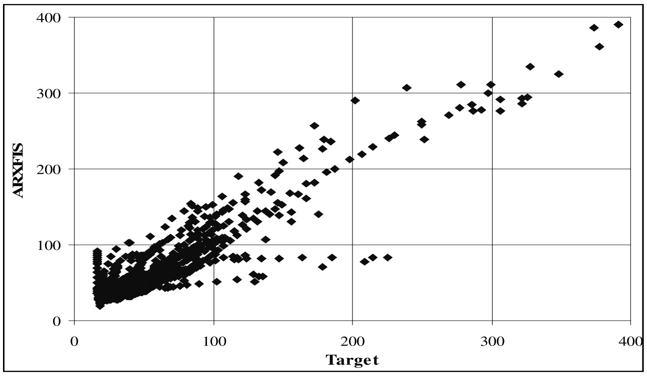


Figure 3 | (a) Observed and simulated hydrograph for training set. (b) Scatterplot of observed and simulated values for training set.

where A, B and C are fuzzy sets (MFs) in the antecedent, while p, q, r and s are the output equation parameters determined by least-squares estimation. In this particular case study, only two expressions or rules were found to be sufficient for the modelling of rainfall-runoff processes, but for more complex catchments there might be several such linear expressions (rules) corresponding to each set of membership functions.

Figure 3(a) shows the observed and simulated hydrograph for the training set. From this figure, it is seen that some of the smaller floods are overestimated but the larger floods are reproduced well. The ARXFIS was able to reproduce the hydrograph shape and general flow patterns. Figure 3(b) shows the scatterplot of observed runoffs against ARXFIS outputs. The scatterplot of observed runoffs against simulated runoffs also shows that the model performs well to estimate higher floods during training giving a high correlation coefficient ($r = 0.917$).

For evaluating the performance of the model for the test set, the following graphical plots and numerical measures were defined:

1. joint plots of observed and computed hydrograph,
2. scatterplot of observed and computed values,
3. residual auto- and cross-correlation functions,
4. Nash & Sutcliffe (1970) coefficient (also known as coefficient of determination),
5. percent bias representing mass balance.

Visual inspection of simple plots (hydrograph, scatterplot) that compare the predictions to actual measurements can provide significant information about how close the predictions are to the observations for different flow regimes.

If the residuals contain no information about past residuals or about the dynamics of the system, it is likely that all information has been extracted from the training set and that the model approximates the system well. To investigate this, the residual auto- and cross-correlation functions are analysed to check whether they are uncorrelated and hence converge to a Gaussian distribution with zero mean and variance $1/N$. Typically, it is checked if the functions for lags in the interval $\tau \in [-20,20]$ are zero or within a 95% confidence interval, i.e. $-1.96/\sqrt{N} < r < 1.96/\sqrt{N}$, where τ represents lags and r represents the auto-/cross-correlation function.

The percent bias (PBIAS) and the coefficient of determination (DC) can be defined respectively as follows:

$$\text{PBIAS} = \frac{\sum_{t=1}^{N} (Q_t^{pred} - Q_t^{obs})}{\sum_{t=1}^{N} Q_t^{obs}} \times 100 \quad (7)$$

$$\text{DC} = 1 - \frac{\sum_{t=1}^{N} (Q_t^{pred} - Q_t^{obs})^2}{\sum_{t=1}^{N} (Q_t^{obs} - Q^{mean})^2} \quad (8)$$

where Q_t^{pred} is predicted flow, Q_t^{obs} is observed flow and Q^{mean} is the mean observed flow. PBIAS measures the

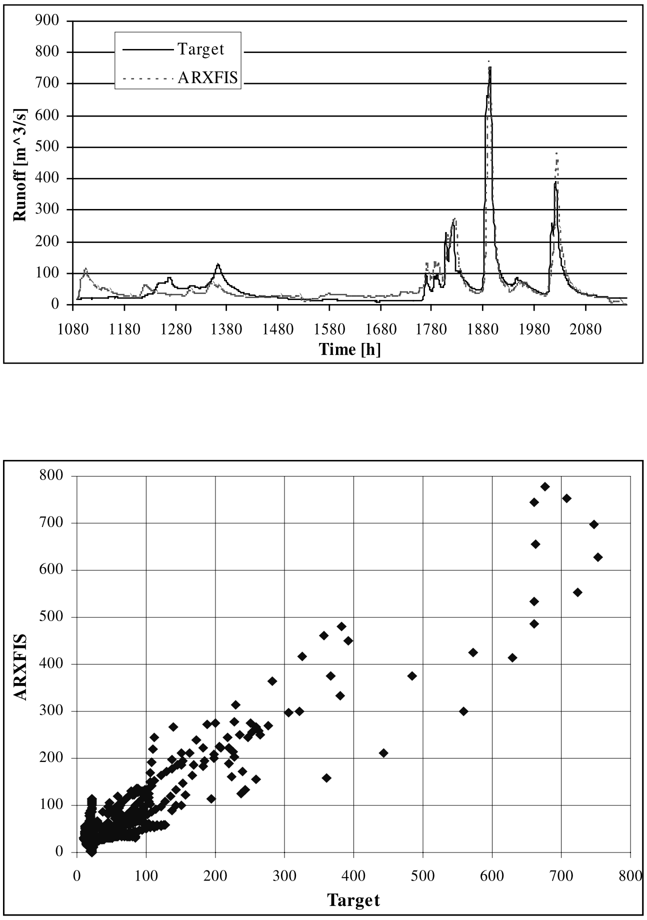


Figure 4 | (a) Observed and simulated hydrograph for test set. (b) Scatterplot of observed and simulated values for test set.

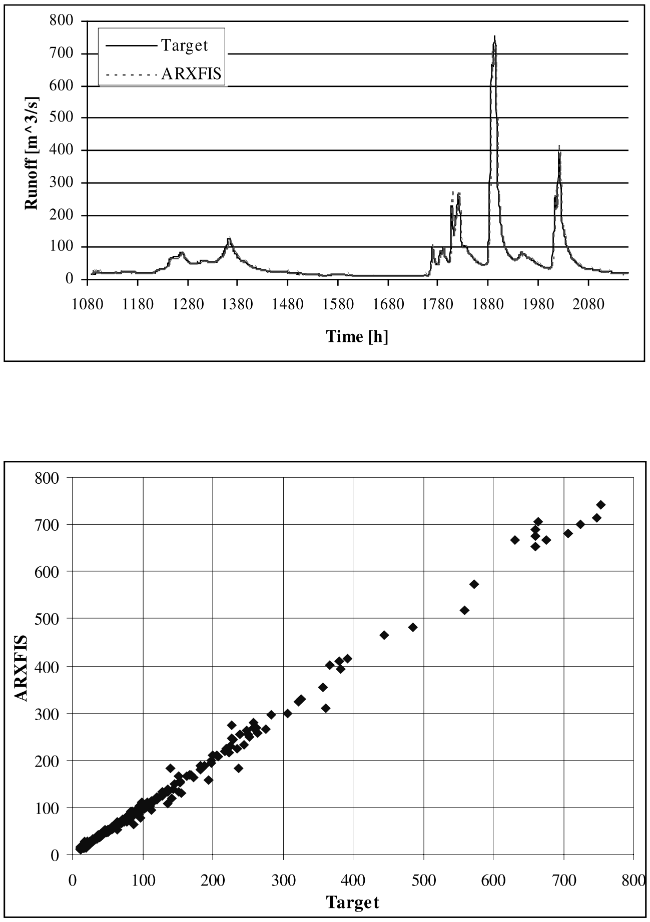


Figure 5 | (a) Observed and one-step predicted hydrograph for test set. (b) Scatterplot of observed and one-step predicted values for test set.

tendency of the predicted flows to be larger or smaller than their observed counterparts. Hence, it is a measure of mass conservation. The optimal value is 0.0, whereas a positive value indicates a tendency of overestimation and a negative value indicates a tendency of underestimation. The coefficient of determination (DC) measures the fraction of the variance of the observed flows explained by the model in terms of the relative magnitude of the residual variance (noise) to the variance of the flows (information). The optimal value is 1.0 and values should be larger than 0.0 to indicate 'minimally acceptable' performance. It is a measure of model efficiency.

Figure 4(a) shows the hydrograph of observed and simulated runoffs during testing and Figure 4(b) shows the scatterplot of observed runoffs against ARXFIS output. These figures show that the model has comparable performance during testing with that during training. The model was quite capable of reproducing the general flow pattern as well as peak flood events. The correlation coefficient of observed and simulated flow was again 0.928.

The model was then employed for one-step-ahead prediction of runoff. Figure 5(a) shows the hydrograph of observed and predicted runoffs and Figure 5(b) shows the scatterplot of observed and predicted runoffs. These figures clearly show the excellent performance of the model during one-step-ahead prediction of river flows with a correlation coefficient of 0.998.

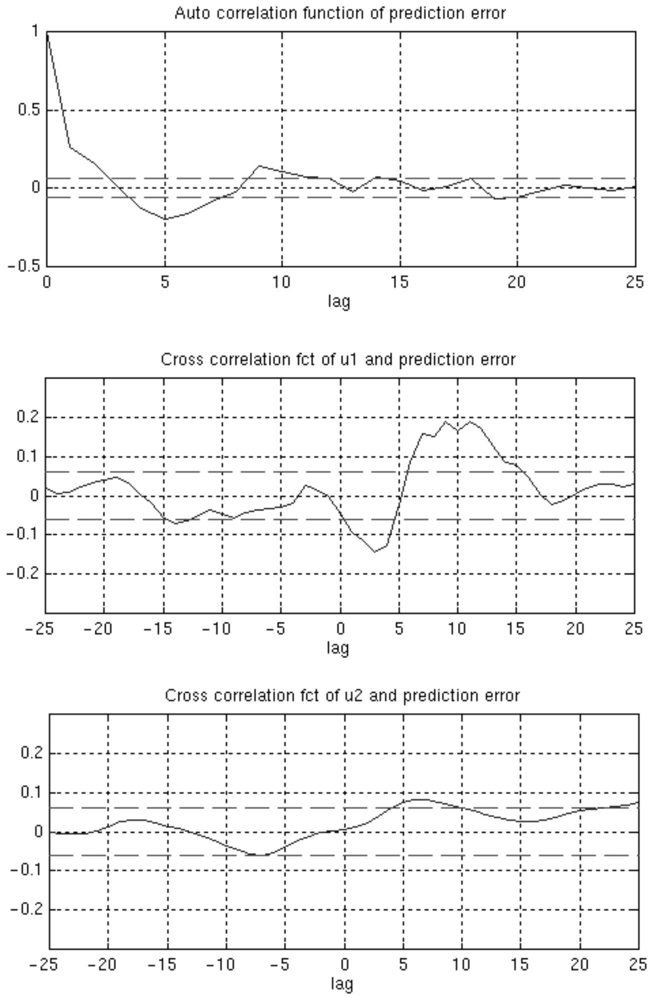


Figure 6 | (a) Autocorrelation function of prediction error. (b) Cross-correlation function of rainfall and prediction error. (c) Cross-correlation function of evapotranspiration and prediction error.

Figure 6(a–c) show the auto-/cross-correlation function of prediction errors (residuals) and inputs. These figures show that, for most of the lag times, the residual auto-/cross-correlation functions lie within the confidence interval.

The percent bias (PBIAS) and the coefficient of determination (DC) as measures of the performance of the model are given in Table 2. The model has very similar and high DC for the training and test set. The model has positive and slightly higher PBIAS during simulation, indicating the overestimation of total flow volume, but during forecasting it has very small PBIAS and very high DC, indicating the excellent performance.

Table 2 | Performance of ARXFIS for simulation and forecasting

Data set	PBIAS (%)	DC
Simulation training	17.992	0.799
Simulation test	7.267	0.859
Forecasting test	0.101	0.996

CONCLUSIONS

The high positive value of PBIAS during simulation indicates that ARXFIS has overestimated the total flow volume, but it has a reasonably good DC. Nevertheless, the hydrographs and scatterplots for the simulation experiments demonstrate that ARXFIS was capable of producing the hydrograph shape reasonably and was able to maintain a good representation of the overall water balance as well as the general flow pattern. It has reproduced the peak floods very well. During simulation, the model trained to capture the relation between the actual observations of the original time series is used in a recurrent form by feeding back the predicted outputs of previous time steps as an input for the next prediction. Hence, the errors which occur for the predicted outputs are propagated and the quality of the simulation will be affected by those errors. When applied to forecast one-step-ahead flows, ARXFIS has shown excellent performance. This provides a sound reason for recommending the ARXFIS as an attractive alternative to complex conceptual models or other empirical models (neural networks, ARX model) for use in rainfall-runoff modelling and real-time flood forecasting.

The successful application of ARXFIS for the modelling of rainfall-runoff processes shows how it can encapsulate the dynamic behaviour of such processes. The case study has clearly demonstrated that ARXFIS can be used for the simulation of rainfall-runoff processes for the conditions that are within the range of the training ensemble. The excellent performance of the model for one-step prediction shows that it will be very valuable for real-time forecasting and control of floods.

ACKNOWLEDGEMENTS

The authors wish to thank M. Franchini for providing the data of Sieve basin, Italy. Thanks are also to D. P. Solomatine for valuable discussions about the application of neural networks and fuzzy logic in the context of hydroinformatics.

NOTATION

$u(t)$	inputs at time t
$y(t)$	output at time t
$\hat{y}(t)$	predicted output at time t
$Q(t)$	flow at time t
$\hat{Q}(t)$	predicted flow at time t
$R(t)$	rainfall at time t
$E(t)$	evapotranspiration at time t
$e(t)$	model error
ANFIS	adaptive network-based fuzzy inference system
ARX	autoregressive with exterior input model
ARXFIS	autoregressive with exterior input fuzzy inference system
FIS	fuzzy inference system
DC	determination coefficient
PBIAS	percent bias
TDL	tapped delay line
n_a, na	order of autoregressive input
n_b, nb	order of exogeneous input
n_k, nk	pure delay time in the system
$\varphi(t)$	regressor vector
θ	parameter vector
g	unknown nonlinear function
r_a	cluster centre's range of influence

REFERENCES

Chiu, S. 1994 Fuzzy model identification based on cluster estimation. *J. Intell. Fuzzy Syst.* **2**(3).

Cunge, J. A., Holly Jr, F. M. & Verwey, A. 1980 *Practical Aspects of Computational River Hydraulics*. Pitman, London.

French, M., Krajewski, W. & Cuykendall, R. R. 1992 Rainfall forecasting in space and time using a neural network. *J. Hydrol.* **137,** 1–31.

Fuzzy Logic Toolbox Version 2 for MATLAB. 1998 The MathWorks, Inc., Natick, MA.

Jang, J.-S. R. 1993 ANFIS: adaptive network-based fuzzy inference systems. *IEEE Trans. Syst., Man Cybern.* **23,** 665–685.

Khondker, M.-Ul-H., Wilson, G. & Klinting, A. 1998 Application of neural networks in real time flash flood forecasting. *Proc. 3rd Int. Conf. on Hydroinformatics, Copenhagen* pp. 777–781. Balkema, Rotterdam.

Luk, K. C., Ball, J. E. & Sharma, A. 1998 Rainfall forecasting through artificial neural networks. *Proc. 3rd Int. Conf. on Hydroinformatics, Copenhagen* pp. 797–804. Balkema, Rotterdam.

Minns, A.W. & Hall, M. J. 1996 Artificial neural networks as rainfall-runoff models. *Hydrol. Sci. J.* **41**(3), 399–417.

Nash, I. E. & Sutcliffe, I. V. 1970 River flow forecasting through conceptual models, I. *J. Hydrol.* **10,** 282–290.

See, L. & Openshaw, S. 1998 Using soft computing techniques to enhance flood forecasting on the River Ouse. *Proc. 3rd Int. Conf. on Hydroinformatics, Copenhagen* pp. 819–824. Balkema, Rotterdam.

Sugeno, M. 1985 *Industrial Applications of Fuzzy Control.* Elsevier, Amsterdam.